

# 先进视觉-3D 识别项目技术报告

学校：西安交通大学 队名：XJTUfly3

参加学生：黄隆宁 雷博书 周靖淦 刘昭宇 杜佩珊 胡思敏

指导教师：武彤晖

## 一. 测试用笔记本电脑配置说明：

本队所使用的笔记本电脑已经无法在京东或者天猫商城找到，所以我们列举了所使用版本和更高配置版本的参数对比，以及更高配置版本的京东购买页面截图。

	测试用笔记本	最新产品（京东）
型号	戴尔 灵越 7560(已停产)	戴尔 灵越 8550
CPU	Intel i7-7500	Intel i7-8550
内存	8G	16G
硬盘	512G 固态+1T 机械	512G 固态+1T 机械
显卡	NVIDIA GeForce 940MX	MX150
价格	/	6399 元





关注 分享 对比 举报

**戴尔 (DELL) 灵越7572-3745燃7000 15.6英寸窄边框独显超薄笔记本电脑 金色:16G/512G固态+1T定制**

【新一代燃7000】8代i7四核,NVME固态,满血版MX150,4G独显我想选14英寸i7

京东价 **¥6399.00** 降价通知

优惠券 **满2000减100**

促销 **赠品**  x1  x1  x1  x1 (条件: 购买1件及以上, 赠完即止)

**加价购** 满2000元以折扣价在购物车换购热销商品, 包邮 (限中国内地)

累计评价 **4100+**

增值业务 **高价回收, 享补贴**

配送至 **陕西西安市高陵区县城内** 有货 免运费

由 **戴尔卓越专卖店** 从 湖北武汉市 发货, 并提供售后服务. 今日12:00前完成下单, 预计8月15日19:00前发货, 8月18日24:00前送达

服务支持 **放心购** **送运费险** **极速审核**

货到付款

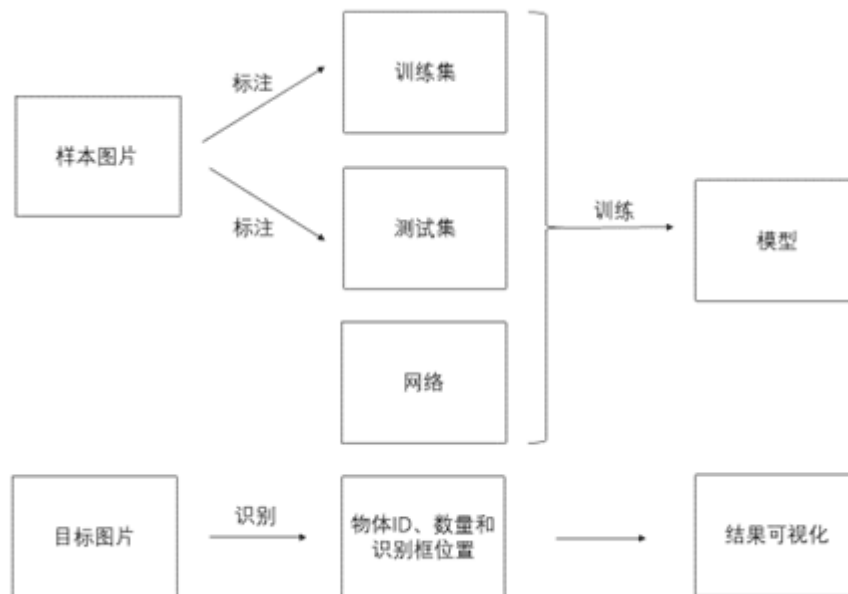
选择颜色

 银色: 8G/128G固态+1T 标配	 银色: 16G/512G固态+1T定制
 银色: 8G/256G固态+1T 定制	 银色: 16G/128G固态+1T定制
 金色: 16G/128G固态+1T定制	 金色: 16G/256G固态+1T定制
 金色: 8G/128G固态+1T 标配	 金色:16G/512G固态+1T定制
 金色 :8G/256G固态+1T 定制	 银色:16G/256G固态+1T定制

最新产品价格截图

## 二. 视觉软件处理流程

1. 可视化界面初始化, 打开摄像头。
2. 加载网络, 并读取训练好的权值参数。
3. 开始识别之后, 拍摄图片并传入模型识别。
4. 将识别到的 ID、数量、位置等结果可视化显示。



软件处理流程

### 三. 视觉软件界面及功能说明

视觉处理软件主要使用 Python 编写完成。实现视觉处理的过程中，为了得到测试样品的彩色及深度图片，我们使用 pyrealsense 库调用 Intel Realsense 深度摄像头的相关接口获得；识别检测部分，我们使用 pytorch 及其辅助模块，在 yolo-v3 的基础上完善改进训练得到所使用的模型；最终通过 tkinter 模块实现实时捕获图片和检测结果的可视化。

#### 软件界面

界面设计：tkinter 模块在 Windows10 系统上实现，左侧为相机拍摄画面实时显示区域，右侧为识别结果输出区域。

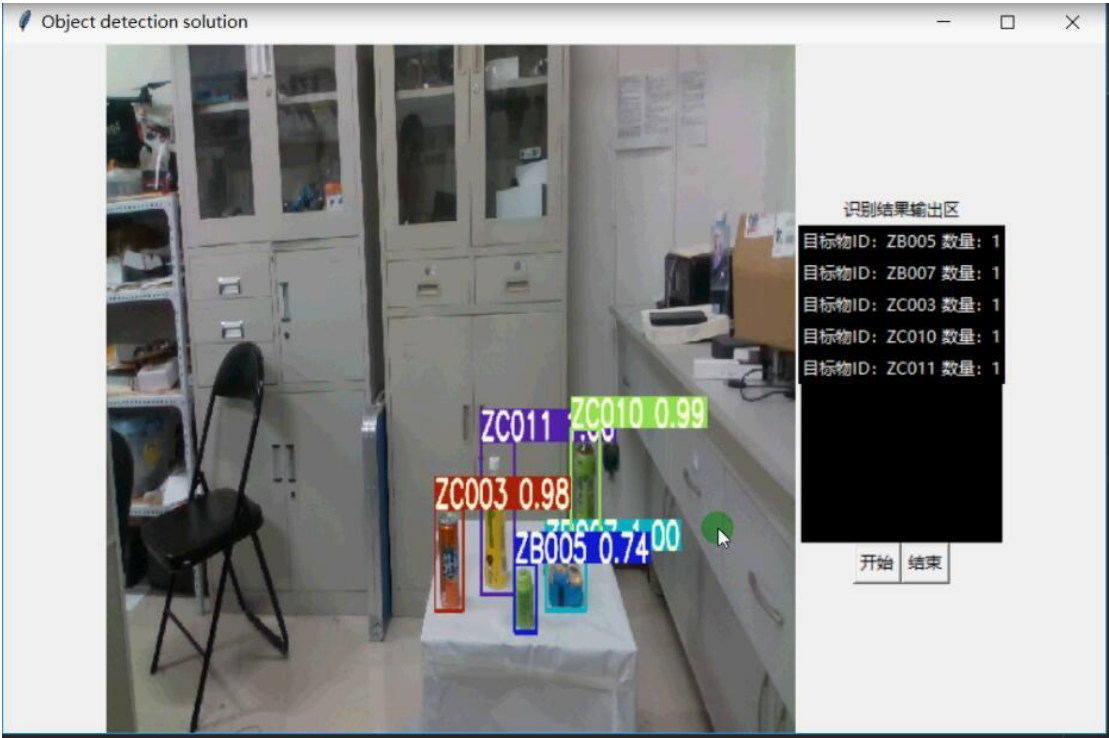
主要功能：

程序开始执行之后，图像界面初始化如下。点击开始按钮即可进行识别，识别过程中，左侧实时显示相机拍摄图片及识别结果，右侧显示

识别物体种类及各自数量。点击结束按钮则识别过程结束。



初始图像界面

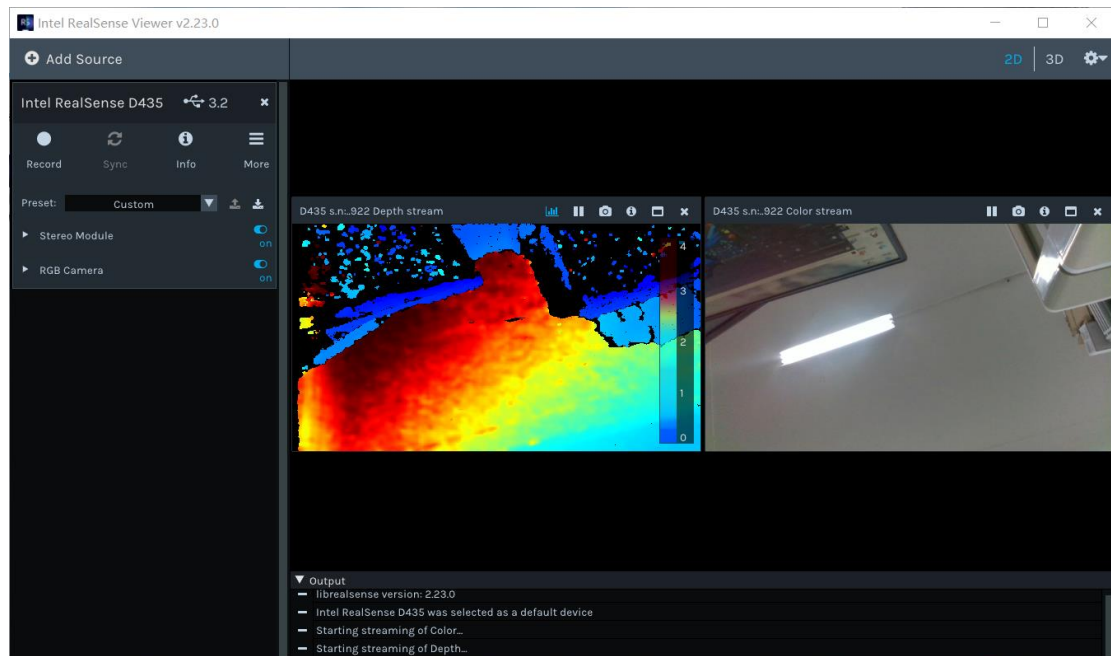


检测效果图

Realsense:

基于 Windows 10 和 vs2017 完成的

主要功能：使用过程中将摄像机和电脑相连接，之后在软件上直接进行相关操作。



Realsense 界面

在网络算法编译过程当中,使用到的库文件 numpy、matplotlib、time。

## 四. 主要算法

### 数据集制作

在不同相机的角度、不同光线强度、不同背景的情况下，共拍摄 8000 多张照片来制作数据集，其中包含仅有一个物体、多个物体、10 个以上物体、有遮挡物体在标定纸上的照片，使用开源软件 labelImg 制作类似于 coco 的数据集，同时制作相对应的标签文件。然后采用 opencv 将图片读入，并对颜色，位置，旋转角度三个方面做了数据扩展。输入到神经网络中。

### 模型训练

采用 pytorch 的 dataloader 加载输入数据。采用 SGD 优化器。设

置 scheduler，在训练到总进度的 0.8，0.9 时将学习率降低到之前的 0.1 倍。采用学习率为 0.001。先设置所有的参数都可以优化。在进行到 1500 批后，每次随机固定一半的参数进行优化。并降低学习率到 0.0001。

## 模型的导出

为了能够使用训练好的模型，需要对模型训练出来的 checkpoint 文件进行整合导出为.pt 文件，该步骤亦使用了 pytorch 所自带的 api 进行导出。

## 模型加载

得到模型的 pt 和标签文件之后，就可以直接使用 api 对输入的图片进行预测，得到类别、概率、检测框位置等信息。

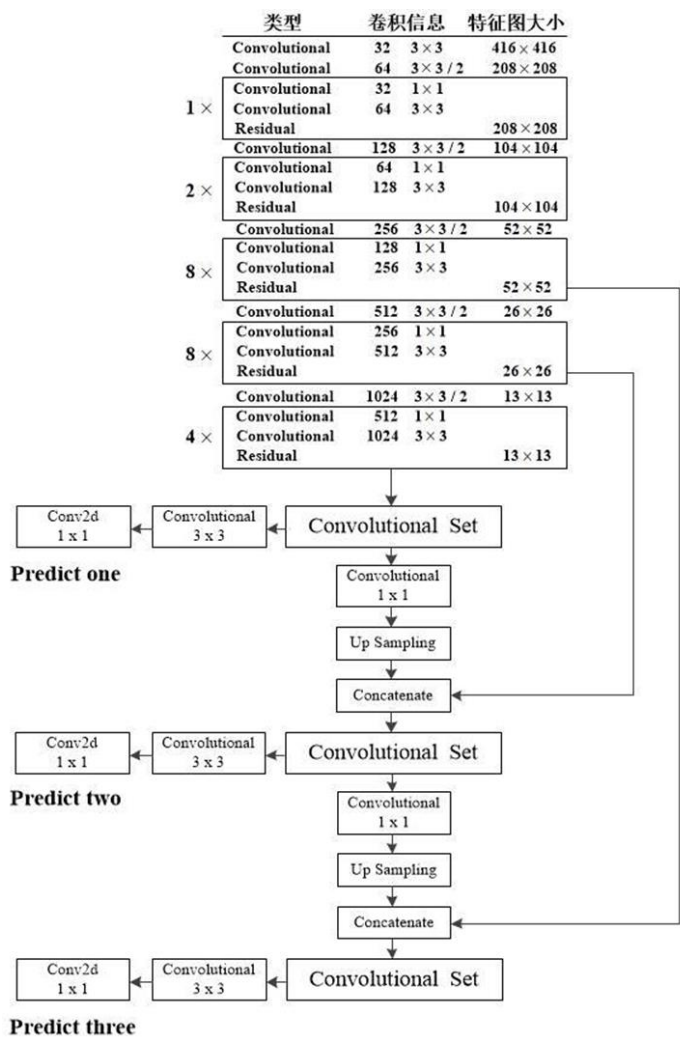
## 基于深度学习的三维图像分类与定位算法

使用 yolov3 算法对 3D 图像进行了图像分类及物体检测。过程中我们对不同类待测样本进行了不同角度不同方位的多样采集。为了训练出的模型更为准确，在采样过程中对物体进行一定程度的遮挡。将数据集进行手动标定，输入 yolov3 的训练器，经过 epoch 训练之后，训练后的参数导出。

## 彩色图与深度图处理--yolov3 网络结构

比赛采用两个网络结构, 分别处理彩色图与深度图。彩色图的处理采用 yolov3 网络结构 Yolo 层设置的 anchor box 的比值为 :10, 13 16, 30 33, 23 30, 61 62, 45 59, 119 116, 90 156, 198 373, 326 之后, 在深度图上截取获得由 yolov3 网络预测的区域。将此伪彩色图转换

到 HSV 空间，并提取 H 通道上的数值，输入到下一个网络中。下一层网络由三层卷积层与一层全连接层构成，负责检测该区域中是否真正存在物体，或是由于彩色图导致前一个网络误判。在保持图像中比例尺寸不变的情况下，将图像放缩到 50, 50 输入到网络中，得到真正存在的概率。并将概率大于一定阈值的结果输出。最后，综合输出得到的分类，边界框。



yolov3 网络结构示意图

### 实物和贴图区分问题的解决方法

针对实物和贴图问题，采用两种解决方法：贴图拟物法，传统深度图法。贴图拟物法：将贴图同样看做实物，然后根据神经网络自身识别

能力将实物与贴图区分开，这一方法可解决一部分区分问题。传统深度图法：获取识别对象各点的深度值，根据实物与贴图的深度不同，从而来判断识别对象为实物或为贴图。将上述两种方法相结合处理实物和贴图区分问题，从而准确识别解决问题。

### **运动物体识别的解决方法**

针对运动物体识别问题，采用如下解决方法：1. 提升识别的速度。提升单次识别速度，缩短单次识别时间，从而准确识别到运动物体；2. 置信度法。识别对象不同区域的置信度不同，可根据置信度的不同识别出运动中的物体；3. 众数、平均数法。将测试结果取众数或平均数得最终结果，将结果和实物相对比，从而识别出运动中的物体。