ISSN 1004-9037,CODEN SCYCE4
Journal of Data Acquisition and Processing Vol. 36,No. 1,Jan. 2021,pp. 122－132
DOI:10. 16337/j. 1004-9037. 2021. 01. 012

http://sjcj. nuaa. edu. cn
E-mail:sjcj@nuaa. edu. cn
Tel/Fax:＋86-025-84892742

# 面向智慧生物实验室的弱外观多目标轻量级跟踪网络

宗佳平[1]，吴　妍[2]，陈建强[3]，张琳娜[3]，张　悦[1]，岑翼刚[1]

(1. 北京交通大学信息科学研究所,北京 100044;2. 贵阳市公安司法鉴定中心,贵阳 550025;3. 贵州大学机械工程学院,贵阳 550025)

**摘　要**：基于监控视频的弱外观多目标跟踪是建设智慧生物实验室的一个重要内容。但是,由于遮挡、目标外观差别细微等因素的影响,容易出现漏检、误检等问题,导致跟踪失败。此外,基于深度学习的相关算法需要大量的计算量,在嵌入式平台上难以达到实时性。因此,本文提出了一种新的轻量级多目标跟踪算法,以YOLOv3作为基础目标检测网络,提出基于归一化层权重评价的层剪枝算法压缩检测网络计算量,以提高该算法在嵌入式平台上的运算速率。同时,基于已有的跟踪结果,对当前帧检测结果进行校正,实现对漏检目标的补偿校正,用于提高检测的准确性。最后利用卷积神经网络来提取目标特征,融合目标特征及候选框与预测框间的交并补(Intersection-over-union, IoU),进行数据关联。实验结果表明,本文提出的轻量级多目标跟踪算法与已有的多目标跟踪算法相比取得了较好的跟踪结果,且在仅损失较少精度的情况下保持较高的网络压缩率,适于嵌入式平台前端实现。

**关键词**：多目标检测;多目标跟踪;压缩剪枝;智慧实验室;嵌入式平台

**中图分类号**：TP391.4　　**文献标志码**:A

## Lightweight Tracking Network of Weak Appearance Multi-object for Intelligent Biology Laboratory

ZONG Jiaping[1], WU Yan[2], CHEN Jianqiang[3], ZHANG Linna[3], ZHANG Yue[1], CEN Yigang[1]

(1. School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China;2. Criminal Examination Center of Guiyang Security Bureau, Guiyang 550025, China;3. School of Mechanical Engineering, Guizhou University, Guiyang 550025, China)

**Abstract**: Multi-object tracking with weak appearances based on the surveillance video is one important issue for intelligent biology laboratory. However, due to the occlusion and subtle differences among objects, missing detection or false detection is prone to cause tracking failure. In addition, computational cost of deep learning is too high to be realized on embedded platforms. Therefore, a new lightweight multi-objects tracking algorithm is proposed, which uses YOLOv3 as the basic object detection network. A batch normalization layer weight evaluation based layer compression pruning algorithm is proposed to reduce the computational cost of the detection network such that the detection speed can be significantly improved on the embedded platform. Besides, according to the previous tracking results, the missing detection results can be corrected for the current frame, which improves the accuracy of the detection results. Furthermore,

**基金项目**:贵州省自然科学基金(黔科合基础[2019]1064)资助项目;安徽省科技重大专项(17030901047)资助项目;国家自然科学基金(62062021, 61872034)资助项目;北京市自然科学基金(4202055)资助项目。
**收稿日期**:2020-07-15;**修订日期**:2020-09-30

the convolutional neural network is employed to extract the object features. Object features and intersection-over-union (IoU) between the candidate frame and the prediction frame are combined for data association. Experimental results show that the proposed lightweight multi-object tracking algorithm achieves a better result compared with others. Especially, the network achieves a high compression rate with only slightly reducing the detection accuracy, which ensures the proposed network can be easily implemented on the embedded platform.

**Key words:** multi-object detection; multi-object tracking; compression pruning; intelligent biology laboratory; embedded platform

## 引　　言

随着信息技术的发展,视频处理也越来越智能化,比如无人超市、车辆检测等。由于视频场景纷繁复杂,几乎没有视频分析算法能够适应所有的应用场景[1]。例如在生化、生物实验室中,工作人员进行实验操作有规范的流程和步骤,对实验室中仪器的使用也有规定的顺序和使用时间。然而由于缺乏实时的跟踪手段,若工作人员疏忽未按规定流程进行仪器操作,则可能造成实验结果不准确。因此,利用监控摄像头实现对室内工作人员的轨迹跟踪成为智慧实验室建设中重要的内容之一,其可以实现对进入实验室内多个工作人员实时运动轨迹的跟踪,并获取各个工作人员在每台机器前停留的时间,以便对工作人员的操作规范进行追溯。然而,由于工作人员统一的着装,使得在基于视频的目标跟踪中外观特征弱化甚至缺失,为准确跟踪带来一定的困难,容易导致漏检及人员ID分配错误(例如多人同时走动相互遮挡时)。因此,本文重点针对智慧实验室建设中基于固定监控摄像头的弱外观特征下多目标检测及前端实现进行研究,提出基于归一化层权重评价的层剪枝算法,并结合了检测结果校正的轻量级多目标跟踪网络,使整体的跟踪算法能更好地运用于摄像头前端实现,最后将本文算法在NVIDIA Jetson AGX Xavier嵌入式开发板上实现,从而使得在实际应用中,通过智能摄像头边缘计算能力极大地减少视频图像传输、存储以及中心服务器的计算压力。

传统的运动目标检测方法主要是通过帧与帧之间图像信息的变化实现对目标的检测,常用的方法有帧差法、光流法和背景差法等[2],一般分为3个阶段:先在给定的图像上选择一些候选的区域;再对这些区域进行特征提取;最后使用训练的分类器进行分类[3]。但是,这类方法所采用的手工设计的特征对于目标多样性的变化缺少很好的稳健性[3]。随着深度学习技术的快速发展,越来越多的基于深度学习的目标检测算法[4-9]被提出。这类算法主要分为两类:(1)双步目标检测算法,如R-CNN[10]系列,将检测分为两个阶段,先使用区域候选网络(Region proposal net,RPN)来提取候选目标信息,再利用检测网络完成对候选目标的位置及类别的预测和识别;(2)单步目标检测算法,如SSD[11]、YOLO系列[12-13]等,采用了端到端的运算,利用神经网络直接生成目标的位置与类别信息,具有更快的检测速度。

传统的多目标跟踪算法从外观模型角度可以将其归类为生成式模型,Mei等[14]首次将稀疏表示方法引入到视频目标跟踪领域中,其核心思想是将跟踪转化为求解$L_1$范数最小化问题[15]。它没有利用背景信息,仅仅是对目标外观数据的内在分布进行刻画,但是在遇到遮挡时,容易因为错误更新将噪声混入模型,从而出现误差以及漂移[5]。另外一种则是判别式模型,利用在线学习或离线训练的检测器来区分前景目标和背景,从而得到前景目标的位置[16],比如相关滤波算法等。通过深度学习训练网络模型,其卷积特征的输出表达能力更强[17-19],如果将该特征直接应用到相关滤波等方法的跟踪框架中,可以获取更好的跟踪结果。Wojke等[20]提出的算法使用类似于GoogLeNet网络来提取128维的特征,并使用cosine距离来度量表观特征[21]。Wang等[22]提出的跟踪模型则是将目标检测和外观嵌入共享结构学

习,将外观嵌入模型合并到单步检测器中,同时输出检测结果和相应的跟踪结果,实现端到端的运算。

多目标跟踪的具体流程可以分为以下几个步骤:目标检测、特征提取/运动预测、数据关联以及分配ID。首先对每一帧图像中出现的目标进行检测,检测结果的好坏会直接影响跟踪器的性能。因此,本文采用YOLOv3算法进行目标检测时,结合前一帧的跟踪结果对检测结果进行修正,确保检测器的准确性。同时,利用ShuffleNet网络,结合特征向量间的余弦距离及检测框与跟踪框的重叠度(Intersection-over-union,IoU),采用匈牙利匹配算法进行多目标跟踪,具体如图1所示。当视频帧输入该系统后,先对其进行目标检测,采用非极大值抑制(Non-maximum suppression,NMS)法获取目标检测结果,并根据该结果对目标进行裁剪。将裁剪完成的目标输入特征提取网络,获取其对应的特征向量,对其进行目标轨迹间的关联匹配,以获取最终的多目标跟踪结果。此外,通过结构化剪枝对YOLOv3网络进行网络压缩,最后在NVIDIA Jetson AGX Xavier嵌入式开发板上实现剪枝压缩后的深度学习网络,对本文提出的目标检测和跟踪方法进行验证实验。

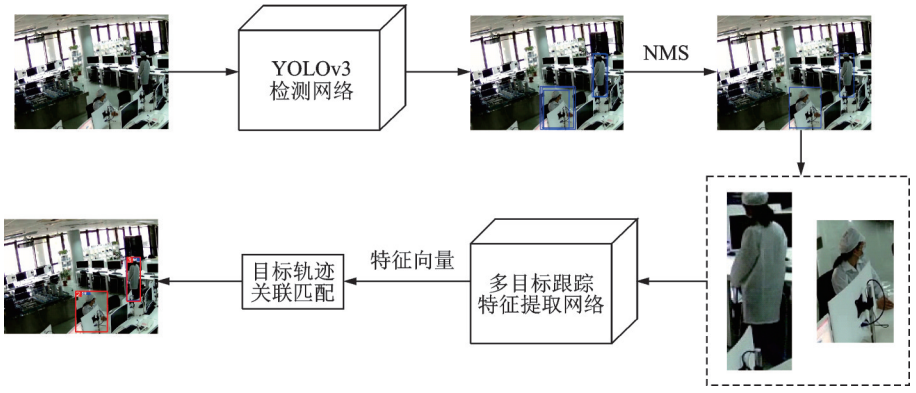



图1 系统框图

Fig.1 System block diagram

## 1 目标检测

自YOLO算法被提出以来,已经过3个版本的更替,在精度和速度上均获得了巨大的提升,在Titan X上检测速度可达到45帧/s,是目前检测速度最快的目标检测算法之一[3]。然而,目前基于卷积神经网络(Convolutional neural network, CNN)的目标检测模型的训练与测试均依赖如Titan X这样的台式GPU计算平台,计算资源消耗大,难以向嵌入式平台移植,因此很难满足工业界对于目标检测实时性和便携性的需求[23-24]。因此本文提出了一种基于YOLOv3的轻量型目标检测网络结构。在该结构中,对原有的特征提取部分的网络层数进行适度压缩。同时,为了弥补在单帧图片进行检测过程中由于置信度低而造成漏检的情况,利用前一帧的跟踪结果,对当前帧进行检测结果校正,确保当前帧目标检测的准确性。

### 1.1 YOLOv3模型压缩

为了嵌入式平台上的目标检测网络能够达到实时性,需要设计轻量级的检测网络模型。因此,本文提出了一种基于归一化层权重评价层剪枝算法,实现对YOLOv3现有的网络模型的剪枝压缩。

在剪枝压缩的过程中可以发现,模型的卷积核权重值的分布大部分处于0的附近,如图2所示。所以,本文先根据卷积核的权重来对网络进行通道剪裁,再根据网络结构中shortcut层前面BN(Batch normalization)层的$\gamma$权重进行层剪枝。通过图2可以发现,卷积核的权重小于0.5的占比较大,所以在进行

模型剪枝时,设定网络裁剪率$P$,从卷积核权重中确定出其对应的数值,将其作为全局阈值$T_1$,用来确定需要修剪的特征通道。此外,为了防止卷积层上的过度修剪并保持网络连接的完整性,需要确保每个卷积层中最大的一个卷积核权重大于全局阈值$T_1$。如果出现某一个卷积层被裁剪掉,则保留其中权值最大的卷积核。

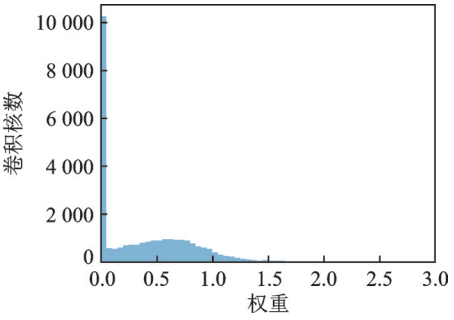



图2 模型权重直方图

Fig.2 Model weight histogram

在YOLOv3网络中,存在23个shortcut结构,为保证其结构的完整性,剪掉1个shortcut时会同时剪掉它前面的2个卷积层。网络中的BN层,即归一化层,其表达式为

$$y=\gamma\times\frac{x-\bar{x}}{\sqrt{\delta^2+\varepsilon}}+\beta \tag{1}$$

式中:$\bar{x}$和$\delta^2$为输入特征的均值和方差;$\gamma$与$\beta$表示可训练尺度因子和偏差;$\varepsilon$为一个很小的正数,目的是为了防止方差为0产生无效计算。

在进行层剪枝时,先对$\gamma$添加$L_1$范数正则化进行稀疏化训练,再针对每一个shortcut前的一个卷积层、BN层及激活层进行评价。对各层的最高$\gamma$值进行排序,取最小$\gamma$值对应层进行层剪枝,从而进一步减少网络的深度,提高网络推理速度。具体的剪枝方法如表1所示。

**表1 模型剪枝算法**

**Table 1 Model pruning algorithm**

| |
|---|
| 输入:网络结构,各卷积核权重$L_1,L_2,\cdots,L_M$,通道剪枝裁剪率$P$,层剪枝数$N$。 |
| 过程: |
| 1: 根据网络结构训练网络模型 |
| 2: 计算全局阈值$T_1$:卷积核权重一共有$M$个,按从小到大顺序排列后,其第$M*P$个数值作为全局阈值 |
| 3: 对每一层的卷积核进行裁剪: |
| 4: 对该层卷积核权重进行从小到大排序 |
| 5: if 该层权重最大值小于阈值: |
| 6: 保留权重最大的卷积核,其余删除 |
| 7: else |
| 8: 删除该层权重低于阈值的卷积核 |
| 9: 裁剪完成后,重新进行训练,并针对BN层的$\gamma$值添加$L_1$正则化,进行稀疏化训练 |
| 10:训练完成后,对每一个shortcut层前BN层的$\gamma$值进行从小到大排序 |
| 11:根据层剪枝数$N$,将最小的$N$个$\gamma$值所对应的shortcut进行裁剪,包括它前面的两个卷积层 |
| 12:将裁剪之后的模型进行重新训练 |
| 输出:裁剪之后的网络结构,新的网络模型文件 |

压缩之后,网络参数量及推理时间等如表2所示。实验时,使用的数据集是从公开数据集中随机提取的只含有person类别的图片,训练集有5 000张,测试集1 300张。表2中,当压缩比例为50%时,仅对网络模型进行了通道裁剪,其网络参数量减少了57%,推理速度提升了近66%,网络精度相比原模型也略有提升。压缩比例为95%时,对网络模型进行了进一步的通道裁剪,同时也对网络层数进行了裁剪。此时,网络参数量减少了将近99%,推理速度提升了70%,而网络精度也仅仅只是降低了8.5%。因此,当该网络部署到嵌入式设备上时,在确保精度的前提下大大提高了物体检测速度。

**表2 通道剪枝与层剪枝前后效果对比**

**Table 2 Comparison of effects before and after channel pruning and layer pruning**

| 压缩比例 | 网络大小/MB | 网络参数量 | 推理时间/ms | 网络精度/mAP |
|---|---|---|---|---|
| 0.00 | 246.3 | 61 523 734 | 30.5 | 0.777 |
| 0.50 | 107.2 | 26 764 372 | 20.2 | 0.812 |
| 0.95 | 4.6 | 1 155 624 | 5.9 | 0.711 |

### 1.2 基于前一帧跟踪结果的检测结果校正

在进行实验室内的目标检测时,室内的人员数量较少,不会出现人员拥挤的情况。但是一般情况下,室内成员大都处于近乎静止状态。同时,室内遮挡也比较严重,比如机器、风扇等。当人员出现同时运动时,也可能产生一定的遮挡,导致检测出的目标置信度比较低,从而出现漏检的情况,造成目标的跟踪失败。因此,为了尽可能降低目标漏检情况的发生概率,结合实验场景的特殊性,本文针对检测结果进行了校正,在对每一帧图像检测完成之后,利用前一帧的跟踪结果与当前帧的检测结果进行对比,判断是否出现了疑似漏检的情况。如果出现该情况,在确保漏检目标确实在监控可视区域的情况下,即目标未曾离开该监控区域,结合前一帧的跟踪结果,对当前检测结果进行修正更新,尽可能提高检测结果的准确性,以确保后面的跟踪算法性能。具体如表3所示。

**表3 检测结果更新算法**

**Table 3 Detection result update algorithm**

输入:前一帧目标跟踪结果集合 $T$,当前帧的目标检测结果集合 $D$

过程:

1:跟踪集合 $T$ 中某一个跟踪结果 $t_1=\{t_{x1}, t_{y1}, t_{x2}, t_{y2}, \text{flag}\}$,与检测结果 $d_1=\{d_{x1}, d_{y1}, d_{x2}, d_{y2}, \text{flag}\}\in D$ 进行对比

2:if abs($t_{x1}-d_{x1}$)<10 and abs($t_{y1}-d_{y1}$)<10

3: 目标在检测区域内,且成功检测,检测结果保持不变, flag = 0

4:else

5: 目标疑似漏检

6: if 目标在可视区域边缘内

7: if 连续漏检帧数flag大于5帧

8: 目标已离开可视区域,检测结果不进行修改,该目标校正完成,进行下一个目标校正

9: else

10: 目标标记为漏检,flag = flag + 1

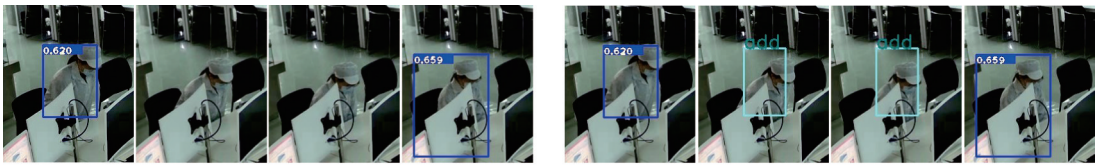
11: 将该目标的跟踪结果作为检测结果,添加进集合 $D$ 中

12:目标校正结束,进行下一个跟踪目标结果校正

输出:校正完成之后的检测结果

假设在前一帧检测跟踪过程中,出现了 $m$ 个目标,用 $T$ 表示,当前帧的检测结果有 $n$ 个目标,用 $D$ 表示,视频帧大小为 $w\times h$,用 $[x_1, y_1, x_2, y_2]$ 表示帧中的某一个目标所处的位置,而 $(x_1, y_1)$ 和 $(x_2, y_2)$ 则分别代表了目标所处矩形区域的左上角坐标和右下角坐标。首先需确保跟踪目标理论上仍然处于室内,并未离开可视区域,防止后面进行数据更新时一直将其默认为处于室内而造成的误检情况。因此,当目标处于边缘位置,即 $x_1<3$ 或 $w-x_2<3$,在进行检测校正时,若作为漏检目标,则需确认其作为漏检目标时连续漏检帧数 $t$。如果连续5帧处于漏检状况,即 $t\geqslant 5$ 时,该目标标记为离开实验室,不再

参与结果校正更新过程;反之在漏检的后5帧中出现某一帧再次检测出目标,则将$t$置0,重新开始计数。遍历前一帧所有跟踪结果,与当前帧的检测结果进行比较,计算结果差用d$x$和d$y$表示。由于是相邻帧的数据进行比较,无论目标运动速度快慢,两帧中的同一个目标坐标差别不大,并且考虑到室内人员不多,所以如果d$x$<10同时d$y$<10,则说明该目标仍然处于室内,并成功检测;反之说明该目标在当前帧被漏检了。因此,当目标被标记为漏检时,利用该目标在前一帧中的跟踪结果修正当前帧的检测结果,进行下一步的跟踪运算。数据修正之后,检测效果如图3所示。




(a) Missed detection images (b) Corrected tracking result

图3 检测结果修正

Fig.3 Correction of test results

图3中所显示的图片是视频中的第110~113帧,图3(a)中的4幅图像是检测器检测出的实际结果,但是因为检测目标置信度低,出现了漏检的情况;图3(b)中的4幅图片则是结合了跟踪器的跟踪结果,对漏检目标进行修正并补充检测框。从图3可以看到,经过本文算法的修正,补充的目标框较为准确,确保了后续多目标跟踪的准确性。

## 2 目标跟踪

与单目标跟踪不同,多目标跟踪会存在新目标进入与旧目标消失的问题。在单目标跟踪中,一般会根据给定的初始框,在后续视频帧中对初始框内的物体进行位置预测。但是在进行多目标跟踪时不考虑初始框,而是采用数据关联的方法进行跟踪,即对每一帧进行目标检测,并利用检测结果进行特征提取,再选用余弦距离作为相似性度量方法进行数据关联,进而实现多目标跟踪。

### 2.1 特征提取

目前,图像特征的提取主要有两种方法:传统图像特征提取方法和深度学习方法。其中,传统的特征提取方法依赖于手工特征设计,常见的特征提取算子有Harris、SIFT和HOG等[25]。相对于传统方法,利用深度学习提取的特征表达能力更强。它是通过一个卷积神经网络对样本进行自动训练,从而获取区分图像的特征分类器。提取特征的好坏会影响轨迹与检测框之间的分配结果,因此为了得到比较好的判别效果,并保证跟踪的准确性与实时性,本文选用ShuffleNet[26]作为基础网络用于提取目标特征。

在实验过程中,根据第1节所提的目标检测网络对每一帧图像进行人员检测,利用检测结果从每一帧图片中截取各目标区域,将其统一变成256像素×128像素大小的图片,作为ShuffleNet网络的输入图像。同时,将该网络最后的全局池化层和全连接层,替换成自适应池化层,使得输出宽高为(8, 4),维度为1 024的张量,然后使用$L_2$正则化,将提取出的特征向量投影在单位超球面上面。

在进行网络训练时,训练集采用行人重识别数据集Market1501[27],该数据集包含在不同视角和不同场景下拍摄的1 501个不同的行人目标,总共32 000张训练图片,其中训练集有751人,包含12 936张图像,采用批量输入的方式进行训练。为了更好地区分正负样本,采用triplet loss损失函数,每次从训练样本中挑选难训练样本进行训练,使得正样本之间的特征向量距离尽量小,负样本之间的特征向量距离尽量大。

### 2.2 相似性度量

在进行多目标跟踪时，目标特征由2.1节所使用的ShuffleNet网络生成，由于余弦距离能够包含经过长时间遮挡后的特征信息，因此在进行相似度计算时，对于每个边界检测框$d_j$，本文计算特征描述符$\boldsymbol{r}_j$并进行归一化处理。通过余弦距离，对当前的特征向量与之前匹配成功的特征向量集进行相似性度量，具体相似度计算公式为

$$f(T_i, D_j^t) = \min\{1 - \boldsymbol{r}_j^{\mathrm{T}} r_k^{(i)} \mid r_k^{(i)} \in \boldsymbol{R}_i\} \tag{2}$$

式中$\boldsymbol{R}_i = \{r_k^{(i)}\}$为对第$i$个跟踪器创建的特征描述符集合，包含其过去成功跟踪后所对应的目标特征向量集，最多只保留100个。$f(T_i, D_j^t)$是计算在第$t$帧中，第$j$个目标检测框与现有的第$i$个跟踪器所跟踪的所有特征向量之间最小余弦距离，并将其作为两者之间的特征相似度。

预测框与候选框之间的IoU的计算函数为

$$\mathrm{IoU}(T_i^t, D_j^t) = \frac{T_i^t \bigcap D_j^t}{T_i^t \bigcup D_j^t} \tag{3}$$

式中：$T_i^t$表示第$i$条轨迹在第$t$帧中的预测框的位置；$D_j^t$表示第$t$帧中的第$j$个候选框。最低重叠率设置为0.3。

考虑到应用场景是实验室，室内人员较少，在处理候选框与预测框之间的关系时，优先使用目标特征对数据进行关联匹配。同时，为提高算法运行速度，针对先前成功匹配的目标与轨迹，采用IoU运算对数据进行关联。具体的运算流程如图4所示。

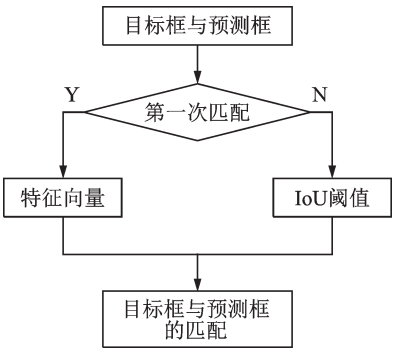



图4 数据关联流程

Fig.4 Data association process

## 3 验证与分析

本文实验中使用联想笔记本电脑进行实验，操作系统为Ubuntu16.04，显卡是GTX1060，显存大小为6 GB。在该电脑上对网络进行训练完成后，再将其部署到嵌入式平台NVIDIA Jetson AGX Xavier上面，用于搭建实时多目标检测跟踪系统。其中，Xavier采用的是8核ARM v8.2 64位CPU，GPU采用的是NVIDIA Volta架构，具有512个NVIDIA CUDA（Compute unified device architecture）核心和64个Tensor核心。

### 3.1 检测结果对比

在目标检测的实验中，本文从公开数据集COCO、VOC中提取了含有person类别的图片，并从中随机选取6 300张图片作为本次实验的数据集，其中5 000张作为训练集，1 300张作为测试集。训练时，采用pytorch框架，将训练集中的图片统一变为416像素×416像素大小输入检测网络进行训练。训练完成后，先对该模型进行通道剪枝，接着对其稀疏化训练并层剪枝，剪枝完成后对网络再一次训练。最后在装有Ubuntu16.04操作系统的笔记本电脑上对网络模型进行测试，测试结果如表4所示。

表4中，第1、2行分别是现有的压缩版YOLOv3模型和原本的YOLOv3模型，第3、4行分别是利用本文1.1节所述的剪枝方法对YOLOv3进行50%和95%压缩后的结果。其中，$F_1$-Score[28]为查准率与召回率的调和平均数，即

$$F_1\text{-Score} = 2 \times \frac{\text{查准率} \times \text{召回率}}{\text{查准率} + \text{召回率}} \tag{4}$$

从表4中可以发现，本文针对YOLOv3网络进行50%压缩率裁剪之后，其网络精度相对原本的网络略有提升，同时推理时间也提高了33%。当压缩率达到95%时，网络大小、参数量及推理速度大大减

表4 目标检测网络裁剪前后效果对比

Table 4 Comparison of effect of target detection network before and after cutting

| 网络名称 | 压缩比例 | 网络大小/MB | 网络参数量 | 推理时间/ms | 网络精度/mAP | 查准率 | 召回率 | $F_1$-Score |
|---|---|---|---|---|---|---|---|---|
| YOLOv3-tiny | | 33.1 | 8 669 876 | 7.5 | 0.584 | 0.756 | 0.683 | 0.699 |
| YOLOv3 | 0.00 | 246.3 | 61 523 734 | 30.5 | 0.777 | 0.864 | 0.799 | 0.830 |
| YOLOv3-0.5 | 0.50 | 107.2 | 26 764 372 | 20.2 | 0.812 | 0.831 | 0.835 | 0.833 |
| YOLOv3-0.95 | 0.95 | 4.6 | 1 155 624 | 5.9 | 0.711 | 0.773 | 0.742 | 0.757 |

少,网络精度、查准率及召回率略有降低,但是与YOLOv3-tiny相比,参数量减少了87%,推理速度提升了2.6 ms,精度提升了约12%,$F_1$-Score也提升了约8%。与原本的YOLOv3网络相比,本文方法在推理速度大大提升的同时,其精度只是略有降低。

### 3.2 跟踪结果对比

在多目标跟踪的实验中,对特征提取网络进行训练,训练集采用公开数据集Market1501,一共包含751个类别。训练完成后,利用该网络提取检测到的目标特征进行多目标跟踪。在多目标跟踪算法测试的实验中,采用的数据集为2D MOT16,该数据集包含14组视频序列,包含固定摄像头和移动摄像头两种拍摄方式。表5是本文方法和现有的一些算法在数据集MOT16上的实验结果,使用的是配有GTX1060显卡的笔记本电脑,操作系统是Ubuntu16.04,在测试过程中所涉及到的相关对比算法,均是采用文献[21]提供的检测结果。表5中MT(Mostly tracked)表示命中的轨迹占总轨迹的占比,ML(Mostly lost)表示丢失的轨迹占总轨迹的占比,FP(False positives)表示目标误检次数,FN(False negativate)表示目标丢失次数,IDs(ID switch)表示ID改变的总次数,FM(Fragmentation)表示跟踪过程中轨迹中断次数,MOTA(Multiple object tracking accuracy)表示多目标跟踪准确度,MOTP(Multiple object tracking precision)表示多目标跟踪精度。

表5 多目标跟踪算法比较

Table 5 Comparison of multi-target tracking algorithms

| 网络名称 | MT (↑) | ML (↓) | FP (↓) | FN (↓) | FM (↓) | IDs (↓) | MOTA (↑) | MOTP (↑) | 跟踪器处理速度/(帧·$s^{-1}$) (↑) |
|---|---|---|---|---|---|---|---|---|---|
| EAMIT[26] | 19.0 | 34.9 | 4 407 | 81 223 | 1 321 | 910 | 52.5 | 78.8 | 12 |
| POI[21] | 34.0 | 20.8 | 5 061 | 55 914 | 3 093 | 805 | 66.1 | 79.5 | 10 |
| SORT[29] | 25.4 | 22.7 | 8 698 | 63 245 | 1 835 | 1 423 | 59.8 | 79.6 | 60 |
| Deep sort[20] | 32.8 | 18.2 | 12 852 | 56 668 | 2 008 | 781 | 61.4 | 79.1 | 40 |
| 本文算法 | 33.4 | 16.8 | 5 060 | 38 413 | 1 225 | 614 | 60.1 | 81.9 | 44 |

从表5中发现,本文算法减少了ID切换的次数,与Deep SORT[20]相比,IDs从781减少到614,减少了约14%,FN减少了约32%;相比于EAMIT[26],FM也降低至1 225。同时,本文提出的跟踪算法在装有GTX1060显卡的笔记本电脑上的运行速度达到44帧/s,达到实时性要求。

### 3.3 实时性对比

当网络训练完成之后,将多目标检测跟踪系统分别部署到上述带有GTX060显卡的笔记本电脑和

嵌入式设备Xavier上进行测试。其中,输入的测试视频的分辨率为640像素 × 480像素,帧率为24帧/s,具体测试结果如表6所示,其中Ubuntu是指在笔记本电脑上的测试结果。从表6中可以发现,本文对YOLOv3网络进行压缩之后减少了模型占用空间,同时提升了模型推理速度。但是由于运行设备不同,在嵌入式设备上使用性能,其实时性能达到13帧/s左右。

在3.2节中算法性能比较时,视频的检测结果已知,采用的均为文献[21]提供的检测结果,利用现有结果进行跟踪性能比较,所以其速度能够达到44帧/s。在进行实时性对比过程中,本文采用的是在线跟踪,需要先获取每一帧的检测结果,再对这一帧进行跟踪。由于受到检测网络推理速度的影响,其速度目前最大只达到20帧/s。

同时,该算法在作者自建的生物实验室数据集上进行了测试。在数据集中,目标均统一着装白大褂、白口罩及白帽子,具体的实验结果如表7所示。表7中的数据是采用表3所描述的检测结果校正算法前后的对比结果。从表7中可以看出,经过表3算法的检测结果修正后,MOTA能够保持在98%左右。具体的实验结果也可以从图5中看出,其中:图5(a)两幅图像是视频中的第736、750帧,目标5进入测试区域并经过目标3后,其各自的跟踪仍然保持不变;图5(b)两幅图片是视频中的第1 216、1 250帧,目标1和3互换位置,位置交换结束后其对应的跟踪结果仍然保持不变。

**表6 嵌入式平台的模型性能对比**

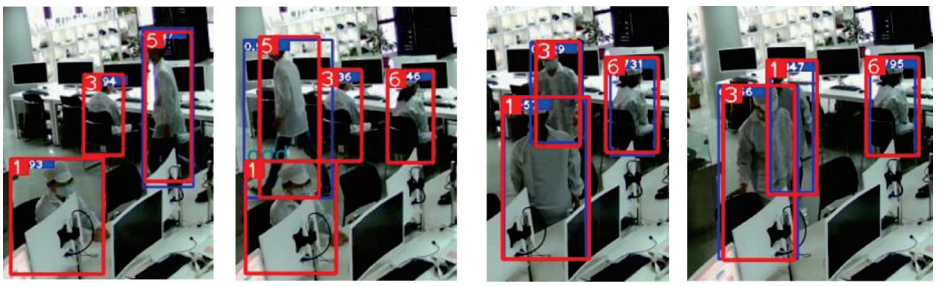
**Table 6 Model performance comparison of embedded platform**

| 压缩比例/% | 网络大小/MB | 运行速度/(帧·s$^{-1}$) | |
|---|---|---|---|
| | | Ubuntu | Xavier |
| 0 | 246.3 | 6 | 4 |
| 50 | 107.2 | 11 | 8 |
| 95 | 4.6 | 20 | 13 |

**表7 检测结果校正前后算法性能对比**

**Table 7 Comparison of algorithm performance before and after test results correction**

| 序号 | 检测校正前 | | | 检测校正后 | | |
|---|---|---|---|---|---|---|
| | FP | FN | MOTA | FP | FN | MOTA |
| 1 | 12 | 17 | 94.1 | 0 | 2 | 99.6 |
| 2 | 114 | 85 | 77.6 | 2 | 9 | 98.7 |
| 3 | 102 | 107 | 90.0 | 12 | 12 | 98.8 |



(a) The 736th and 750th frame images in video (b) The 1 216th and 1 250th frame images in video

图5 测试视频中的部分图像帧

Fig.5 Some image frames in the test video

## 4 结束语

本文提出一种新的轻量级多目标跟踪算法,该算法通过对YOLOv3网络模型进行剪枝压缩,在保证其检测精度的同时降低了模型占用空间,提升推理速度。再利用ShuffleNet网络提取目标特征,结合了目标框与预测结果间的IOU以及特征相似度,对结果进行数据关联。最后将多目标检测跟踪系统部署到嵌入式设备上。从实验结果可以发现,嵌入式设备上的模型推理速度相比于在GTX1060上较慢,

和网络剪枝之前的模型相比,速度虽提升3倍,但还未达到实时性的要求。其中,网络运行时间大多是消耗在模块检测上,因此还需对检测网络进行进一步优化以达到实时性要求。

**参考文献:**


[1] 任陶瑞. 室内监控中多目标检测及跟踪设计与实现[D]. 南京:南京邮电大学, 2016.
REN Taorui. Design and realization of multi-target detection and tracking in indoor surveillance[D]. Nanjing:Nanjing University of Posts and Telecommunications, 2016.

[2] 李云鹏, 侯凌燕, 王超. 基于YOLOv3的自动驾驶中运动目标检测[J]. 计算机工程与设计, 2019(4): 38.
LI Yunpeng, HOU Lingyan, WANG Chao. Moving target detection in autonomous driving based on YOLOv3[J]. Computer Engineering and Design, 2019 (4): 38.

[3] 王晓青, 王向军. 应用于嵌入式图形处理器的实时目标检测方法[J]. 光学学报, 2019, 39(3): 274-280.
WANG Xiaoqing, WANG Xiangjun. Real-time target detection method applied to embedded graphics processor[J]. Acta Optics, 2019, 39(3): 274-280.

[4] 华夏, 王新晴, 王东, 等. 基于改进SSD的交通大场景多目标检测[J]. 光学学报, 2018,38(12): 221-231.
HUA Xia, WANG Xinqing, WANG Dong, et al. Multi-target detection in large traffic scenes based on improved SSD[J]. Acta Optics, 2018,38(12): 221-231.

[5] 王文秀, 傅雨田, 董峰, 等. 基于深度卷积神经网络的红外船只目标检测方法[J]. 光学学报, 2018, 38(7): 160-166.
WANG Wenxiu, FU Yutian, DONG Feng, et al. Infrared ship target detection method based on deep convolutional neural network[J]. Acta Optics, 2018, 38(7): 160-166.

[6] WU X, SAHOO D, HOI S C H. Recent advances in deep learning for object detection[J]. Neurocomputing, 2020, 396: 39-64.

[7] LIU L, OUYANG W, WANG X, et al. Deep learning for generic object detection: A survey[J]. International Journal of Computer Vision, 2020, 28(2): 261-318.

[8] AMIT Y, FELZENSZWALB P, GIRSHICK R. Object detection[J]. Computer Vision: A Reference Guide, 2020: 1-9.

[9] 韦皓瀚, 曹国, 尚岩峰,等. 一种改进聚合通道特征的行人检测方法[J]. 数据采集与处理, 2018, 33 (3): 521-529.
WEI Haohan, CAO Guo, SHANG Yanfeng, et al. Improved pedestrian detection method of modified aggregate channel feature[J]. Journal of Data Acquisition and Processing, 2018, 33(3): 521-529.

[10] 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. [S. l.]: IEEE, 2014: 580-587.

[11] LIU W, ANGUELOV D, ERHAN D, et al. SSD:Single shot multibox detector[C]//Proceedings of European Conference on Computer Vision. Amsterdam, The Netherlands: Springer, 2016: 21-37.

[12] REDMON J, FARHADI A. YOLOv3:An incremental improvement[EB/OL]. (2018-04-08) [2020-03-15]. https://arxiv.org/abs/1804.02767.

[13] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. YOLOv4:Optimal speed and accuracy of object detection[EB/OL]. (2020-04-23)[2020-07-01]. https://arxiv.org/abs/2004.10934.

[14] MEI X, LING H B. Robust visual tracking using L1 minimization[C]//Proceedings of the 12th IEEE International Conference on Computer Vision. Kyoto: IEEE, 2009: 1436-1443.

[15] 管皓, 薛向阳, 安志勇. 在线单目标视频跟踪算法综述[J]. 小型微型计算机系统, 2017, 38(1): 147-153.
GUAN Hao, XUE Xiangyang, AN Zhiyong. Overview of online single-target video tracking algorithms[J]. Small Microcomputer System, 2017, 38(1): 147-153.

[16] 王飞. 基于视频图像处理的公交客流统计技术研究[D]. 南京: 东南大学, 2018.
WANG Fei. Research on bus passenger flow statistics technology based on video image processing[D]. Nanjing: Southeast University, 2018.

[17] SANCHEZ-MATILLA R, POIESI F, CAVALLARO A. Online multitarget tracking with strong and weak detections[C]//Proceedings of European Conference on Computer Vision. [S.l.]: Springer, 2016: 84-99.

[18] ZHANG J, JIN X, SUN J, et al. Spatial and semantic convolutional features for robust visual object tracking[J]. Multimedia Tools and Applications, 2020, 79(21): 15095-15115.

[19] CIAPARRONE G, SÁNCHEZ F L, TABIK S, et al. Deep learning in video multi-object tracking: A survey[J]. Neurocomputing, 2020, 381: 61-88.

[20] WOJKE N, BEWLEY A, PAULUS D. Simple online and realtime tracking with a deep association metric[C]//Proceedings of IEEE International Conference on Image Processing (ICIP). [S.l.]: IEEE, 2017: 3645-3649.

[21] YU F, LI W, LI Q, et al. POI: Multiple object tracking with high performance detection and appearance feature[C]//Proceedings of European Conference on Computer Vision. Cham, Switzerland:Springer, 2016: 36-42.

[22] WANG Z, ZHENG L, LIU Y, et al. Towards real-time multi-object tracking[EB/OL]. (2019-09-27) [2020-03-15]. https://arxiv.org/abs/1909.12605.

[23] 冯小雨, 梅卫, 胡大帅. 基于改进Faster R-CNN的空中目标检测[J]. 光学学报, 2018, 38(6): 250-258.
FENG Xiaoyu, MEI Wei, HU Dashuai. Aerial target detection based on improved Faster R-CNN[J]. Acta Optics, 2018, 38(6): 250-258.

[24] 辛鹏, 许悦雷, 唐红, 等. 全卷积网络多层特征融合的飞机快速检测[J]. 光学学报, 2018,38(3): 337-343.
XIN Peng, XU Yuelei, TANG Hong, et al. Fast aircraft detection based on multi-layer feature fusion of full convolutional network[J]. Acta Optica Sinica, 2018,38(3): 337-343.

[25] 张建明, 王伟, 陆朝铨, 等. 基于压缩卷积神经网络的交通标志分类算法[J]. 华中科技大学学报(自然科学版), 2019, 47(1): 108-113.
ZHANG Jianming, WANG Wei, LU Chaoquan, et al. Traffic sign classification algorithm based on compressed convolutional neural network[J]. Journal of Huazhong University of Science and Technology (Natural Science Edition), 2019, 47(1): 108-113.

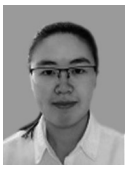
[26] ZHANG X, ZHOU X, LIN M, et al. ShuffleNet:An extremely efficient convolutional neural network for mobile devices[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). [S.l.]: IEEE, 2018: 6848-6856.

[27] ZHENG L, SHEN L Y, TIAN L, et al. Scalable person re-identification: A benchmark[C]//Proceedings of 2015 IEEE International Conference on Computer Vision (ICCV). Santiago: IEEE, 2015: 1116-1124.

[28] ENRIQUE A, GONZALO J, ARTILES J, et al. A comparison of extrinsic clustering evaluation metrics based on formal constraints[J]. Information Retrieval, 2009, 12(4): 461-486.

[29] BEWLEY A, GE Z, OTT L, et al. Simple online and realtime tracking[C]//IEEE International Conference on Image Processing (ICIP). [S.l.]:IEEE, 2016: 3464-3468.

**作者简介:**



**宗佳平**(1995-),女,硕士研究生,研究方向:目标检测、目标跟踪等,E-mail:1138880236@qq.com。

**吴妍**(1978-),女,副主任法医师,研究方向:法医物证检验,E-mail: 304714080@qq.com。

**陈建强**(1974-),男,硕士,副教授,研究方向:机械设计、故障诊断,E-mail:200875600@qq.com。

**张琳娜**(1977-),女,讲师,硕士生导师,研究方向:工业产品缺陷检测、机器视觉等,E-mail:zln77080 @163.com。

**张悦**(1990-),女,博士研究生,研究方向:深度学习、行人重识别、模式识别等,E-mail: 17112065@bjtu.edu.cn。

**岑翼刚**(1978-),**通信作者**,男,教授,博士生导师,研究方向:低秩矩阵重构、小波分析、异常检测等,E-mail:ygcen@bjtu.edu.cn。

(编辑:张黄群)