

# 基于卷积神经网络(CNN)的动物识别

代昕冉<sup>1</sup>, 朱恩泽<sup>1</sup>, 张杰铭<sup>1</sup>

指导教师: 赵俊<sup>2</sup>

XINRAN DAI<sup>1</sup>, ENZE ZHU<sup>1</sup>, JIEMING ZHANG<sup>1</sup>

Supervisor: JUN ZHAO<sup>2</sup>

2025 年 5 月

<sup>1</sup> XJTU-YGP

<sup>2</sup> SHSSIP

## 摘要

近年来,为了解决更复杂的问题,人工智能技术的实现大多都采用机器学习和深度学习的相关算法。图像识别是人工智能领域一个很重要的分支,而分类任务是图像识别领域的一个重点问题。本文通过搭建神经网络在本组获取的数据集上完成分类识别任务,并通过实验分析各种操作对最终的拟合与准确率产生的影响。

**关键词:** 卷积神经网络; 分类任务; 动物识别

## 一、研究背景

近年来,随着人工智能技术的发展,图像识别和分类已经发展成为一个重要的课题,而深度学习中的神经网络在处理图像识别和分类问题领域中展现出了卓越的性能。因此,本项目希望通过研究性学习的形式完成一个简单的识别任务并以此学习有关神经网络的相关知识,同时将以学习为主,而非追求创新性的研究成果。

人类拥有着强大的视觉神经系统,可以在一瞬间识别出大部分动物的图像。然而,由于动物种类的及其丰富的多样性,人类意识会陷入先天的直觉、后天的见识限制和特殊情况的困境中。为此,可以尝试通过机器学习和深度学习的方式来学习不同的规则特性,让计算机代替人脑自主做出判断。由于卷积神经网络处理图像问题的优越性,本项目选择利用卷积神经网络来尝试实现动物识别。

本项目从卷积神经网络的基本原理入手,逐步了解其网络结构、训练过程、优化方法及经典模型。同时结合具体的动物识别任务,通过实践操作加深对卷积神经网络的理解。

## 二、卷积神经网络原理

通过初期探索,我们认识到此次研究性学习的本质在于学习而非严格意义上的科研。基

于进一步深入理解卷积神经网络的内在动机，本项目决定采用控制变量法，系统分析不同策略对模型性能的影响。考虑到研究条件与标准化、高质量数据的要求，本研究主要依托文献资料及现有大型模型进行探讨。为兼顾实验可行性与结果有效性，需选择一个在性能与计算资源需求上均处于中等水平的模型：若模型规模过小或资源需求过低，可能导致结果不具备足够讨论价值；而若规模过大或资源需求过高，则易受到随机因素干扰，影响研究的科学性与可靠性，同时亦受限于实际可用资源。因此，诸如 LeNet-5、ResNet 等网络结构因性能相对不足或过高而被排除。最终，选定 VGG 系列中结构简明、对超参数及优化策略调整较为敏感、且易于出现过拟合现象的 VGG16 模型。该模型在性能与复杂度之间取得平衡，有助于放大不同处理方式带来的效应差异，从而便于对比分析。然而，在后续训练过程中发现，VGG16 对计算资源的需求较高，因此亦引入 MobileNet 网络进行对比研究，以期在可接受的资源消耗下，通过适度调整模型规模实现更高效的实验流程。

## 2.1 基本组成

不同于一般的神经网络，卷积神经网络由卷积层、激活函数、池化层与全连接层组成。其中卷积层利用提前设置好的权重与偏置进行加权计算（卷积运算），得出对应的特征值，并输出特征值组，基于这种操作实现所谓的“特征提取”，即抓取图像间相邻区域的通性。

激活函数处理接收到的数据，并进行非线性变换，令数据符合卷积运算的范围并提高数据的“现实性”，以防止过拟合并赋予神经网络解决非线性问题的能力。

池化层将数据整合，保存主要信息并删除冗余信息，降低数据的宽度和高度，大幅减小参数的数量，提高对相似特征图像的识别能力并防止过拟合。

全连接层全连接处理后的特征值与最终的输出结果，降维的同时整合前述的特征，并完成分类任务。

除此之外，在处理分类任务时选择了交叉熵损失函数与 Adam 优化器。

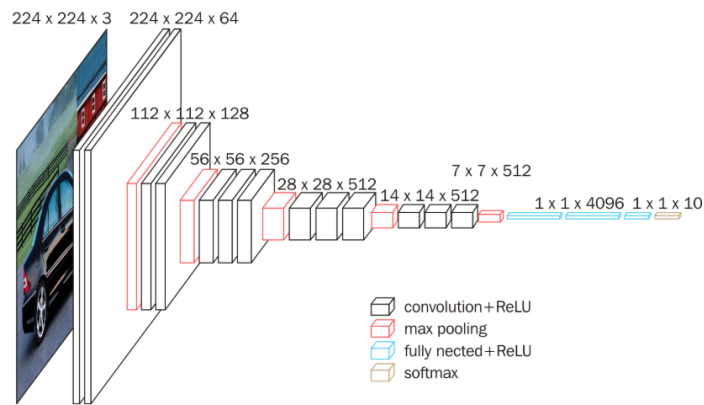
交叉熵损失函数表示了真实概率分布与预测概率分布之间的差异，公示如下：

$$L = \frac{1}{N} \sum_i L_i = -\frac{1}{N} \sum_i \sum_{c=1}^M y_{ic} \log(p_{ic})$$

Adam 优化器追踪之前的梯度，分析梯度的变化，进而动态改变学习率，这是一种自适应学习率的梯度下降算法。

## 2.2 VGG16 模型结构

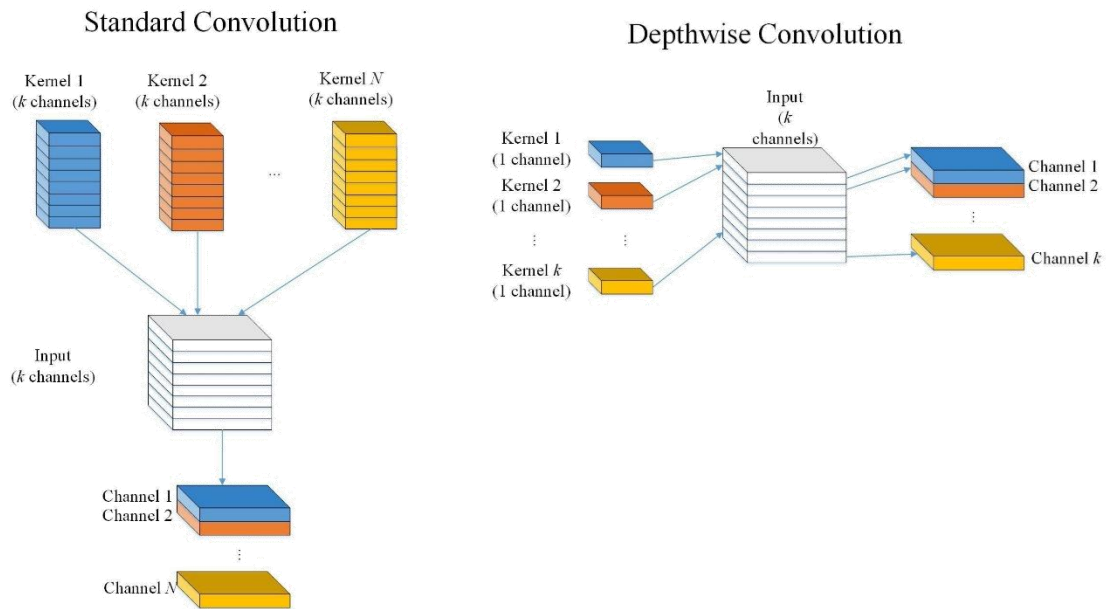
VGG16 模型采用以下结构：



VGG16 模型结构

## 2.3 MobileNet 模型结构

MobileNet（这里的 MobileNet 指的是 V1，下同）模型采用以下结构：



MobileNet 模型结构

需要说明的是，MobileNet 的核心思想是通过卷积分解来降低标准卷积核的参数量。它用深度可分离卷积使得每一个卷积核只关于一个输入特征矩阵，并用  $1 \times 1$  的传统卷积层将所有独立的输出信息相联系起来。

## 2.4 通用原理

整个训练过程旨在读取到每一类中的独有特征，并把结果向同类型未知数据泛化。每一个卷积层都在读取它们可以感受到的图像相邻特征，并尝试利用优化器更新迭代，再利用反向传播缓慢逼近正确的极小点。线性的卷积层特征提取与非线性的激活函数、池化层泛化是卷积神经网络读取特征的关键所在，学习率、损失函数与优化器则控制着模型的收敛速度。

## 三、学习阐释与分析

### 3.1 数据集的搜集与处理

本数据集是由小组成员使用脚本爬取的动物识别数据集。该数据集包含了 11 种不同的动物，覆盖了常见的猫、狗等家庭宠物。每种动物都有大量的图片，总数达到了八千余张，平均每种约 760 张。这些图片都经过了组员的人工筛选和处理，一定程度上保证了数据的质量和准确性，但由于各种图片背景的干扰和质量上不可避免的差距，我们的数据集质量不够理想。



“鹅”的部分数据集如图所示

为了进一步处理数据集，我们用 `getdata` 程序将其储存在 `data` 文件夹中，并进一步按照 8:2 的比例分为 `train`(训练集)与 `test`(测试集)，每一种动物种类所含图片放入不同子文件夹中。这样，我们可以在主程序中直接将该数据集中的图片自动标记标签并统一处理为 `vgg16` 可调用的数据集格式，且方便直接处理为张量并进行各种数据增强，这有助于我们的下一步操作。

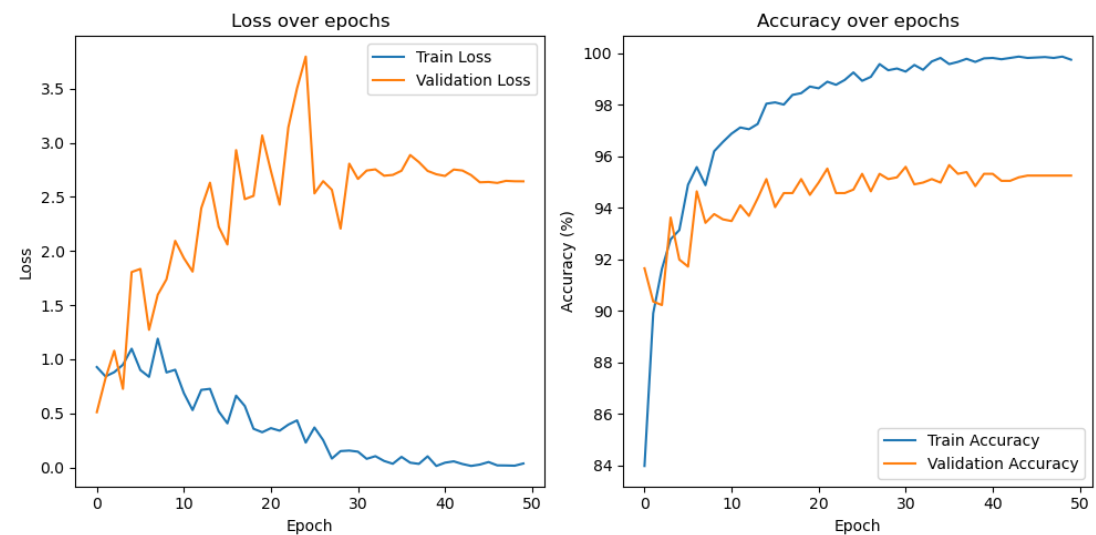
```
9 original_data_dir = r'C:\Data_name'
10
11 base_dir = r'C:\data'
12 os.makedirs(base_dir, exist_ok=True)
13 train_dir = os.path.join(base_dir, 'train')
14 test_dir = os.path.join(base_dir, 'test')
15
16 for i in range(0, size):
17     os.makedirs(train_dir, exist_ok=True)
18     os.makedirs(os.path.join(train_dir, catalo[i]), exist_ok=True)
19     os.makedirs(test_dir, exist_ok=True)
20     os.makedirs(os.path.join(test_dir, catalo[i]), exist_ok=True)
21
22 files = []
23 for i in range(0, size):
24     files.append([f for f in os.listdir(os.path.join(original_data_dir, catalo[i])) if
25                  os.path.isfile(os.path.join(original_data_dir, catalo[i], f))])
26
27 all_files = (files[0] + files[1] + files[2] + files[3] + files[4] + files[5] + files[6] + files[7] + files[8] + files[9] + files[10])
28 train_files, test_files = train_test_split(*arrays: all_files, test_size=0.2, random_state=42)
```

`getdata` 部分代码

### 3.2 VGG16 的搭建与探索

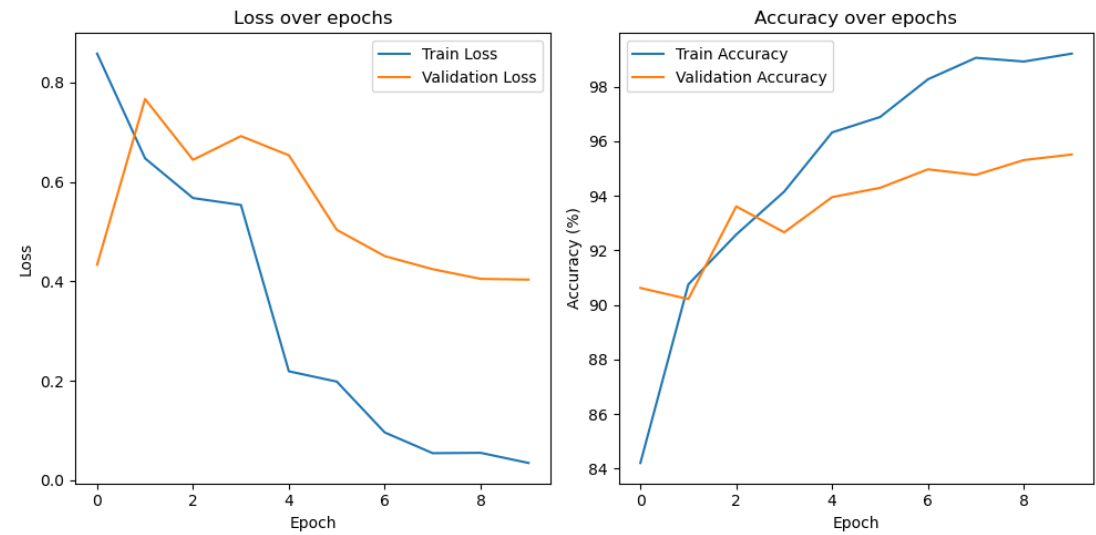
首先，我们尝试搭建了 VGG16 大模型。

初期，我们的模型在仅有 4 个分类任务下正确率仅有 65.03%，且 loss 较高。在咨询了文心一言后，我们尝试添加了一些措施，成功将该分类任务的测试正确率提升至 98.94%，这证明了我们思路的可行性。紧接着，我们修改参数至 11 类，并得到了以下结果：



模型的训练和测试结果如图所示

其中，我们采用了冻结特征提取、正则化（L2 正则化、dropout）、学习率衰减策略、早停、数据增强等一系列措施缓解过拟合并提高模型准确率，最终的实验结果显示训练数据 loss 极低、acc 极高，但测试数据 loss 极高、acc 稍高，准确率可观但呈过拟合趋势。



模型局部的训练和测试结果（已进行重新训练）如图所示

当我们将视线聚焦到训练初期时，模型质量近乎不变的情况下过拟合情况有了一定的缓解。因此，在我们的数据集及模型的前景下，epoch=10 为一个不错的训练轮次。为了筛选更

加有效的措施，接下来，我们将会以 epoch 为 10 的训练结果定为标准组（对照组），并更改模型参数或措施来观察训练结果的改变。

粗略分析下，我们决定设置以下实验组：

取消学习率衰减策略，其余保持不变。

取消 L2 正则化，其余保持不变。

取消 dropout，其余保持不变。

取消冻结，其余保持不变。

5. 取消早停，其余保持不变。

6. 更改学习率 lr（标准为 0.001，尝试更改为 0.0005 与 0.002），其余保持不变。

7. 增强（随机裁剪翻转）/取消（仅中心裁剪）数据增强，其余保持不变。

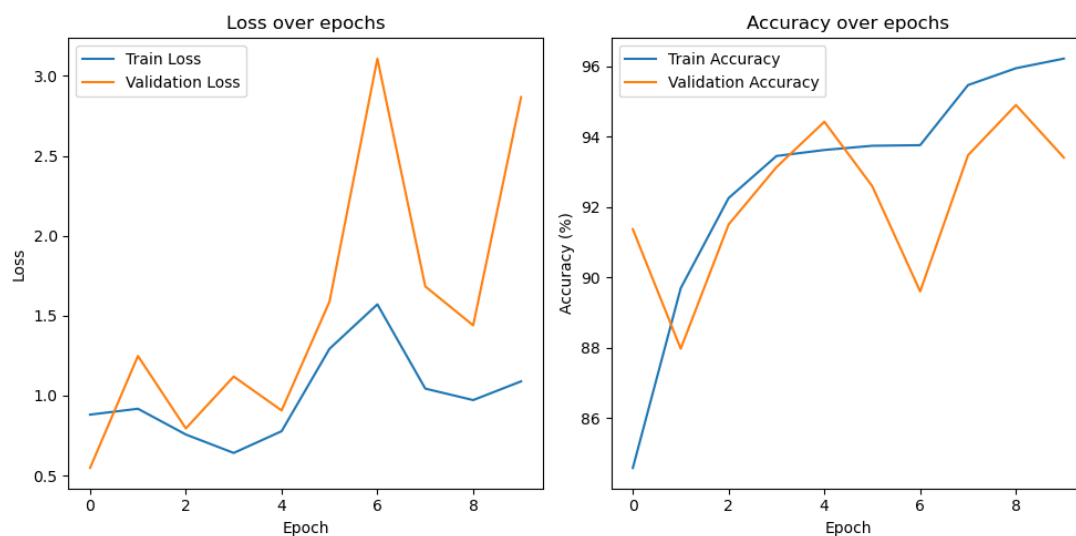
8. 改变数据集分类比例（标准为 8:2，尝试更改为 9:1），其余保持不变。

## ①取消学习率衰减策略

理论分析：由于 Adam 自带了动态调整，学习率衰减策略反而会因为学习率的下降降低后期收敛速度，但是，即使存在动态调整，失去了学习率衰减策略模型的 VGG16 也极易在极小值处震荡导致所得性能下降，只是相对于其他优化器振幅较小。

当我们拉大训练轮次后，多层次学习率衰减策略可能会避免陷入局部最优解的陷阱，在这里体现不大。

实验结果：



实验分析：与预计相似，损失提高，准确率下降，有全程震荡的可能性。

## ②取消 L2 正则化

理论分析：失去 L2 正则化后，模型拟合噪声的趋向将放大，过拟合可能会加重，训练集上的表现提高，而测试机上的表现减少。但考虑到仍有 Dropout 控制，整体变化不会过于明显

### ③取消 Dropout

理论分析：失去 L2 正则化后，模型拟合噪声的趋向将放大，过拟合可能会加重，训练集上的表现提高，而测试机上的表现减少。考虑到仍有 L2 正则化控制，整体变化不会过于明显，但 Dropout 的使用范围大于 L2 正则化，L2 正则化控制权重变化的能力在我们权重参数较弱的前提下太弱，这会导致变化明显程度大于②。

### ④取消冻结

理论分析：VGG16 的预训练权重本身就基于 1000 个分类任务产生的，这意味着微调比直接训练迅速且有效，因此，取消冻结大概率会降低整体的表现性能，同时一定程度上拉长训练时间。

### ⑤取消早停

理论分析：早停在多轮次训练中可以准确终止在局部最优处，在少轮次训练中的作用约等于 0，因此，取消与否对我们的模型无影响。

### ⑥更改学习率 lr（标准为 0.001，尝试更改为 0.0005 与 0.002）

理论分析：在学习率衰减策略与 Adam 优化器的双重保障下，一定范围内增大学习率不会对展示结果产生较大影响，但会加快开始时的收敛速度，整体波动也会更加明显。在冻结的保障下，一定范围内降低学习率可能更容易将模型收敛在最优解处，但模型的收敛速度会降低，整体波动也会更加平缓。

### ⑦增强（随机裁剪翻转）/取消（仅中心裁剪）数据增强

理论分析：数据增强会降低模型的过拟合程度，但会放大数据集中的噪声比例与混乱程度，可能会降低模型抓取同类特征的能力。数据不增强会增大模型的过拟合，放大 VGG16 模型复杂的特征，这也意味着模型抓取同类特征的能力的上升，展示性能应会有一定上升。

### ⑧改变数据集分类比例（标准为 8:2，尝试更改为 9:1）

整体上，我们发觉学习率衰减策略、Dropout、冻结、lr=0.001、数据增强产生了明显的正收益。

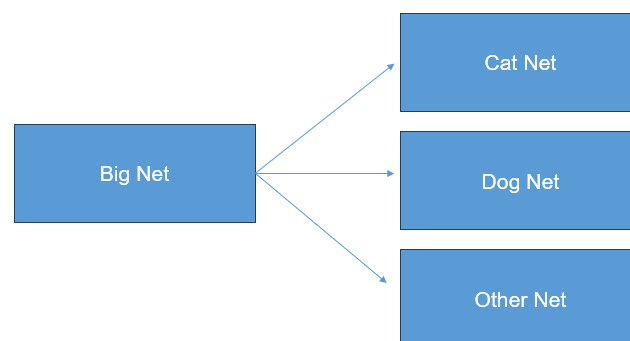
同时，在本次探索中，我们发现 VGG16 模型在图像分类任务上表现出色，但需要大量的调试调整，训练时间也稍长。

### 3.3 MobileNet 的搭建

其次，我们搭建了 MobileNet，并利用 VGG16 的经验添加了部分优化措施，它在 14 项分类任务中表现出色，且训练时间有着明显地减少。

### 3.4 Nets

在上述探索中，我们发现面对种类较少的分类任务时模型的性能表现优良，在种类较多时性能较差，于是，我们想先构建一个网络识别大类，再根据结果用下一个模型识别小类，利用多个模型保证分类任务的准确性。



在我们利用 MobileNet 的测试中，大类网络识别的准确率达到了 99.07%，在猫识别中准确率仅有 96.45%，在这个测试角度上，这种思路是可行的。

## 四、总结

通过搭建卷积神经网络的方法实现了对单个动物个体的分类识别，并在整个过程中学习了卷积神经网络的基本信息，学习了 VGG16 模型与 MobileNet 模型，在 VGG16 模型上设计了对比实验，并尝试构建了一种算法思路。

本次项目实践增强了我们对机器学习和人工智能的认识，同时也让我们更加全面的理解了卷积神经网络的基本原理和设计方法、参数的作用。

## 参考文献

- [1] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization.
- [2] Ilya Loshchilov and Frank Hutter. DECOUPLED WEIGHT DECAY REGULARIZATION.
- [3] 知乎：MobileNet 系列网络结构详解