

代码说明

1.代码结构及说明

- **checkpoint**: 用于保存每个epoch判别器、生成器的模型训练结果。
- **data**: 包含训练数据、验证数据、测试数据。
- **dev-tools**: 对于验证集和测试集的图片预处理的代码。
- **log**: 代码的训练过程, 可通过tensorboard在浏览器中进行可视化, 来实时监控模型的训练。

注意: 该文件夹下只能有一个log文件, 否则可能会无法生成曲线。

- **pytorch_ssim**: 计算ssim评估指标的代码。
- **vis_val**: 每一个epoch验证集的超分辨率结果, 用于监控图片的SR效果, 如“epoch_102_index_12.png”表示第102个epoch的第12张图片的结果, 该图片从左向右依次是bicubic/Original/SR。
- **vis_test**: 测试集的超分辨率结果, 用于测试模型的泛化性能, 该图片从左向右依次是bicubic/Original/SR。
- **eval.py**: 对从第1个epoch到第 100 个纪元的所有检查点进行测试, 并打印结果, 如页面底部的结果。
- **eval-compare.py**: 用于比较SRGAN、SRWGAN-GP等模型的SR结果, 图片从左到右依次是bicubic/SRGAN/SRWGAN/SRWGAN-GP/Original。
- **model.py**: 生成器、判别器、残差块等相关模块的定义, 即SRGAN、SRWGAN、SRWGAN-GP等模型的定义。
- **sr.py**: 单幅图像的SR测试。
- **train.py**: 训练SRGAN的函数入口。
- **train-wgangp.py**: 训练SRWGAN-GP的函数入口。
- **utils.py**: 训练图片的预处理、获取生成器和判别器的梯度和参数等其他函数。

2.环境搭建与依赖库安装

2.1开始之前

- 数据集可以从以下网站[新数据集 \(seeprettyface.com\)](http://seeprettyface.com)找到, 这里也给出百度网盘的数据集下载地址: [百度网盘](https://pan.baidu.com/s/1JIMD) 提取码: JIMD
- 环境推荐使用anaconda的虚拟环境, 来避免和已安装库的版本冲突问题, 因此你需要根据以下相关教程安装anaconda:

[\(177条消息\) Anaconda超详细安装教程 \(Windows环境下\) 菜鸟1号!! 的博客-CSDN博客windows安装anaconda](#)

- 如果你遇到 `pip` 或者 `conda` 命令安装库失败的情况, 通常是网络的问题, 一种解决办法是更换默认源:

[Python 修改 pip 源为国内源 - 137point5 - 博客园 \(cnblogs.com\)](#)

[conda换国内源&pip换国内源 - 知乎 \(zhihu.com\)](#)

2.2anaconda虚拟环境创建与激活

创建并激活一个anaconda的虚拟环境，你需要以下命令：

```
conda create --name SRGAN_Pytorch python=3.9# 创建新环境
conda activate SRGAN_Pytorch# 激活SRGAN_Pytorch环境
```

2.3依赖库安装

1. pytorch（最新版即可）

首先是pytorch下载，以GPU版本为例：首先你需要确定自己的NVIDIA的显卡版本（我的CUDA型号是11.7），接着进入pytorch官网[PyTorch](https://pytorch.org/)，找到如下页面并选择，一般来说使用 pip 比 conda 安装速度要快的多：

PyTorch Build	Stable (1.13.0)		Preview (Nightly)	
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python		C++ / Java	
Compute Platform	CUDA 11.6	CUDA 11.7	ROCm 5.2	CPU
Run this Command:	pip3 install torch torchvision torchaudio --extra-index-url https://download.pytorch.org/whl/cu117			

复制给出的命令即可安装GPU版本的pytorch：

```
pip3 install torch torchvision torchaudio --extra-index-url
https://download.pytorch.org/whl/cu117
```

2. 安装其他库

执行如下命令即可：

```
pip install torchvision # pytorch需要用到的数据处理库
pip install tensorboard_logger #为tensorflow训练可视化开发的，但是pytorch也可以用
pip install tqdm # 训练过程中的进度条打印
```

3.程序运行

1. 训练原始的SRGAN模型，其中 train_set 是训练数据集的相对路径：

```
python train.py --train_set=data/train
```

2. 训练原始的SRWGAN-GP（WGAN with gradient penalty）模型，其中 train_set 是训练数据集的相对路径：

```
python train-wgangp.py --train_set=data/train
```

3. 测试训练好的模型，其中 val_set 指定验证集的地址，在第1个epoch到第100个epoch的模型上测试（存放在checkpoint中），epoch的interval为1：

```
python eval.py --val_set=data/val --start=1 --end=100 --interval=1
```

4. 对单幅照片进行超分辨, `1r.png` 是指定的图片:

```
python sr.py --1r=1r.png
```

5. 比较所有的模型, 其中 `val_set` 是最终测试图片的相对路径:

```
python eval-compare.py --val_set=results/Original
```

4.训练可视化

如果你想要将训练过程可视化, 你需要额外安装 `tensorboard`, (最新版即可) 命令如下:

```
pip install tensorboard
```

完成后再命令行输入如下命令启动, 出现如下页面表明启动成功:

注意: 一定要首先定位到项目log的保存位置,

```
tensorboard --logdir=./log
```

```
TensorFlow installation not found - running with reduced feature set.  
Serving TensorBoard on localhost; to expose to the network, use a proxy or pass --bind_all  
TensorBoard 2.11.0 at http://localhost:6006/ (Press CTRL+C to quit)
```

浏览器打开<http://localhost:6006/#timeseries>即可实时的观测到训练过程:

注意: log文件夹中只能有一个log文件

