# 周报

|  |  |  |  |
| --- | --- | --- | --- |
| **学生姓名** | 何洋程 | **学号** | 17301123 |
| **项目名称** | 基于情感分析的养老院智能陪护系统 | | |
| **本周工作时间** | 7月5日 8:30-11:30, 14:00-17:00，19:00-21:00  7月6日 8:30-11:30, 14:00-17:00，19:00-21:00  7月7 日 8:30-11:30, 14:00-17:00，19:00-21:00  7月8 日 8:30-11:30, 14:00-17:00，19:00-21:00  7月9 日 8:30-11:30, 14:00-17:00，19:00-21:00  7月10 日 8:30-11:30, 14:00-17:00，19:00-21:00  7月11 日 8:30-11:30, 14:00-17:00，19:00-21:00  7月12 日 8:30-11:30, 14:00-17:00，19:00-21:00  共计64小时 | | |
| **本周工作内容** | 1. 人脸表情模型重构、调优 2. 实现人脸识别功能 3. 人脸与表情识别、返回数据 4. 文档编写 | | |
| **本周体会** | 之前人脸识别使用的是resnet神经网络来训练模型，但是识别率低、加载缓慢，最后又更换成了cnn卷积神经网络来训练模型，在训练模型的时候，因为最先使用了ckpt来保存模型，需要加载tf库，运行速度慢，就尝试使用其他方法保存模型，发现模型可以用多种方式保存，我尝试过了hdf5、pb、ckpt三种方式，最后发现模型的存储方式与加载速度无关，因为他们都是基于TensorFlow来保存模型，虽然储存方式不一样，但是加载速度和识别率是一样的。然后进行人脸识别功能的实现，一开始我打算使用cnn来训来模型，后来在查询资料时发现Python有相关的库可以直接识别人脸，那就是face\_recognition库，它是基于dlib库的，但是dlib库和face\_recognition库似乎对Windows并不是十分友好，难以安装，我查询了很多教程，最后还是使用在pypi官网下载的whl文件才安装成功，但即便如此，它在Windows上的加载速度也还是很慢；官方文档说明它在Linux好macOS上运行会很快。将两个功能分别写好后，下一步就是将两个模块放在一起运行，两个模块的实现方式不同，但是原理相似，都是先识别人脸的位置，在进行预测，经过一部分时间的摸索就将两个模块连接起来，但是发现预测结果有问题，表情识别与人脸识别本身都没有问题，但是人物与相应的表情不对应，花费了我很长时间，发现是表情识别与人脸识别在进行人脸位置识别是的方式不一样，前者是从左往右识别，而后者是从右往左识别，经过修改，能够正确预测。再返回数据时，要执行一个if语句，来区分不同的人脸，又发现返回的数据一直是义工，即使人物并不是义工，经过各种尝试后，才发现是预测的结果后面有一个空格，下次一定要注意生成的结果是否含有特殊字符。 | | |