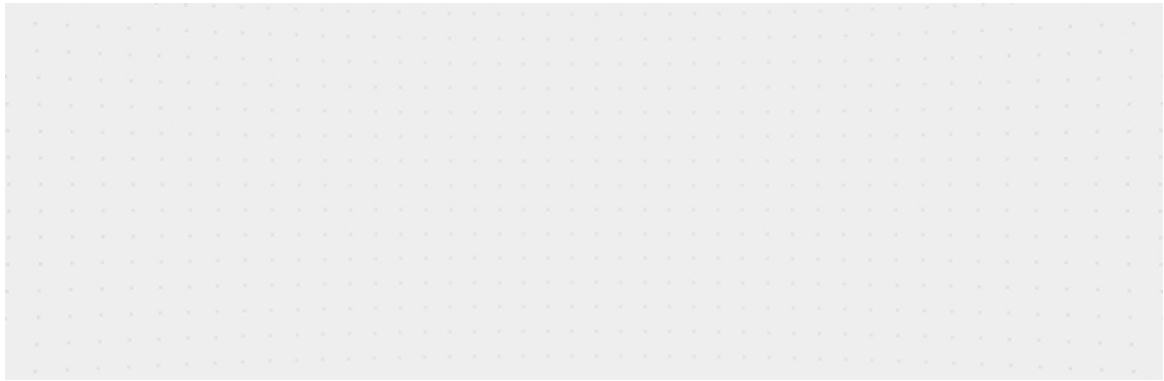


厨师工作服/工作帽识别冠军方案 | PRCV2022

原创 CV开发者都爱看的 极市平台 2022-11-04 22:00:51 发表于广东 手机阅读 𠄎

↑ 点击蓝字 关注极市平台



作者 | 杨新

编辑 | 极市平台

极市导读

PRCV2022厨师工作服/工作帽识别的冠军方案解读，作者展示了其模型思路全过程。 >>加入极市CV技术交流群，走在计算机视觉的最前沿

任务介绍：随着食品安全问题的日益严峻，后厨的卫生也成为了当下人们关心的问题。后厨人员的卫生穿戴习惯直接影响到食品的卫生品质，因此规范后厨穿着逐渐成为后厨人员卫生管理重要一环。算法应用场景：厨房监控摄像头取像，识别厨房人员的穿着情况，识别到厨房人员没有识别对象，进行告警。



数据集：



训练集：17725个 测试集：7776个 目标类别数目：31类

本赛题采用的评审指标如下：

评审指标指标		
说明权重分		
算法精度	F-score	85
算法性能值	性能分=fps/100, 如果 fps>100 则当满分 100 计算	15
总分	Score=算法精度*85+算法性能值*15	

注意：

- 1.PRCV赛题中对厨师工作服/工作帽识别模型榜，有一个最低分数要求，成绩分需达到0.9
- 2.本赛题最终得分采取准确率、算法性能绝对值综合得分的形式，其中IoU使用0.4

解决方案：

算法选择：

Method	Input Size	AP ^{val}	AP ^{val} ₅₀	FPS (bs=1)	FPS (bs=32)	Latency (bs=1)	Params	FLOPs
YOLOv5-N [10]	640	28.0%	45.7%	602	735	1.7 ms	1.9 M	4.5 G
YOLOv5-S [10]	640	37.4%	56.8%	376	444	2.7 ms	7.2 M	16.5 G
YOLOv5-M [10]	640	45.4%	64.1%	182	209	5.5 ms	21.2 M	49.0 G
YOLOv5-L [10]	640	49.0%	67.3%	113	126	8.8 ms	46.5 M	109.1 G
YOLOX-Tiny [7]	416	32.8%	50.3%*	717	1143	1.4 ms	5.1 M	6.5 G
YOLOX-S [7]	640	40.5%	59.3%*	333	396	3.0 ms	9.0 M	26.8 G
YOLOX-M [7]	640	46.9%	65.6%*	155	179	6.4 ms	25.3 M	73.8 G
YOLOX-L [7]	640	49.7%	68.0%*	94	103	10.6 ms	54.2 M	155.6 G
PPYOLOE-S [45]	640	43.1%	59.6%	327	419	3.1 ms	7.9 M	17.4 G
PPYOLOE-M [45]	640	49.0%	65.9%	152	189	6.6 ms	23.4 M	49.9 G
PPYOLOE-L [45]	640	51.4%	68.6%	101	127	10.1 ms	52.2 M	110.1 G
YOLOv7-Tiny [42]	416	33.3%*	49.9%*	787	1196	1.3 ms	6.2 M	5.8 G
YOLOv7-Tiny [42]	640	37.4%*	55.2%*	424	519	2.4 ms	6.2 M	13.7 G*
YOLOv7 [42]	640	51.2%	69.7%	110	122	9.0 ms	36.9 M	104.7 G
YOLOv6-N	640	35.9%	51.2%	802	1234	1.2 ms	4.3 M	11.1 G
YOLOv6-T	640	40.3%	56.6%	449	659	2.2 ms	15.0 M	36.7 G
YOLOv6-S	640	43.5%	60.4%	358	495	2.8 ms	17.2 M	44.2 G
YOLOv6-M [‡]	640	49.5%	66.8%	179	233	5.6 ms	34.3 M	82.2 G
YOLOv6-L-ReLU [‡]	640	51.7%	69.2%	113	149	8.8 ms	58.5 M	144.0 G
YOLOv6-L [‡]	640	52.5%	70.0%	98	121	10.2 ms	58.5 M	144.0 G

上图为yolov6文章中的几个yolo算法的性能指标比对，yolov6主要基于repvgg结构进行改进，以及采用anchorfree方式，笔者最开始使用的是yolov5m模型进行训练，速度可以达到100fps，但是调整训练和测试参数都无法达到赛题中要求的0.9的基准值，还是模型精度不够好，对比看yolov5-m和yolov6-m的性能指标，几乎同样的推演延时情况下，yolov6-m的mAP值比yolov5-L的精度还要高，因此笔者最终采用的是yolov6-m进行训练。

模型训练：

yolov6在代码结构上数据集上以及训练方法上有很多都和yolov5很类似。

数据集的生成：

由于笔者最开始使用的是yolov5进行训练测试的，yolov6的数据生成都是和之前yolov5一样的，参考了<https://cvmart.net/document>其中《基于YOLOV5的新手任务实践》中的步骤，训练集和测试集的比例采用的是9:1的比例。

数据配置文件如下：

```
train: /project/train/src_repo/dataset/images/train
val: /project/train/src_repo/dataset/images/val

nc: 31 # number of classes
names: ["person", "short_sleeve_red", "short_sleeve_black", "short_sleeve_white", "short_sleeve_blue", "short_sleeve_dark_blue", "long_sleeve_red", "long_sleeve_black", "long_sleeve_blue", "non_uniform", "other_uniform", "chef_hat_red", "chef_hat_black", "chef_hat_white", "peaked_cap_beige", "disposable_cap_white", "disposable_cap_blue", "head", "other_hat", "c
```

模型配置文件如下：

```
YOLOv6m model
model = dict(
  type='YOLOv6m',
  pretrained='yolov6m.pt',
  depth_multiple=0.60,
  width_multiple=0.75,
  backbone=dict(
    type='CSPBepBackbone',
    num_repeats=[1, 6, 12, 18, 6],
    out_channels=[64, 128, 256, 512, 1024],
    csp_e=float(2)/3,
```

```
    ),
    neck=dict(
        type='CSPRepPANNeck',
        num_repeats=[12, 12, 12, 12],
        out_channels=[256, 128, 128, 256, 256, 512],
        csp_e=float(2)/3,
    ),
    head=dict(
        type='EffiDeHead',
        in_channels=[128, 256, 512],
        num_layers=3,
        begin_indices=24,
        anchors=1,
        out_indices=[17, 20, 23],
        strides=[8, 16, 32],
        iou_type='giou',
        use_dfl=True,
        reg_max=16, #if use_dfl is False, please set reg_max to 0
        distill_weight={
            'class': 1.0,
            'dfl': 1.0,
        },
    )
)

solver=dict(
    optim='SGD',
    lr_scheduler='Cosine',
    lr0=0.01,
    lrf=0.01,
    momentum=0.937,
    weight_decay=0.0005,
    warmup_epochs=3.0,
    warmup_momentum=0.8,
    warmup_bias_lr=0.1
)

data_aug = dict(
    hsv_h=0.015,
    hsv_s=0.7,
    hsv_v=0.4,
    degrees=0.0,
    translate=0.1,
    scale=0.9,
    shear=0.0,
    flipud=0.0,
    fliplr=0.5,
    mosaic=1.0,
    mixup=0.1,
)
```


训练代码修改：

由于yolov6训练的模型太大，平台上最多只能存1G的模型，在代码engine.py中将self.tblogger = None 以及tblogger相关的屏蔽掉防止其占用太多的空间。

笔者采用yolov6训练的时候到截止提交时间没剩几天，将最大epochs改为了200，这里有可能使用默认的400效果更好。训练命令如下：

```
python /project/train/src_repo/YOLOv6/tools/train.py
--data /project/train/src_repo/yolov5/data/object.yaml
--img 640
--conf configs/yolov6m.py
--epochs 200
--output-dir /project/train/models/train \
--batch 16
```

推理测试：

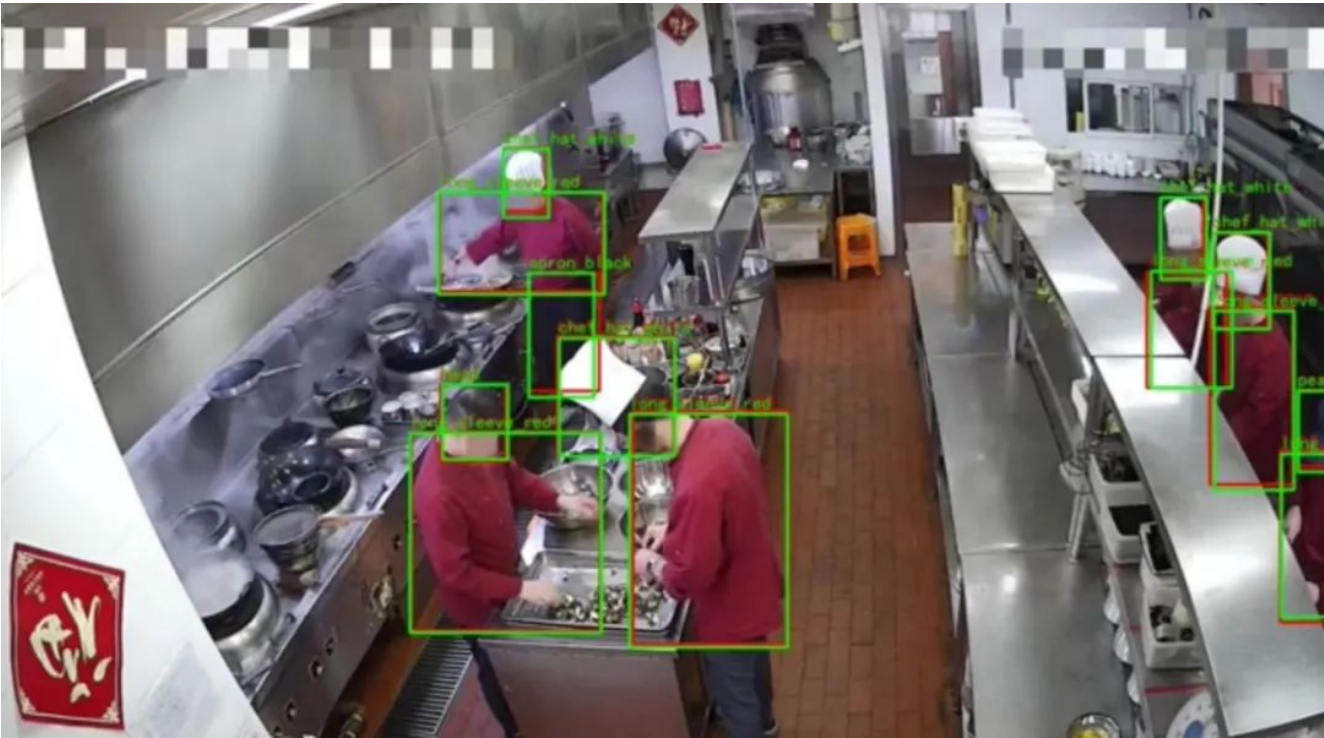
推理代码主要基于的是极市封装程序的代码https://github.com/ExtremeMart/ev_sdk_demo 4.0_pedestrian_intrusion_yolov5

为了方便进行测试，这里在cpp代码中增加python代码用于方便导出onnx模型

```
std::system("cd /project/train/src_repo/YOLOv6 && python /project/train/src_repo/YOLOv6/deploy/ONNX/export_onnx.py --weights /project/train/models/train/exp/weights/best_ckpt.pt --simplify --inplace --img-size 640 640"),
```

笔者使用这种方法在测试代码中直接导出onnx，后面的部分ev_sdk中包含使用onnx转换为tensorrt的代码。前面的测评标准中有写到使用IOU 0.4，因此在后处理做非极大抑制的时候将阈值改成了0.4，这个地方多次测试对比会对结果有影响。

推理效果：



测试结果：

算法类型	f-score	性能分	成绩分
yolov5-m	0.8791	116.6498	0.8972
yolov6-m	0.8863	104.632	0.9034

总结：

- 1.基于yolov6-m进行训练，在同样推理速度的情况下比yolov5-m的精度要好，这里模型的选择很重要。
- 2.注意测评标准中有提到IOU用的是0.4，因此我们在推演的时候也使用的是0.4的IOU进行测试，这里对精度也有影响。
- 3.基于tensorrt fp16进行推演，同时为了要满足推演速度，需要调整合适的尺寸满足这个条件。

<https://github.com/ultralytics/yolov5>

<https://github.com/meituan/YOLOv6>

https://github.com/ExtremeMart/ev_sdk_demo4.0_pedestrian_intrusion_yolov5

公众号后台回复“直播”获取极市直播系列PPT下载~



2022
高通人工智能创新应用大赛

50W 15W+ 3大
总奖金池 脱敏数据 真实场景赛题

免费算力 码上开发

TIPS
若获奖团队并注册落户金牛区，更有机会获得政府10万元现金补助（与赛题奖项可重复获得哦）



极市平台

为计算机视觉开发者提供全流程算法开发训练平台，以及大咖技术分享、社区交流、竞...
848篇原创内容

公众号

△点击卡片关注极市平台，获取最新CV干货

极市干货

算法竞赛：往届获奖方案总结以及经验详解 | ACCV2022国际细粒度图像分析挑战赛

技术综述：BEV 学术界和工业界方案、优化方法与tricks综述 | PyTorch下的可视化工具（网络结构/训练过程可视化）

极视角动态：极视角与华为联合发布基于昇腾AI的「AICE赋能行业解决方案」 | 算法误报怎么办？自训练工具使得算法迭代效率提升50%！

CV技术社群邀请函

△长按添加极市小助手

添加极市小助手微信 (ID : cvmart2)

备注：姓名-学校/公司-研究方向-城市（如：小极-北大-目标检测-深圳）

即可申请加入极市目标检测/图像分割/工业检测/人脸/医学影像/3D/SLAM/自动驾驶/超分辨率/姿态估计/ReID/GAN/图像增强/OCR/视频理解等技术交流群

极市&深大CV技术交流群已创建，欢迎深大校友加入，在群内自由交流学术心得，分享学术讯息，共建良好的技术交流氛围。

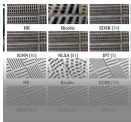
// 点击阅读原文进入CV社区
收获更多技术干货

阅读原文 文章已于2022-11-07修改

喜欢此内容的人还喜欢

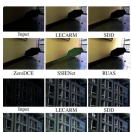
ICCV 2023 | 南开程明明团队提出适用于SR任务的新颖注意力机制（已开源）

极市平台



ICCV23 | 将隐式神经表征用于低光增强，北大张健团队提出NeRCo

极市平台



YOLOv5帮助母猪产仔？南京农业大学研发母猪产仔检测模型并部署到Jetson Nano开发板



