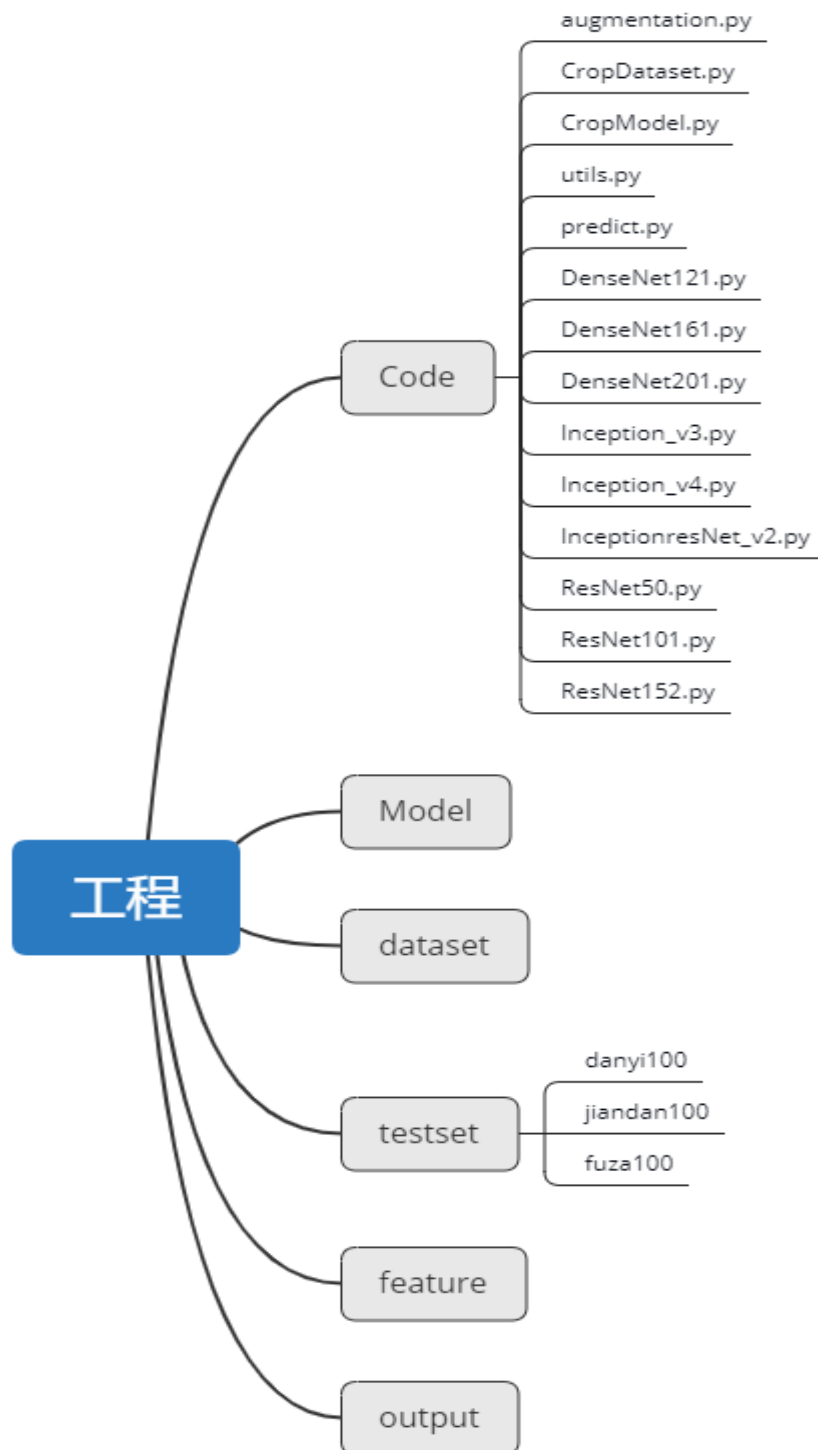


1. 前言

本工程是应用于参加 PRCV2019-大田作物病害图像识别技术挑战赛, 对 3000 张按照类别分成 6 类的水稻和小麦叶片病虫害照片进行有监督训练, 最后使用三种不同背景条件下的图像各 100 张共 300 张进行测试。使用 Pytorch1.1 作为学习框架, 操作系统为 ubuntu16.04。

2. 工程组织结构



Code: 存放项目的代码文件。CropModel.py 对训练使用到的模型进行封装; CropDataset.py 存放数据加载类和 transform 方法; utils.py 存放使用到的工具方法; 模型名.py 为对应模型的训练程序。

Model: 存放训练好的模型权重。每个模型有 acc_best.pth 和 loss_best.pth 两个权重文件, 预测的时候只加载其中一个。

dataset: 存放训练数据集。有 6 个文件夹, 文件夹名字对应类别标签。总共 3000 样本。

testset: 存放测试数据集。有 3 个文件夹, 对应单一、简单和复杂背景图片各 100 张。

feature: 存储 TTA 之后生成的结果, 也是网络预测后的原始输出。

output: 存放要提交结果的文件。

3. 操作指南

本节是使用训练好的模型对 danyi100、jiandan100、fuza100 三个文件夹图片进行预测, 生成预测文件用 json 格式保存在 output 文件夹中。

步骤:

1. 搭建运行环境

1.1 安装虚拟环境工具

a) 进入 ubuntu16.04 系统后, 首先更新 apt-get, 在终端命令行输入:

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

b) 然后安装 python3.5, 在终端命令行输入:

```
sudo apt-get install python3.5-dev
```

c) 然后安装虚拟环境管理工具, 在命令行输入:

```
sudo pip install virtualenv virtualenvwrapper
```

d) 将 virtualenv 添加到环境变量中, 在命令行输入:

```
sudo gedit ~/.bashrc
```

e) 把以下内容添加到 .bashrc 文件末尾, 保存

```
export WORKON_HOME=$HOME/.virtualenvs
```

```
source /usr/local/bin/virtualenvwrapper.sh
```

f) 执行以下命令更新 .bashrc 文件:

```
source ~/.bashrc
```

1.2 创建虚拟环境

a) 进入主目录 (或自定义路径), 在终端命令行输入:

```
mkvirtualenv Torch -p python3
```

若出现以下信息, 则创建成功

```

klb@klb-B360M-DS3H:~$ mkvirtualenv Torch -p python3
Already using interpreter /usr/bin/python3
Using base prefix '/usr'
New python executable in /home/klb/.virtualenvs/Torch/bin/python3
Also creating executable in /home/klb/.virtualenvs/Torch/bin/python
Installing setuptools, pip, wheel...
done.
virtualenvwrapper.user_scripts creating /home/klb/.virtualenvs/Torch/bin/predeactivate
virtualenvwrapper.user_scripts creating /home/klb/.virtualenvs/Torch/bin/postdeactivate
virtualenvwrapper.user_scripts creating /home/klb/.virtualenvs/Torch/bin/preactivate
virtualenvwrapper.user_scripts creating /home/klb/.virtualenvs/Torch/bin/postactivate
virtualenvwrapper.user_scripts creating /home/klb/.virtualenvs/Torch/bin/get_env_details
(Torch) klb@klb-B360M-DS3H:~$

```

若遇见以下错误:

```

yiqiu@ubuntu:~$ mkvirtualenv Torch110 -p python3
Traceback (most recent call last):
  File "/usr/lib/command-not-found", line 27, in <module>
    from CommandNotFound.util import crash_guard
ImportError: No module named 'CommandNotFound'

```

则, 在终端命令行输入:

```
virtualenv Torch110 -p python3
```

虚拟环境建立成功, 在当前路径 (本文档的路径为/home/yiqiu/) 生成文件夹 'Torch110'
如下图所示:

```

yiqiu@ubuntu:~$ virtualenv Torch110 -p python3
Already using interpreter /usr/bin/python3
Using real prefix '/usr'
New python executable in /home/yiqiu/Torch110/bin/python3
Also creating executable in /home/yiqiu/Torch110/bin/python
Installing setuptools, pip, wheel...done.

```

在建立虚拟环境的路径下, 终端输入以下命令行:

```
source Torch110/bin/activate/
```

若激活成功, 则 user 显示出虚拟环境名称, 如下图所示:

```

yiqiu@ubuntu:~$ source Torch110/bin/activate
(Torch110) yiqiu@ubuntu:~$

```

2. 安装必要的包

环境搭建完毕后进入虚拟环境 (user 前面显示虚拟环境名称则成功进入虚拟环境)

```

klb@klb-B360M-DS3H:~$ mkvirtualenv Torch -p python3
Already using interpreter /usr/bin/python3
Using base prefix '/usr'
New python executable in /home/klb/.virtualenvs/Torch/bin/python3
Also creating executable in /home/klb/.virtualenvs/Torch/bin/python
Installing setuptools, pip, wheel...
done.
virtualenvwrapper.user_scripts creating /home/klb/.virtualenvs/Torch/bin/predeactivate
virtualenvwrapper.user_scripts creating /home/klb/.virtualenvs/Torch/bin/postdeactivate
virtualenvwrapper.user_scripts creating /home/klb/.virtualenvs/Torch/bin/preactivate
virtualenvwrapper.user_scripts creating /home/klb/.virtualenvs/Torch/bin/postactivate
virtualenvwrapper.user_scripts creating /home/klb/.virtualenvs/Torch/bin/get_env_details
(Torch) klb@klb-B360M-DS3H:~$

```

执行以下命令安装必要的包:

```
pip install torch==1.1
```

```
pip install torchvision==0.3.0
```

```
pip install pretrainedmodels
```

```
pip install scipy
```

若没有 VPN, 会出现下载速度慢的情况, 可以在每个命令行后面添加:

```
-i https://pypi.douban.com/simple/
```

例如：

```
pip install torch==1.1 -i https://pypi.douban.com/simple/  
pip install torchvision==0.3.0 -i https://pypi.douban.com/simple/  
pip install pretrainedmodels -i https://pypi.douban.com/simple/  
pip install scipy -i https://pypi.douban.com/simple/
```

3. 执行预测文件 predict.py

进入工程文件夹后依次执行以下步骤进行预测：

[1] 预测 danyi100 文件夹的图片，在终端输入命令行：

```
python predict.py --mode all --input ../testset/danyi100  
--output ../output/danyi.json
```

[2] 预测 jiandan100 文件夹的图片，在终端输入命令行：

```
python predict.py --mode all --input ../testset/jiandan100  
--output ../output/jiandan.json
```

[3] 预测 fuza100 文件夹的图片，在终端输入命令行：

```
python predict.py --mode all --input ../testset/fuza100  
--output ../output/fuza.json
```

执行完即可在 outpu 文件夹看到对应的 json 文件。