基于 Openpose 和 Yolo 的手持物体分析算法

贺文涛，黄学宇，李　瑶
（空军工程大学防空反导学院，西安，710051）

摘要　针对当前模式识别领域少有专门针对手持物体识别的研究，提出了可实时全局分析人体手持物体状态及手持物体类别的分析算法。以人体姿态估计网络 Openpose 和物体检测网络 Yolo 为基础对图像进行初步处理，利用 C＋＋ API 将二者获取到的人体关节点坐标和目标物体坐标进行信息融合，然后针对不同尺寸的物体进行分类并分别设计了判定法则，融合交并比(IOU)算法作为手持状态的辅助判断，最终实现了人体手持物体行为分析算法。采集手持物体的视频流制成数据集，使用多种方法进行数据增强并训练，最终算法识别出手持物体状态的同时，正确识别手持物体类别的准确率可达 91.2％左右，相较于传统方法提高了大约 1.3％，且运行速度可达 13 fps，验证了算法的准确性。试验证明该算法对手持刀具、枪支等危险品的异常行为检测具有较高应用价值。

关键词　Openpose；Yolov4；手持识别；数据增强
DOI　10.3969/j.issn.1009-3516.2021.06.013
中图分类号　TP391.41　　**文献标志码**　A　　**文章编号**　1009-3516(2021)06-0082-08

Hand Held Object Analysis Algorithm Based on Openpose and Yolo

HE Wentao, HUANG Xueyu, LI Yao
(Air Defense and Missile Defense College, Air Force Engineering University, Xi'an 710051)

Abstract　For the current pattern recognition field, there are few researches specifically aimed at hand-held object recognition, and an analysis algorithm that can analyze the state of human hand-held objects and the types of hand-held objects in real-time and globally is proposed. Preliminary processing of the image based on the human pose estimation network Openpose and the object detection network Yolo, the C＋＋ API is used to fuse the coordinates of the human body joint points and the target object coordinates obtained by the two, and then classify and separate objects of different sizes. The judgment rule is designed, and the IOU algorithm is used as the auxiliary judgment of the hand-held state, and finally the behavior analysis algorithm of the human hand-held object is realized. Collect the video stream of the handheld object into a data set, and use a variety of methods for data enhancement and training. The final algorithm recognizes the state of the handheld object, and at the same time the accuracy of correctly identifying the category of the handheld object can reach about 91.2％, compared with the traditional method it has increased by about 1.3％, and the running rate can reach 13 fps, which verifies the accuracy of the algorithm. The algorithm has high application value for the detection of abnormal behaviors of dangerous goods such as handheld knives and guns.

收稿日期：2021-08-16
作者简介：贺文涛(1996—)，男，湖南衡阳人，硕士生，研究方向为图像处理、机器视觉。E-mail：1058574635@qq.com

引用格式：贺文涛，黄学宇，李瑶. 基于 Openpose 和 Yolo 的手持物体分析算法[J]. 空军工程大学学报(自然科学版)，2021，22(6)：82-89.
HE Wentao, HUANG Xueyu, LI Yao. Hand Held Object Analysis Algorithm Based on Openpose and Yolo[J]. Journal of Air Force Engineering University (Natural Science Edition), 2021, 22(6): 82-89.

Key words　Openpose；Yolov4；hand held identification；data enhancement

随着模式识别和图像处理技术的快速发展，人体分析作为该领域一个分支近来也受到了科研人员和工业应用的广泛关注，但当前的大多数分析算法仅将人体自身作为研究点，通过对人体的全身姿态或动作捕捉来进行相关应用，少有针对某一特定身体部位进行的分析研究，对人体与物体的关联研究也比较少。

现有的人体分析算法可按实现方式大致分为两类，自上而下（top-down）和自下而上（bottom-up）。自上而下的方式指的是在每次的检测中首先使用人物检测器识别图片中的人员，然后在此基础上进行分析，如 Newell[1]，WEI[2]等使用的姿态估计方法。这种方式的缺点是需要很高的前期人员识别准确率保证，一旦无法识别人物，姿态估计就会失效；另外，由于对每个人都要使用一个检测器，随着图片中人物数量的增加，其计算成本会随之成倍增加。相较之下，自下而上的方式则可以弥补这些缺点，其首先在全局进行关节热点图的提取，然后根据向量关系进行连通，从而为前期保证提供了高可靠性。但早期的自下而上方法，如 Pishchulin[3]，Insafutdinov[4]等提出的方法，由于最终的解析仍需要复杂的全局推断，因此并未提高效率。文献[5]用贪心算法将关键点连接起来，大大提升了效率，使得实时的人体分析成为现实。

当前的目标检测器通常由三部分组成：第一部分是在拥有海量的图片集如 ImageNet 上经过预训练的主干网络（backbone）；第二部分是用于预测类别和物体方框位置的头（head），主要可分为二阶检测器（R-CNN[6-10]系列）和单阶检测器（YOLO[11-14]，SSD[15]和 RetinaNet[16]等）；第三部分是 Neck 网络，指的是在 backbone 和 head 之间插入的连接层，用于连接不同阶段的特征图。

近年来，手持物体分析正受到越来越多的关注和应用，但大多应用通常有着严格的条件和区域限制。如文献[17]中驾驶员手持通话检测，首先摄像头需要正对驾驶员，然后进行人脸检测，估算耳部位置划出感兴趣区域，再根据区域内手部存在时间加上唇部张合状态来综合判断手持通话状态，步骤相当繁琐且需要很高的前期识别准确保证。

为补充专用于手持物体分析算法的缺失，以及更为准确稳定的识别结果，本文提出了一种可全局分析的手持物体行为分析算法。

1　OPENPOSE 人体姿态估计方法

Openpose[5]是由卡耐基梅隆大学（CMU）感知实验室发布的一种实时多人姿态估计方法。该方法采用一种非参数的表达方式，即局部亲和矢量场（part affinity fields，PAF），来学习将各个身体部位与图片中的各个人体相关联。其体系结构通过对全局的内容进行编码，从而自下而上地用贪心算法进行解析，无论图片中有多少人，都能在保持高精度的同时保证实时性。其网络结构主要是通过一个连续的预测过程的两个分支来同时学习关键点的定位以及它们之间的关联。

Openpose 的网络结构图如图 1 所示。

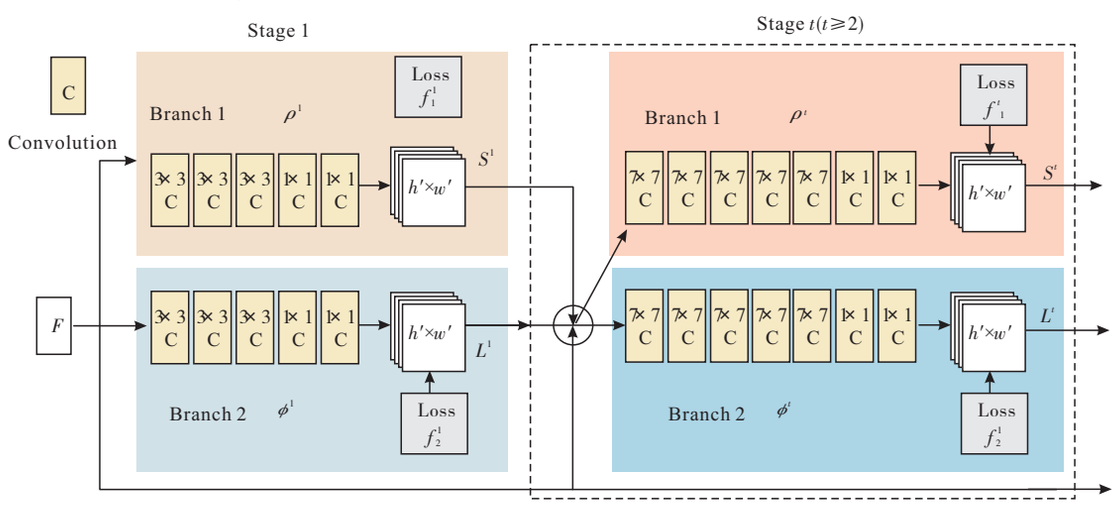

图 1　多阶段的双支路网络结构图

图 1 中 F 是由经过微调的 VGG-19 卷积网络的前 10 层通过对输入图片的分析后生成的特征图集合，然后将 F 作为两个分支第一阶段（Stage 1）的输入，其中分支一（Branch 1）用于预测置信图 S^t，分

支二(Branch 2)用于预测 PAFs-L^t。在每个阶段之后,两个分支的预测结果会被合并作为下一个阶段的输入,并重复上一阶段的操作。

通过上述重复操作,即可预测关键点位置及其置信图。最后,通过贪心算法将这些关键点连接起来即可获得人体的骨架图。如图 2 所示。

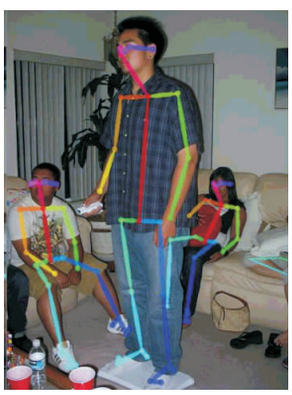

图 2 Openpose 检测结果

其中,两个相邻关键点 d_{j1} 和 d_{j2} 的关联性的评估是通过计算相应矢量场中向量间的线积分来实现的:

$$E=\int_{u=0}^{u=1} L_C(P(u))\frac{d_{j2}-d_{j1}}{d_{j2}-d_{j1}}\mathrm{d}u \tag{1}$$

式中:$P(u)$表示 d_{j1} 和 d_{j2} 之间的点。

$$P(u)=(1-u)d_{j1}+ud_{j2} \tag{2}$$

2 Yolov4 目标检测算法

Yolov4 是由 Alexey Bochkovskiy[14] 等人于 2020 年 4 月发布的目标检测算法。通过将时下最为先进的网络调优方法,如加权残差连接(WRC)、跨阶段部分连接(CSP)、跨小批量归一化(CmBN)、自对抗训练(SAT)、Mish 激活函数、马赛克数据增强、DropBlock 正则化、CIoU Loss 等,在 Yolov3 的基础上进行对比改进实验,最终获得了检测速度与检测精度的最佳平衡的目标检测器——Yolov4。

图 3 为 Yolov4 与当前其他最先进方法在 COCO 数据集上的检测速度、精度对比图。

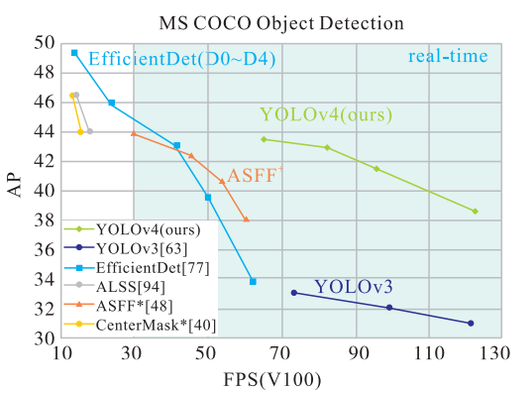

图 3 Yolov4 检测效果对比[11]

由图 3 可知,检测速度差不多时 Yolov4 的检测精度更高;检测精度差不多时,Yolov4 则更快。最终其结构如下:

backbone:CSPDarknet53[18]

neck:SPP[19],PAN[20]

head:Yolov3[13]

此外,网络结构针对单 GPU 训练做了优化,不需要额外的训练成本即可复现其优良性能,因此本文选择 Yolov4 作为物体检测器的基础框架,在其基础上搭建本文算法。

3 基于 Openpose 和 Yolo 的手持物体分析算法

本文的算法流程如图 4,以右手为例进行说明。

输入图片首先经过 Openpose 处理后,提取出图片中的人员骨架关节点,其输出如下(Body_25 模型),见图 5。

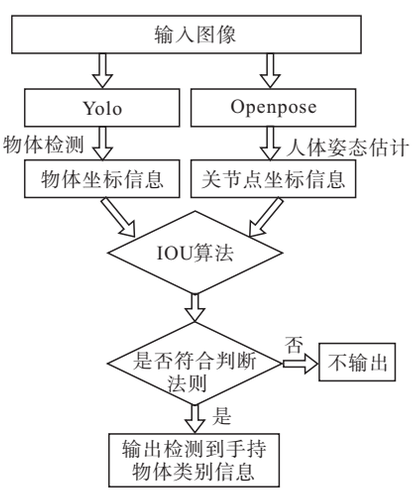

图 4 算法流程图

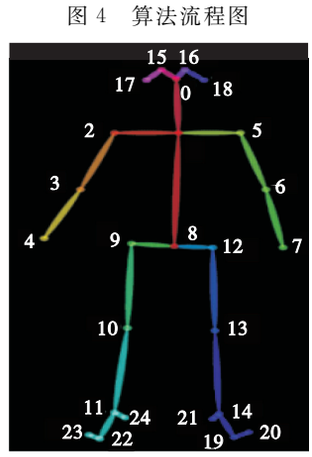

图 5 BODY_25 输出关节图

Body_25 检测模型下与关节点的对应关系如表 1 所示,共有 25 个关节点,每个节点位置输出(x,y,*score*),分别为关节点横坐标、关节点纵坐标以及置信度。

表 1　输出编号对应关节

编号	对应关节	编号	对应关节
0	鼻子	13	左膝盖
1	脖子	14	左脚踝
2	右肩	15	右眼
3	右手肘	16	左眼
4	右手腕	17	右耳
5	左肩	18	左耳
6	左手肘	19	左脚内
7	左手腕	20	左脚外
8	胯中心	21	左脚跟
9	右臀部	22	右脚内
10	右膝盖	23	右脚外
11	右脚踝	24	右脚跟
12	左臀部		

同时，输入图片经 Yolo 处理，检测出图片中感兴趣物体的类别与位置，画出方框。其输出包含（$X_{\text{top\_left}}$，$Y_{\text{top\_left}}$，w，h），obj_id 等，分别为方框左上角横-纵坐标，方框宽-高以及物体类别对应 ID。

经过上述处理，融合两部分输出信息，使用 IOU 算法进行处理，最后进入手持物体行为分析算法的逻辑执行部分。实际手持场景中，不同尺寸大小的物体对手持位置与逻辑判定关系均有不同影响。按手持物体的尺寸大小分为小型物体和中、大型物体两类。

3.1　手持小型物体判定法则

小型物体的边长小于手的长度（15 cm），特点是尺寸较小，形状较规则，宽高比较小，手持时通常握在物体中心，如手机、小刀等，因此当手腕节点与感兴趣物体中心（X_{center}，Y_{center}）的距离小于手的长度即可认为该物体被手持。

如图 6 所示，通过 Openpose 获得右手腕（x_4，y_4），右手肘（x_3，y_3）的位置信息，取右手腕到右手肘距离的一半作为手的长度 L_{hand}（由于手指的关节数量众多，直接提取其关节位置信息会大大增加计算成本，增加算法复杂度），可得：

$$L_{\text{hand}}=\frac{\sqrt{(x_4-x_3)^2+(y_4-y_3)^2}}{2} \tag{3}$$

使用 Yolo 获得感兴趣物体的类别和位置并画出方框，输出方框左上角点坐标（$X_{\text{top\_left}}$，$Y_{\text{top\_left}}$），方框宽 w，方框高 h，可得物体中心点坐标为：

$$X_{\text{center}}=\frac{X_{\text{top\_left}}+w}{2} \tag{4}$$

$$Y_{\text{center}}=\frac{Y_{\text{top\_left}}+h}{2} \tag{5}$$

由式（3～5）可得手持小型物体的逻辑判断条件为：

$$D=\sqrt{(X_{\text{center}}-x_4)^2+(Y_{\text{center}}-y_4)^2}\leqslant L_{\text{hand}} \tag{6}$$

当满足式（6）时，可认为当前人员手持物体。

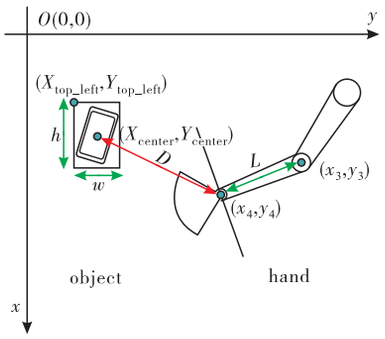

图 6　手持物体逻辑判断图

3.2　手持中、大型物体判定法则

中、大型物体指边长分别大于手的 1 倍（15 cm）和 3 倍（45 cm）以上，特点是尺寸较大，形状不规则，宽高比变化多，手持中心位置浮动较大，如书本、晾衣杆等。此时小型物体的逻辑判断法则不再完全适用。如出现图 7 手持状态时，手腕位置与物体中心距离远大于 L_{hand}，按照小物体逻辑判断法则（6），此时图 7 是未手持状态，因此出现漏检。因此，需要对小物体逻辑判断法则进行扩充。令手腕到物体中心的距离小于物体检测框最长边长的一半，即：

$$D_{\text{mid\_big}}\leqslant\frac{\text{Max}(w,h)}{2} \tag{7}$$

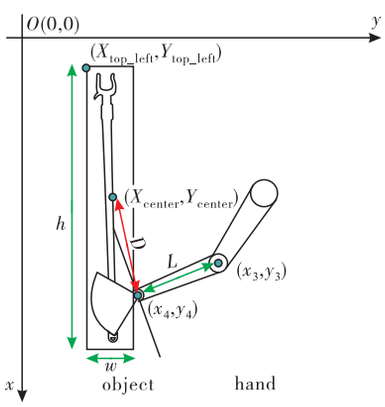

图 7　手持中、大型物体情况 1

根据式（7），虽然提高了图 7 情况手持物体的识别准确率，但对图 8 所示情况的误检（未手持却检出手持）会增多。为解决图 8 所示误检情况，需要对物体与手之间的关联性进行约束，因此参考图像检测中的交并比（IOU）算法进行补充。

交并比（intersection over union）是用于目标检测任务中计算图像重叠比例的算法，主要用于生成候选框的置信度排序。在本文的算法中，利用交并比来判断手与感兴趣物体的关联性大小。

如图 9 所示，A 为 Yolo 检测物体后生成的矩形框，B 为以右手腕关节点（x_4，y_4）为中心，以手长的两倍 $2L_{\text{hand}}$ 为边长绘制的矩形，蓝色部分 C 为 A 与

B 的交叉部分。

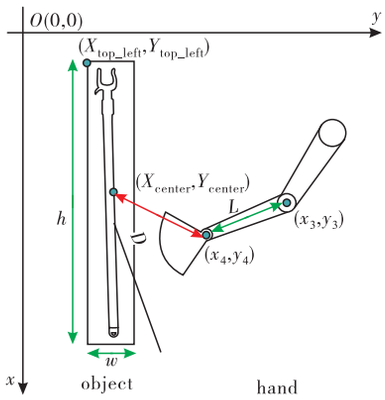

图 8 手持中、大型物体情况 2

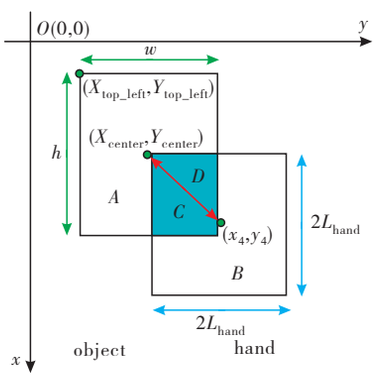

图 9 交并比法则

设 A 左上与右下坐标为(X_{11}, Y_{11}),(X_{12}, Y_{12}),设 B 左上与右下坐标为(X_{21}, Y_{21}),(X_{22}, Y_{22}),则有:

$$X_{11} = X_{top\_left} \tag{8}$$

$$Y_{11} = Y_{top\_left} \tag{9}$$

$$X_{12} = X_{top\_left} + w \tag{10}$$

$$Y_{12} = Y_{top\_left} + h \tag{11}$$

$$X_{21} = X_4 - L_{hand} \tag{12}$$

$$Y_{21} = Y_4 - L_{hand} \tag{13}$$

$$X_{22} = X_4 + L_{hand} \tag{14}$$

$$Y_{22} = Y_4 + L_{hand} \tag{15}$$

可得:

$$x_A = \mathrm{Max}(X_{11}, X_{21}) \tag{16}$$

$$y_A = \mathrm{Max}(Y_{11}, Y_{21}) \tag{17}$$

$$x_B = \mathrm{Max}(X_{12}, X_{22}) \tag{18}$$

$$y_B = \mathrm{Max}(Y_{12}, Y_{22}) \tag{19}$$

式中:x_A,y_A,x_B,y_B 分别为交叉部分左上角与右下角点,故有矩形 A、矩形 B、交叉部分 Intersection 的各部分面积分别为:

$$S_A = (X_{12} - X_{11})(Y_{12} - Y_{11}) \tag{20}$$

$$S_B = (X_{22} - X_{21})(Y_{22} - Y_{21}) \tag{21}$$

$$S_{inter} = \mathrm{Max}(x_B - x_A, 0)\mathrm{Max}(y_B - y_A, 0) \tag{22}$$

则交并比为:

$$IOU = \frac{S_{inter}}{S_A + S_B - S_{inter}} \tag{23}$$

由式(6)~(7)和式(23)得到算法的最终逻辑判断法则为:

$$\begin{cases} D \leqslant L_{hand} \\ D \leqslant \dfrac{\mathrm{Max}(w, h)}{2} \\ IOU > Limit \end{cases} \tag{24}$$

其中 Limit 为自定义的 IOU 阈值。

通过手持状态判定结果结合物体类别识别结果即可输出手持物体类别结果。

4 实验及结果分析

首先对 Yolo 进行训练。按照小型、中型和大型 3 种不同的尺寸大小,包括小刀、手机、水杯、书本、扫帚、晾衣撑共 6 种类别,包含了危险品和日常家庭使用的各种物品,分辨率为 1 280×720,采集了共计 1 489 张图片制成数据集。具体组成如表 2,图片数据输入网络前统一缩放成 512×512 的尺寸大小,将训练集与测试集按照 4∶1 的比列分配。

表 2 数据采集类别及数量

参数	小型		中型		大型	
	0	1	2	3	4	5
类别	knife	Cell phone	Cup	book	broom	Drying rack
数量	233	213	242	244	320	237

为了提高网络的鲁棒性,对训练数据使用了随机旋转,随机缩放,改变色相、对比度、曝光度和马赛克数据增强等方法。其中马赛克数据增强是一种新的数据增强方法,将 4 张图片数据按随机比例拼成一张,这样就能将 4 张图片的内容混合,使得在目标检测时能超出原有的内容范围。图 10 为马赛克数据增强效果图。通过这一方法与未使用该方法相比,最终物体检出率提高了大约 0.3%。

图 10 马赛克数据增强

本文的实验环境配置为：硬件：CPU 为 Inter Core i5-10600KF @ 4.10 GHz，GPU 为 NVIDIA GeForce RTX3070(8G)，16 GB 内存，软件：Windows 10 操作系统，安装 CUDA 11.1，CUDNN 8.0.5，使用 Visual Studio 2019 作为编辑器，OPENCV 4.2.0 用于结果显示。

为适应手持物体的特点，加快算法运行速度以及方便后期将算法移植到轻型计算设备，对网络作出适应性调整。将 Yolov4-tiny 的基于 Resnet 的预训练模型 yolov4-tiny.conv.29 替换成了基于 EfficientNet-B0 的 enetb0-coco.conv.132，称为 Effinet-Yolo；同时在训练时将输入网络尺寸设置为 416×416，在检测时放大至 512×512，以提高小目标物体检出率。分别使用原生的 Yolov4，Yolov4-tiny，以及调整过的 Effinet-Yolo 进行训练，预设迭代次数为 12 000 次，初始学习率分别设置为 0.001 3，0.002 6，0.002 6，学习策略为 step。当 average loss 降到 0.05 至 3.0 以内，或者经过多次迭代后 average loss 不再下降时，停止训练。

在本文实验中迭代至 6 000 次时损失函数即不再下降，停止训练。分别用 3 种模型对验证数据集进行测试，其结果如表 3 所示。

表 3　模型训练结果对比

网络模型	准确率/%	召回率/%	检测速度/fps	网络权重/MB
Yolov4	95.3	98.8	69	245
Yolov4-tiny	89.1	94.2	168	23.1
Effinet-Yolo	92.6	95.7	124	18.3

由表 3 可知 Yolov4 拥有最高的准确率和召回率，但由于其网络层数最深导致其网络权重也最大，运行速度最慢；Yolov4-tiny 由于削减了主干网络，运行速度最快，但准确率最低，而准确率较低的主要原因是小型物体的检出率不足。Effinet-Yolo 在替换了 backbone 后，使得网络权重相对于 Yolov4-tiny 减少了 4.8 MB，从而降低了算法的复杂度；另外通过在训练时减小网络输入尺寸、在检测时增大网络尺寸的方式，使得小型物体的检出率有所提高，相较于 Yolov4-tiny 准确率提高了 3.5%左右。因此，在满足检测性能的前提下，最终选用权重最小最易部署的 Effinet-Yolo 网络模型作为本文算法的物体检测器。

然后对 Openpose 的关节点输出进行筛选，仅保留手腕和手肘的关节信息。使用 C++ API 进行 Openpose 和 Yolo 的坐标信息融合，并用 IOU 算法进行处理，最后进行识别结果验证。将 Yolo 封装成动态链接库使用，并在 VS 上进行本文算法的代码运行。分别使用表中所示模型搭配以及传统的方法，均采集手持不同种类的物体，交替左右手，采用单手或双手，改变身体位姿等视频数据集进行验证结果对比实验。

不同模型搭配本文算法结果如表 4。由第一组结果可知，Yolov4 与 body_25 的组合，虽然准确率最高，但是运行速率只有 6 fps，无法满足实时性要求；由第二组与第三组结果对照可知，body_25 模型运行速度更快，这是因为相较于 coco 模型虽然提取更多的关节点，但是其参数量少，并且由于使用了 CUDA 技术进行加速，因此其运行速度更快。另外，由于本文算法需要依赖于物体检测器 Yolo 和关节提取器 Openpose 的准确率，且二者同时运行，而 Openpose 运行相对较慢，因此限制了算法的运行速率。另外，相较于表中第 4 组传统方法即文献[20]模型，本文算法的准确率有所提高，且本文所提算法不需要划定感兴趣区域，可直接对全局进行分析，表现出更好的泛用性。由表 4 综合考量，Effinet-Yolo 和 body_25 组合为最优搭配。

表 4　检测结果对比

模型	准确率/%	检测速度/fps	参数量/MB
Yolov4+body_25	92.7	6	344.8
Effinet-Yolo+body_25	91.2	13	118.1
Effinet-Yolo+coco	88.3	10	217.3
文献[20]	89.9		

算法运行效果如图 11，最终在图片上输出人体骨架及节点，物体检测框以及左上角的手持状态以及左右手的手持物体类别，由图 11(a)可知算法对前文所提手持小、中、大型物体均有较好识别效果，且能区分左右手，非手持状态也能准确识别；图 11(b)在双手持物、背面以及部分遮挡重叠时也能保持较高准确率，验证了算法的鲁棒性。

本文算法潜在应用场景如下：在工地中，可识别工人是否佩戴安全手套；在家庭的安全监察中，可对儿童拿起刀具等危险品的行为进行识别警告；可用于手持危险品行为检测，如在地铁站或者火车站等闸口用监控视频进行危险行为检测以补充安全检测遗漏；在战场中可通过是否手持武器来区分敌我。

(a) 单手持物

(b) 双手持物

图 11 算法运行效果图

综上,Effinet-Yolo 和 Body_25 组合的模型组合结合算法的判定法则,可使正确识别手持状态同时识别出手持物体类别的准确率达到 91.2%,运行速度可达 13 fps,并且总参数量最少,仅有 118.1 MB。因此将其作为本文的实时手持物体行为分析算法的最终框架。相较于传统手持物体识别的思路,以姿态估计和目标检测为基础进行手持物体识别具有更高的准确度和泛用性,手持定位更为准确,同时本文算法不需要严格的前期人员识别保证,不需要划定感兴趣区域即可进行全局分析。

5 结语

本文针对当前人体分析算法很少以某一特定部位作为研究点的问题,提出了专用于手部的全局实时手持物体识别算法。通过使用 Openpose 和 Yolo 对图片做预处理,然后使用 C++ API 进行二者的坐标信息融合。根据手持物体的尺寸大小分为小型,中型和大型两类情况进行分类,参考交并比(IOU)算法进行处理并作为手持物体状态的辅助判断,最终分别设计出了判断法则,实现了手持物体行为分析算法,并以提高运行速率和方便部署为目的做了适应性调整。在采集的手持物体视频流数据集上,最终识别手持状态的同时准确识别手持物体类别的准确率可达到 91.2%,通过插帧等方式基本可达到实时运行的要求。相较于传统思路方法,本文所提以姿态估计和目标检测为基础的算法定位更为精准,识别准确率更高,算法在民用以及军用等多种场景均具有良好的潜在应用价值。

下一步的工作是使用多线程调度机制等方法,进一步提高算法的运行效率,同时将算法分析的范围扩展到脚部、头部等其他身体部位,形成一套完整的人-物交互分析系统,最终将其应用到无人机监查、智能监控等工程应用中。

参考文献：

[1] NEWELL A, YANG K, DENG J. Stacked Hourglass Networks for Human Pose Estimation[C]//European Conference on Computer Vision. [S. l.]: Springer, 2016: 483-499.

[2] WEI S E, RAMAKRISHNA V, KANADE T, et al. Convolutional Pose Machines[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 4724-4732.

[3] PISHCHULIN L, INSAFUTDINOV E, TANG S, et al. Deepcut: Joint Subset Partition and Labeling for Multi Person Pose Estimation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 4929-4937.

[4] INSAFUTDINOV E, PISHCHULIN L, ANDRES B, et al. Deepercut: A Deeper, Stronger, and Faster Multi-person Pose Estimation Model[C]//European Conference on Computer Vision. Springer, Cham, 2016: 34-50.

[5] CAO Z, SIMON T, WEI S E, et al. Realtime Multi-person 2d Pose Estimation Using Part Affinity Fields [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 7291-7299.

[6] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 580-587.

[7] GIRSHICK R. Fast R-CNN[C]//. Proceedings of the IEEE International Conference on Computer Vision. 2015:1440-1448.

[8] REN S, HE K, GIRSHICK R, et al. Faster F-cnn: Towards Real-time Object Detection with Region Proposal Networks[J]. Advances in Neural Information Processing Systems, 2015, 28: 91-99..

[9] DAI J, LI Y, HE K, et al. R-fcn: Object Detection via Region-Based Fully Convolutional Networks[C]// Advances in Neural Information Processing Systems. 2016: 379-387.

[10] PANG J, CHEN K, SHI J, et al. Libra r-cnn: Towards Balanced Learning for Object Detection[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 821-830.

[11] REDMON J, DIVVALA S, GIRSHICK R, et al. You Only Look Once: Unified, Real-time Object Detection [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 779-788.

[12] REDMON J, FARHADI A. YOLO9000: Better, Faster, Stronger[C]//IEEE Conference on Computer Vision & Pattern Recognition. IEEE, 2017:6517-6525.

[13] FARHADI A, REDMON J. Yolov3: An Incremental Improvement[C]//Computer Vision and Pattern Recognition. Berlin/Heidelberg, Germany: Springer, 2018: 1804-2767.

[14] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. Yolov4: Optimal Speed and Accuracy of Object Detection[J]. arXiv preprint arXiv:2004.10934, 2020.

[15] LIU W, ANGUELOV D, ERHAN D, et al. Ssd: Single Shot Multibox Detector[C]//European Conference on Computer Vision. [S. l.]: Springer, 2016: 21-37.

[16] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal Loss for Dense Object Detection[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 2980-2988.

[17] 王涛. 基于视频分析的驾驶员手持通话行为检测[D]. 哈尔滨：哈尔滨工业大学，2019.

[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]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. 2020: 390-391.

[19] HE K, ZHANG X, REN S, 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, et al. Path Aggregation Network for Instance Segmentation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 8759-8768.

（编辑：徐敏）