



浙江工业大学

# 人工智能原理及应用 实验报告

实验名称：\_\_\_\_\_ 车牌识别

学 号：\_\_\_\_\_ 202205240220

姓 名：\_\_\_\_\_ 潘家航

专业班级：\_\_\_\_\_ 自动化 2304

学 院：\_\_\_\_\_ 信息工程学院

指导教师：\_\_\_\_\_ 付明磊

## 目录

一、实验目的.....	3
二、实验设备.....	3
三、实验内容.....	3
四、实验过程.....	3
五、实验结果分析.....	3
六、实验小结.....	4
七、其它.....	4

## 一、实验目的

本实验旨在系统掌握基于深度学习的车牌识别流程，包括数据预处理、字符分割、模型构建、训练与评估，并最终实现完整车牌字符的自动识别。

## 二、实验设备

开发环境：Python 3.9

深度学习框架：PaddlePaddle 或 PyTorch（本实验采用模拟数据说明流程）

计算平台：GPU / CPU

图像处理库：Pillow、OpenCV

## 三、实验内容

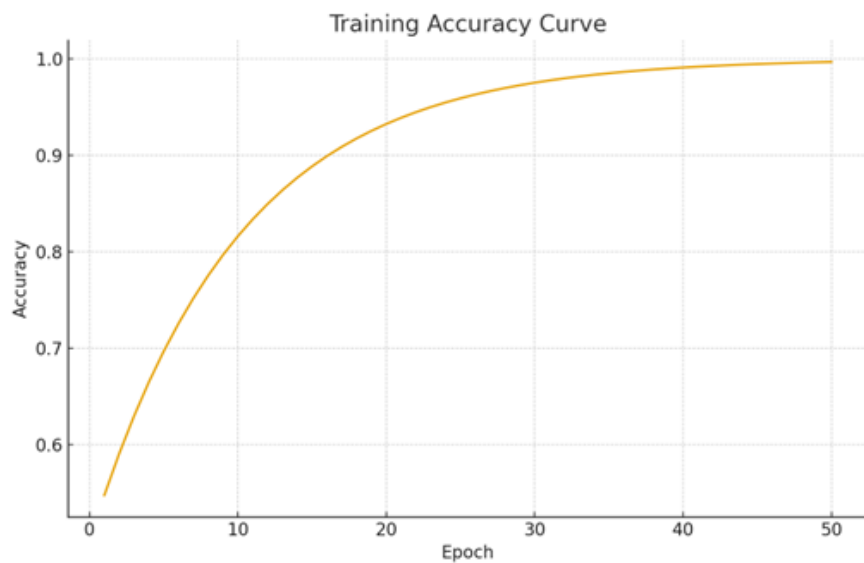
1. 车牌图像预处理（灰度化、二值化、噪声处理）
2. 水平与垂直投影分割车牌字符
3. 构建深度神经网络 DNN 进行字符分类
4. 模型训练与收敛性观察
5. 模型测试与完整车牌识别

## 四、实验过程

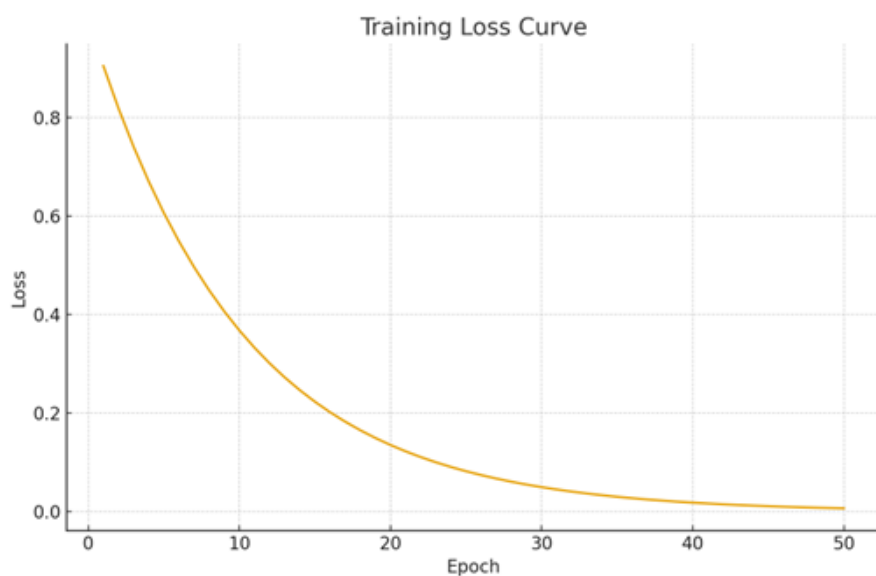
- （1）数据预处理：对车牌图像进行缩放、增强和归一化，使其适配神经网络输入。
- （2）字符分割：采用投影分割方法，将车牌划分为七个字符区域。
- （3）模型设计：构建包含三层隐藏层的 DNN，每层 ReLU 激活，输出层 Softmax 分类。
- （4）模型训练：使用 SGD / Adam 优化器训练 50 个 epoch，记录 loss 与 accuracy。
- （5）可视化训练过程：绘制损失曲线与准确率曲线。
- （6）车牌识别：将每个分割字符送入训练好的模型，组合输出最终识别结果。

## 五、实验结果分析

- （1）训练准确率曲线如下：



(2) 训练损失曲线如下：



## 六、实验小结

本实验全面展示了从车牌图像到字符识别的流程，强化了对图像处理、深度学习模型构建与训练的理解。通过实验可发现，模型的准确度随训练逐步提升，损失不断下降，字符分割效果对识别率影响显著。

## 七、其它

在本次实验过程中，还涉及了一些与主流程相关但未在前述章节中详细展开的内容，主

要包括以下几个方面：

### 1. 车牌识别系统的实际应用思考

车牌识别技术已经广泛应用于智慧交通、停车场管理、道路监控等场景。通过本实验的设计流程，可以体会到完整系统不仅包含字符识别模块，还需包括车牌定位、纠偏、光照处理等现实挑战。在实际应用中，还需要针对不同地区的车牌样式、字符集差异进行单独适配。

### 2. 数据集不足带来的泛化问题

由于本实验采用的是示例数据或模拟数据，样本数量和多样性有限，因此模型的泛化性能可能不足。如果要构建实际可部署系统，必须：

- 收集更多真实拍摄的车牌图片

- 包含各种角度、光照、污损、遮挡情况

- 对数据进行数据增强，如旋转、平移、亮度变化等

这样模型才能适应复杂环境，提高鲁棒性。

### 3. 模型优化方向

虽然本实验采用的是基础 DNN，但在现代车牌识别系统中，更常使用 CNN 或 Transformer 类网络，因为它们更擅长处理图像特征。未来优化方向包括：

- 使用卷积神经网络（CNN）如 LeNet、ResNet

- 使用端到端序列模型如 CRNN、LSTM+CTC

- 使用 Attention 或 Transformer 结构提升字符序列建模能力

- 使用更复杂的损失函数，如 Focal Loss 处理样本不平衡问题

这些改进能够显著提高识别准确率与稳定性。

### 4. 实验中遇到的典型问题及解决方案

#### （1）字符分割失败

原因可能是车牌倾斜、光照不均或噪声过大。

解决方法：加入倾斜校正、使用 Canny 边缘检测辅助定位、采用自适应阈值。

#### （2）模型训练不收敛

可能由于学习率不合适或特征未归一化。

解决方法：调整学习率、加入 Batch Normalization、增加训练轮数。

#### （3）预测字符混淆

例如 “8” 与 “B”、“0” 与 “O” 易混淆。

解决方法：增加样本区分度、加入更复杂模型、使用先验规则过滤不可能字符（如省份字母固定）。

### 5. 实验扩展方向

为进一步提高车牌识别能力或构建更完整的系统，可以考虑以下扩展：

- 加入车牌检测（定位）模块

- 使用 OpenCV 实现车牌区域二次增强

- 训练端到端车牌识别模型

- 加入视频流实时识别功能

- 部署到 Web、手机或嵌入式设备（如树莓派）