

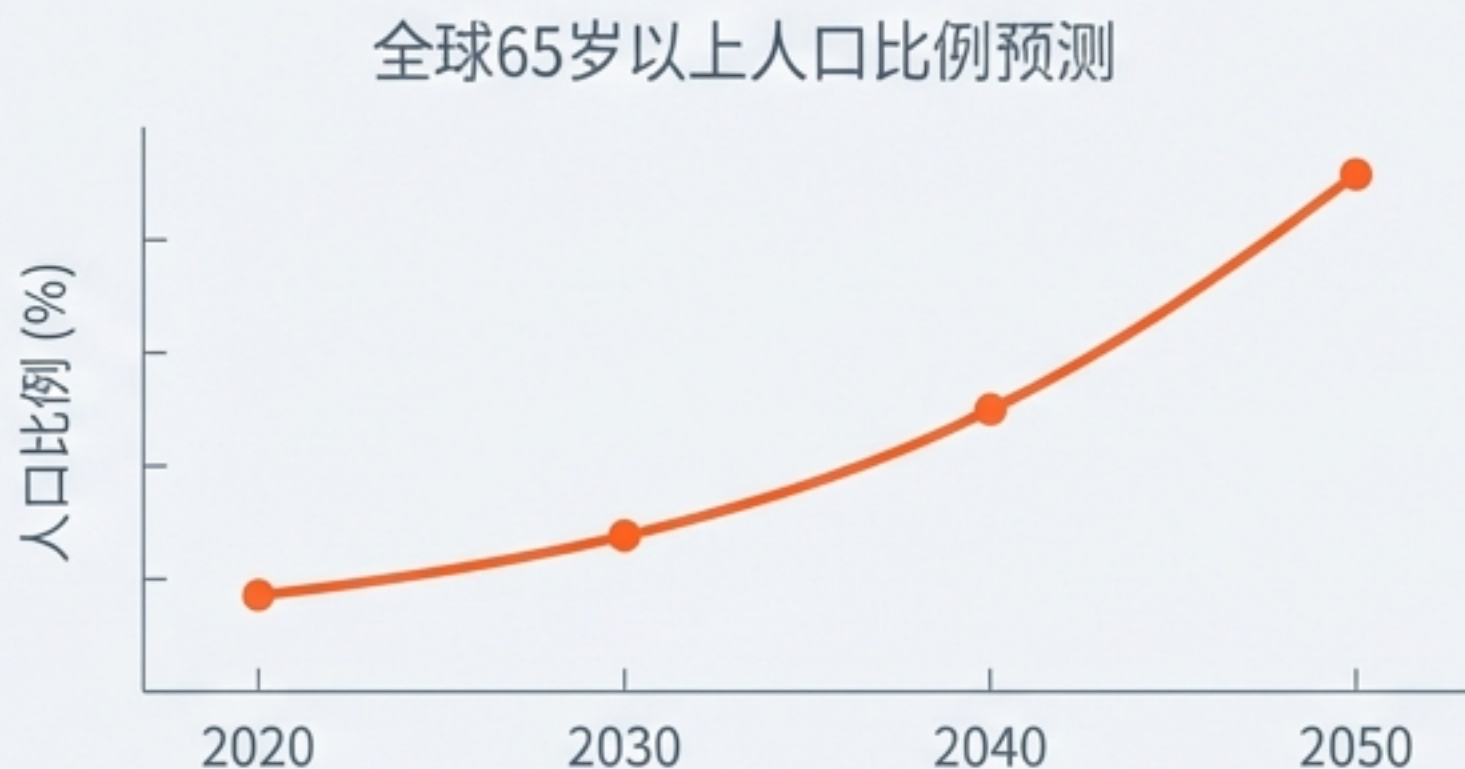
用技术守护安全：构建基于AI的老人监护系统

一份面向开发者的工程实践手册

我们面临的挑战：日益严峻的居家养老安全问题

人口老龄化加剧

全球人口结构正在迅速变化，老年人口比例持续上升。这带来了巨大的社会和家庭照护压力。



跌倒是老年人最常见的伤害风险

意外跌倒可能导致严重后果，包括身体伤害、心理恐惧和独立生活能力的丧失。及时发现并响应至关重要。



每年有超过 **30%** 的长者至少经历一次跌倒。



跌倒是导致老年人意外伤害致死的主要原因。

我们的解决方案：一个非侵入、实时的智能监护系统

利用计算机视觉和人工智能，我们能够将任何普通摄像头转变为一个主动式的安全哨兵。系统在保护隐私的前提下，24/7不间断地分析实时视频流，以识别潜在的危险情况。



非侵入式 (Non-Invasive)

无需穿戴任何设备，不改变老人的生活习惯。



实时响应 (Real-Time)

毫秒级检测，确保在意外发生时能立即触发警报。



高准确性 (High Accuracy)

基于深度学习模型，能有效区分正常活动与真实跌倒。

系统架构一览：从像素到警报的全流程



实时视频输入



YOLO 目标检测

在画面中精确定位人形目标。



姿态分析

提取人体关键点，理解身体姿态。



FSM 状态判断

基于姿态序列，判定当前行为状态（站立、行走、跌倒）。



异常事件警报

当检测到‘倒地’状态时，立即触发警报。

核心技术栈



Python

角色：主要开发语言。其丰富的生态系统和简洁的语法，是连接所有模块的粘合剂。



OpenCV

角色：计算机视觉库。负责处理所有底层的图像和视频流操作，如读取摄像头、图像预处理等。



YOLO (You Only Look Once)

角色：实时目标检测算法。用于在视频帧中快速、准确地识别人体。



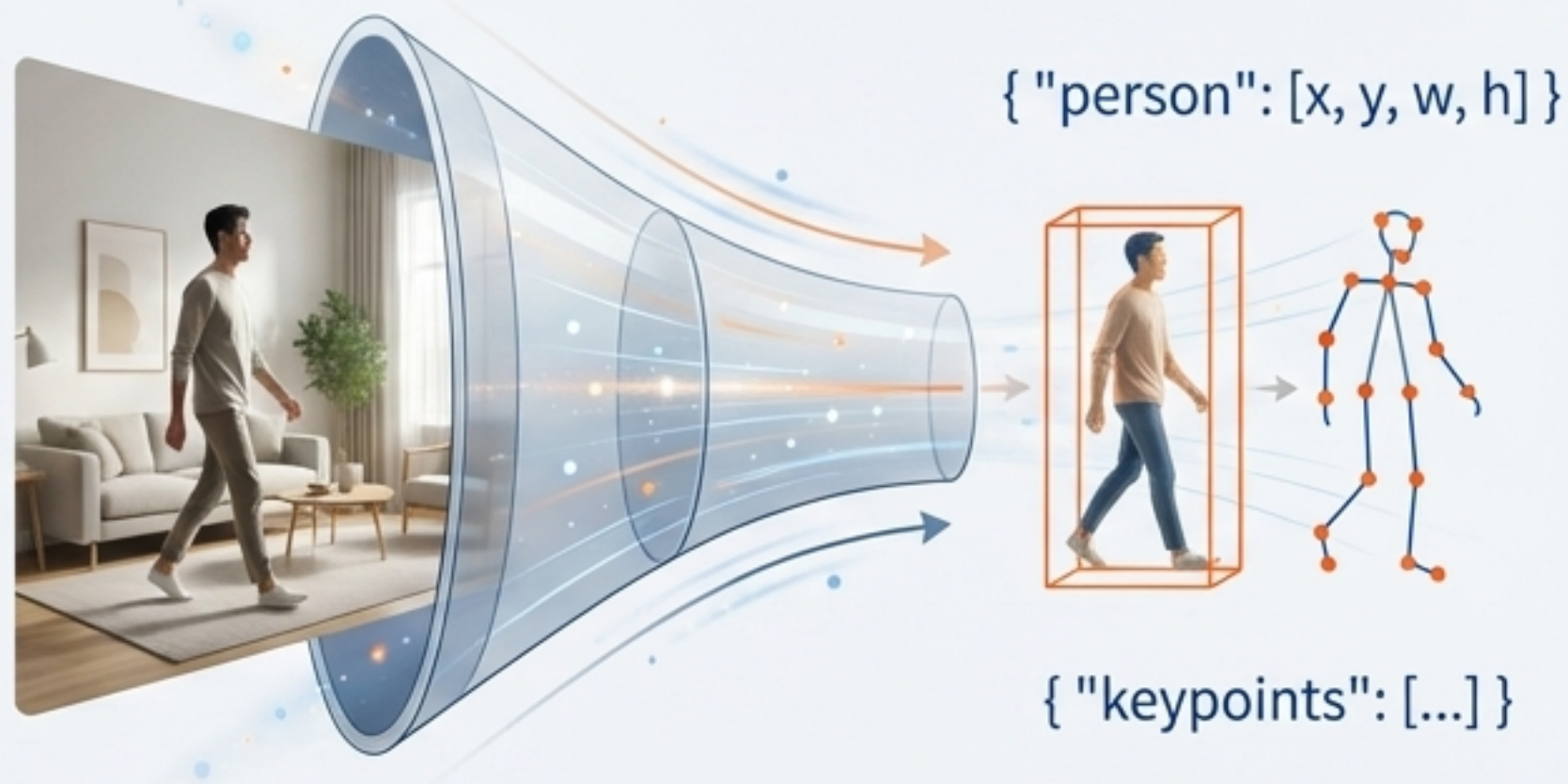
FSM (Finite State Machine)

角色：核心逻辑模型。用于对人的行为状态进行建模和判断，是系统决策的大脑。

看见并理解场景

机器如何从连续的视频帧中，不仅“看到”有人，而且“理解”这个人的位置和姿态？

1. **定位 (Localization)**：首先，我们需要在复杂的背景中找到目标。
2. **分析 (Analysis)**：接着，我们需要分析目标的身体结构和姿态。



关键技术一：YOLO——实现高效实时的目标检测

Why YOLO?

对于跌倒检测系统，速度就是生命。YOLO算法能在单次网络前向传播中同时预测边界框和类别概率，实现了速度与精度的卓越平衡，完美满足了我们的实时性需求。

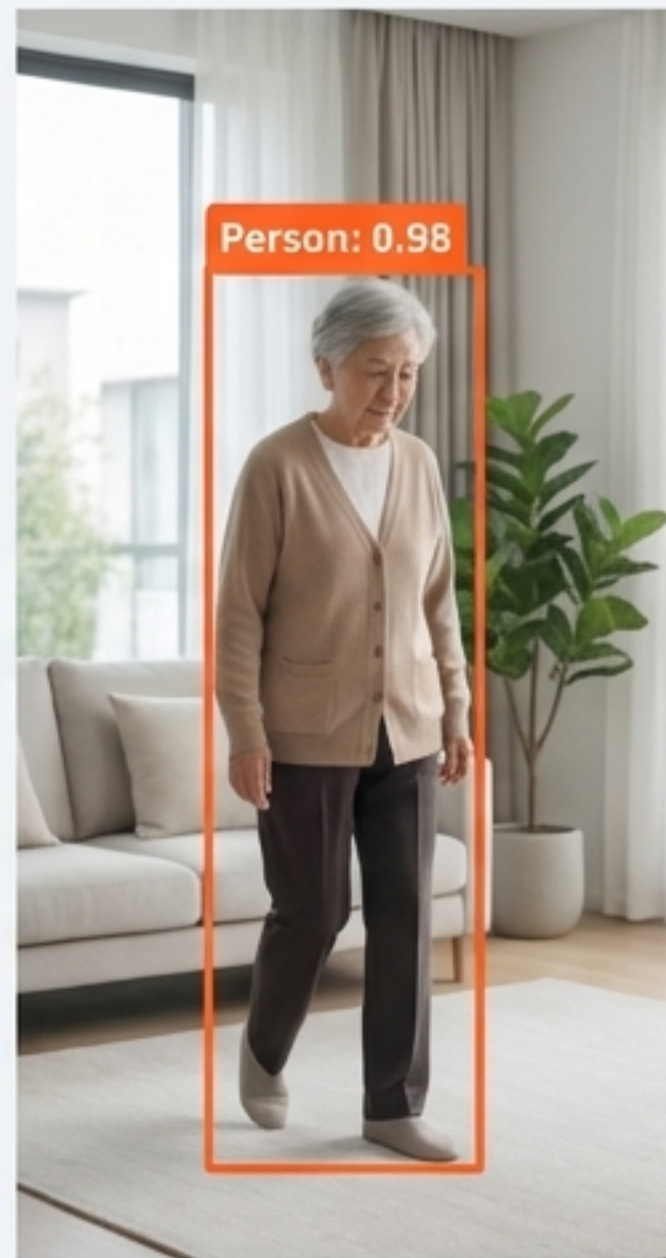
YOLO in Action

- **输入 (Input)** : 原始视频帧 (Raw Video Frame)
- **处理 (Process)** : YOLO 模型分析
- **输出 (Output)** : 带有边界框和置信度的人体目标 (Person, 98%)



输入 (Input)

YOLO 处理



输出 (Output)

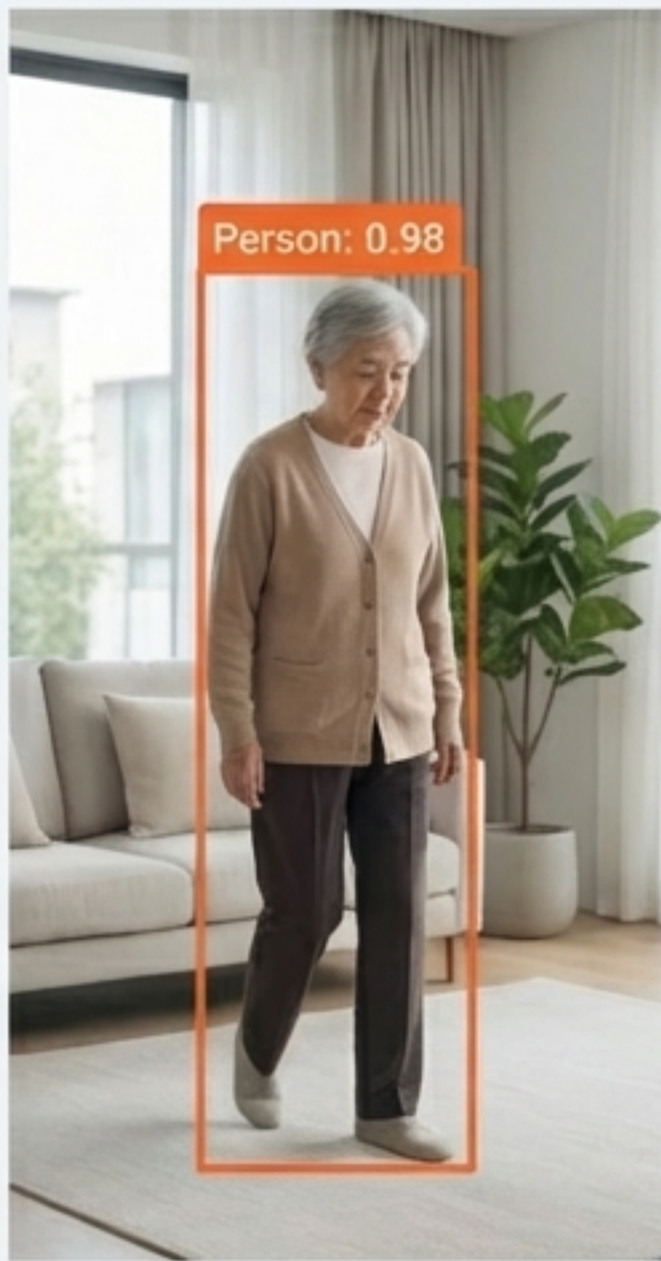
从定位到理解：利用姿态估计提取人体关键点

Rationale

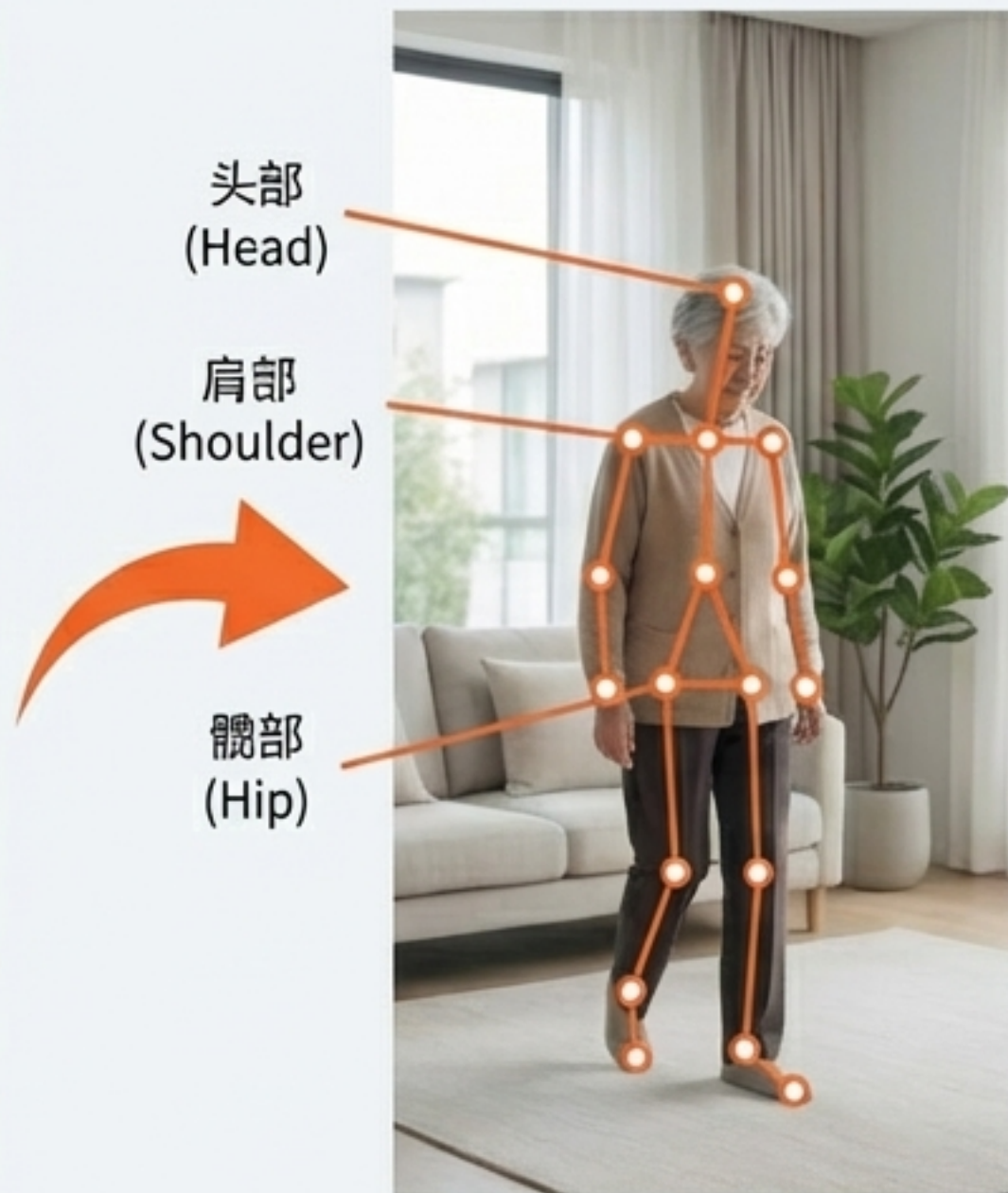
仅有边界框不足以判断跌倒。我们需要更精细的数据——**人体的骨骼结构**。姿态估计技术能从边界框内的像素中识别出人体的**关键关节点点**（如头、肩、肘、髋、膝等）。

The Role of OpenCV

OpenCV 提供了强大的工具集，用于处理YOLO输出的图像数据，并为后续的姿态估计算法提供输入，是实现整个视觉处理流程的基石。



步骤 1：目标定位 (YOLO)



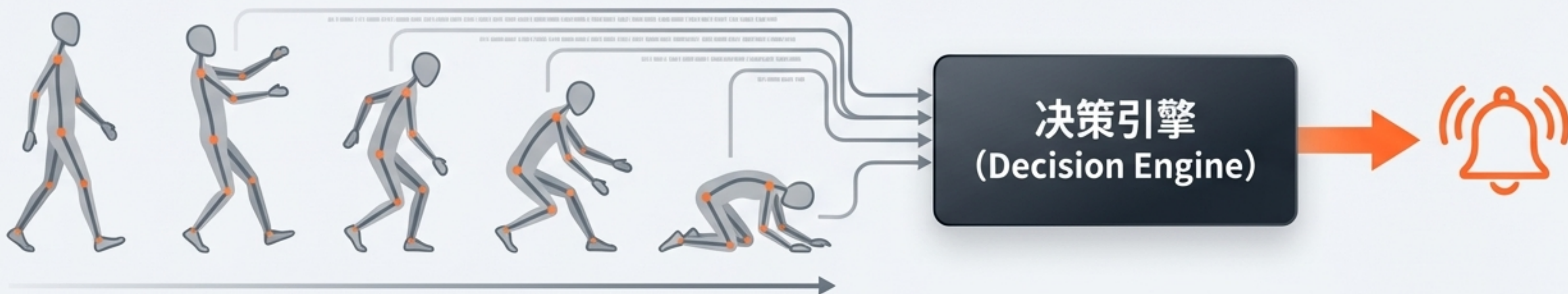
步骤 2：姿态估计

深度解析 Part 2

分析行为并判断状态

我们已经拥有了每一帧的人体姿态数据流。如何基于这些时序数据，构建一个稳定、可靠的逻辑引擎来判断“跌倒”这一复杂事件？

“跌倒”不是一个瞬时动作，而是一个包含了姿态、速度和位置剧烈变化的过程。我们需要一个能够捕捉这种动态过程的模型。



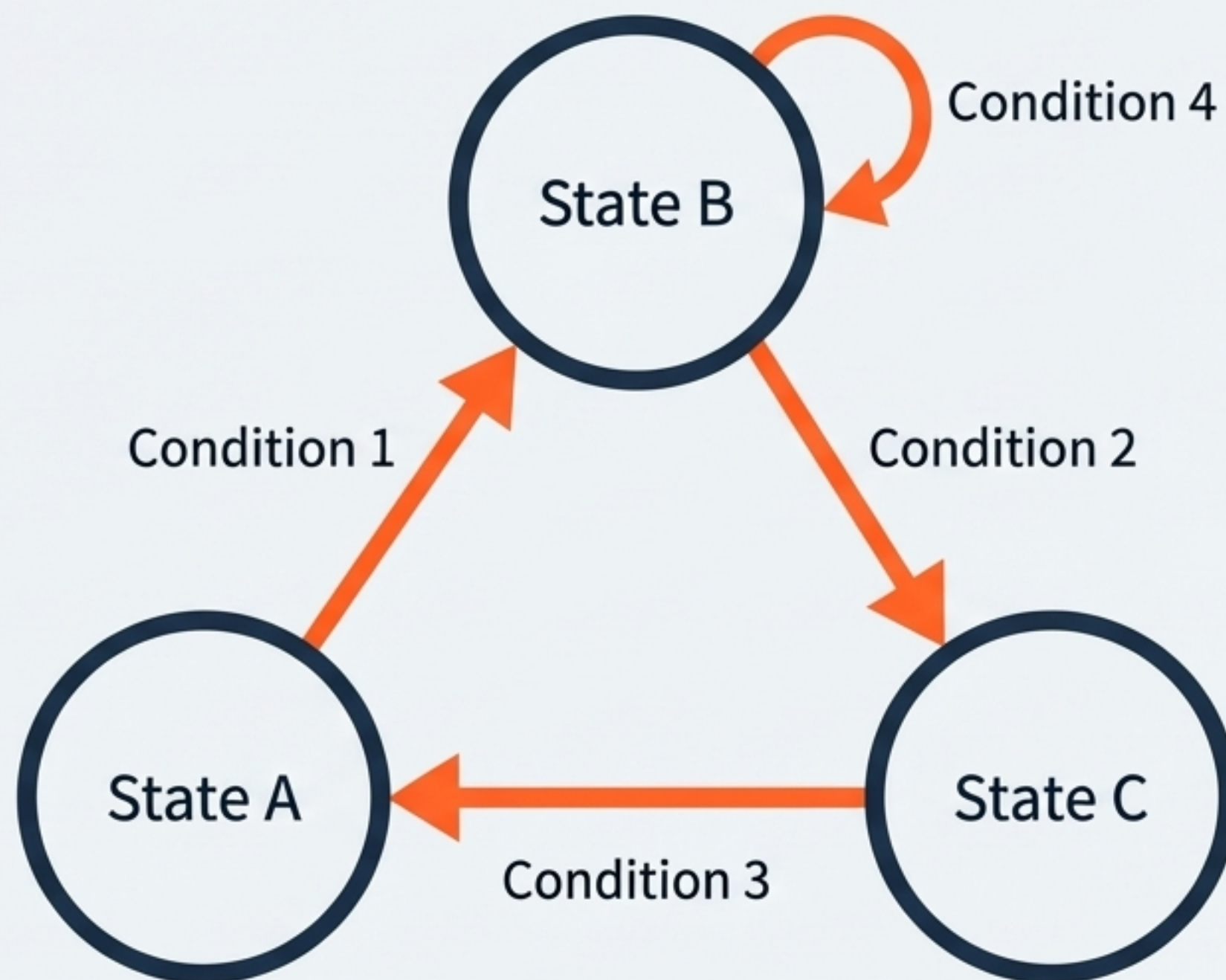
关键技术二：有限状态机（FSM）——系统的决策核心

What is an FSM?

有限状态机是一个数学模型，它定义了一系列有限的“状态”以及在这些状态之间转换的“规则”。系统在任何给定时间点都只能处于其中一个状态。

Why FSM for this task?

- **鲁棒性 (Robustness)**：FSM 的行为是可预测和确定的，能有效避免因瞬时数据抖动导致的误判。
- **可解释性 (Interpretability)**：状态（如站立、行走、倒地）和转换规则（如“髋部垂直速度过快”）非常直观，易于调试和优化。
- **高效性 (Efficiency)**：FSM 的计算开销极小，非常适合在边缘设备上运行。



状态机在行动：跌倒检测的逻辑流程



代码实现：将状态机逻辑转化为 Python 代码

使用 Python，我们可以清晰地定义状态和转换逻辑。以下代码片段展示了状态机中一个核心转换函数的简化示例。

```
# Simplified transition logic example
def check_fall_transition(current_pose, previous_pose, state):

    # Calculate vertical velocity of the hip keypoint
    hip_velocity = calculate_vertical_velocity(current_pose.hip, previous_pose.hip)

    # Check for transition from 'Normal' to 'Falling'
    if state == 'Normal' and hip_velocity > VELOCITY_THRESHOLD:
        return 'Falling' # Transition to the Falling state

    # ... other transition checks ...

    return state # No state change
```

核心指标：计算关键点的瞬时速度

可调参数：通过实验确定的关键阈值

状态转换：满足条件时，更新到新状态

全流程演示：从正常活动到触发警报



行走 (Walking)

当前状态：常规



跌倒中 (Falling)

当前状态：疑似跌倒



已倒地 (Fallen)

当前状态：倒地



警报 (Alert)

动作：触发远程警报

总结与展望

总结



我们构建了什么 (What We Built) :
一个端到端的、基于计算机视觉的自动化老人监护系统。



核心优势 (Core Advantages) : 结合了 YOLO 的实时检测能力和 FSM 的稳定决策逻辑,实现了高效、可靠的跌倒检测。



项目价值 (Value) : 为居家养老安全提供了一种低成本、高尊严、技术驱动的解决方案。

展望



功能扩展 (More Features) : 增加对其他异常行为的检测,如长时间静止、深夜徘徊等。



多模态融合 (Multi-modal Fusion) : 结合音频传感器 (如呼救声检测) 以提高系统准确性。



模型优化 (Model Optimization) : 持续优化模型,降低功耗,使其更易于部署在低成本的边缘计算设备上。

技术向善：AI赋能智慧养老新未来

