

2019 夏程序设计实践

大作业(2) 文档

牛浩宇 软件 72 2017010729

I. 开发环境:

Python: 3.7

Django: 2.2.4

数据库: SQLite3

GitHub 地址: https://github.com/nhy17-thu/FinalProject_Upgrade

IDE: PyCharm

Python 外部库:

pytorch 1.2.0: <https://github.com/pytorch/pytorch>, BSD-style licensed

torchvision 0.4.0: <https://github.com/pytorch/vision/tree/master/torchvision>, BSD 3-Clause License

numpy 1.16.4: <https://github.com/numpy/numpy>, BSD 3-Clause License

opencv 3.4.2: <https://github.com/opencv/opencv>, BSD 3-Clause License

dlib 19.17: <https://github.com/davisking/dlib>, Boost Software License

前端外部库:

bootstrap 4.0.0: <https://github.com/twbs/bootstrap>, MIT License

jquery 3.4.1: <https://github.com/jquery/jquery>, MIT License

(上两者已包含在作业文件夹中)

II. 主要功能升级:

在周一提交的大作业基础上（文档请见 ./Document(last_submission).pdf），我在本次提交中独立地实现了如下主要功能：

- A. 提交页面与处理分离，在提交后立刻渲染新页面，再利用 AJAX 技术实时返回处理好的图片/结果。
- B. 添加了基于 ImageNet 数据集的 ResNet-152 网络，实现 1000 类大图分类。
- C. 添加了基于 ResNet-152 分类结果的新结果浏览页面，用户现在可以依据上传图片的整图分类结果来查看/删除操作历史了。
- D. 添加了基于 pytorch 例程的神经网络风格转换，可以将任意尺寸的输入图片转换成镶嵌风格等多种不同风格的同尺寸图片。
- E. 添加了基于 dlib 的正面人脸识别和聚类处理，可以自动识别上传图片中的正面人脸，并依照人脸的不同特征划分为不同人员。

- F. 添加了基于 dlib 人脸聚类结果的结果浏览页面，用户可以一次性地查看上传的所有图片中出现的所有不同人物，并可以方便地查看原图及相应处理结果。
- G. 添加了新的查看所有结果接口及相应模版，用户现在可以在各个查询记录页面一次性查看存储在系统中的所有结果。
- H. 系统易用性方面：进行了提示文字等方面的升级；添加了错误处理机制，对可能造成 500 等错误的风险操作使用 `try except` 语句进行处理，不影响用户体验。
- I. 系统安全性方面：上述所有接口均有登陆检测机制，登陆信息错误的用户将被拒绝访问；在用户删除记录时，所有处理结果（图片）也将被删除，防止信息泄露。

III. 使用方法

本系统的使用方法与上一个版本相同，只需在 `manage.py` 所在的文件夹下执行

```
python manage.py runserver
```

即可在系统默认的 <http://127.0.0.1:8000/> 路径打开本系统。

请注意，此时运行的 `python` 环境需要安装有开发环境部分所提到的所有前端以外的外部库，否则可能会导致运行错误。

此外，在 `test_images/` 目录下准备了上百张测试照片可用于上传测试，还特别为人脸聚类任务准备了 `test_images/FaceClusteringImages/` 目录。

IV. 详细功能说明

本部分将对 II. 主要功能升级 部分中的部分细节较为复杂的功能进行更详细的解释。

- A. 实时返回处理结果：现在系统将在用户点击提交按钮以后立刻刷新页面，并在新的模版页面中发起图片处理请求，再利用 AJAX 技术在新页面中实时追踪并接受服务器后台处理好的每一张图片。即，用户现在可以实时地每隔一段时间看见一个新的处理结果了。
- B. ResNet-152：利用 `pytorch` 提供的 ResNet-152 网络模型在标准 1000 类 ImageNet 数据集上进行训练，并将训练好的模型部署在本网站上，Top-1 error 仅为 21.69%。用户在提交完图片/查看历史记录时将可以看到更细更准确的分类结果。
- C. 基于 ResNet-152 分类结果的新结果浏览页面：系统现在在首页提供新的历史记录浏览方式，即依照整图分类结果来浏览所有上传图片，与先前的时间顺序相比更有利于用户找到自己想要的图片记录。同时，该页面还支持打开原图、查看所有处理结果、删除记录及对应图片等功能，点击图片下方的相应按钮即可完成。

D. 风格转换：基于 pytorch 在 GitHub 上的例程实现，可以将任意尺寸的输入图片转换成多种不同风格的同尺寸图片，系统默认为镶嵌风格。若要更改风格，只需更改 users/neural_style.py 文件中的 state_dict 参数，风格文件均存储在 users/config/ 路径下，提供 candy/mosaic/rain_princess/udnie 四种选择。该网络处理结果将与其他网络的处理结果一并显示。

E. 正面人脸识别和聚类处理：本功能基于 dlib 和 opencv 实现，可以自动检测识别上传图片中的所有正面人脸，并将其在结果页面中合并成一张新图片，便于用户浏览。此外，我们还利用人脸识别模型得到每一张人脸中的 68 个关键点，并将其转化为 128 维特征向量，再利用 CW (Chinese Whisper) 聚类算法实现人脸聚类，帮助用户轻松查看照片中多次出现过的不同好友。

F. 人脸聚类结果的结果浏览页面：系统现在在首页提供新的历史记录浏览方式，即按照人脸聚类结果排列所有操作记录。用户可以一次性查看上传所有图片中多次出现的所有不同人物，并可以方便地查看原图及相应处理结果，同样可以选择删除这条记录。

G. 新的查看所有结果接口及相应模版：按原来大作业要求，系统只支持查看用户上传的原始图片。升级后，用户可以在上述各个查询记录页面直接打开新页面，一站式查看存储在数据库中的相应所有处理结果，不必再次上传或等待重新处理。包含图片在内的所有结果均以二进制数据流方式从服务器数据库直接发送到用户浏览器。

H. 系统易用性升级：在各个需要用户交互的页面都进行了提示文字等方面的升级，并灵活运用 django 与前端页面交互，用简单的 html 语句展示了含有复杂信息的界面。此外，系统还添加了错误处理机制，在服务器端对可能造成 500 等内部错误的典型风险操作（如文件读写等）使用 try except 语句进行处理，使得用户操作不规范时（典型操作如上传时不选中文件，直接上传）也不会导致服务器返回错误页面，而是正常跳转回首页等相关页面，从而优化了用户体验。

I. 系统安全性升级：上述所有新的功能接口均有登陆检测机制，登陆信息错误的请求将被拒绝，请求任何别的用户的私人信息也将返回错误信息。此外，无论用户在哪个界面点击删除结果按钮，与之相关的上传图片、处理后图片/结果均会被一起删除，从而加强了用户信息的安全性。

V. 系统界面展示

鉴于在上一次大作业报告中已经展示了部分系统部署后的运行界面，因此下面将只展示有新增功能的页面。为了显示更多内容，图片尺寸经过缩放。

A. 图像上传页面

[back to main page](#)

建议上传.jpg格式图片，大小不超过500KB，否则可能出现错误

pic: No file chosen

url:

如下是你上传的图片：



Detection处理结果如下：



Style Transfer处理结果如下：

(图像较大时处理可能较慢或显示不完全，请耐心等待或至查询记录页面查看完整结果)



在CIFAR-10数据集上训练的ResNet-18给出的分类预测结果如下：

Predicted: bird

(CIFAR-10数据集共有: plane, car, bird, cat, deer, dog, frog, horse, ship, truck十类)

在1000类ImageNet数据集上训练的ResNet-152给出的分类预测结果如下：

Predicted: apron

上传图像中所有正面人脸如下：



B. 一站式结果查看页面:

[back to records page](#)

Detection处理结果如下:



Style Transfer处理结果如下:



在CIFAR-10数据集上训练的ResNet-18给出的分类预测结果如下:

Predicted: deer

(CIFAR-10数据集共有: plane, car, bird, cat, deer, dog, frog, horse, ship, truck十类)

在ImageNet数据集上训练的ResNet-152给出的分类预测结果如下:

Predicted: barbershop

(ImageNet数据集共有1000类)

上传图像中所有正面人脸如下:



C. 按 ResNet-152 分类结果浏览操作记录：

[back to main page](#)

本页面将按在ImageNet上训练的ResNet-152分类结果，自动展示不同类别的上传图片。

出于排版考虑，所有图片均已调整尺寸，但均支持右键在新窗口打开原图。

如下是ResNet-152分类为 apron 的所有图像：



[查看对应处理结果](#) [删除这条记录及相应图片](#)

如下是ResNet-152分类为 assault rifle, assault gun 的所有图像：



[查看对应处理结果](#) [删除这条记录及相应图片](#)



[查看对应处理结果](#) [删除这条记录及相应图片](#)

如下是ResNet-152分类为 barbershop 的所有图像：



[查看对应处理结果](#) [删除这条记录及相应图片](#)

如下是ResNet-152分类为 laptop, laptop computer 的所有图像：



[查看对应处理结果](#) [删除这条记录及相应图片](#)

如下是ResNet-152分类为 maillot 的所有图像：



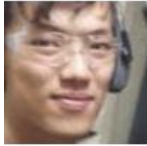







D. 按人脸聚类结果浏览操作记录：

[back to main page](#)

本页面将按基于dlib的人脸聚合分类结果，自动计算并展示上传图片中的不同人脸及其所属图片。

出于排版考虑，原始图片均已调整尺寸，但均支持右键在新窗口打开原图。

如下是自动编号为0的人的所有图片：

| | | | |
|---|---|--------------------------|-----------------------------|
|  |  | 查看对应处理结果 | 删除这条记录及相应图片 |
|  |  | 查看对应处理结果 | 删除这条记录及相应图片 |
|  |  | 查看对应处理结果 | 删除这条记录及相应图片 |
|  |  | 查看对应处理结果 | 删除这条记录及相应图片 |

如下是自动编号为1的人的所有图片：

| | | | |
|---|---|--------------------------|-----------------------------|
|  |  | 查看对应处理结果 | 删除这条记录及相应图片 |
|  |  | 查看对应处理结果 | 删除这条记录及相应图片 |
|  |  | 查看对应处理结果 | 删除这条记录及相应图片 |

VI. 结语

以上就是本次代考大作业的全部报告内容，源代码中也有具体的注释和说明。

尽管由于交换的时间安排原因，我无法在国内完成考试，但这也给了我机会去探索更多自己在上一次大作业中想要实现而没有来得及实现的功能和 feature，典型的比如给予 dlib 的人脸聚类，其灵感来源于 iPhone 相册应用中的 People 功能，也可以自动识别所有照片中的人脸，并显示属于某个人的所有照片。本系统也完成了相同的功能。

此外，这样一次独特的经历也督促我在短时间内快速自主地学习，进一步增强了我的自学能力和抗压能力，显著地增进了我对 python 相关知识，尤其是 pytorch 和 django 等方面的了解。

最后，感谢丁贵广老师善解人意，用灵活的考核方式为学生解决问题；也感谢助教的仔细阅读和认真批改，在提交成绩前夕还专门为我批改大作业。

谢谢！

牛浩宇
软件 72

niuhaoyu17@gmail.com