# 一 环境及数据集介绍

## 1.1 代码环境

以<https://github.com/BobLiu20/mtcnn_tf>为base版本.

## 1.2 训练数据集

### 1.2.1 WIDER train数据集.

WIDER face dataset数据集.( **[WIDER FACE: A Face Detection Benchmark](http://mmlab.ie.cuhk.edu.hk/projects/WIDERFace/)**[)](http://mmlab.ie.cuhk.edu.hk/projects/WIDERFace/)

特点:

1. Face detection benchmark datasets, 图片均为公共图片
2. 含有32203张图, 有393703个人脸(scale, pose,occlusion(遮盖)等多变性图片).
3. 含有61分类事件. 40%/10%/50%的train,val,test占比.
4. 对于test图片,并不release对应的boundingbox ground truth.用户需要提交预测文件.由WIDER来做预测.



1.2.2 landmark 数据集

对于人脸关键点数据集采用的是:[CNN\_FacePoint](http://mmlab.ie.cuhk.edu.hk/archive/CNN_FacePoint.htm)其特点是:

1. 含有5590个LFW图像和7876个网络下载的图像.
2. trainImageList.txt和testImageList.txt分别保存有train,val数据集.

其格式是: 图片名字+矩形框+landmark的5个关键点(five facial points).

# 二 训练MTCNN

2.1 训练命令

|  |
| --- |
| #!/bin/bash  **set** -e  ### All of your tmp data will be saved in ./tmp folder  **echo** "Hello! I will prepare training data and starting to training step by step."  # 1. checking dataset if OK  **if** **[** **!** **-d** "./dataset/WIDER\_train/images" **];** **then**  **echo** "Error: The WIDER\_train/images is not exist. Read dataset/README.md to get useful info."  **exit**  **fi**  **if** **[** **!** **-d** "./dataset/lfw\_5590" **];** **then**  **echo** "Error: The lfw\_5590 is not exist. Read dataset/README.md to get useful info."  **exit**  **fi**  **echo** "Checking dataset pass."  **if** **[** **-d** "./tmp" **];** **then**  **echo** "Warning: The tmp folder is not empty. A good idea is to run ./clearAll.sh to clear it before training."  **fi**  # 2. stage: P-Net  ### generate training data(Face Detection Part) for PNet  **echo** "Preparing P-Net training data: bbox"  **python** prepare\_data**/**gen\_hard\_bbox\_pnet.py  ### generate training data(Face Landmark Detection Part) for PNet  **echo** "Preparing P-Net training data: landmark"  **python** prepare\_data**/**gen\_landmark\_aug.py **--**stage**=**pnet  ### generate tfrecord file for tf training  **echo** "Preparing P-Net tfrecord file"  **python** prepare\_data**/**gen\_tfrecords.py **--**stage**=**pnet  ### start to training P-Net  **echo** "Start to training P-Net"  **python** training**/**train.py **--**stage**=**pnet  # 3. stage: R-Net  ### generate training data(Face Detection Part) for RNet  **echo** "Preparing R-Net training data: bbox"  **python** prepare\_data**/**gen\_hard\_bbox\_rnet\_onet.py **--**stage**=**rnet  ### generate training data(Face Landmark Detection Part) for RNet  **echo** "Preparing R-Net training data: landmark"  **python** prepare\_data**/**gen\_landmark\_aug.py **--**stage**=**rnet  ### generate tfrecord file for tf training  **echo** "Preparing R-Net tfrecord file"  **python** prepare\_data**/**gen\_tfrecords.py **--**stage**=**rnet  ### start to training R-Net  **echo** "Start to training R-Net"  **python** training**/**train.py **--**stage**=**rnet  # 4. stage: O-Net  ### generate training data(Face Detection Part) for ONet  **echo** "Preparing O-Net training data: bbox"  **python** prepare\_data**/**gen\_hard\_bbox\_rnet\_onet.py **--**stage**=**onet  ### generate training data(Face Landmark Detection Part) for ONet  **echo** "Preparing O-Net training data: landmark"  **python** prepare\_data**/**gen\_landmark\_aug.py **--**stage**=**onet  ### generate tfrecord file for tf training  **echo** "Preparing O-Net tfrecord file"  **python** prepare\_data**/**gen\_tfrecords.py **--**stage**=**onet  ### start to training O-Net  **echo** "Start to training O-Net"  **python** training**/**train.py **--**stage**=**onet  # 5. Done  **echo** "Congratulation! All stages had been done. Now you can going to testing and hope you enjoy your result."  **echo** "haha...bye bye" |