

OpenCV4 DNN对象检测

- 贾志刚

内容概述

- DNN模型介绍
- 支持的对象检测模型
- 通用对象检测/自定义对象检测
- 人脸检测

开发环境搭建

- OpenVINO2022.x
- 开发IDE: VS2017
- OpenCV4.x

学习前提与基础

- 理论加实践!
- 实践是最好的理论!
- CV=OpenCV+OpenVINO+Pytorch
- 语言C++/Python
- <https://www.bilibili.com/video/BV1i54y1m7tw>

DNN模块介绍

- DNN-Deep Neural Network
- OpenCV3.3 开始发布
- 支持VOC与COCO数据集的对象检测模型
- 包括SSD/Faster-RCNN/YOLOv4等
- 支持自定义对象检测
- 支持人脸检测
- **请别在用HAAR人脸检测**

对象检测模型支持

- 支持SSD/Faster-RCNN
- 支持YOLOv4
- 模型的输入与输出，格式解析文件
- models.yml
- 内置人脸检测模型支持

DNN支持框架

- DNN模块来源-tinydnn
- 支持tensorflow/caffe/onnx模型
- 支持OpenVINO加速
- 缺点：支持的模型数量有限/类型有限

函数知识

- 读取模型
- 转换数据与设置
- 推理输出

```
Net net = readNetFromTensorflow(model, config); // 支持tensorflow
```

```
Net net = readNetFromCaffe(config, model); // 支持caffe
```

```
Net net = readNetFromONNX(onnxfile);
```

```
// 读取数据
```

```
Mat image = imread("D:/images/example.png");
```

```
Mat blob_img = blobFromImage(image, scalefactor, size, mean, swapRB);
```

```
net.setInput(blob_img);
```

```
// 推理输出
```

```
Mat result = net.forward();
```


后处理/输出解析

- 不同网络的输出不同
- 如何解析要根据模型输出
- 对象检测网络SSD/Faster-RCNN解析

SSD的输出解析

- http://download.tensorflow.org/models/object_detection/ssd_mobilenet_v1_coco_2017_11_17.tar.gz
- 1x1xNx7 – DetectOutput
- [image_id, label, conf, x_min, y_min, x_max, y_max]

管理员: 命令提示符

Microsoft Windows [版本 10.0.16299.15]

(c) 2017 Microsoft Corporation。保留所有权利。

C:\Users\Administrator>cd /d D:\opencv-4.5.1\opencv\sources\samples\dnn

D:\opencv-4.5.1\opencv\sources\samples\dnn>tf_text_graph_ssd.py

usage: tf_text_graph_ssd.py [-h] --input INPUT --output OUTPUT --config CONFIG

tf_text_graph_ssd.py: error: the following arguments are required: --input, --output, --config

D:\opencv-4.5.1\opencv\sources\samples\dnn>tf_text_graph_ssd.py --input D:\tensorflow\ssd_mobilenet_v2_coco_2018_03_29\frozen_inference_graph.pb --output D:\tensorflow\ssd_mobilenet_v2_coco_2018_03_29\frozen_inference_graph.pbtxt --config D:\tensorflow\ssd_mobilenet_v2_coco_2018_03_29\pipeline.config

Scale: [0.200000-0.950000]

Aspect ratios: [1.0, 2.0, 0.5, 3.0, 0.333299994469]

Reduce boxes in the lowest layer: True

Number of classes: 90

Number of layers: 6

box predictor: convolutional

Input image size: 300x300

D:\opencv-4.5.1\opencv\sources\samples\dnn>_

Faster-RCNN输出解析

- http://download.tensorflow.org/models/object_detection/faster_rcnn_inception_v2_coco_2018_01_28.tar.gz
- 1x1xNx7 – DetectOutput
- [image_id, label, conf, x_min, y_min, x_max, y_max]

Microsoft Windows [版本 10.0.16299.15]

(c) 2017 Microsoft Corporation。保留所有权利。

C:\Users\Administrator>cd /d D:\opencv-4.5.1\opencv\sources\samples\dnn

D:\opencv-4.5.1\opencv\sources\samples\dnn>tf_text_graph_faster_rcnn.py

usage: tf_text_graph_faster_rcnn.py [-h] --input INPUT --output OUTPUT
--config CONFIG

tf_text_graph_faster_rcnn.py: error: the following arguments are required: --input, --output, --config

D:\opencv-4.5.1\opencv\sources\samples\dnn>tf_text_graph_faster_rcnn.py --input D:\tensorflow\faster_rcnn_resnet50_coco_2018_01_28\frozen_inference_graph.pb --output D:\tensorflow\faster_rcnn_resnet50_coco_2018_01_28\frozen_inference_graph.pbtxt --config D:\tensorflow\faster_rcnn_resnet50_coco_2018_01_28\pipeline.config

Number of classes: 90

Scales: [0.25, 0.5, 1.0, 2.0]

Aspect ratios: [0.5, 1.0, 2.0]

Width stride: 16.000000

Height stride: 16.000000

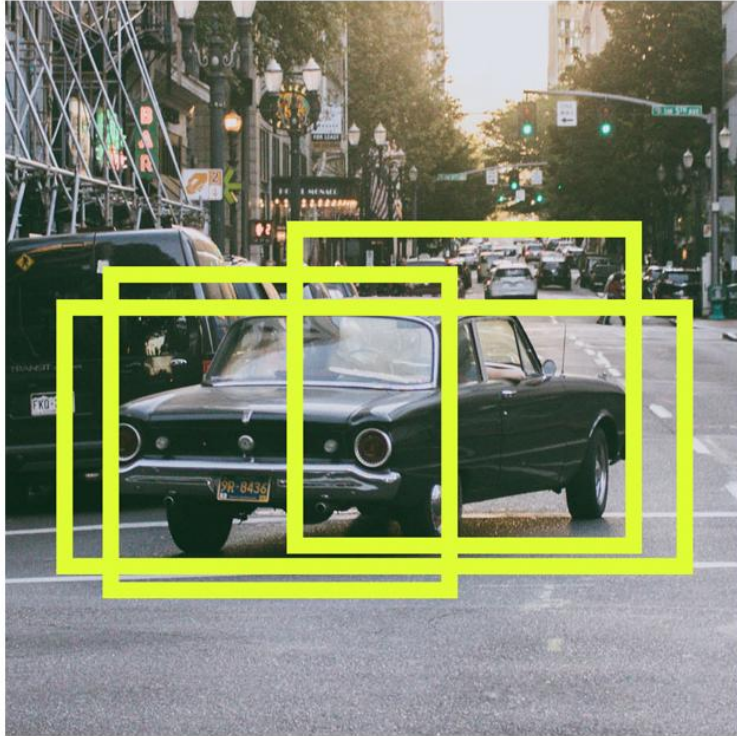
Features stride: 16.000000

D:\opencv-4.5.1\opencv\sources\samples\dnn>_

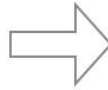
YOLOv4输出解析

- 解析多个输出层, 80个类别-Nxwxhxd
- 4+80 预测, 三个输出层
- centerx, centery, width, height
- $13 \times 13 \times 255 = 13 \times 13 \times 3 \times 85$
- NMS
- <https://github.com/AlexeyAB/darknet/wiki/YOLOv4-model-zoo>

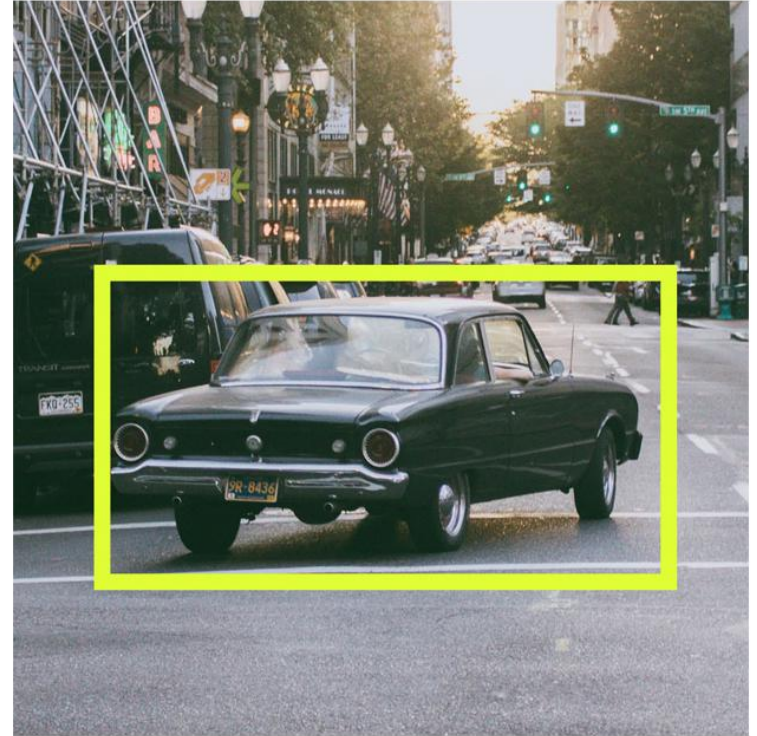
Before non-max suppression



**Non-Max
Suppression**

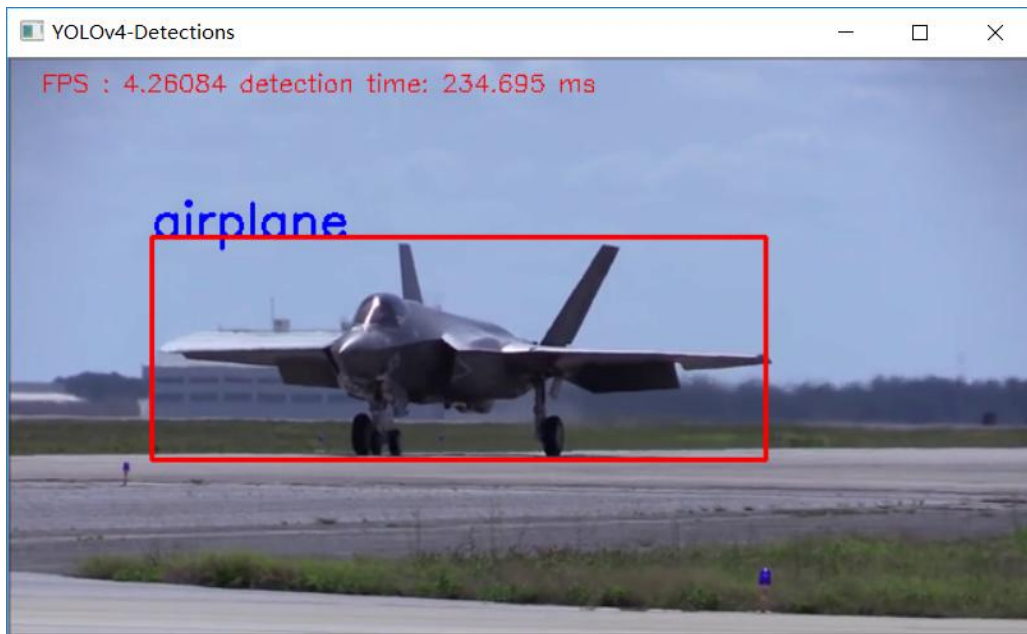


After non-max suppression



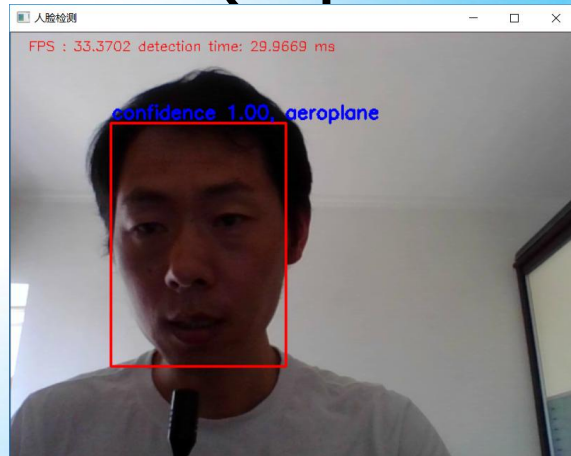
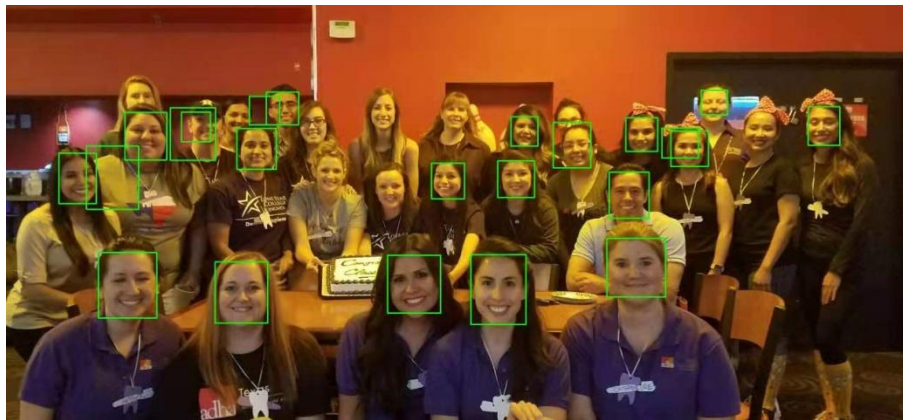
YOLOv4对象检测

- OpenCV YOLOv4对象检测



人脸检测

- 人脸检测模型SSD，支持tf与caffe
- 实时检测
- 模型加速OpenCV DNN+IE(OpenVINO)



YOLOv5推理

INPUTS

images

name: **images**

tensor: float32[1,3,640,640]

OUTPUTS

output0

name: **output0**

tensor: float32[1,25200,85]

- 推理时只需要关注模型的输入与输出
- 输入格式: RGB, 0~1
- 输出格式: 25200x85, 5+80, 其中 cx,cy,w,h,conf, 80类别

YOLOv8推理

INPUTS

images

name: **images**

tensor: float32[1,3,640,640]

OUTPUTS

output0

name: **output0**

tensor: float32[1,84,8400]

- 推理时只需要关注模型的输入与输出
- 输入格式: BGR, 0~1
- 输出格式: 84x8400, 4+80, 其中 cx,cy,w,h 80类别

ANY QUESTION?

