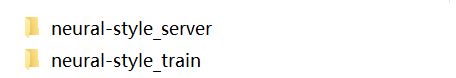
**训练模型**

# 准备工作

## 说明

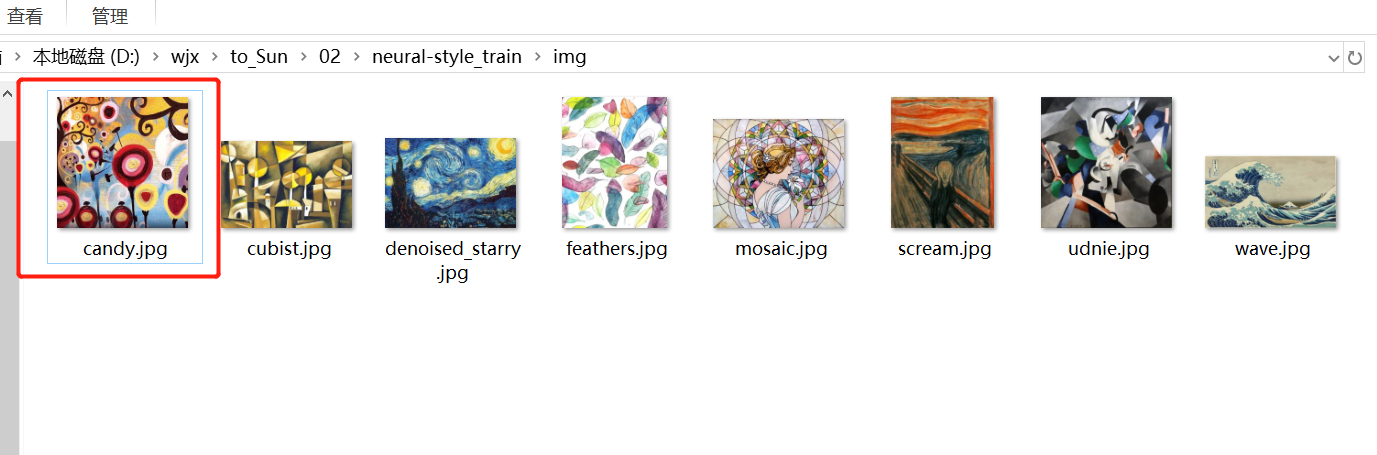


\_server和之前使用与修改一下，具体参考《优化服务器使用说明》

\_train是训练代码

## 准备数据

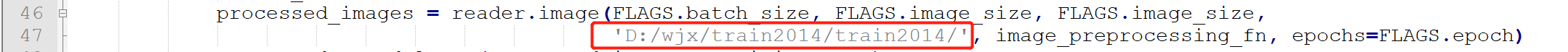
1. 网盘下载trian2014并解压到某一路径
2. 新准备一张风格图片保存到neural-style\_train\img，将其名字改为有意义的英文名，后面以candy为例，建议招风格强烈的图片。



# 二、训练参数修改

## neural-style\_train\train.py

**1.数据集路径修改：**

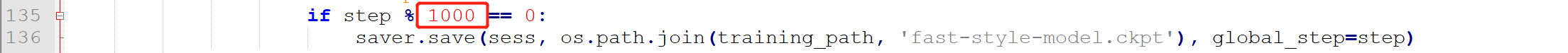


将‘D:/wjx/train2014/train2014/’修改为你解压后的train2014路径

**2)学习率（不建议乱改）:**



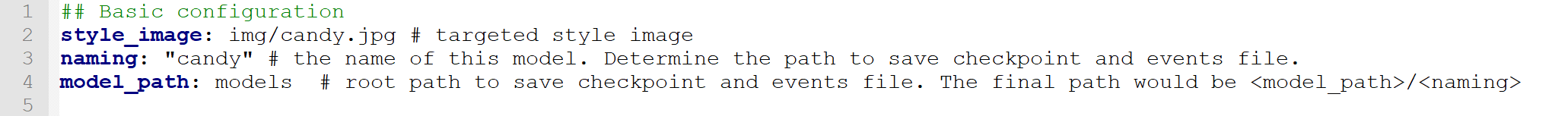
**3）每多少次保存一次模型：**



可以稍微改大一点

## 填写neural-style\_server\conf\candy.yml

**1）基础配置**

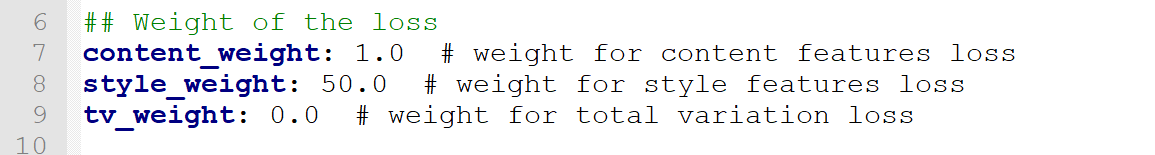


style\_image:风格图片的路径

madel\_path:模型保存的路径

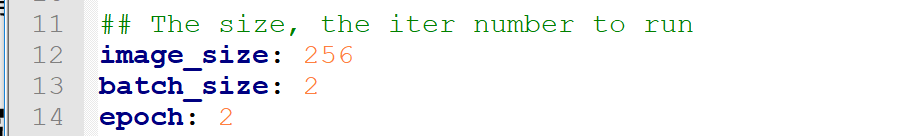
naming：模型保存的文件夹名，这里填“candy”，成功训练后你会在model/candy中看到对应模型的生成

**2）权重的配置**



**建议只改style\_weight，50-300**

**3)其他配置**



image\_size不建议修改

batch\_size：训练批数量，根据gpu的好坏增大或减小，我这边机器不好2或4勉强能接受，建议2的次方。

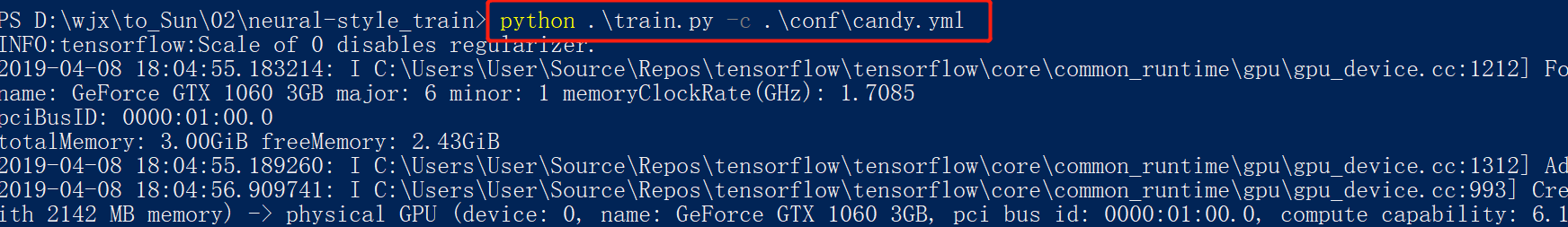
epoch：把训练集也就是train2014，的每张图片完整的训练两遍。建议在资源和时间充沛的情况下，可将其调大,那保存模型的频率可以在train.py将其调大。

**剩下的不建议修改，可以参考conf文件夹下的yml文件**

# 三、训练

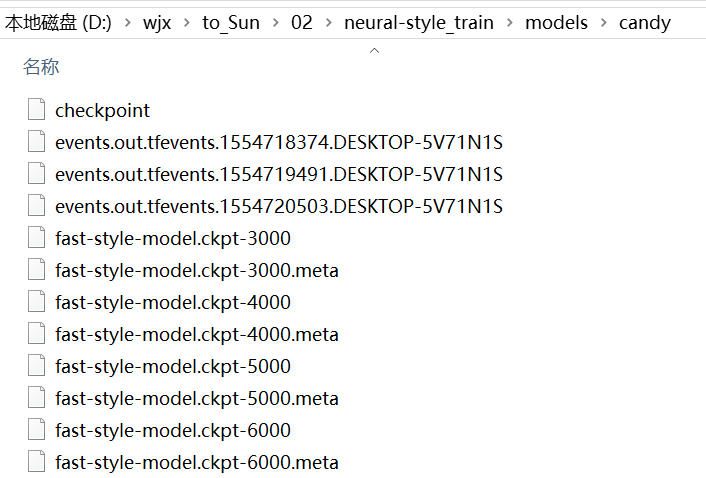
## 命令

python train.py -c conf/candy.yml



报错的话，要么gpu不够给力，建议调小bitch\_size，要么路径之类的不对，好好检查一下。

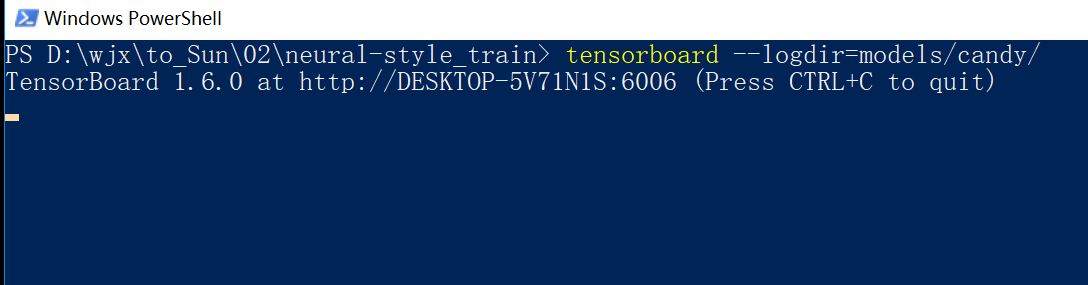
生成模型类似于下图：



# 四、Tensorboard

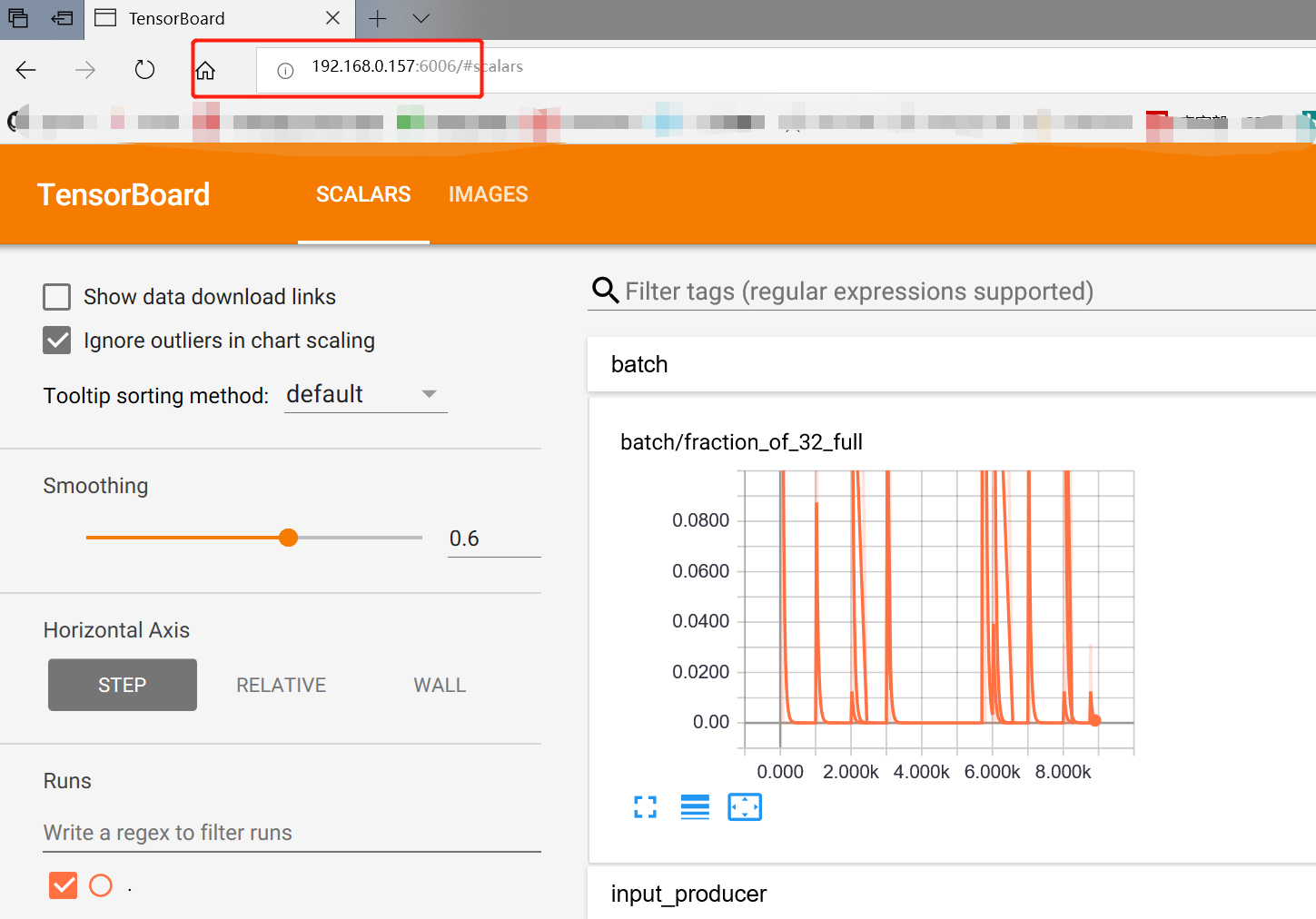
命令行：

tensorboard --logdir models/candy/



浏览器访问：

<http://ip地址:6006>



# 五、测试

将上图的fast-style-model.ckpt-xxxx文件（19633kb）的那个文件，xxxx表示训练了多少张图片，复制到**服务器代码**：neural-style\_server\models\，并重命名为candy.ckpt-done，然后用服务器python server.py，model\_file=candy，测试模型效果。一定要改名成xxxx.ckpt-done，如果你想测多个模型，比如candy5000和6000，你就改名成candy5000.ckpt-done,candy6000.ckpt-done，测试的时候model\_file= candy5000 或candy6000，建议最后保留一个模型

# 六、最后

训练终端了开启训练命令，可以继续训练（除非你把之前的models/candy删掉了或者epoch完成）。一个好的模型可能要训练和调参很多次，多测试找出最合适的模型。