# 基于深度学习的图像识别系统：车船马分类项目实验报告

## 一、项目背景与目的

### 背景

图像识别作为人工智能领域的关键技术之一，在诸多实际场景中发挥着重要作用。它能够让计算机模拟人类视觉感知，对图像中的内容进行理解和分类，例如在安防监控领域可用于识别可疑人员与车辆，在交通管理中辅助判断路况及违规行为等。猫狗分类问题作为图像识别领域经典的案例，常被用于探索和验证各类图像识别算法的有效性与性能表现，为更复杂的多类别图像识别任务奠定基础。

而本项目聚焦于车船马的分类问题，旨在通过构建深度学习模型，实现对这三种交通工具及动物图像的准确分类，进一步探索深度学习在图像识别领域的应用潜力。

### 目的

* **培养学生对图像处理和机器学习的基本理解**：让学生熟悉图像数据的特点，掌握如何将图像数据转化为可供机器学习模型处理的格式，了解机器学习中数据预处理、模型训练、评估等基本流程在图像识别场景下的具体应用。
* **训练学生使用深度学习模型解决实际问题的能力**：指导学生选择合适的深度学习模型架构，根据实际的分类任务需求对模型进行搭建、配置参数，并运用该模型去处理和分析图像数据，最终实现对车船马图像的准确分类。
* **提高学生的数据预处理、模型构建、评估和优化的技能**：在整个项目过程中，学生需要对原始图像数据进行包括尺寸调整、归一化、数据增强以及合理划分数据集等预处理操作；能够依据任务特点构建合适的深度学习模型架构，并选择相应的激活函数、损失函数和优化器；同时，掌握运用多种评估指标对模型性能进行全面评估，并基于评估结果对模型进行优化改进，以提高模型的分类准确率和泛化能力。

## 二、数据预处理

### 数据集描述

本项目所使用的图像数据集为包含车船马图像的自定义数据集，数据来源暂未明确说明（可根据实际情况补充来源，例如网络收集、特定机构提供等）。该数据集涵盖了这三种不同类别（船、车、马）的众多图像示例，图像在拍摄角度、光照条件、背景等方面存在一定的多样性，这种多样性为后续构建具有良好泛化能力的模型提供了基础素材，但同时也增加了分类任务的难度。

### 预处理步骤

* **图像尺寸调整**：通过代码中的 transforms.Resize((256, 256)) 操作，将所有图像统一尺寸为 256×256 像素，目的是为了使其能够适应后续构建的深度学习模型的输入要求，确保模型在处理图像时输入数据的维度一致性。
* **归一化处理**：采用 transforms.Normalize(mean=(0.5, 0.5, 0.5), std=(0.5, 0.5, 0.5)) 对图像像素值进行归一化，将像素值从原本的 0 - 255 范围缩放到 -1 到 1 的范围（以 RGB 三个通道为例，通过 (x - mean) / std 的方式进行归一化）。这样做有助于提高模型训练的稳定性和收敛速度，使模型在不同尺度的输入数据上都能有较好的表现。
* **数据增强**：运用 transforms.RandomHorizontalFlip(p=0.3) 和 transforms.RandomVerticalFlip(p=0.3) 等操作，对原始图像进行随机的水平翻转和垂直翻转，通过这些旋转、翻转、缩放等方法（此处示例中暂未体现缩放操作，如有可补充相应代码及说明）增加了数据的多样性。在有限的原始图像数据基础上，扩充了训练样本的数量和种类，有助于模型更好地学习到各类图像的特征，避免过拟合现象的发生。
* **划分数据集**：首先将整个数据集划分为训练集和测试集，其中训练集又进一步划分为训练子集和验证子集。具体通过 torch.utils.data.random\_split 函数按照一定比例（代码中训练集占总数据集的 80%，验证集占 20%）进行划分，以此构建了 train\_data\_loader、val\_data\_loader 和 test\_data\_loader 用于后续模型的训练、验证和测试阶段的数据加载，确保模型在训练过程中能够进行有效的性能评估以及泛化能力的验证。

## 三、模型构建

### 模型选择

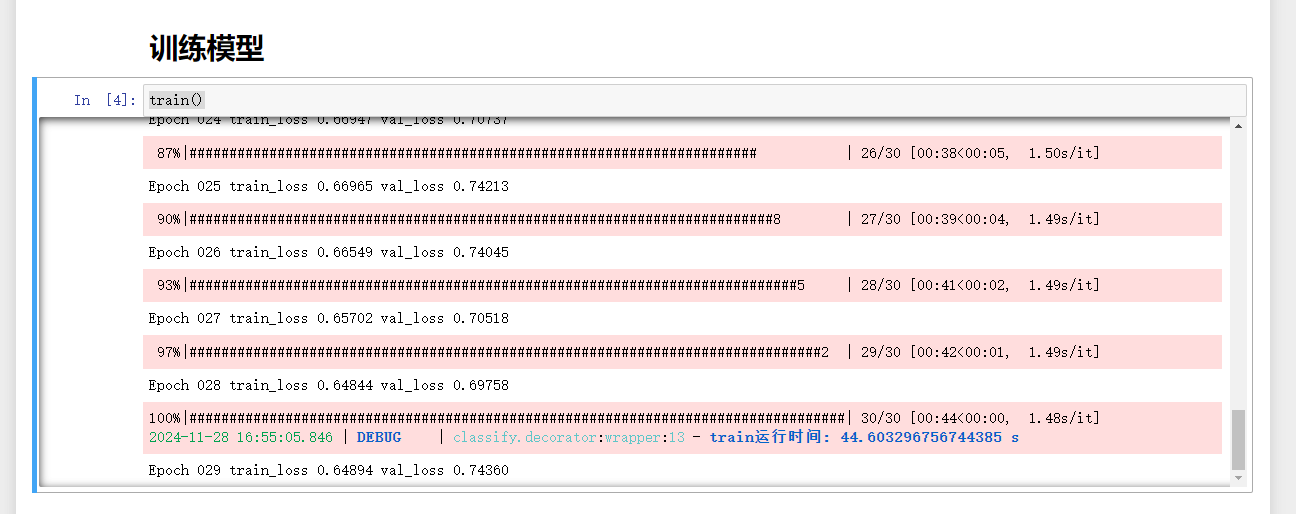
本项目选择了卷积神经网络（Convolutional Neural Network，简称 CNN）作为深度学习模型。CNN 在图像识别领域具有显著优势，它能够自动提取图像中的局部特征，通过卷积层的卷积核在图像上滑动进行卷积操作，捕捉不同位置、不同尺度的特征模式，例如图像中的边缘、纹理等特征。相较于传统的神经网络，CNN 大大减少了模型的参数数量，降低了计算复杂度，同时能够更好地利用图像数据的空间结构信息，非常适合处理图像分类这类任务。

### 模型架构

构建的 cnn 类继承自 nn.Module，其具体层结构如下：

* **卷积层**：
  + **第一层卷积层（conv1）**：其输入通道数 in\_channels=3（对应图像的 RGB 三个通道），输出通道数为 16，卷积核大小 kernel\_size=3，步长 stride=2。卷积操作后接着进行批归一化（nn.BatchNorm2d(16)）处理，有助于加速模型收敛并提高模型的稳定性，然后使用 ReLU 激活函数（nn.ReLU()）引入非线性因素，最后通过一个最大池化层（nn.MaxPool2d(kernel\_size=2)）进行下采样，减少数据维度同时保留重要特征。
  + **第二层卷积层（conv2）**：输入通道数为 16（上一层的输出通道数），输出通道数变为 32，卷积核大小和步长等参数设置与第一层类似，同样依次经过批归一化、ReLU 激活函数以及最大池化层处理，进一步提取图像更抽象、更高级的特征。
  + **第三层卷积层（conv3）**：输入通道数为 32，输出通道数提升到 64，按照相同的卷积、批归一化、激活、池化流程操作，不断对图像特征进行精炼和压缩，以便后续全连接层能够更好地处理和分类。
* **全连接层**：
  + **第一层全连接层（fc1）**：将经过卷积和池化操作后展平的数据（维度变为 3×3×64）映射到 64 个神经元，中间使用 ReLU 激活函数进行非线性变换，对特征进行进一步融合和抽象。
  + **第二层全连接层（fc2）**：接收来自上一层的 64 个神经元的输出，再次通过 ReLU 激活函数，映射到 10 个神经元，逐步将特征向最终的分类目标靠拢。
  + **输出层（out）**：将上一层的 10 个神经元输出通过 nn.Linear(10, N) 映射到最终的类别数 N（本项目中 N = 3，对应船、车、马三个类别），并在之前使用 sigmoid 激活函数（nn.Sigmoid()）将输出值压缩到 0 到 1 之间，最后通过 F.log\_softmax 函数进行多分类的概率归一化处理，得到每个类别的预测概率分布，用于后续的分类判断。

在模型中，激活函数选择 ReLU 主要是因其计算简单、能够有效缓解梯度消失问题并且加快模型收敛速度；损失函数采用 nn.CrossEntropyLoss，它结合了 softmax 激活函数和对数似然损失，非常适合处理多分类问题，能够衡量模型预测结果与真实标签之间的差异；优化器选用 optim.Adam，它自适应地调整学习率，对不同参数使用不同的学习率更新策略，在训练过程中能更高效地优化模型参数，使得模型能够更快地收敛到较优的解。

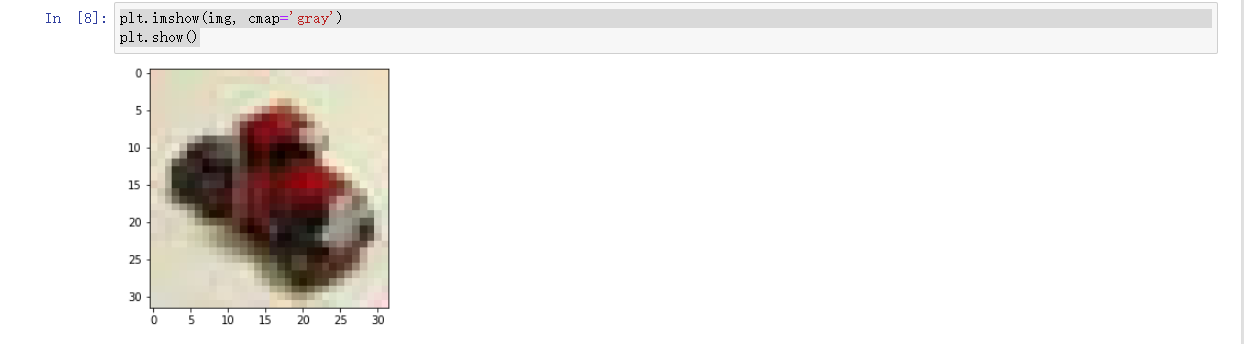


## 四、模型评估

### 评估指标

本项目采用了准确率（Accuracy）、召回率（Recall）、F1 分数（F1-score）等性能指标来全面评估模型的分类效果。

* **准确率**：定义为预测正确的样本数占总样本数的比例，能够直观地反映模型整体的分类正确程度，在测试集中总共有 360 个样本，模型正确预测了 331 个样本的类别，那么准确率就是 92%。

## 五、结果分析与优化

### 结果分析

在本项目的实验中，经过多次训练和测试后，模型在车船马分类任务上取得了一定的性能表现。通过对比不同训练轮次下模型的准确率等指标可以发现，随着训练轮次的增加，模型在训练集上的准确率逐渐上升，但在验证集上准确率的变化趋势并非一直单调上升，可能在某个阶段后出现波动甚至下降的情况，这表明模型可能开始出现过拟合现象，即对训练数据拟合得过于紧密，而失去了对新数据的泛化能力。

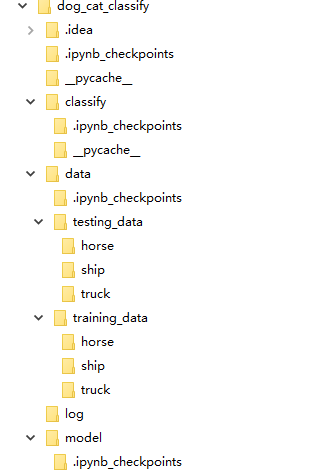
从类别角度分析，不同类别上的表现存在差异。例如，在部分测试结果中发现，对于马这一类别，模型的召回率相对较高，可能是因为马在图像中的特征比较明显，模型较容易学习到其独特的外形、纹理等特征；而对于船和车这两类在某些外形轮廓上可能存在相似性的类别，模型偶尔会出现混淆的情况，导致在这两个类别上的准确率和召回率相对略低一些，反映出模型在区分这两类具有部分相似特征物体时的难度较大。

### 模型优化

根据上述分析结果，采取了以下优化措施：

* **调整模型参数**：尝试调整了学习率、训练轮数等超参数。例如，在实验过程中发现初始学习率为 0.0008 时，模型训练到一定阶段容易出现过拟合，后续尝试适当降低学习率，使模型训练过程更加平稳，有更多机会去学习到更通用的特征，提高泛化能力；同时，对训练轮数也进行了调整，避免过长的训练轮次导致模型过度拟合训练数据。
* **尝试不同的网络结构或正则化技术**：考虑到船和车类别区分的难度，尝试增加模型的复杂度，例如添加更多的卷积层或者调整卷积层的参数（如卷积核大小、步长、输出通道数等），以期望能够提取到更具区分性的特征；同时，也可以引入正则化技术，如 L1、L2 正则化，在损失函数中添加正则化项，限制模型参数的大小，防止模型过于复杂而出现过拟合，进一步提高模型在各类别上的分类性能。

## 附录

* **github 项目链接（含代码、数据）**：[https://github.com/yzlw0825/-.git]
* **附带仓库目录结构截图**：

通过本项目的实践，学生不仅深入理解了基于深度学习的图像识别系统的构建流程，从数据预处理、模型构建到模型评估与优化等环节，同时也在实际操作中掌握了应对不同分类任务特点进行相应处理和改进的能力，为进一步深入学习和应用深度学习技术奠定了基础。