**2 车牌识别**

1. 目的

利用TensorFlow构造卷积神经网络模型，实现普通机动车车牌字符的识别。

1. 原理

仅考虑最简单的中国普通机动车车牌的识别，车牌示意如图1所示，将车牌划分为**三个区域**：

（1）省、自治区、直辖市简称

（2）发牌机关代号

（3）车牌序号

**默认车牌字符已经得到划分**，车牌识别分解为三个区域的字符识别任务（多分类任务），共实现**7个字符**的识别。



图1 车牌示意图

1. 工具及数据集

数据集及结果在百度网盘下载（https://pan.baidu.com/s/1csUMt4u9siDaeRRitYOReA），目录划分如下：

dataset

|— train

|— province

|— area

|— letter

|— val

|— province

|— area

|— letter

数据集数据量不大，因此仅划分为训练集train和验证集val，均包含province、area和letter三个目录，且每个目录下按照其所包含的字符划分目录存储对应字符的灰度图，大小为20\*20。三个区域包含的字符如下。

Province：

("皖", "沪", "津", "渝", "冀", "晋", "蒙", "辽", "吉", "黑", "苏", "浙", "京", "闽", "赣", "鲁", "豫", "鄂", "湘", "粤", "桂", "琼", "川", "贵", "云", "藏", "陕", "甘", "青", "宁", "新")

Area：("A","B","C","D","E","F","G","H","I","J","K","L","M","N","O","P","Q","R","S","T","U","V","W","X","Y","Z")

Letter：

("0","1","2","3","4","5","6","7","8","9","A","B","C","D","E","F","G","H","J","K","L","M","N","P","Q","R","S","T","U","V","W","X","Y","Z")

1. 步骤

三个区域识别分类采用同一模型结构，利用对应的数据分别进行训练，模型结构如图2所示。



图2 模型结构图

模型结构分为4层：输入为20\*20的二值图像，第一层为卷积层，包含16个卷积核，尺寸为3\*3，卷积步长为1，后接一个尺寸为2\*2，步长为2的最大池化；第二层为卷积层，包含32个卷积核，尺寸为3\*3，卷积步长为1，后接一个尺寸为1\*1，步长为1的最大池化；然后将第二层输出扁平化，第三层为全连接层，将图像特征映射到512维向量；第四层为全连接Softmax分类层，Num为三个区域包含类别数。

数据准备、模型构建和模型训练三个步骤。

（1）数据准备

数据集为一系列20\*20的灰度图像，将图像读入并转换为TensorFlow能够进行处理的数据。使用OpenCV库或者PIL库遍历数据集对应目录，打开图像。

（2）模型构建

基于TensorFlow，**采用分类任务常用的交叉熵损失**；训练优化采用**Adam**，**初始学习率设置为1e-4**。

（3）模型训练

模型的训练过程则是一个循环过程，每一步循环准备训练的数据并送入模型。为了检测模型训练过程，每间隔一段时间步，将验证集数据送入模型中，获取模型在验证集上的准确率并输出。（环境：tensorflow -gpu+ python3.5）

1. 代码结构

Dataset：数据集；

SimpleNetwork.py：利用TensorFlow框架定义了卷积网络结构；

train\_test\_model.py ：车牌三部分的数据处理、网络训练及预测模块

1. 训练和测试命令

训练命令：

Python train\_test\_model.py area train

Python train\_test\_model.py letter train

Python train\_test\_model.py province train

测试命令：

Python train\_test\_model.py area predict ./test\_images/K.bmp

Python train\_test\_model.py letter predict

Python train\_test\_model.py province predict ./test\_images/thu.jpg

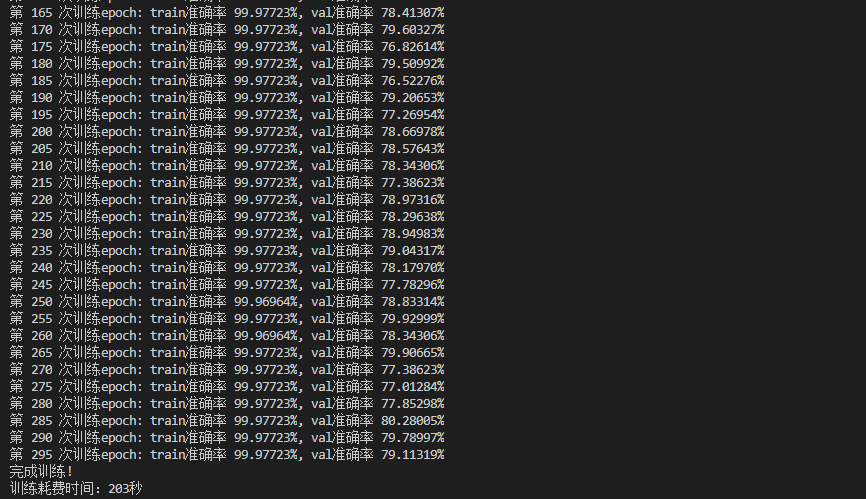
1. 结果

1.Letter

训练周期：300epoch

训练时间：203秒

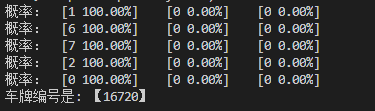
训练集精度：99.977%，验证集精度：79.113%



预测图片

34567

预测结果

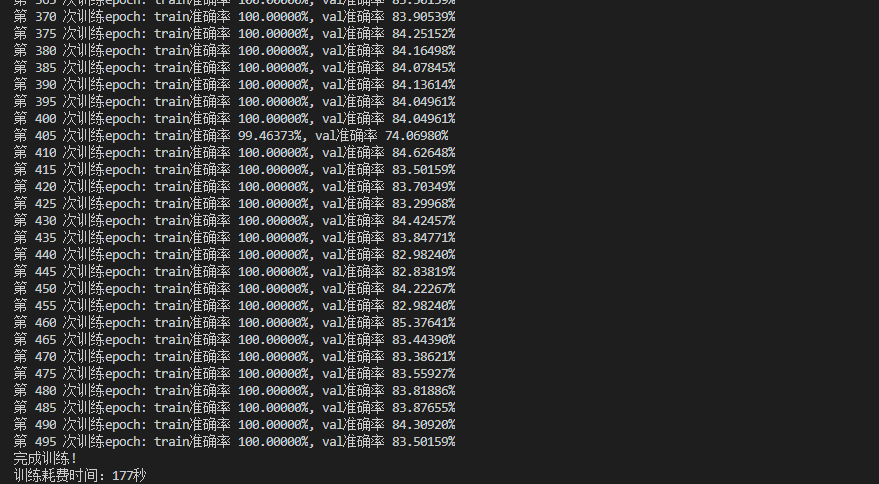


2.Area

训练周期：500epoch

训练时间：177秒

训练集精度：100%，验证集精度：83.502%



预测图片

K

预测结果

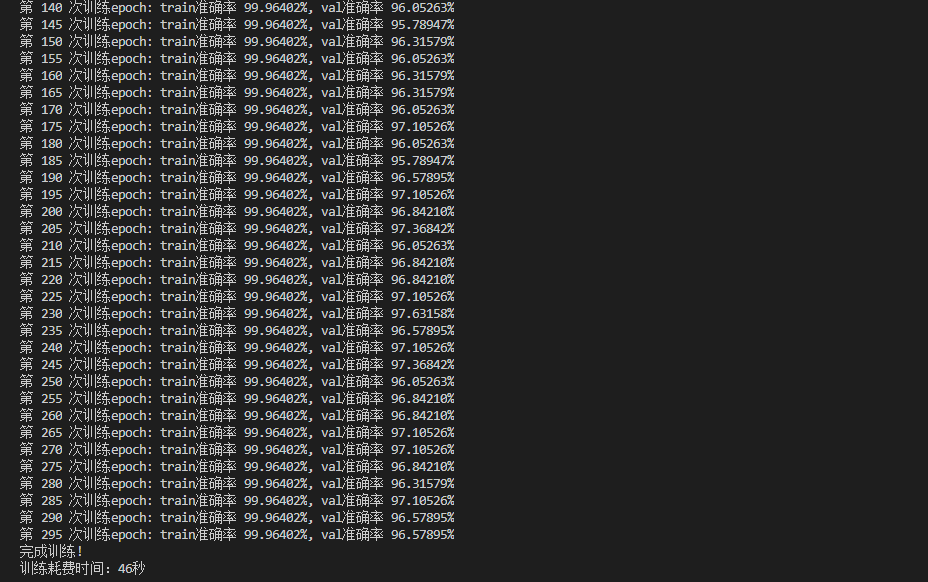


3.Province

训练周期：300epoch

训练时间：46秒

训练集精度：99.9640%，验证集精度：96.5789%



预测图片

thu

预测结果

