

模型训练代码完全来源于K210_Yolo_framework, 为了能在 win 上训练模型, 你需要简单修改它的代码, 我已经完成修改可以在 win 上运行

你可以直接使用我修改后的代码Yolo-for-k210, 或者你尝试使用原来的代码自行修改

@版权所有->刘威

LICENSE: MIT License

个人博客: <https://seasky-master.github.io/SEASKY-Master/>

K210 YOLO V3 框架

这是一个清晰的、可扩展的 yolo v3 框架。

1. Real-time display recall and precision
2. Easy to use with other datasets
3. Support multiple model backbones and expand more
4. Support n number of output layers and m anchors
5. Support model weight pruning
6. Portable model to kendryte K210 chip

VOC 数据集训练-开发环境

原作者在 ubuntu 18.04 - Python 3.7.1 中进行测试, 本人尝试可以在 windows 正常训练, 你需要安装 requirements.txt 中的内容 优先安装 tensorflow-gpu==1.15.0, 如果你的电脑不支持 GPU 版本, 请安装 tensorflow==1.15.0

```
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'tensorflow' is not defined
>>> import tensorflow as tf
2020-07-05 21:31:58.414910: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library cudart64_100.dll
>>> tf.__version__
'1.15.2'
>>> -
```

请在 tensorflow 环境搭建完成后继续向下操作, tensorflow 环境搭建参见[百度](#)

然后使用 `pip install -r requirements.txt` 安装其他工具

准备数据集

首次使用 (确保你获取到了数据集):

准备数据集

数据集须存放在/Train_Image/MyImage

数据集结构

```
Train_Image
-> VOCdevkit
  --> Annotations
    -- 000001.xml
    -- 000002.xml
    -- 000003.xml
    -- ...
  --> ImageSets
    --> Main
      -- test.txt
      -- train.txt
      -- val.txt
    --> JPEGImages
      -- 000001.jpg
      -- 000002.jpg
      -- 000003.jpg
      -- ...
-> MyImage
  -->
```

你可以将自己的数据集放在MyImage，但是需要保证不会缺少以上列出来的VOC数据集结构的文件，同时你最好检查一下test.txt、train.txt、val.txt中的文本格式是否和VOC数据集中的一致。

然后使用python生成需要的文件

datamaking.py

仅根据Annotations和JPEGImages目录
生成pascalvoc.txt、train.txt、val.txt、test.txt
同时会删除多余或不配对的.xml .jpg 运行时间较长
数据按7: 2: 1分配

datamakingv2.py

仅根据Annotations目录下文件的.xml文件生成需要的pascalvoc.txt、train.txt、val.txt、test.txt
运行时间较短，适用于确认数据集一一对应的情况

生成label

```
python voc_label.py
cat MyImage_train.txt MyImage_val.txt> train.txt    Linux使用此命令
```

```
type MyImage_train.txt MyImage_val.txt> train.txt
```

 windows使用此命令

注意:

- 改变路径后重新训练需从`python voc_label.py`从新开始
- win 不支持 wget, 因此你需要安装相关工具, 或直接在浏览器中输入 wget 后面的网址, 下载后复制到改目录

然后将 IMG 路径和注释合并到一个 NPY 文件

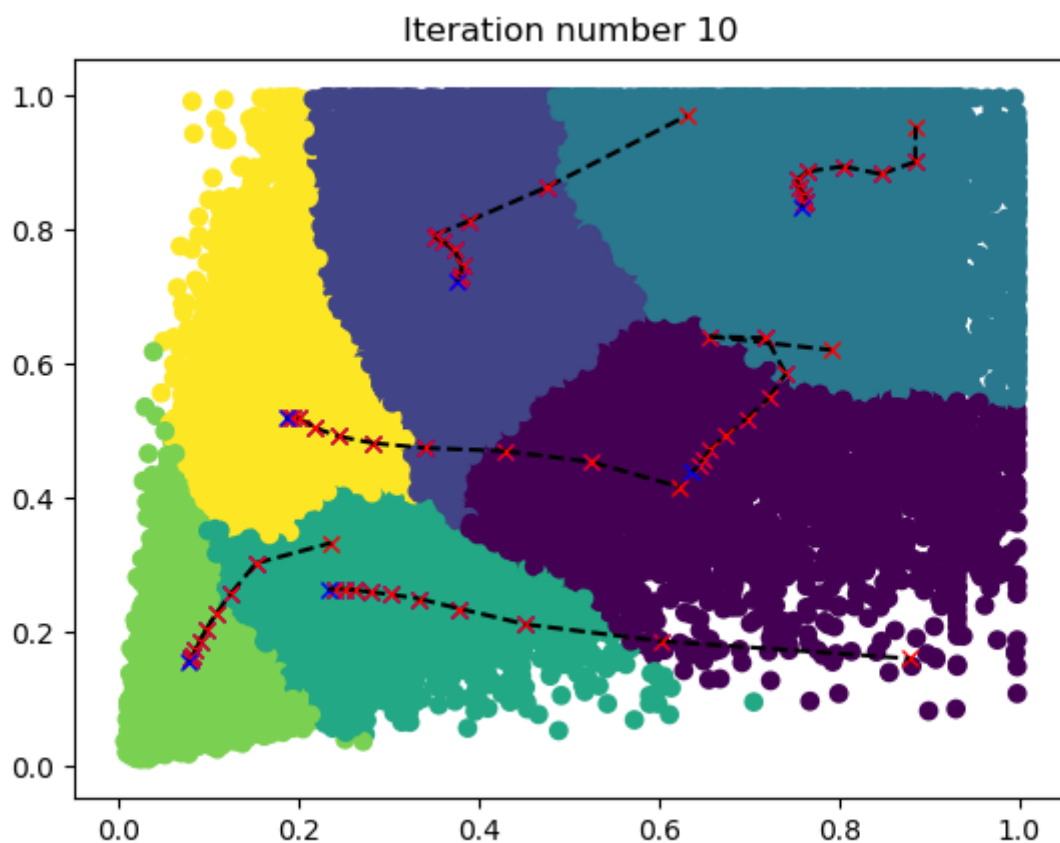
```
python make_voc_list.py train.txt data/voc_img_ann.npy
```

生成 anchors

加载注释生成 anchors(LOW 和 HIGH 视数据集的分布而定):

```
make anchors DATASET=voc ANCNUM=3 LOW="0.0 0.0" HIGH="1.0 1.0"
```

当你成功的时候, 你会看到这样以下内容:



注: 结果是随机的。当你有错误时, 就重新运行它。

如果要使用自定义数据集，只需编写脚本并生成 `data/{dataset_name}_img_ann.npy`，然后使用 `make anchors DATASET=dataset_name`。更多选项请参见 `python3 ./make_anchor_list.py -h`

如果要更改输出层的数目，则应修改 OUTSIZE 在 Makefile

下载预训练模型

你必须下载您想要训练的模型权重，因为默认情况下会加载训练前的权重。把文件放进 `./data` 目录。

MODEL	DEPTHMUL	Url	Url
yolo_mobilev1	0.5	google drive	weiyun
yolo_mobilev1	0.75	google drive	weiyun
yolo_mobilev1	1.0	google drive	weiyun
yolo_mobilev2	0.5	google drive	weiyun
yolo_mobilev2	0.75	google drive	weiyun
yolo_mobilev2	1.0	google drive	weiyun
tiny_yolo		google drive	weiyun
yolo		google drive	weiyun

注：mobilev 不是原创的，原作者有修改它适合 K210

Train

使用 Mobileenet 时，需要指定 DEPTHMUL 参数。使用 tiny yolo 或 yolo 你不需要设定 DEPTHMUL。

1. 集MODEL和DEPTHMUL开始训练： `make train MODEL=yolo_mobilev1 DEPTHMUL=0.75 MAXEP=10 ILR=0.001 DATASET=voc CLSNUM=20 IAA=False BATCH=8`
2. 集CKPT继续训练:

```
make train MODEL=yolo_mobilev1 DEPTHMUL=0.75 MAXEP=10 ILR=0.001 DATASET=voc
CLSNUM=20 IAA=False BATCH=8 CKPT=log/xxxxxxxxx/yolo_model.h5
```

3. 集IAA为了增加数据:

```
make train MODEL=xxxx DEPTHMUL=xx MAXEP=10 ILR=0.0001 DATASET=voc CLSNUM=20
IAA=True BATCH=16 CKPT=log/xxxxxxxxx/yolo_model.h5
```

4. 使用 tensorboard:

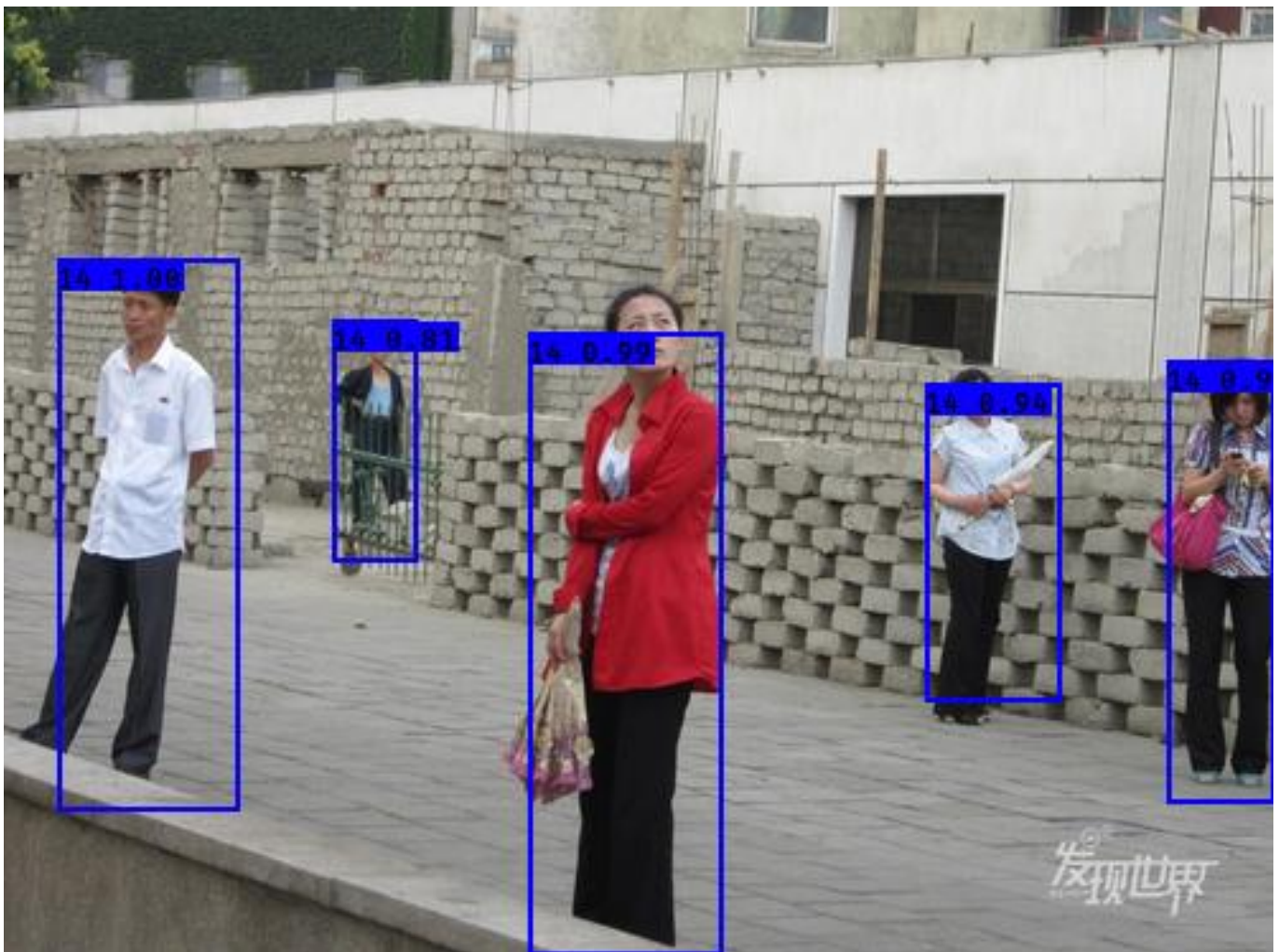
```
tensorboard --logdir log
```

推论

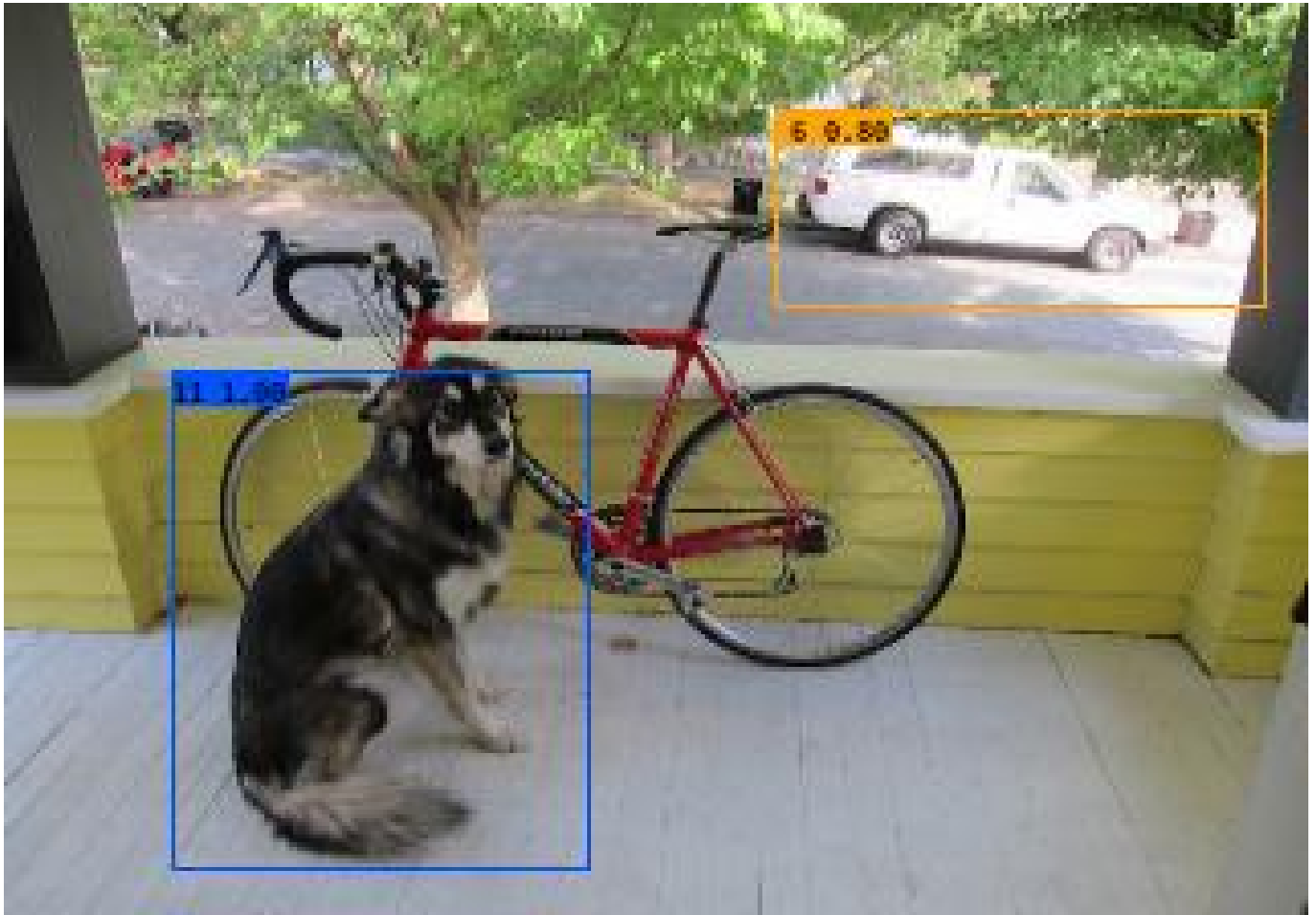
```
make inference MODEL=yolo_mobilev1 DEPTHMUL=0.75 CLSNUM=20  
CKPT=log/xxxxxx/yolo_model.h5 IMG=data/people.jpg
```

你可以尝试我的模型：

```
make inference MODEL=yolo_mobilev1 DEPTHMUL=0.75 CKPT=asset/yolo_model.h5  
IMG=data/people.jpg
```



```
make inference MODEL=yolo_mobilev1 DEPTHMUL=0.75 CKPT=asset/yolo_model.h5 IMG=data/dog.jpg
```



注：由于 anchors 是随机生成的，如果您的结果与上面的图像不同，你只需要加载这个模型并继续训练一段时间。

更多选项请参见 `python3 ./keras_inference.py -h`

修剪模型

```
make train MODEL=xxxx MAXEP=1 ILR=0.0003 DATASET=voc CLSNUM=20 BATCH=16 PRUNE=True  
CKPT=log/xxxxxx/yolo_model.h5 END_EPOCH=1
```

训练结束时，将模型保存为 `log/xxxxxx/yolo_prune_model.h5`。

Freeze

```
toco --output_file mobile_yolo.tflite --keras_model_file log/xxxxxx/yolo_model.h5
```

现在你有了 `mobile_yolo.tflite`

转换 Kmodel

Please refer `nncase v0.1.0-RC5 example`

```
ncc mobile_yolo.tflite mobile_yolo.kmodel -i tflite -o k210model --dataset  
nncase_images
```

将 Kmodel 部署到 K210

- 见另一个文档

2020/7/5 21:04:35