学校代码：11517

学　　号：201810916202

河南工程学院

HENAN INSTITUTE OF ENGINEERING

毕业设计（论文）

题　　目　基于计算机视觉的公共场所防暴检测系统

学生姓名　梁 焯 炫

专业班级　物联网工程 1842

学　　号　201810916202

院（部）　计 算 机 学 院

指导教师(职称)　王　禹（副教授）

完成时间　2022 年 4 月 29 日

河南工程学院

毕业设计（论文）版权使用授权书

本人完全了解河南工程学院关于收集、保存、使用学位毕业设计（论文）的规定，同意如下各项内容：按照学校要求提交毕业设计（论文）的印刷本和电子版本；学校有权保存毕业设计（论文）的印刷本和电子版，并采用影印、缩印、扫描、数字化或其它手段保存毕业设计（论文）；学校有权提供目录检索以及提供本毕业设计（论文）全文或者部分的阅览服务；学校有权按有关规定向国家有关部门或者机构送交毕业设计（论文）的复印件和电子版；在不以赢利为目的的前提下，学校可以适当复制毕业设计（论文）的部分或全部内容用于学术活动。

毕业设计（论文）作者签名：

年　月　日

河南工程学院

毕业设计(论文)原创性声明

本人郑重声明：所呈交的毕业设计（论文），是本人在指导教师指导下，进行研究工作所取得的成果。除文中已经注明引用的内容外，本毕业设计（论文）的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本毕业设计（论文）所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位毕业设计（论文）原创性声明的法律责任由本人承担。

毕业设计（论文）作者签名：

年　　月　　日

目　录

摘 要

社会治安是一个重大的社会问题和政治问题，提高社会治安力度，有效地预防和打击违法犯罪，保护广大人民的生命财产安全，创造良好的社会环境，是全中国人民的强烈愿望。本着协助预防公共场所暴力犯罪工作的想法，本文主要实现基于计算机视觉的公共场所暴力犯罪检测应用。

防暴检测系统中，行为的识别依赖于 Pytorch 框架，应用 YOLOV4-Tiny 算法，判定行人是否持有刀具、棍棒等武器，利用 Mediapipe 中的 Blazepose 算法对行人进行人体姿态估计并用 KNN Regression 算法展开预测，综合各算法的预测结果，评估图像中行人的犯罪倾向指数。为了模拟实地应用，上述系统部署于树莓派中，能够实现快速、广泛的应用。

关键词：防暴检测；Pytorch；YOLOV4-Tiny；Blazepose；KNN Regression；树莓派

ABSTRACT

Public security is a vital social and political problem. Enhancing social security force, preventing and combating ctimes effectively, protecting lives and property of masses of people, creating a good social environment is a strong desire of chinese people. In view of giving assistance to prevent violence crime in public place, This paper mainly implements the application of violence crime recognition in public place which bases on computer vision.

In this paper, a violence crime recognition system based on Pytorh is designed. Applying YOLOV4-Tiny algorithm to detect whether person or arms such as knife and stick in a image.Use Blazepose algorithm to estimate human pose and use KNN Regression algorithm to predict violence crime trend index for each person in a image.And deploy the system into Raspberrry Pi to run the program in hardware.

Key words: violence crime recognition, Pytorch, YOLOV4-Tiny, Blazepose, KNN Regression, Raspberry Pi

第 1 章 绪论

1.1 背景及意义

社会治安地一个重大的社会问题和政治问题。加强社会治安综合治理，有效地预防和打击违法犯罪，保护广大人民的生命财产安全，为先进生产力和先进文化的发展创造良好的社会环境，为实现和发展广大人民的根本利益提供有力保证，是落实“三个代表”重要思想的必然要求[1]。我国曾经发生过危害性极大的公共场所暴力犯罪事件，如 3·1 昆明火车站暴力恐怖案、4·30 乌鲁木齐火车站恐怖袭击案。

及时有效地发现有暴力犯罪倾向的人员，对其行为进行干预是解决这个问题的关键。本毕业设计以此为出发点，编写可以部署在微型计算机上的系统，只要是有摄像头的地方，将摄像头与微型计算机对接便可运行此系统。此系统使用计算机技术对图像进行处理，预测公共场景中可能会进行暴力犯罪的人员并进行反馈，缩短安保人员发现犯罪行为的时间，更及时有效地干预暴力犯罪行为，以此来加强社会治安。

1.2 国内外研究发展现状

1.2.1 国外研究和现状

Bermejo 和 Deniz（2011）提出计算视频对应的 STIP 和 MoSIFT，以及用词袋法 BOW 对视频序列进行特征创建，以此作为特征训练分类模型 SVM[2]，来判断视频中是否有暴力行为的出现。Martin（2012）等人使用多尺度的局部二相模式直方图进行暴力检测[3]。在神经网络算法流行起来之前，研究人员主要使用传统的机器学习算法来预测图像中的行人是否存在的暴力行为。

随着 AI 技术的发展，深度学习在计算机视觉、自然语言处理等领域大放光彩。Sarthak 使用深度神经网路 CNN 和 LSTM 对视频中人的行为进行暴力行为的识别[4]。Amarjot（2018）使用 FPN 从无人机拍摄的图像中检测人群，针对图像中的有人区域，利用 ScatterNet 混合深度学习网络（SHDL）进行人体姿态估计，再根据估计的四肢之间的方向确认施暴个体[5]。现今，深度学习 Deep Learning 在计算机视觉领域有着举足轻重的地位，深度学习模型海量的参数以及无比强大的拟合能力是其最大的优势所在。

1.2.2 国内研究和现状

相较于国外的研究发展，国内在暴力行为识别领域的研究起步较晚且发展较慢。周智和朱明等人提出了 3D-CNN 模型，使用三维的卷积神经网络，无需手动创建特征，就能够很好的提取视频中图像的的时空特征信息, 从而进行暴力行为的检测[6]。王晓龙提出利用 ORB 算法提取运动目标的兴趣特征点,并采用 FLANN 算法对每相邻帧进行特征点的匹配，获得大量的表示目标运动趋势的兴趣点轨迹，作为区分暴力行为与非暴力行为的模式特征。然后采用了短轨迹优化和多段最小二乘法等措施，针对暴力行为轨迹进行了优化，以加快训练速度，使得算法的识别率进一步提升。在得到兴趣点运动轨迹后，利用 BRIEF 和马尔科夫链模型，分别从外形层次和几何层次两方面提取出这些轨迹的特征向量，并采用词袋模型进行数学建模，再把提取到的特征投入到支持向量机 SVM 中进行训练和分类[7]。胡琼和秦磊通过提取视频图像的静态特征、动态特征、时空特征和描述性特征来构造特征向量，通过基于相似度的方法匹配模板，以此来定性图像中行人的行为[8]。

1.3 研究内容

1. 使用 mAP 指标对比随机初始化参数的模型，和使用参数初始化文件初始化的模型的预测结果。探索图片数据的采集途径、剔除无用数据的方法以及模型参数的设定。

2. 在拍摄暴力犯罪倾向的行人数据集时，为了抽象人体姿态并进行人体姿态估计，考虑拍摄角度、要做什么样的动作等。探索人体关键点的特征构建方法，使用三维的像素坐标可以构建哪些有用的特征来抽象出人体姿态的隐含信息。Blazepose 模型可以提取出人体的 33 个关键点信息，是否需要将 33 个关键点全部用于特征构建。

3. 尝试往树莓派上烧录不同的 Linux 操作系统，并寻找一个最适合于计算机视觉程序部署的操作系统。探索在树莓派上配置基于 Python 的 Pytorch 开发环境，使用哪个版本的 Python、Pytorch 等。

4. 尝试在树莓派上安装外设，以及在哪些版本的操作系统中可以安装对应外设的驱动。

第 2 章 相关技术与理论基础

2.1 Python 语言概述

Python 由荷兰数学和计算机科学研究学会（CWI）的 Guido van Rossum 于 1990 年代初设计，作为一门叫做 ABC 的语言的替代品[11]。Python 是一个高级的面向对象的程序设计语言，使用 Python 编写的程序具有很好的可读性，其简化了很多语法结构，对初学者十分友好。如果你精通别的编程语言，那么你上手 Python 将毫无难度。

Python 的特点：

（1） Python 在进行程序开发时不需要像 C 语言那样进行编译，是一种解释型的语言，十分适合用于编写自动化脚本。

（2）Python 具有很强的交互性，使用者可以在键入一小段代码后直接执行得到运行结果。

（3）Python 是面向对象的，使用者可以使用继承、封装、多态等面向对象的编程方法进行面向对象的编程。

（4）Python 是很多编程学者都觊觎的语言，对于编程初学者来说，得 Python 如鱼得水，初学者使用 Python 编写一个小游戏或是一个网络服务程序是毫无压力的。

2.2 Pytorch 框架简介

Pytorch 是主流的开源机器学习库，Pytorch 前身是 Torch，在 Torch 的基础上重写了很多内容，使得其更加灵活，Pytorch 提供 C++、Python 等主流编程语言的接口，不同于使用静态图的 TensorFlow 框架，Pytorch 框架使用动态图使得其更加的灵活。结合强大的 GPU 加速，Pytorch 正逐步成为最流行的机器学习框架。

Pytorch 拥有像 numpy 一样的张量计算功能，但比 numpy 强大的是，其张量运算支持使用 GPU 加速，同时其支持自动求导功能。现今，Pytorch 已经得到了广泛的使用[9] 。

Pytoch 框架主要有三个特点[10]：

（1）支持分布式训练

Pytoch 通过 torch.distributed 后端，支持使用者在研究和生产中进行可伸缩（Scalable）规模的分布式训练和优化模型性能。

（2）拥有高鲁棒性的生态系统

Pytorch 拥有丰富的扩展，得以支持开发者在计算机视觉、自然语言处理等领域的开发工作。

（3）提供云支持

Pytorch 对于主流的云平台提供良好的支持，对开发者十分友好。

2.3 卷积神经网络

卷积神经网络（convolutional neural network，CNN）是一类十分使用于处理图像数据的神经网络。在计算机视觉领域中，以卷积神经网络为基础设计的深度学习模型已经占据了半壁江山，现今大多数计算机视觉相关的工业界应用和学术研究都离不开卷积神经网络。

卷积神经网络一般包括三个部分：全连接层、卷积层和池化层。

卷积层（Convolutions layer），使用卷积核对上层输出的矩阵进行卷积运算，同时用激活函数处理得到最终输出。

池化层（Pooling layer），可以对输出张量进行降维，减少出现过拟合的可能性。典型的池化操作有最大化池化与平均池化。

全连接层（Full connected layer），对卷积层和池化层进行合并，生成更多的全连接层，以提升神经网络的预测能力。

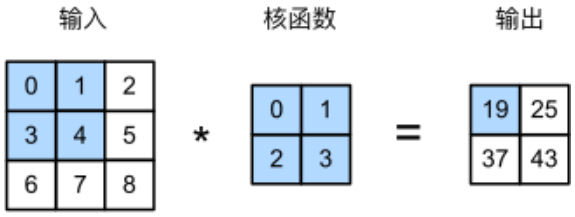

图 1 卷积运算示意图

在卷积层中会进行卷积运算，卷积运算将输入张量和卷积核相互运算，产生输出张量[12]。此处以二维的卷积运算为例进行介绍，如图 1 所示，输入张量是高度和宽度均为 3 的二维张量，卷积核是高度和宽度均为 2 的二维张量，卷积窗口和卷积核的尺寸相同为 2x2。在二维的卷积方法运算中,卷积运算窗口从输入张量的左上角开始，由左向右、自上而下地滑动。卷积窗口每滑动经过一个新的位置，

对应窗口中的张量就会和卷积核按元素进行相乘，并且对这些乘积值进行相加得到此位置输出的张量值。如图 1 中蓝色部分所示，其进行卷积运算为 $0*0+1*1+2*3+4*3=19$ 即对应于输出张量对应位置的元素值。

2.4 YOLO 系列模型与 YOLOV4-Tiny

（1）残差网络 ResNets

随着神经网络层数的增加，训练会变得越来越困难，网络的优化也会变得越来越困难，残差网络的诞生使得此问题得以解决。残差网络在原来神经网络的基础上加入捷径连接（shortcut connection），使得网络不容易过拟合。其中包含一个捷径连接的几个网络层称残差块，如图 2 所示。

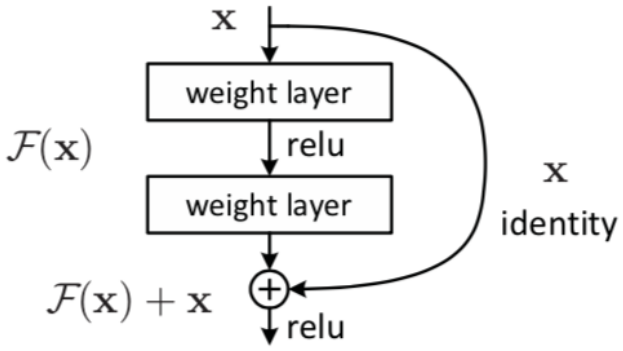

图 2 残差块示意图

如图 2 所示，x表示输入的张量，$F(x)$表示残差块第二层的输出，当没有捷径连接时，残差块就是一个普通的 2 层网络。设第二层网络在激活函数之前的输出为 $H(x)$。如果在该 2 层网络中，最优的输出就是输入 x，那么对于没有捷径连接的网络，就需要将其优化成 $H(x)=x$；对于有捷径连接的网络，如果最优输出是x，则只需要将$F(x)=H(x)-x$优化为 0 即可，显然对后者进行优化比对前者进行优化简单。

（2）YOLOV3

YOLOV3 使用 Darknet53 进行特征提取，Darknet53 对比于其之前提出的网络，最大的优势就是使用了残差网络，这使得网络便于优化且不易过拟合。Darknet53 的每一个卷积部分使用了特有的 DarknetConv2D 结构，在卷积时会进行 L2 正则化，完成卷积后进行标准化并用激活函数 Leaky ReLU 进行处理。普

通的 ReLU 是将负值都更改为零，Leaky ReLU 则是对负值进行一个降权，对其除以一个不为零的数值。Leaky ReLU 数学表示如式(2-1)。

$$y_i = \begin{cases} x_i, x_i \geq 0 \\ x_i / a_i, x_i < 0 \end{cases} \quad \text{式(2-1)}$$

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

图 3 Darknet53 结构图

YOLOV3 进行预测的过程可分成两部分，一是使用 FPN 加强特征的提取，二是用 Yolo Head 对三个特征层进行预测。FPN 可以融合尺寸相异的特征层，提取对预测更有利的特征。

（3）YOLOV4

YOLOV4 可以看成是在 YOLOV3 的基础上结合一系列改进的版本。YOLOV4 的主干特征提取网络使用 CSPDarkNet53 代替原来的 DarkNet53。一是将 DarkNet53 中二维卷积层的激活函数由 Leaky ReLU 改成 Mish；二是使用了 CSPNet 结构[13]，CSPNet 将残差块的堆叠拆成了两部分，如图 4 所示，Part1 部分按原来的位置进行堆叠，Part2 部分使用 ResNet 方法处理后再进行堆叠。

$$Mish = x \times \tanh\left(\ln(1 + e^x)\right) \quad \text{式(2-2)}$$

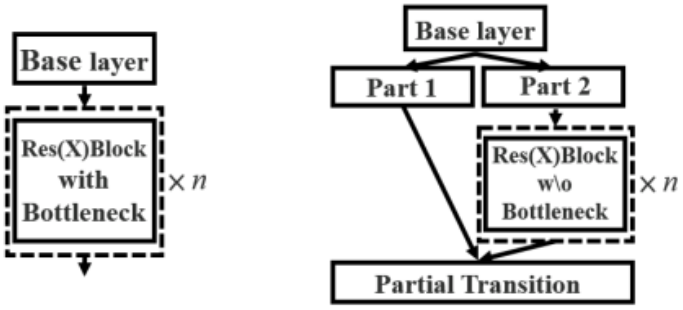

图 4 CSPNet 结构图

YOLOV4 在特征提取层使用了 SPP 结构和 PANet 结构[14]。SPP 结构对 CSPDarkNet53 的输出进行四种不同尺寸的最大池化。PANet 的结构如图 5，PANet 会对图像的特征进行反复地提取在图 5（a）中的特征金字塔由下往上进行特征提取后，还要实现图 5（b）从上往下的特征提取。

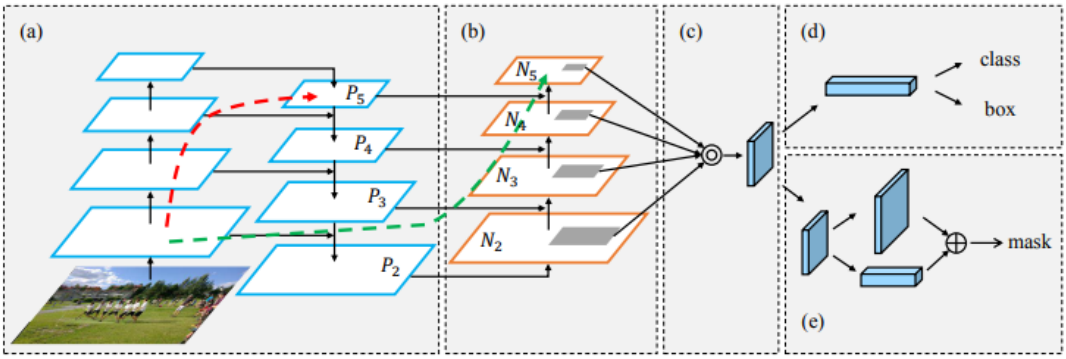

图 5 PANet 结构图

（4）YOLOV4-Tiny

YOLOV4-Tiny 是 YOLOV4 的精简版，参数量只有 YOLOV4 的十分之一，是轻量化的模型，适用于移动端。YOLOV4-Tiny 同样使用 CSPNet 的结构，只使用两个特征层进行分类与回归预测，使用 Leaky ReLU 作为激活函数，大大加快了模型的预测速度。

2.5 Mediapipe 库及 BlazePose 算法

Mediapipe 库是由 Google 公司开源的，提供可支持交叉平台、高度定制化机器学习解决方案的，应用于实时、流媒体检测的算法库。其具有提供端到端的预测及时性、一次构建模型可支持多平台部署、免费且开源等特点。Mediapipe 库支持多种机器学习解决方案如人脸检测、手语加测、人体姿态估计、实时运动追踪等[15]，BlazePose 算法就是 Mediapipe 库中提供的一种进行人体姿态估计的算法。

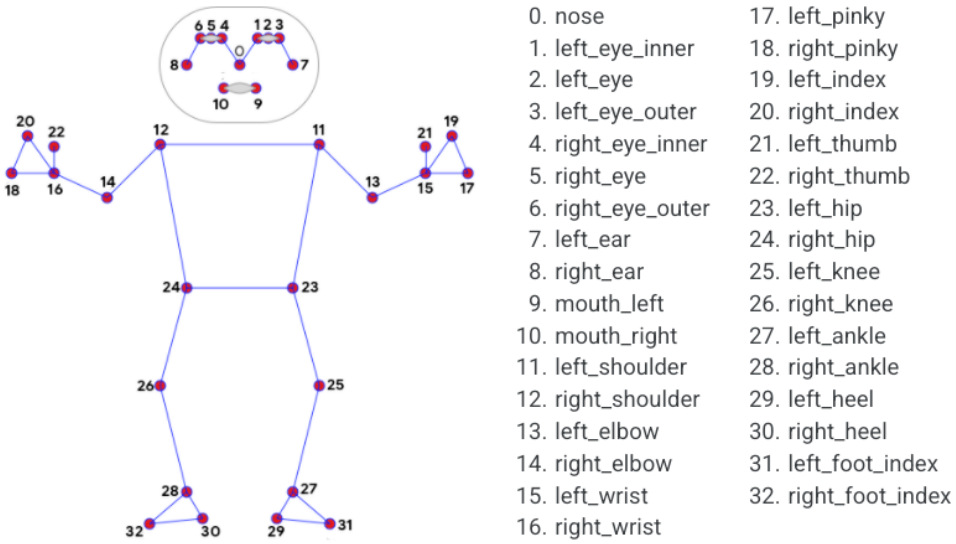

图 6 BlazePose 算法可检测出的人体 33 个关键点

对图像或视频的人体姿态估计技术，在各式各样的应用如人体健康追踪、肢体语言识别、手语识别中起着至关重要的作用。BlazePose[16]是一个用于人体姿态估计，为移动端设备量身定做的轻量化卷积神经网络。此算法可以预测出人体 33 个关键点的像素坐标，如图 6 所示。BlazePose 算法使用了两种技术：heatmaps 和 regression。heatmaps 技术会预测出人体每个关键点的概率，及每个关键点需要精调的偏移量。regression 技术是直接使用神经网络回归预测出人体关键点的坐标，计算量相对较小。结合两种技术的 BlazePose 算法更适用于移动端的实时预测，且预测精度要优于仅使用 heatmaps 技术的算法。

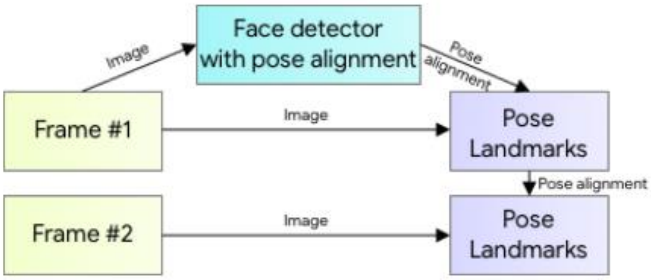

图 7 BlazePose 算法连续帧预测结构图

BlazePoze 算法使用了编码器-解码器（encoder-decoder）网络结构，在训练时，同时训练 heatmaps 和 regression 网络，在预测时，只使用 regression 网络。使用 BlazePose 算法进行连续帧的预测时，会首先使用 Face detector 进行人脸的检测，以此找出人体所在的图像候选区域，如图 7 所示。然后使用 Pose Landmarks

模型对人体候选区域进行检测出人体的 33 个关键点坐标，对于下一帧所使用的候选框是根据上一帧所预测出的关键点坐标计算生成的，提升了模型的预测速度。

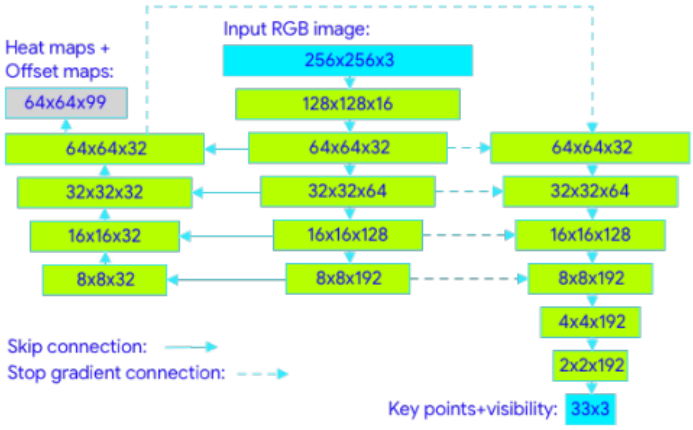

图 8 BlazePose 网络结构示意图

在传统的目标检测解决方案中，在预测的后处理阶段都会使用非极大值抑制算法（Non-Maximum Suppression, NMS），来选择对于同一个物体的 IOU 超过某个阈值且置信度最大的先验框，但是这种算法只适用于刚性的物体（rigid-objects），即不易发生形变、自由度很低的物体。对于人体这种高自由度，易发生形变的物体，BlazePose 的解决方法是，先检测出人体相对刚性、与人体的其他部位具有高对比度的部位：人脸，并用对齐算法找到人体候选区域。因为输入到模型的人体图像同样需要进行对齐，BlazePose 先用内置的 Face Detector 检测出人脸、人体肩关节的中点、髋关节的中点、人体的外接圆，以人体肩宽节和髋关节的中点构造直线，以此直线为基准来对齐人体，如图 9 所示。

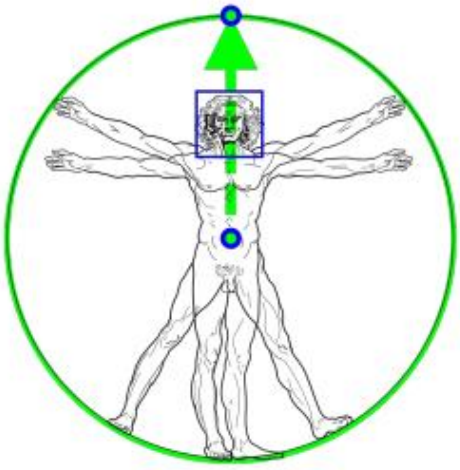

图 9 BlazePose 的维特鲁威人对齐算法示意图

2.6 K 最邻近回归算法

本程序的最终任务是要预测出输入图像中的每个人的暴力犯罪指数，暴力犯罪指数是一个连续值，需要使用回归模型进行预测。K 最邻近回归算法是一种基于欧式距离进行特征度量的、轻量的回归算法。假设特征向量为 n 维，K 最近邻回归算法的原理，是找到在 n 维向量空间中，预测数据的 n 为特征向量与训练数据的所有特征向量欧式距离最近的 k 个样本，其中 k 为需要设定的模型参数，并对所找到的 k 个样本的标签值求平均数，得到最终的预测值。

第 3 章 数据来源

3.1 行人、危险物品数据集

含有行人、危险物品（刀具、棍棒）的图片数据主要来源于网络视频截图，在 Bilibili、优酷、爱奇艺等视频网站上键入关键词“持刀”、“持棍”、“刀具”等，播放对应搜索到的视频，并对含有行人、有人持刀或持棍等场景进行截图保存，最终收集到训练图片 268 张。含持刀、持棍的人对比于单独的行人和刀具的图片更符合目标检测模型 YOLOV4-Tiny 的预测场景，因此训练数据侧重于有人持刀、持棍的场景。

图 10 行人持刀图

图 11 行人持棍图

如图 10 和图 11，是在视频网站上根据关键字播放视频，对视频帧进行截图得到的，用于训练目标检测模型 YOLOV4-Tiny 的图片数据。

3.2 暴力犯罪倾向数据集

人体姿态是人行为倾向最直接的表达方式，不同的人体姿态会表示出人不同的意图。在同一个地点，不同的角度拍摄正常人以及有暴力犯罪倾向的人员可能做出的各种动作，如举臂、行走、站立等，总计 128 张图片如图 12 所示。

图 12 有暴力犯罪倾向的行人、正常行人姿态图

3.3 数据集划分

对于行人、刀具棍棒数据集，首先使用 labelimg 软件对数据集进行标注，生成标注文件，标签分为“人”和“武器”两类，对于每张标注的图片生成一个对应的 xml 文件。使用数据集的百分之九十作为训练集，剩下的百分之十作为测试集，以此来训练 YOLOV4-Tiny 模型。

图 13 标记软件 labelimg 展示图

图 13 是使用 labelimg 软件进行图片数据标记的图片，蓝色框框选的是人，紫色框框选的是危险物品。

```
<segmented>0</segmented>
<object>
    <name>person</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
        <xmin>281</xmin>
        <ymin>89</ymin>
        <xmax>526</xmax>
        <ymax>610</ymax>
    </bndbox>
</object>
```

图 14 行人、刀具棍棒标注文件内容

图 14 是使用 labelimg 标注图片生成对应的 xml 文件的内容图，在 object 标签里面，name 标签标记的是分类的名字，bndbox 里面 xmin、ymin、xmax、ymax 分别表示在此图片中候选框左下角和右上角的 x 轴、y 轴的像素坐标。

第 4 章 系统设计与实现

4.1 总体设计

基于计算机视觉的防暴检测系统，主要具备以下的功能模块：

（1）硬件模块：使用树莓派接收摄像头的图像输入，并运行防暴检测系统对进行图像处理。

（2）行人、刀具棍棒检测模块：可以框选出行人、刀具棍棒在图像中的位置，并预测出对应候选框的类别。

（3）人体姿态估计模块：预测出人体的 33 个关键点在其候选框中的位置。

（4）暴力犯罪倾向评估模块：预测出图像中的每一个行人的暴力犯罪指数，用于表示其可能会进行暴力犯罪的可能性。

（5）预测结果反馈模块：将预测出的行人、刀具棍棒的候选框、行人人体的关键点、每个人的暴力犯罪指数在图像中标记出来。

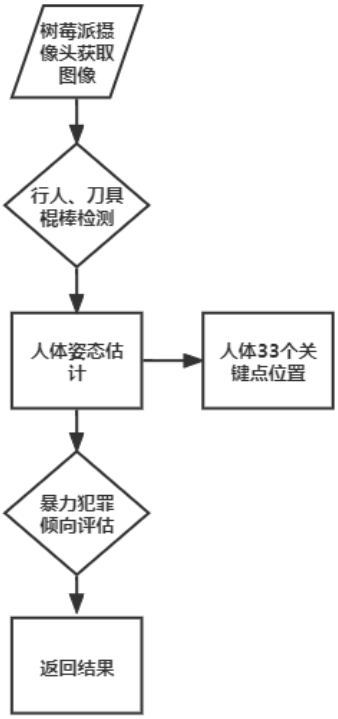

图 15 系统功能模块及流程图

4.2 详细设计

4.2.1 硬件模块

（1）树莓派初始化配置

本系统使用内存为4GB的树莓派4B，使用树莓派镜像烧录器将32位Legacy版的树莓派官方操作系统烧录到16GB的SD卡中，配置树莓派的wifi模块、SSH。连接树莓派电源，设置PC的网络为共享模式，用Advance IP Scanner搜索名称为raspberrypi的主机对应IP地址，根据这个IP地址用SSH连接上树莓派，通过sudo raspi-config命令打开系统设置，开启VNC。用HDMI线对接PC和树莓派，在PC上使用软件VNC Viewer登录桌面版的系统，接入官方售卖的摄像头Raspberry Pi Camera Module 2。

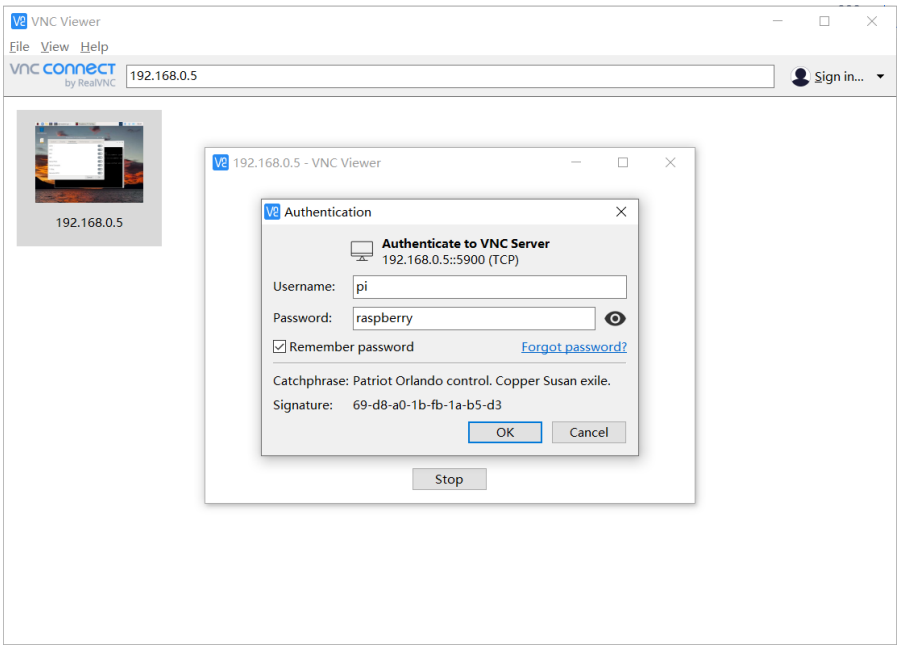

图16 使用VNC Viewer进入Raspberry Pi OS展示图

如图16所示，Raspberry Pi的IP地址是192.268.0.5，操作系统初始的用户名是pi，初始的密码是raspberry。根据IP地址、用户名和密码即可进入桌面版的Raspberry Pi操作系统。

（2）树莓派配置

设置系统默认的Python为Python3.7.3，安装依赖库Python3-opencv、Pytorch1.3、torchvision0.6、pillow、numpy、matplotlib、mediapipe、pickle、sklearn、tqdm。

如图17所示，使用HDMI线连接树莓派和PC的HDMI接口，往树莓派上接入网线，接通树莓派的5V3A type-c电源并打开电源开关，待树莓派的红色和黄色提示灯亮起便成功地启动了树莓派。

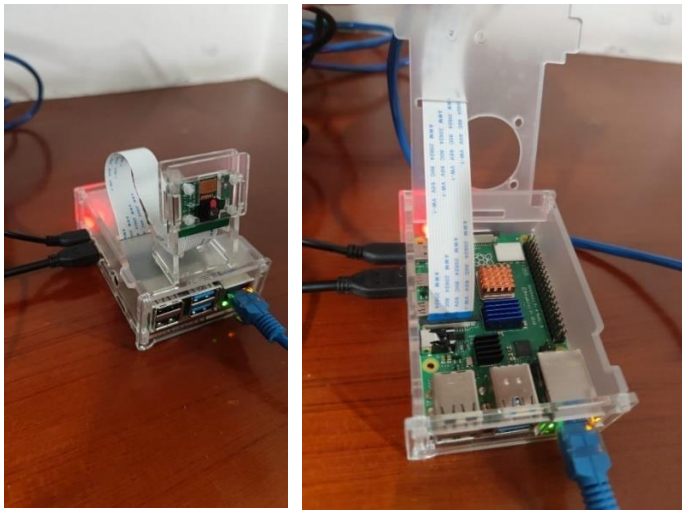

图 17 带摄像头的树莓派 4B 实物图

4.2.2 目标检测模块

（1）模型训练所使用的硬件

由于收集到的图片数据有限，只有几百张图片。考虑到训练数据集较小，以及购买 GPU 的成本过高，故使用 PC 进行目标检测模型 YOLOV4-Tiny 的训练。YOLOV4-Tiny 模型训练所使用的硬件为 Intel i5-8300H 处理器、8GB 内存的 PC 电脑。

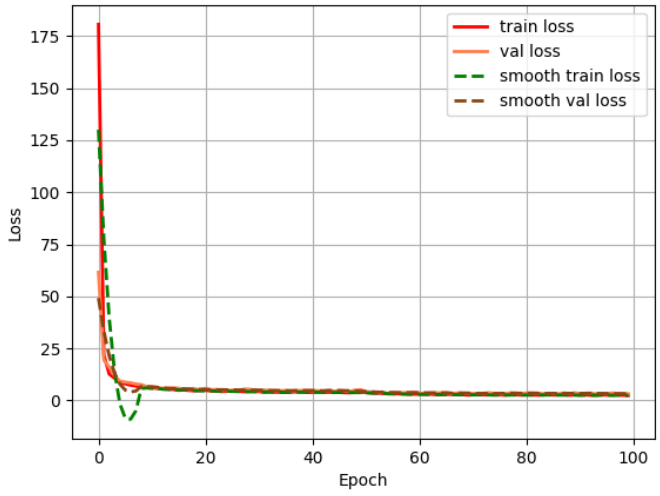

图 18 模型训练过程中 loss 的变化图

（2）模型训练

在 YOLOV4-Tiny 模型的训练中使用的是 VOC2007 数据集的形式，使用 Python 库 labelimg 的桌面应用程序标注需要用于训练的图片数据，生成对应的

标注文件。按 9:1 的比例划分训练集和测试集。在这里并没有选择随机初始化参数，如果这么做，模型训练效果会欠佳，因此使用开源项目提供的参数文件来初始化模型的参数。

图 18 为模型在 epoch 为 100 时，训练过程中 loss 变化值的折线图。可以发现，模型的训练损失 train loss 和验证损失 val loss 在 epoch 为 0 到 8 时有明显的下降，往后随着 epoch 的增大，模型的训练损失和验证损失的下降趋势趋于平缓。

（3）模型预测

模型训练完成之后会生成多个以.pth 为后缀的参数文件，选取 epoch 为 100 的参数文件，用于模型的预测。使用 OpenCV 捕获摄像头传入的图像或给定一张图片，以此作为输入，用模型对其进行预测，并在输入图像上绘制候选框进行反馈。

4.2.3 人体姿态估计模块

由于 BlazePose 算法只支持单人的人体姿态估计，故先使用 YOLOV4-Tiny 预测出输入图像中每个人所在的候选框，截取出每个人对应的候选框，再使用 BlazePose 算法对 YOLOV4-Tiny 框选出的行人进行人体关键点检测，并返回每个人对应的多个关键点坐标。

4.2.4 暴力犯罪倾向评估模块

（1）对关键点坐标的特征工程

人为评估行人数据集中每张图片人体姿态所表现出的暴力犯罪倾向，给每张图片赋予一个数值在 0 到 1 的浮点数作为标签，此数值越接近于 1，表示候选框中的人越有可能进行暴力犯罪。使用 BlazePose 算法预测每张图片中的人体 33 个关键点在图像中的像素坐标。因为不同的图片的长宽可能不同，所以用图片的长乘以关键点的纵坐标，用图片的宽乘以关键点的横坐标，来对关键点坐标进行数值归一化，使得用不同图片预测出的关键点坐标数值大小保持相对稳定，利于提升基于欧式距离进行相似度计算的 KNN 模型的预测效果。

在 33 个关键点中，选取位于头部的左眼内侧关键点、肩关节、肘关节、腕关节、髋关节和膝关节共 11 个关键点进行特征工程。为了减小进行坐标点特征工程计算的时间复杂度以及训练模型的特征向量的维度，只选取头部的一个关键点表示头部的位置、选取人体躯干的主要关键点来进行特征工程。计算出头部、

肩部、腕部、肘部、髋部之间的欧式距离$distance$，以及三个关键点组成的两个空间向量的夹角$angle$，对应于每张图片共构建一个 110 维的特征向量，$distance$和$angle$的计算公式如（4-1）所示。

$$distance = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad 式(4\text{-}1)$$

其中用x_i, y_i, z_i表示序号为i的人体关键点的三维坐标，其中x_i表示i号关键点距离其所在图片左边界的像素距离，y_i表示i号关键点距离其所在图片下边界的像素距离，z_i表示i号关键点距离摄像头的距离。

$$\overrightarrow{vec1} = (x_1 - x_2, y_1 - y_2, z_1 - z_2)$$
$$\overrightarrow{vec2} = (x_1 - x_3, y_1 - y_3, z_1 - z_3)$$
$$cos\theta = \frac{\overrightarrow{vec1} \cdot \overrightarrow{vec2}}{|\overrightarrow{vec1}| \times |\overrightarrow{vec2}|}$$
$$angle' = \arccos(cos\theta)$$
$$angle = angle' \times \frac{180}{\pi}$$

式(4-2) 向量夹角 angle 计算公式

其中θ表示两向量间的夹角，$vec1$和$vec2$表示三个关键点组成的两个向量，$angle'$是用弧度制表示的角度，$angle$是使用角度制表示的角度，这里的特征构建选择使用角度制表示角度，因为使用角度制表示的角度，数值更接近于别的特征的数值，利于提升模型的精度。

```python
def caculate_vector_angle(vector1, vector2):
    '''计算两个向量的夹角，为提升模型的预测效果，使用角度制'''
    vector1 = np.array(vector1)
    vector2 = np.array(vector2)

    mode_1 = np.sqrt(vector1.dot(vector1))
    mode_2 = np.sqrt(vector2.dot(vector2))
    dot_value = vector1.dot(vector2)

    cos_value = dot_value / (mode_1 * mode_2)
    angle_value = np.arccos(cos_value) # 默认弧度制

    # 角度修正
    # 使用角度制，这样数值更接近于 距离特征的值 利于 KNN 模型的预测
    angle_value = angle_value * 180 / np.pi

    return angle_value
```

图 19 角度制的向量夹角计算代码

（2）训练及评估 KNN Regression 模型

使用 sklearn 库的 KneighborsRegressor 进行模型训练，设置需要进行网格搜索的参数为 n_neighbors，网格搜索参数 n_neighbors 的数值范围为 1 到 7，使用网格搜索和 10 折交叉验证来训练和调优模型。

（3）使用训练好的 KNN Regression 模型进行实时预测

在程序实测中，有时候 Blazepose 算法无法检测出人体的某些关键点，在这种情况下，构建的特征向量会出现缺失值。为处理这种缺失值，对暴力犯罪倾向数据集构建的 128 个样本构建的特征向量进行求平均，用对应特征的平均值来填充缺失值。使用 KNN Regression 模型预测从每个人的人体姿态关键点构造特征得到的特征向量对应的暴力犯罪倾向指数。

4.2.5 预测结果反馈模块

使用 Python Pillow 库中的 write 模块进行矩形候选框的绘制，首先使用 YOLOV4-Tiny 模型预测出每个行人候选框的四个顶点坐标，然后用 write 模块的 rectangle 函数根据四个顶点坐标，设置颜色、框体的宽度等参数来绘制矩形候选框。用 Mediapipe 库内置的绘图工具 drawing_utils 来绘制每个人人体的关键点及关键点之间的连线。候选框左上角的浮点数表示候选框内，人体姿态表现出的暴力犯罪倾向指数，此数值越接近于 1 表示框中的人越有可能进行暴力犯罪。

```
for key, value in person_coo_mean.items():
    _distance = (value[0] - arms_coo_mean[0])**2 +\
                (value[1] - arms_coo_mean[1])**2
    if _distance > max_distance:
        max_distance = _distance
        violent_person_key = key

person_arms[violent_person_key] = True  # 标记持有武器的人
```

图 20 判断武器属于行人代码

若图像中被检测出有武器，则会计算每个武器候选框的四个顶点坐标的中点，到每个人候选框的四个顶点坐标的中点的距离，将距离最近者视为持有对应武器。使用绿框框选没有持武器的行人，使用蓝色框框选武器。若判断得候选框内的行人持有武器，且预测得该行人暴力犯罪倾向指数大于 0.5，则该行人的候选框会变为红色，候选框左上角的文本变为 warning，表示此人很有可能会进行暴力犯罪或正在进行暴力犯罪。

```
# 若判断为超过阈值的人，则框的颜色有绿色转换为红色
for ii in range(thickness):
    draw.rectangle([left + ii, top + ii, right - ii, bottom - ii], outline=color_object)
draw.rectangle([tuple(text_origin), tuple(text_origin + label_size)], fill=color_object)
# label 改为暴力犯罪指数和是否持有刀具棍棒的加权值
if warn and c==0:
    if person_violence_value[i]:
        if person_violence_value[i] + 0.5 > 1:
            label = "warning"
```

图 21 绘制候选框代码

4.3 系统测试

使用评估指标 mAP 对目标检测模型 YOLOV4-Tiny 的预测效果进行评估。mAP 用于表示不同的预测分类，当 IOU 从 0 到 1 取不同的用于将预测结果视为正样本的阈值时，以召回率（Recall）作为横坐标，精确率（Precision）作为纵坐标描绘的曲线所围成的面积的平均值。在最初进行模型训练时， YOLOV4-Tiny 模型对行人的识别准确率是令人满意的，但对武器的识别效果欠佳，武器的识别准确率仅为 0.33，猜测是由于模型需要预测的武器中刀具、棍棒的种类繁多，模型的训练数据不够所导致的。在筛除训练集中武器模糊的图片、调整模型的 phi、MINOVERLAP 等参数以及加大了训练的 epoch 之后，YOLOV4-Tiny 对武器的预测准确率提升至 0.51。

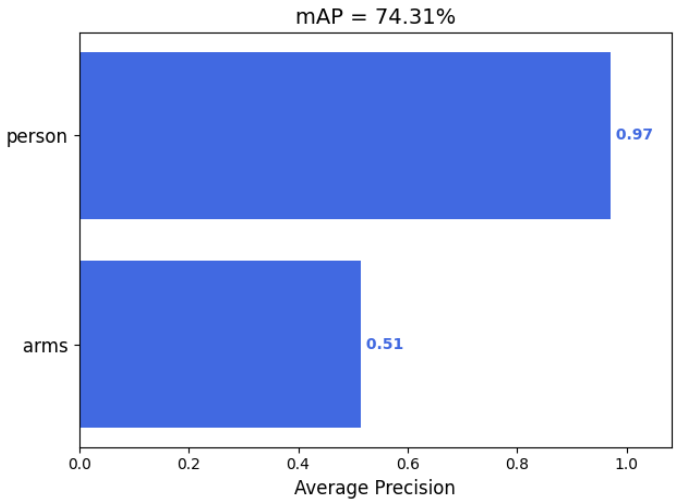

图 22 mAP 柱状图

图 22 为 person 分类和 arms 分类对应的 mAP 柱状图，YOLOV4-Tiny 模型对行人的识别准确率为百分之 97，对武器的识别准确率为百分之 51。mAP 为百分之 74。

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - y\_pred_i| \quad \text{式(4-3)}$$

使用 MAE（Mean Absolute Error）指标来评估模型的预测准确度，MAE 表示模型的所有预测值与真实值之差的平均数，其计算公式如式（4-3），是常用的回归模型评估指标。可视化不同的 n_neighbors 参数对应的 MAE 如图 23，可以发现 n_neighbots 为 5 时，模型的效果最佳。

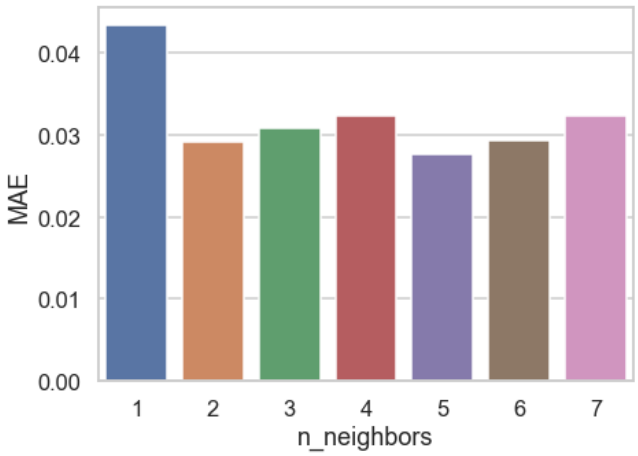

图 23 参数 n_neighbors 不同的数值计算得模型的 MAE

为了测试本系统在公共场所的预测效果，选择在本校的操场进行系统测试时这是一个空旷的、非单人的公共场景，十分适合用于进行本系统的测试。需要进行测试的场景有两个，其一是安排两个人自由走动，模拟正常、没有危害发生的场景；其二是安排两个人，其中一人自由走动，另外一人手持武器进行挥舞，模拟有犯罪分子进行暴力犯罪的场景。

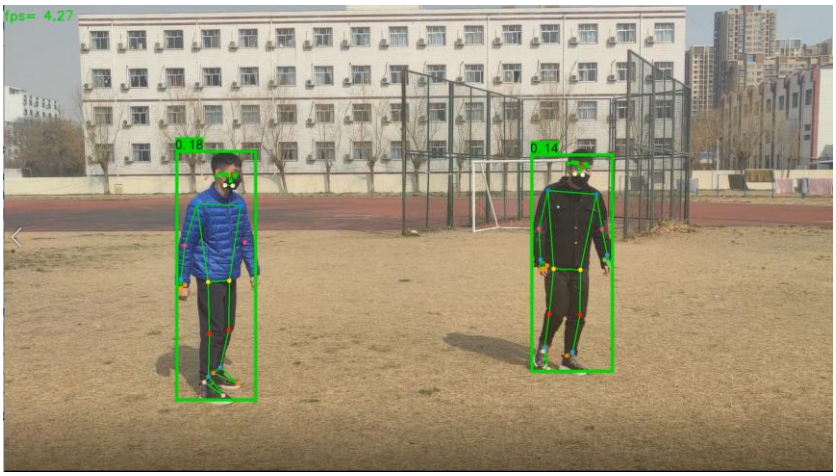

图 24 系统检测效果展示图

图 24 是本系统对模拟正常情况下，公共场景的预测效果图，绿色框框选出的是不会进行暴力犯罪的行人，每个候选框左上角的数值表示框内行人会进行暴

力犯罪的概率，此处两人均展示出正常的行走姿态且没有手持危险器械，故此概率是较低的。框内对人体的躯干进行了可视化，使得用户对每个行人的人体姿态有更直观的感受。

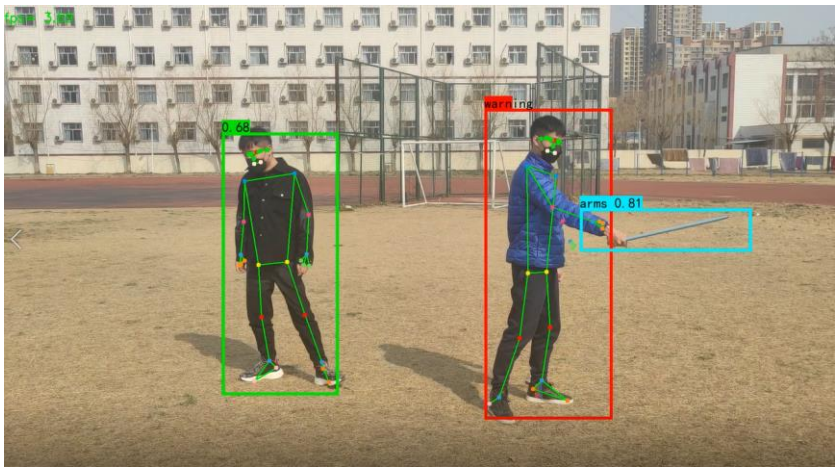

图 25 系统检测效果展示图

图 25 是本系统对模拟有暴力犯罪发生的情况下，公共场景的预测效果图，靠图像右方的行人持有短棒，且有明显的抬臂动作，其候选框被标记为红色，且对应候选框的左上角标记有 warning 的文本，警告用户此人有暴力犯罪倾向。

4.4 心得体会

最初进行 YOLOV4-Tiny 的训练时，没有使用参数初始化文件，测试时发现预测效果欠佳，仔细对比多个优秀的目标检测项目里面模型的训练流程之后，发现需要先下载别人提供的初始化参数文件，并用此文件初始化模型的参数，这样模型的预测效果才会有提升。

对于有暴力犯罪倾向行人数据集的构建，最开始拍摄时只拍摄了和右手动作有关的人体姿态图片，在测试时发现当只用左手做动作时，预测效果欠佳，于是加拍了和左手动作有关的人体姿态图片。在最初对关键点进行特征工程时，是将人体的 33 个关键点都用于特征的创建，包括计算两点的距离、三个点组成的向量的夹角作为特征，但发现回归模型的预测效果不佳。经过仔细思考，发现人体一些关键点的距离是固定的，如头部到左右肩关节的距离、左右肩关节到左右髋关节的距离等，去除掉相应的特征之后发现模型的预测效果有小幅提升。

最初往树莓派上烧录的是 Raspberry Pi OS 的开发版，这个版本内置的是 Python3.9 及很多最新版本的软件，缺乏稳定性。在调试摄像头的时候，发现系

统无法正常使用摄像头，经过网上大量的查阅资料，发现是开发版的 Raspberry Pi OS 不支持调用摄像头。最后使用 Raspberry Pi OS 的 Legacy 版本即稳定版，此问题得以解决。在 Raspberry Pi OS 上安装 OpenCV 时，尝试过使用 Python 的 pip 命令安装、源文件编译安装等方法都无法安装成功，最后发现直接用 apt-get 命令安装 Python3-opencv 包即可。

结束语

本文是基于计算机视觉的公共场所防暴检测系统，使用 Python 语言进行目标检测模型 YOLOV4-Tiny、回归模型 KNN Regression 的训练。使用 Mediapipe 库中的 Blazepose 算法获取人体的关键点坐标，使用关键点坐标进行特征创建，以此训练回归模型 KNN Regression 来预测人体姿态所表现出来的暴力犯罪倾向指数，最后将此系统部署在树莓派上，以实现移动端的解决方案。

在评测用于移动端的目标检测模型时，最开始使用 YOLOV4 模型，由于树莓派 4B 的性能欠佳而无法运行起来，更换了更为轻量化的 YOLOV4-Tiny 模型后，在树莓派 4B 上得以勉强运行。在构建暴力犯罪倾向数据集时加上和左手有关的人体姿态图片，以及在利用人体关键点坐标进行特征创建时去除方差较小的特征，使得 KNN 模型的预测效果得到了提升。在配置树莓派环境时，尝试了多种 Linux 环境都无法使得摄像头正常接入，在尝试安装了 Legacy 版本的 Raspberry Pi OS 后问题得以解决。

最后，希望本文能对研究暴力行为识别的学者有所贡献。

参考文献

[1] 国务院. 中共中央国务院关于进一步加强社会治安综合治理的意见[EB/OL]. 2001. http://www.gov.cn/gongbao/content/2001/content_61190.htm.

[2] Enrique Bermejo Nievas, Oscar Deniz Suarez. Violence detection in video using computer vision techniques[J]. Computer analysis of images and patterns, 2011, 332-339: 1-4.

[3] Hassner T, Itcher Y, Kliper-Gross O. Violent flows: Real-time detection of violent crowd behavior[J]. Computer Vision and Pattern Recognition Workshops, 2012: 1-6.

[4] Sharma, Sudharsan, Naraharisetti. A fully integrated violence detection system using CNN and LSTM[J]. International Journal of Electrical and Computer Engineering, 2021, 3374-3380: 1-2.

[5] Amarjot Singh, Devendra Patil. Eye in the Sky: Real-Time Drone Surveillance System (DSS) for Violent Individuals Identification Using ScatterNet Hybrid Deep Learning Network[J]. Computer Vision and Pattern Recognition, 2018, 10.1109: 1-7.

[6] 周智, 朱明. 基于 3D-CNN 的暴力行为检测[J]. 计算机系统应用, 2017, 207-211: 1-2.

[7] 王晓龙. 基于轨迹分析的暴力行为识别算法研究[D]. 硕士学位论文. 2015: 1-3.

[8] 胡琼, 秦磊. 基于视觉的人体动作识别综述[J]. 计算机学报, 2013, 36(12): 2512-2524: 1-3.

[9] 廖星宇. 深度学习入门之 PyTorch[M]. 北京:电子工业出版社, 2017: 11-13.

[10] Pytorch 官方. KEY FEATURES & CAPABILITIES OF PYTORCH[EB/OL]. 2021. https://pytorch.org/.

[11] Python 官方. 历史和许可证-Python3.9.1 文档[EB/OL]. 2020. https://docs.python.org/zh-cn/3/license.html.

[12] 李沐. 动手学深度学习[EB/OL]. 2019. https://zh-v2.d2l.ai/chapter_convolutional-neural-networks/conv-layer.html#id2.

[13] Chien-Yao Wang, Hong-Yuan Mark Liao. CSPNet: A New Backbone that can Enhance Learning Capability of CNN[J]. Computer Vision and Pattern Recognition Workshop, 2020, 1571-1580: 4-5.

[14] Shu Liu, Lu Qi, Haifang Qin. Path aggregation network for instance segmentation[J]. Computer Vision and Pattern Recognition, 2018, 8759–8768: 1-2.

[15] Mediapipe 官方. Live ML anywhere[EB/OL]. 2021. https://google.github.io/mediapipe/.

[16] Valentin Bazarevsky, Ivan Grishchenko. BlazePose: On-device Real-time Body Pose tracking [J]. Computer Vision and Pattern Recognition, 2020, arXiv:2006.10204: 1-3.

致 谢

在此十分感谢王禹老师在我的论文撰写过程中提供的帮助与指导。由于我对人体姿态估计十分感兴趣，在进行论文的选题时，我向王老师询问是否可以做这方面的课题，王老师就选题新颖性、技术可行性等方面与我进行了深入的讨论，老师建议我选择暴力犯罪检测相关的题目。在确定了此题目后，王老师耐心地向我介绍了当前目标识别常用的算法，同时考虑到要在树莓派上流畅运行，他建议我使用 YOLOV4-Tiny 算法来进行目标检测。

在论文的撰写过程中，王老师向我推荐了许多优秀的论文与期刊，拓展了我撰写论文的思路。在配置树莓派环境和模型训练的过程中，遇到了许多问题，王老师一一耐心对我进行指导。在最后的论文撰写中，王老师对于我的论文格式与内容进行了严格的把关。再次，衷心地感谢王老师在我的毕业设计制作和论文编写过程中的悉心指导和帮助。