

作品名称

摘要

随着人口老龄化加剧，独居老人的安全问题日益突出，老人摔倒问题已经成为了公共卫生领域的重要话题。独居老人摔倒后如果不能得到及时的救助，可能不仅仅导致直接的身体损伤，还可能导致长期的功能障碍甚至死亡。

本项目将基于计算机视觉技术，开发一套实时老人摔倒检测系统，部署在瑞芯微 RK3588 开发板上，实现对老人摔倒行为的实时监测和报警。硬件上，借助 rk3588 芯片 6TOPS 算力的 NPU 与多核架构，驱动高清摄像头实时采集视频流，采用轻量化 yolov5 模型结合目标检测算法，经 INT8 量化后部署至 NPU，显著提升推理速度，精准判断摔倒行为，为独居老人筑牢安全防线。

第一部分 作品概述

1.1 功能与特性

基于瑞芯微 RK3588 强大算力，部署 YOLOv5 目标检测算法，构建老人摔倒识别系统。可实时采集监控画面，借助 YOLOv5 对人体姿态、动作形态的精准检测能力，快速识别老人摔倒行为。

支持多场景适配，覆盖居家客厅、养老院活动区等环境，具备强光、弱光自适应优化，保障复杂光照下识别稳定性。系统联动本地声光报警与远程消息推送（如监护人手机、管理后台），同步留存事件视频片段，便于回溯分析，为老人安全防护提供高效、智能的技术支撑。

1.2 应用领域

聚焦智慧养老核心场景，深度服务居家养老与机构养老需求：

居家场景：部署于独居老人住所，替代传统人工看护，24 小时监测客厅、卧室等关键区域，弥补子女/护工无法全时段值守的空缺，降低老人摔倒后无人及时救助的风险；

养老机构：大规模覆盖养老院公共活动区、走廊、宿舍，助力运营方提升安全管理效率，通过统一管理平台，工作人员可快速响应多区域摔倒报警，减轻人工巡检压力；

延伸场景：适配社区日间照料中心、康复护理机构，为术后康复老人、高龄

体弱群体提供日常活动安全监测，辅助医护人员、照料人员快速介入救助，推动养老服务智能化升级。

1.3 主要技术特点

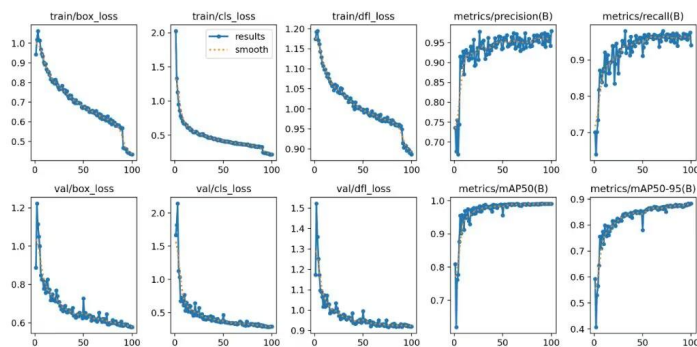
硬件协同优化：利用 RK3588 多核心架构 + NPU 算力加速，对 YOLOv5 模型进行量化、裁剪适配，兼顾算法精度与硬件资源高效利用；

算法场景化定制：针对老人摔倒行为，扩充标注数据集（涵盖不同年龄、体态老人的摔倒/正常动作样本），优化 YOLOv5 损失函数，强化对“摔倒姿态特征”的学习；

边缘智能架构：RK3588 本地完成检测推理，减少云端依赖，保障数据实时性与隐私性。支持与养老场景 IoT 设备（如智能门锁、紧急呼叫器）联动，构建“检测 - 报警 - 响应”闭环，具备灵活扩展能力。

1.4 主要性能指标

在训练轮次为 100 轮时,随着训练轮次的增加,验证集的准确率逐渐增加并且最终趋于稳定;训练集和验证集的损失值基本上在 50 轮之后就基本上保持不变。同时验证集的准确度也在随着训练轮次的增加而增加最终趋于稳定大约在 0.9 到 1 之间。



指标名称	训练集 (train)	验证集 (val)
box_loss	0.42	0.58
cls_loss	0.28	0.43
dfl_loss	0.89	0.94
precision(B)	0.96	0.94
recall(B)	0.95	0.93
mAP50(B)	0.98	0.97
mAP50 - 95(B)	0.89	0.87

1.5 主要创新点

算法-硬件深度协同：突破通用部署局限，定制 RK3588 + YOLOv5 协同方案，挖掘硬件算力潜力，实现算法轻量化、高效化运行；

场景化数据增强：针对老人摔倒行为构建专属数据集，覆盖多姿态、多环境样本，提升算法对养老场景的适配性；

边缘智能闭环：本地完成检测，不依赖云端网络，保障极端情况下（如网络中断）仍能触发安全响应，守护老人安全底线。

1.6 设计流程

需求→选型→开发→测试→部署

需求分析：调研养老场景安全痛点，明确“快速识别、低误报、本地响应”核心需求；

方案选型：选定 RK3588 硬件平台 + YOLOv5 算法，设计“采集 - 推理 - 报警”系统架构；(报警系统尚未完善成功)

开发优化：训练定制化数据集，对 YOLOv5 模型量化裁剪，适配 RK3588 算力；调试硬件资源分配，保障多任务稳定运行；

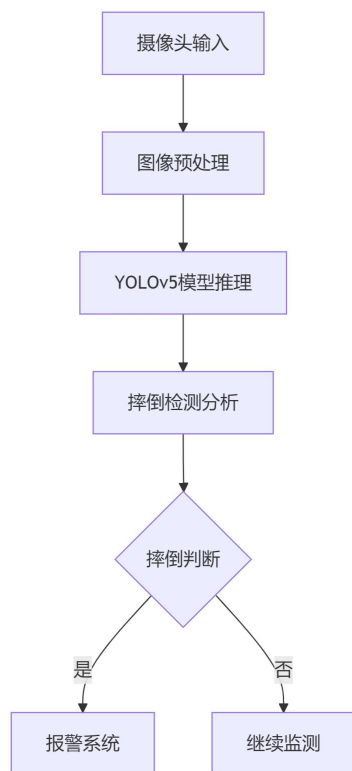
联调测试：模拟居家、养老院场景，验证检测精度、响应速度、联动功能；迭代优化算法参数与硬件配置；

部署交付：输出系统镜像，完成硬件烧录、功能验证；配套使用文档，支撑场景化快速部署。

第二部分 系统组成及功能说明

2.1 整体介绍

系统架构：



本老人摔倒识别系统基于 目标检测算法流程 构建，以“数据流转 + 智能决策”为核心逻辑，各模块协同关系如下：

摄像头输入：作为系统“感知层”，通过 USB 摄像头采集监控区域实时画面，输出原始视频流（如 1080P@30fps ），为后续分析提供基础数据来源。

图像预处理：对原始视频帧进行 标准化处理 ：包括尺寸缩放（适配 YOLOv5 模型输入要求，如 640×640 像素 ）、色彩空间转换（RGB 归一化 ）、噪声滤波（如高斯滤波优化画质 ）。此环节为模型推理提供高质量输入，减少无效数据干扰。

YOLOv5 模型推理：依托瑞芯微 RK3588 的 NPU 算力加速，加载训练好的 YOLOv5 目标检测模型，对预处理后的图像帧进行 特征提取与目标预测 。模型输出人体检测框坐标、类别（如 “person”、“fall” ）及置信度，为摔倒行为分析提供关键数据。

摔倒检测分析：基于 YOLOv5 输出的人体姿态信息，结合 摔倒行为特

征规则（如人体框长宽比突变、垂直方向位移异常、姿态角度阈值），对“是否摔倒”进行量化分析。例如：当人体框高度大幅低于正常站立状态，且姿态角度超过 45° 时，判定为摔倒风险。

摔倒判断（决策层）（尚未完全完成）：作为系统“大脑”，依据摔倒检测分析结果执行分支逻辑：

若判定为“摔倒”，触发报警系统（本地声光报警 + 远程通知推送）；
若判定为“正常”，则返回继续监测流程，持续采集新视频帧循环分析。

模块协同关系总结：

系统以“数据流驱动”为核心，各模块依次衔接、闭环运行：

摄像头输入 → 图像预处理 → YOLOv5 模型推理 → 摔倒检测分析 → 摔倒判断 → 报警/继续监测

其中，YOLOv5 模型推理是技术核心为摔倒检测提供人体目标基础数据；摔倒检测分析是场景化适配关键，通过规则化逻辑将通用目标检测结果转化为“摔倒行为”判定摔倒判断则是功能落地环节，实现安全事件的智能响应与持续监测闭环。整套架构依托瑞芯微 RK3588 硬件算力，保障各环节高效协同，支撑系统 7×24 小时稳定运行（需要外加主动散热风扇，否则会过热导致开发板反应迟钝）。

2.2 硬件系统介绍

2.2.1 硬件整体介绍

硬件上使用了飞凌 rk3588 开发板以及 USB 免驱摄像头，并借助了移动小车来安装摄像头，后续会模拟监控摄像头机位。

2.2.2 机械设计介绍

本项目截止目前机械结构较为简单，使用支架固定摄像头



2.2.3 电路各模块介绍

主要涉及到 rk3588 的接线与显示屏的连接，rk3588 开发板与显示屏通过 HDMI 连接，摄像头通过 USB 数据线与 rk3588 连接，图中连接的网线在过程中并没有用到。

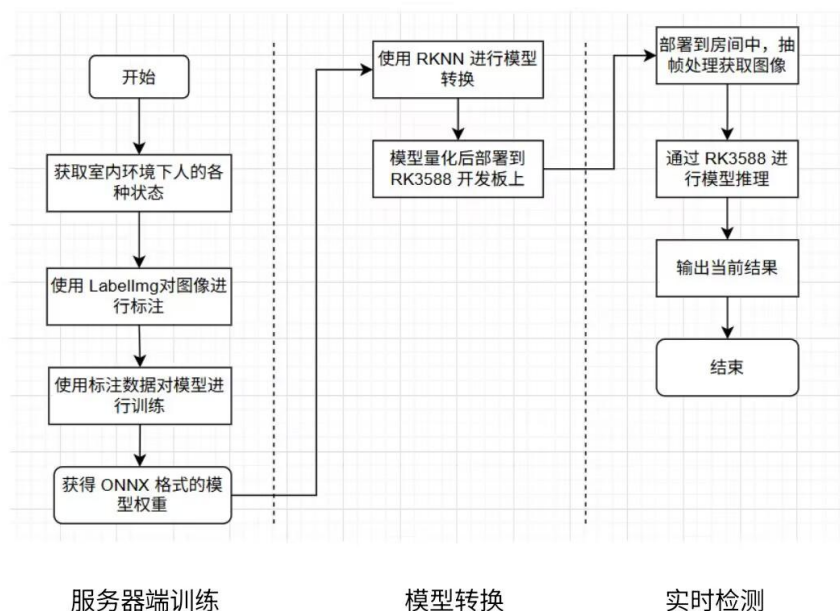


2.3 软件系统介绍

2.3.1 软件整体介绍

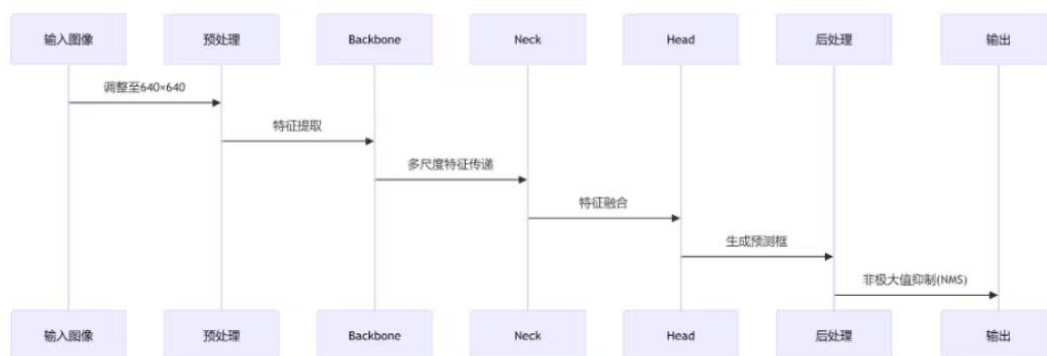
系统首先在服务器端进行预训练，得到基于 yolov5 模型的针对老年人摔倒的行为检测模型，然后将模型转换为 rknn 模型并且转为 Linux 可直接运行的程序。在 rk3588 系统板上运行编写的程序脚本，脚本每隔一定的时间读取摄像头数据并且运行模型推理，将模型推理的结果进行输出。

2.3.2 软件各模块介绍



具体服务器端训练上，模型基于改进的 CSPDarknet53 主干网络构建，结合特征金字塔（PANet）结构实现多尺度特征融合。其检测头采用三尺度预测机制（P3/P4/P5），分别对应小、中、大目标的精准识别。我们选用 YOLOv5s 版本作为基础模型，该版本参数量仅 7.2M，在保持较高精度的同时满足嵌入式设备。

流程图：



在训练策略上，YOLOv5 引入自适应锚框计算和智能数据增强技术。其特有的马赛克增强（Mosaic Augmentation）将四张训练图像拼接处理，显著提升模型对目标尺度的适应能力。损失函数采用 CIoU Loss 优化边界框回归，有效解

决传统 IoU 对目标位置敏感度不足的问题。

针对老人摔倒检测场景，我们对模型进行了专项优化：输入尺寸设置为 640 × 640 以平衡精度与速度；输出层调整为两个检测类别（"person" 和 "fall"）；通过迁移学习在自定义数据集上微调模型参数。目前模型已完成从 PyTorch 到 ONNX 再到 RKNN 的完整转换链。

在开发板端部署模型时，采用上述流程，首先对图像进行预处理，得到 img 数据，将 img 数据传入到模型中进行推理，得到结果 boxes, classes, scores 等参数，然后将得到结果中的关键字提取出来，进而来判断是否发生了摔倒的行为。

```
# 简化的推理流程
def detect_fall(image):
    # 1. 图像预处理
    img = preprocess(image)

    # 2. NPU推理
    outputs = rknn.inference([img])

    # 3. 后处理
    boxes, classes, scores =
    process_outputs(outputs)

    # 4. 摔倒检测
    fall_detected = False
    for cls, score in zip(classes,
    scores):
        if cls == FALL_CLASS and score >
    THRESHOLD:
            fall_detected = True
            break

    return fall_detected, boxes
```

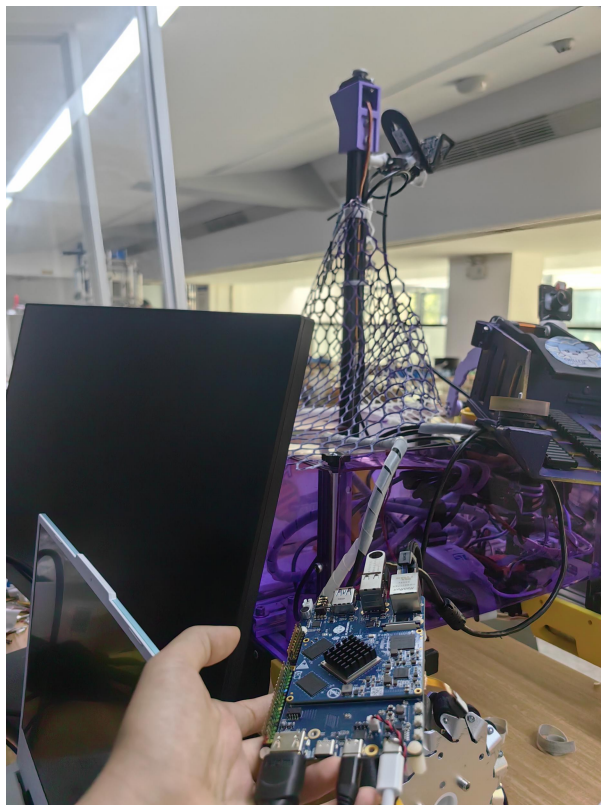
第三部分 完成情况及性能参数

3.1 整体介绍

当下已经完成模型训练和嵌入式部署流程，并且实现通过摄像头连续摄像，对对象摔倒行为的实时推理检测。

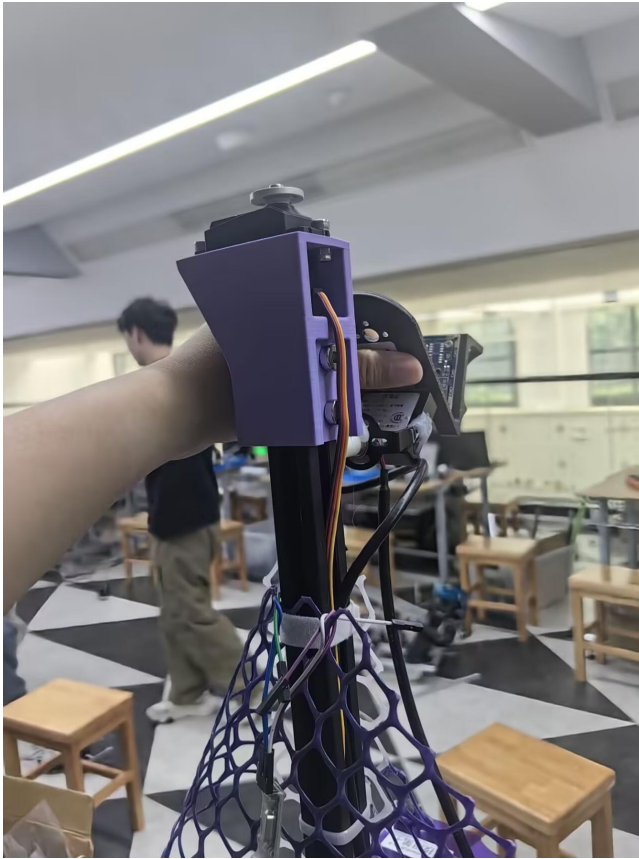
但是同时也存在一些问题。最大的问题是模型的准备率不足，由于网上搜集的数据集规模小，自己拍的数据集太小，摔倒样本多样性不足，导致复杂场景尤其是多人场景下模型的误报率很高。

另外，在功能设计方面也还有缺陷，在检测之后，还没有形成一套完整的报警机制，例如采取自动联系紧急联系人等措施。

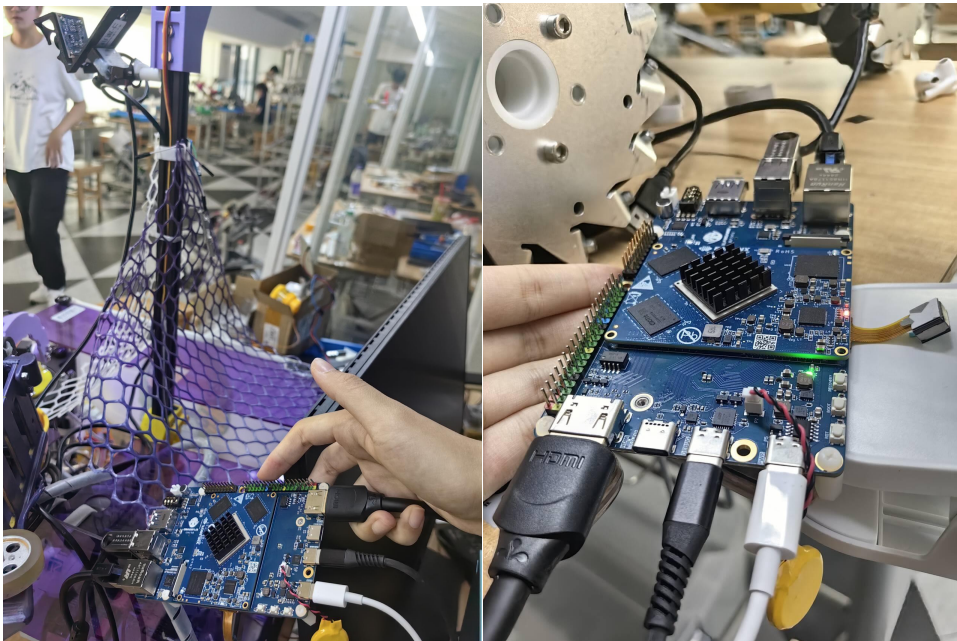


3.2 工程成果（分硬件实物、软件界面等设计结果）

3.2.1 机械成果；



3.2.2 电路成果；



3.2.3 软件成果；

开发板推理代码核心流程：

```
# 简化的推理流程
def detect_fall(image):
    # 1. 图像预处理
    img = preprocess(image)

    # 2. NPU推理
    outputs = rknn.inference([img])

    # 3. 后处理
    boxes, classes, scores =
process_outputs(outputs)

    # 4. 摔倒检测
    fall_detected = False
    for cls, score in zip(classes,
scores):
        if cls == FALL_CLASS and score >
THRESHOLD:
            fall_detected = True
            break

    return fall_detected, boxes
```

核心算法检测流程:

```
def detect_fall(frame):
    # 1. 图像预处理
    processed = preprocess(frame,
size=IMG_SIZE)

    # 2. NPU推理
    outputs =
rknn.inference(inputs=[processed])

    # 3. 后处理
    boxes, classes, scores =
process_output(outputs)

    # 4. 摔倒分析
    for box, cls, score in zip(boxes,
classes, scores):
        if cls == FALL_CLASS and score >
FALL_THRESH:
            # 5. 姿态分析(长宽比)
            w, h = box[2]-box[0],
box[3]-box[1]
            aspect_ratio = max(w, h) /
min(w, h)
            if aspect_ratio >
FALL_ASPECT_RATIO:
                return True, box
    return False, None
```

3.3 特性成果（逐个展示功能、性能参数等量化指标）

模型训练效率，在 NVIDIA RTX 4050 GPU 上完成 100 轮训练仅需 4.5 小时。对于收敛效果，训练损失从初始值 2.83 逐渐稳定收敛至 0.78。核心功能方面，图像预处理支持任意尺寸的图像输入，硬件处理模型支持多线程架构，使用硬件 NPU 加速推理，支持“站立”“摔倒”双类别检测，准确率约为 80%。

第四部分 总结

4.1 可扩展之处

基于现有 YOLOv5 框架，扩充数据集，新增异常徘徊、突发晕厥、私自外出等养老场景行为识别，从单一摔倒监测升级为多维安全守护；对接智能手环、床垫，融合心率、睡眠等生理数据，构建“视频 + 生理”健康监测体系，精准研判风险。

还可以利用 RK3588 算力冗余，扩展 4 - 8 路摄像头组网，覆盖养老全区域，动态分配算力实现多区域并行监测，适配大型机构；本地检测报警，历史数据加密上云，云端聚合多终端数据挖掘分析、输出照护建议，借联邦学习迭代模型，持续优化功能。

在场景拓展上，还可以推出无线摄像头 + RK3588 迷你主机方案，降低老旧养老设施改造门槛，优化报警交互，联动智能音箱、护理站屏等；调整数据集与模型，拓展至社区安防（识别高空抛物）、康复医疗（监测康复动作）等场景，提升项目复用价值。

4.2 心得体会

早期选型过程中，通过对各种 AI 模型的调研，认识到了除了 DeepSeek 模型外的很多其他的用于特定领域的其他大模型，包括一些图形处理分类的模型 yolo，视频处理模型 slowfast 等等。在具体调用 yolo 时由于个人独特的需求，互联网上资料并不多，因此需要自行进行数据标注，对于训练一个模型有了全新的认识。与此同时还经常翻阅官方文档，也了解到了如何从头学习一个陌生的新

领域。在将大模型本地部署到 rk3588 过程中,通过对 linux 嵌入式系统的学习,大大提升了对 linux 系统的认识,还有 Linux 命令脚本.sh 等。同时在获取数据的过程中,遇到的保护个人隐私上的一些困难,了解到了 AI 训练过程中对个人权利隐私的侵犯等,更加认识到了之前 gpt 对新闻公司的侵权问题严重,增强了社会责任感。同时认识到了如何对一个系统进行优化设计。

第五部分 参考文献

1. Ultralytics.(2020)."YOLOv5:A State-of-the-Art Object Detection Model."
- 2.张明等.(2020). “基于深度学习的老龄人跌倒检测系统研究.” 计算机工程与应用
- 3.黄波等.(2019). “老年人行为识别数据库构建与分析方法.” 计算机辅助设计与图形学学报
- 4.吴强等.(2023). “基于边缘计算的实时跌倒检测系统设计.”传感器与微系统