**2017级机器学习方向第三次竞赛说明文档**

报告人：安博洋

**第一次尝试：**

当我拿到数据集时，最开始想到的算法模型便是神经网络，由于图片数据量很大，而且输出项也十分的复杂，所以我感觉只有用神经网络模型才是合适的。最开始，我觉得因为输出结果比较复杂，是一个多分类并且是多类别输出的模型，所以，我在想可不可以把这个输出多类别的问题转换为一个多分类但输出一个类别的模型，就像softmax模型一样。从网上查阅资料，有的方法说可以把一张验证码图片分割为5个部分，相当于把这个问题变为一个62类别的分类问题（26个小写英文字母，26个大写英文字母和10个数字），神经网络模型的输出层只输出一个类别。于是，我便开始着手做。因为验证码是一个彩色的三通道的图片，含有的像素信息太大，干扰项也太多，于是，我便决定把彩色图片转换为单通道的灰度图像。然后还有一个问题，对于将要识别的验证码，显然，里面出现了很多用于干扰作用的灰色线条。我便通过设定灰度阈值（默认100），对图像中大于阈值的像素，赋值为255（灰度图像中像素值范围是0~255，其中255是白色，0是黑色）。然后又通过滤波去了一下噪声点，然后切割图片，分别保存在62个对应的文件夹中，用于后期的训练。当我做完这些后，发现一个问题，这些给的数据集不太好，转换为灰度，又通过滤波去噪，好多有用的信息字符也被去掉了，效果很不好，对后期的训练肯定会有很大的影响，于是便开始想其他图像的处理方法。

**第二次尝试：**

后来，我去问了一下数字图像处理的张老师，如何对这种图片进行处理，她给我提出了一个建议，可以运用聚类。于是，下去我便自己开始是实现。我运用DBSCAN（密度聚类）的方法，通过调节参数查看聚类最好的效果，最终实现了，用人眼看上去效果挺好的。我看到他周围还是有一些线条，我便用了一个暴力的方法，将每个字母数字目标区域四周4个像素值全部置为背景色，然后进行训练，结果最后效果还是不理想。

**第三次尝试：**

后来，听同学说，他们的正确率可以达到50%多，经过请教，他们用的神经网络是一个叫vggNet的模型，而且不用对图片进行处理，直接用原来的猜测彩色图像进行训练。于是，我便参考他们的方法，进行训练和预测，形成最终的结果。

代码中各个函数，类的功能简单介绍如下：（核心代码解释在代码.py 文件中以注释方式写出）

**class** Codeimg(Dataset):

*# 读取训练集的图片数据，并整理为规定的格式*

**class** ResidualBlock(nn.Module):

*# 卷积层（用于提取图片中的特征信息）*

**class** ResNet(nn.Module):

*# vggNet 神经网络*

**def** ResNet18():

*# 实例化 vggNet 神经网络*

**class** UnCodeimg(Dataset):

*# 读取测试集中的图片数据，进行预测*