

计算机视觉在学生体测中的应用探索

谢跃¹ 唐闰臣¹ 梁瑞宇¹ 尤海发²

(1. 南京工程学院通信工程学院, 江苏南京 211167;

2. 南京读动信息科技有限公司, 江苏南京 210037)

摘要: 促进学生体质健康发展是学校体育教育的核心要义, 为激励学生积极进行身体锻炼, 体测结果已被纳入学业考试范围。但传统人工记录体测结果的方式效率低下, 也难以契合后续的体测大数据分析, 为此提出使用人工智能算法实现体测过程中数据的智能化测量。首先利用计算机视觉辨识学生举手准备动作, 同时采集人脸数据, 进入脸识别完成身份认证工作, 其次进行各体测项目的智能测试, 例如跳绳测试中, 利用关节的周期性运动规律进行跳绳计数, 跳远测试中进行人体姿态估计检测最终落地点并进行距离测量, 最后将测试结果进行上传保存以便后续分析。该系统不仅可以减轻人工监测的压力, 提高传统体测效率, 还便于科学化管理分析体测数据。

关键词: 学生体测; 人工智能; 计算机视觉; 人体姿态估计

为建立健全国家学生体质健康监测评价机制, 激励学生积极参加身体锻炼, 教育部印发《国家学生体质健康标准(2014年修订)》, 要求各学校每学年开展覆盖本校各年级学生的《标准》测试工作, 并将体测结果纳入学业考试范围。而随着人工智能在各个领域的推广应用, 其在学生体测数据的采集与分析中也得到了较为广泛的关注和研究。

传统体测一般需要老师提前准备学生信息名单, 测试结果需要通过纸质记录, 最后安纸质记录结果录入电脑系统进行上传保存。测试过程较为烦琐, 耗费人力的同时也容易产生认为误差。而人工智能与大数据的介入, 不仅可以减少人力支出, 还能确保数据的科学性与规范性。例如利用人脸识别技术完成学生身份认证的过程, 就无须事先准备学生信息名单了[1, 2]。在获得体测数据后, 还可以利用机器学习和人工智能算法对体测成绩进行深度分析。在体测过程中, 通过视频分析技术实时监测测试过程中的各种突发事件, 并发出报警信息。

但是随着人工智能技术, 尤其是计算机视觉技术的成熟, 利用其进行体测数据的智能化检测已成为可能。本系统为实现智能化、无人化、规范化的体测, 提出了一套基于计算机视觉的体测流程。包括学生准备就绪检测、身份认证、体测数据的自动化测量, 体测成绩的保存汇总等, 并在跳绳、立定跳远等体测项目中得到了成功应用

一、系统结构

智能体测场景设计如图1所示, 由两部分构成, 图中右侧为学生运动区域, 在该区域中划分了不同体测项目的测试区域。图中左侧为智能体测柱, 主要由摄像头和工控板组成。右侧的运动区域和左侧的智能体测柱要保持合适距离, 确保摄像头能正好完整采集到运动区域的画面。工控板负责数据的传输以及给管理员提供一个交互界面。

系统测试流程如下: 学生首先进入检测区域, 面对摄像头举手示意自己已准备就绪。举手动作的判别由计算机视觉中的人体姿态估计完成, 如果一段时间内未检测到举手示意, 会有语音提示学生进行准备示意。在检测举手示意的同时, 会利用人脸检测算法捕获画面中的人脸图像数据, 进行身份认证, 以便后续测试成绩与个人信息一一对应。如果身份认证失败, 会有语音提示并退系统, 此时需要与管理员联系, 确认系统中有无该学生的待测信

息。身份认证成功, 语音会提示学生, 可以开始进行项目测试。根据不同体测项目, 设计不同的人工智能算法进行智能化测试, 各体测项目的智能化测试方法详见第三节。测试完成后, 系统会自动保存测试成绩结果, 并语音提示学生测试结束。

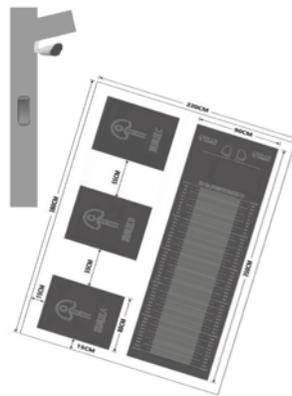


图1. 体测场景效果图

二、计算机视觉在体测中的应用

本文主要以跳绳和立定跳远为例, 结合智能化测试流程, 探讨计算机视觉在体测成绩测评中的应用。主要涉及人体关键点检测和人脸识别技术。

(一) 人体关键点检测

人体关键点检测主要采用了开源项目 Openpose, 它是首个基于深度学习的实时多人二维姿态估计应用。可检测 18 个人体关键点, 包括左右耳、左右眼、鼻、左右肩、左右肘、首、左右手、左右腰、左右膝、左右脚。

1. 举手检测

在正式体测开始前, 需要学生通过举手示意来表示自己已经准备就绪。这一过程要求学生进入待测区域, 面对摄像头完成。举手示意的判别依据人体关键点的相对位置由计算机完成识别判断。人在举手时, 手腕的位置相对高于肩膀的位置, 即右/左手关键点所在的纵坐标位置小于右/左肩的坐标位置(以图像的左上角为原点)。当这种相对位置关系在持续一段时间内均能检测到时, 则判定举手成功。如图2所示, 人体上的红色线段和绿色点为人体关键点检测的结果, 人形外围的紫蓝色方框表示是否检测到有人。

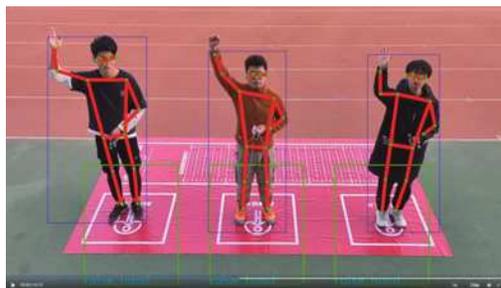


图2. 举手判别效果图

2. 跳绳实时计数

人在跳绳过程中,人体的关键点是上下起伏的,符合一定的周期性运动规律,检测这种周期性运动规律是实现跳绳实时计数的关键。但由于跳绳姿势的差异,脚腕关节的变化差异较大,因此通过检测肩和腰部关键点更具有普适性。通过检测肩部关键点在跳绳过程中出现的最高点与最低点的次数完成一次计数,其中最高点代表起跳后的最高位置,最低点表示落地后的位置,每出现一对最高点与最低点则计数一次。但由于跳绳过程中,前后运动也会产生类似的高低点对。而对于普通的单目摄像机而言,为避免这类误检的产生,还需要通过计算关键点在纵向上的梯度,其反应的是运动速度。人在跳绳时,在纵向上运动速度要明显比前后运动时的速度快。这样可以有效规避断跳后,学生因身体的前后运动而产生的误检。检测效果如图3所示。

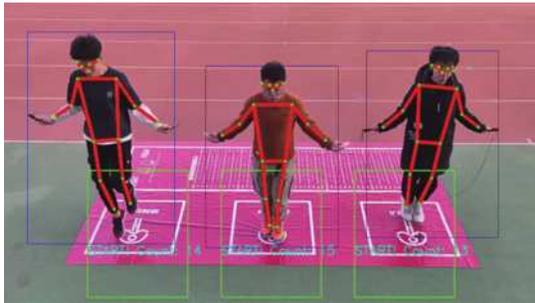


图3. 跳绳实时计数

3. 跳远中落地点检测

在立定跳远项目中可以利用人体关键点检测来获取起跳后的落地点。其关键是检测脚腕关键点位置在水平方向上的运动。运动的距离由视频中相邻两帧的帧差计算获得。当起跳后,人体关键点在水平方向的运动是比较明显的,相邻帧的差值也较大。但准备起跳和落地静止后,相邻帧的差值趋于零,以此来捕获落地时刻。在立定跳远中,学生的落地姿势存在各种差异,可能会存在落地后,人体后仰,导致手掌撑地等情况发生,致使落地点不再是脚后跟。为此落地后还需要持续检测全身所有在脚部后侧的骨骼点,一旦有骨骼点在纵向上接近两只脚踝纵向坐标值,则视为该骨骼点触地。即持续计算落地后所有帧,一旦有骨骼点触地,则对比该触地点是否为最左边的触地点,如果是,则更新触地点。图4为手掌撑地状态下的落地姿势效果检测。



图4. 手掌撑地

(二) 人脸检测与识别

鉴于学校学生群体往往较大,很难将所有学生的身份信息全部同步到资源有限的终端设备中,因此终端设备只负责采集人脸数据,然后将人脸数据发送给后台,进行人脸识别,完成身份认证。

人脸检测使用了MTCNN模型,其核心包含了三个网络结构:P-Net、R-Net和O-Net。三个网络具有类似的结构,输出信息都包含了人脸关键点、人脸框位置以及是否含有人脸。但三个网络

的复杂程度逐步提高的。P-Net网络复杂度最低,其输出的结果比较粗糙,因此需要R-Net对其进行进一步调优,消除P-Net中很多误判。O-Net的复杂度最高,它会对余下的候选人脸区域进行最终判别,得到最后的检测结果。

后台从终端识别上接受到人脸图片后,会先将其转成一个向量来表示,也就是所谓的“特征”。人脸识别的本质就是检测数据库中给定人脸图片最相似的人,相似度可通过向量之间的距离来衡量。在训练人脸识别模型时,采用了中心损失,它不仅保留了原有的分类模型,还给每个类指定了一个类别中心。同一类的图像对应的特征都应该尽量靠近自己的类别中心,不同类的类别中心尽量远离。由此训练的模型部署于后台上,用于提取人脸特征,从而识别人脸身份信息,并将识别结果反馈给终端设备。

(三) 跳远测距

鉴于立定跳远测距的精度要求较高,虽然单目摄像机也有测距算法,但精度很难达到体测要求。为了能够达到较高的测距精度,同时不增加算法的复杂度,系统在跳远运动区域中绘制了距离刻度线,摄像机只需要识别刻度线与落地点位置即可获得跳远成绩。

从图4中可以看出,摄像头非是从垂直的俯视图拍摄跳远运动区域的,为能够获得较好的测距结果,首相需要通过透视变化,将运动区域转变为垂直的俯视角度。然后通过数字图像中的形态学操作,获取刻度线的位置信息。距离的读取不是依赖于地表上的数字刻度,而是每个刻度线在图像中从起跳线开始依次排列的顺序,因为数字符号的地表在距离较远的摄像机中很难辨识,但较粗的刻度线相对容易识别。所以即使看不清图像中的数字符号,也能够进行距离测量。结合图4中落地点检测结果,可获得图5所示的测量结果。



图5. 落地点标注

三、总结

利用计算机视觉与人工智能算法为实现学生体测的科学化、智能化提供一种新的思路。本文以跳绳计数和立定跳远测距为例,设计了一整套智能化测试流程和方法。涉及以人体姿态估计完成的举手示意检测、跳绳计数与跳远的落地点检测,以及通过人脸识别算法完成身份认证,利用计算机视觉完成跳远测距等内容。该系统能够提高体测效率,便于实现体测数据的科学化管理,具有较好的应用前景。

参考文献:

- [1] 吴菲,基于人脸识别的体测系统设计与实现.大众标准化,2021(21):17-19.
- [2] 曹明利等.基于计算机视觉的体测测评打分系统.in天津市电子工业协会2022年年会.2022.中国天津.
- [3] 叶子健, et al., 基于神经网络和支持向量机的体测分析模型.科学技术创新,2021(34):55-57.
- [4] 秦川,李宏伟, and 孙沐钰,DT时代下智能体测服务设备的开发.科技风,2020(15):17.

基金项目:南京工程学院高等教育研究课题(2020YB25),江苏省研究生教育教学研究课题(JGKT22_C049)