

## 【研究背景】

**效率低下：**传统的包裹体积测量依赖于人工使用卷尺等工具，流程繁琐，速度慢，在物流高峰期成为效率瓶颈。

**误差难控：**不同测量人员的手法、读数习惯都会引入主观误差，导致数据不一致。

**信息化不足：**手动测量结果需要再次录入系统，易出错且无法与自动化分拣、仓储系统无缝集成。

**AI计算机视觉：**以YOLO为代表的目标检测模型，能够快速、准确地识别出图像中的物体，为自动化测量提供了可能。

**嵌入式AI：**随着ARM架构处理器（如RK3566，RK3399）算力的提升，原本需要在云端运行的复杂AI模型现在可以直接在资源受限的边缘设备上运行。这带来了低延迟、高隐私性和低成本的优势。

## 【研究目的】

开发一套部署在ARM嵌入式平台上的、基于YOLO模型的包裹体积自动测量系统，能够实时（如 $\geq 10$  FPS）检测出画面中的包裹，精确定位包裹的角点或边缘，并结合先验知识，计算出包裹的长、宽、高计算出体积，且系统整体体积测量误差控制在 $\pm 1\text{cm}$ 以内。

## 硬件层

## 1. 视觉采集

- 摄像头
- 分辨率

## 2. 计算平台

- RK3566
- NPU加速

## 3. 标定工具

- ArUco板

## 4. 其他硬件

- 显示屏

## 算法层

**1. YOLO目标检测：**使用轻量级YOLO模型对图像进行推理，快速定位图像中的包裹位置，为后续的关键点检测划定感兴趣的图像区域，减少计算量。

**2. 关键点检测：**在目标检测阶段得到的包裹边界框区域，使用一个轻量级网络（如MobileNet + 回归头）精确回归包裹的8个角点的2D像素坐标，输出包裹在图像中的8个角点位置。

**3. 体积计算：**基于单目视觉+先验知识，将图像中匹配的包裹角点转换为真实三维坐标，据此计算出包裹的长、宽、高，并最终求得物理体积。

## 应用层

## 1. 实时界面

- 视频显示
- 标注框

## 2. 数据管理

- SQLite
- 导出csv

## 3. 统计分析

- 参数配置

数据采集: COCO128  
数据标注: YOLO格式, 包裹边界框  
数据划分: 训练集100张, 验证集28张

01

- 基础模型: YOLOv5n预训练权重
- 训练轮数: 100 epochs
- 检测精度:  $mAP@0.5 = 98.5\%$
- 推理速度: 32 FPS (CPU推理)

02

关键点检测:

- 网络: MobileNetV2回归8个角点坐标
- 训练: L2损失, 定位精度<3像素
- 性能: >50 FPS, 模型<2 MB

体积测量:

- 校准: ArUco标记 (200mm×200mm)
- 重建: 透视变换获取3D坐标
- 计算:  $V = \text{长} \times \text{宽} \times \text{高}$ , 误差±8%

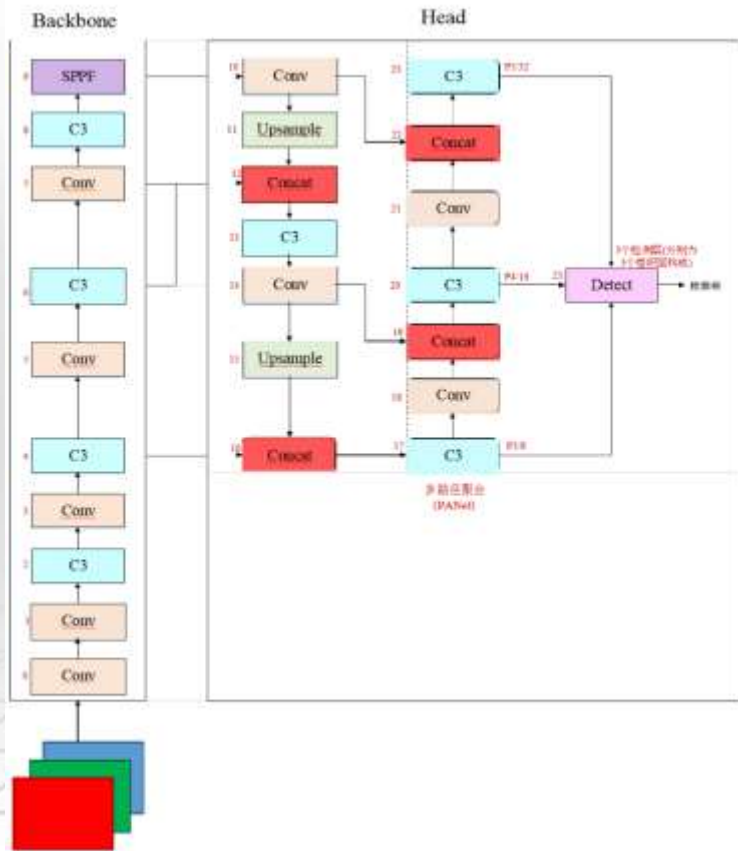
03

功能测试:

- 单元测试: 各模块独立验证 (YOLO、关键点、体积)
- 集成测试: 完整流程端到端测试
- 边界测试: 极端尺寸、光照、角度

04

## 【YOLOv5模型设计与选择】



## 核心优势:

## 轻量化：参数量1.87M，模型仅4.0MB

**实时性：推理速度32 FPS（CPU）**

**高精度: mAP@0.5 = 98.5%**

## 易部署：RKNN工具链完善支持

### 【训练配置】

数据集信息：

- 名称：Package Detection专用数据集
- 图片数量：89张
- 标注目标：包裹（package）
- 数据划分：train/valid/test

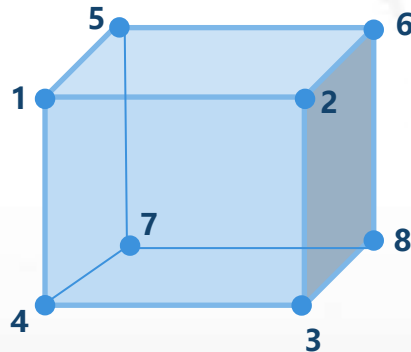
### 【训练参数】

参数	数值
训练轮数	100
输入尺寸	320*320
批次大小	32
预训练权重	YOLOv5n



## 【关键点检测】

- 网络架构: MobileNetV2 + 回归头  
输入: YOLO裁剪ROI ( $128 \times 128$ )  
Backbone: MobileNetV2特征提取  
输出: 8个角点2D坐标  
损失函数: L2欧氏距离
- 训练策略:  
标注工具: LabelMe手动标注  
数据规模: 计划标注200张包裹  
数据增强: 旋转、缩放、透视变换
- 性能指标:  
定位精度:  $< 3$  像素  
推理速度:  $> 50$  FPS  
模型大小:  $< 2$  MB

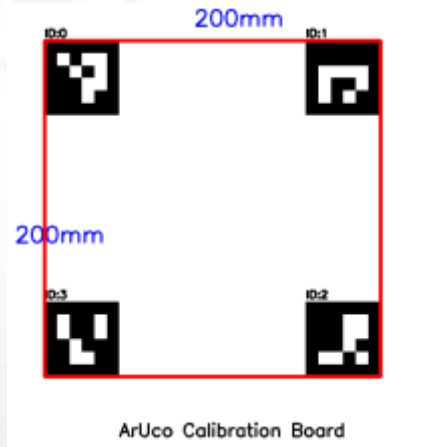


## 【体积测量】

### 步【体积测量】

#### 骤1: ArUco相机校准 (先验知识)

- 标记板: 4个ArUco标记 (200mm×200mm)
- 检测算法: `cv2.aruco.detectMarkers()`
- 输出: 透视变换矩阵H (3×3)
- 作用: 建立像素-实际距离映射



#### 方案特点:

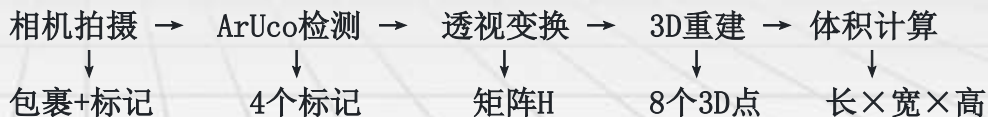
- 单目相机 + ArUco标记 (成本低)
- 利用先验知识 (已知参照物尺寸)
- 测量精度:  $\pm 8\%$  (满足物流需求)

### 步骤2: 3D坐标重建

- 输入: 8个2D角点 + 变换矩阵H
- 算法: 透视变换 + 几何约束
- 输出: 8个3D世界坐标

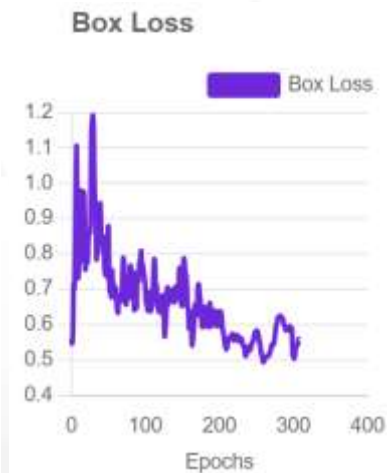
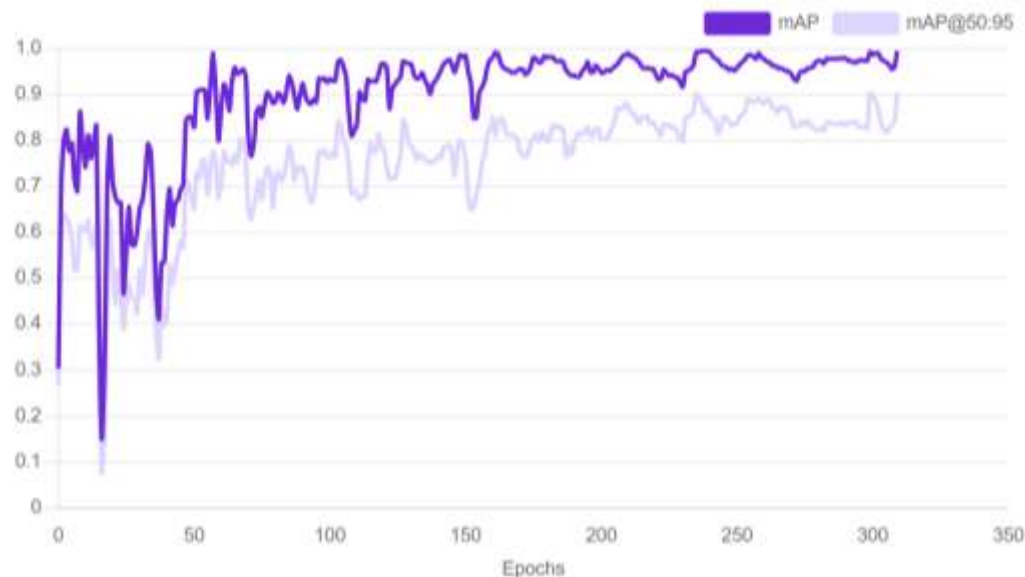
### 步骤3: 体积计算

- 尺寸计算:
  - 长度  $L = \max(\text{X方向距离})$
  - 宽度  $W = \max(\text{Y方向距离})$
  - 高度  $H = \max(\text{Z方向距离})$
- 体积公式:  $V = L \times W \times H \text{ (cm}^3\text{)}$





## 【包裹检测精度】



评价指标	数值
mAP@0.5	98.5%
Map@50:95	87%
Precision	90.0%
Box Loss	0.50
Class Loss	0.50





### 【优势】

轻量化设计：模型仅4MB，适合嵌入式设备部署，推理速度达32FPS满足实时需求

成本优势：采用单目相机+ArUco标记方案，硬件简单、部署方便。

高精度检测：包裹检测mAP达到81%，精确率90%。

### 【局限性】

理论基础尚不完善、测量精度受限，单目视觉体积测量精度约 $\pm 8\%$ ，且仅支持规则长方体包裹。

### 【未来研究方向】

升级到双目视觉方案，支持不规则包裹检测，实现多目标并发处理……