注：alexnet\_no\_flip为数据未扩充结果，训练样本数为：5200 测试样本数为2716

其余四条曲线均为数据扩充后效果， 训练样本数为：10361 测试样本数为 5432



钢轨类型分类训练主要尝试了四种：分别是lenet\_origin，lenet\_dropout，sketch\_a-net，alexnet。

1、lenet\_origin模型使用的是caffe官方提供的lenet模型。只有卷积和池化层。

2、lenet\_dropout 模型则是参考lecun98年论文lenet，在原网络的基础上增加了norm局部响应归一化，并在最后全连接层增加dropout层防止过拟合。

3、sketch\_a\_net 参考论文[1] 。网络与alexnet相似仅仅是中间网络层一些节点数不一样。

4、alexnet则使用alexnet网络，并使用caffe官方提供的参数模型进行fine-tuning。

结果分析：

1、使用的四种网络中，lenet分类准确率在0.9左右，通过测试训练样本数据可以发现此时已经出现过拟合现象，于是lenet\_dropout增加了dropout网络层，并将测试准确率提高了三到四个百分点。sketch\_a\_net和alexnet，两种网络模型接近，效果接近。分类准确率在0.97左右。

2、对数据集进行翻转扩充，使用alexnet网络进行比较，扩充数据后准确率明显优于未扩充前。

3、通过各类别分类准确率数据发现，ZD类别出错率高。ZD类别样本数量较少，且是经过多次复制的数据，样本缺乏多样性，未能学习到有效特征，故出错率较高。

[1]Yu Q, Yang Y, Liu F, et al. Sketch-a-Net: A Deep Neural Network that Beats Humans[J]. International Journal of Computer Vision, 2017, 122(3):411-425.

文档说明

训练基于caffe框架，代码位于/home/CODE/ProfileNet。

1、data、data3:数据文件，其中data为未进行扩充的原始数据，data3为进行扩充后的数据。每个文件夹下均有get\_train\_val\_list.sh 脚本生成数据列表和标签列表，接着运行create\_rail\_lmdb.sh脚本生成lmdb格式数据，并保存在rail\_train\_lmdb和rail\_val\_lmdb文件夹下（运行前需要删除旧的lmdb文件夹）。

2、根目录下分别有train\_lenet.sh,train\_alexnet.sh,train\_sketch\_a\_net.sh分别为训练四种网络的启动脚本。训练脚本调用“网络名称\_solver.prototxt”训练配置文件，并在配置文件中调用各自的“网络名称\_.prototxt”训练网络结构文件。另外每种网络对应一个deploy.prototxt测试网络结构文件，供单独进行图片分类测试使用。deploy文件仅需要在对应的训练网络结构文件中，重新定义输入层，删除每一层的参数初始化，删除准确率层和loss层，并且输出层改用softemax类型。

3、训练网络中临时保存的网络模型和参数文件位于model文件夹下，在mymodel文件夹下则存有各种网络一个训练效果较好的模型。此外，对alexnet进行微调训练使用的官方model：bvlc\_alexnet.caffemodel也存在此文件夹下。

4、log文件夹下存放训练时的训练日志。可以通过./xxxxx.sh >& xxx.log & 在启动训练时将训练日志重定向到log文件。并且训练完成后，可以通过~/caffe/tools/extra/parse\_log.py 程序对log文件进行解析，分别提取训练和测试的迭代数、学习率、准确率、损失函数等信息。

5、python文件夹下主要有以下几个程序：

①image\_agu.py 此程序对原始数据进行翻转，扩充数据，需要安装python-opencv。

②my\_classefication.py 此程序可以进行数据分类测试。对于输入的每一张图片进行分类，输出属于每一标签的概率值，取概率最大的一个标签作为分类结果。程序需要设置每个网络的deploy.prototxt测试网络结构文件，以及每种网络保存的训练模型.caffemodel。程序最终统计测试集中每一个类别的准确率。

③result\_show.py此程序通过parse\_log解析出的log.train和log.test文件，绘画测试准确率和损失函数曲线。