

山东师范大学

Noise2Noise 图像修复实验报告

《计算机视觉》 Assignment 3

姓	名:	董霁兴
学	号:	2024317181
班	级:	计算机技术 2 班
学	院:	信息科学与工程学院

2024 年 12 月 9 日

目录

1 引言	2
2 实验目的	2
3 实验环境	2
3.1 硬件环境	2
3.2 软件环境	2
3.3 数据集	2
4 Noise2Noise 模型设计与实现	2
4.1 整体架构	2
4.1.1 网络结构	3
4.1.2 跳跃连接设计	3
4.2 训练策略	3
4.2.1 超参数调优	3
4.2.2 数据增强方法	3
5 实验结果分析	4
5.1 训练过程分析	4
5.2 训练效率分析	4
5.3 模型推理效果展示	5
6 实验结论	5
7 总结与思考	6
7.1 Noise2Noise 的特点	6
7.1.1 模型结构特点	6
7.1.2 优势与局限性	6
7.2 改进方向	6
7.2.1 可能的优化方向	6

1 引言

图像在采集、传输和存储过程中往往会受到噪声的干扰，导致图像质量下降，影响后续的图像分析和处理任务。因此，图像去噪成为图像处理中的一个重要研究课题。传统的图像去噪方法如均值滤波、中值滤波等在一定程度上能够减少噪声，但往往会模糊图像的细节。近年来，基于深度学习的方法如 noise2noise 为图像去噪带来了新的思路和突破。

本实验将通过实现 Noise2Noise 模型，深入理解其基本原理和实现方法。

2 实验目的

本次实验主要围绕 Noise2Noise 图像修复模型展开，通过理论学习和实践操作，达到以下目的：

1. **理解 Noise2Noise 的核心设计思想：**掌握 noise2noise 的去噪方法，理解其相比传统去噪方法的优势。
2. **理解 Noise2Noise 的架构设计：**理解编码器-解码器结构在图像重建中的作用，掌握跳跃连接对特征保留的重要性。
3. **实现简单的 Noise2Noise 模型：**手动实现 Noise2Noise 的核心组件，在一个较简单的数据集上训练和测试模型。

通过完成上述目标，加深对图像修复任务的理解，掌握深度学习模型设计和实现的基本框架，为今后进一步学习更复杂的图像修复模型奠定基础。

3 实验环境

3.1 硬件环境

本实验使用个人笔记本电脑进行训练，i7 14650HX 处理器和 RTX 4060 Laptop 显卡的设备，CUDA 版本为 12.6

3.2 软件环境

实验采用 Python 3.9，和 PyTorch 深度学习框架。自行实现了一些用于展示损失曲线的工具类。

3.3 数据集

本次实验使用的训练数据来自 ImageNet32x32 中抽取的 6250 张图像。每张图片随即添加了高斯噪声，为方便加载数据集，使用了 pytorch 的 pickle 文件格式对数据集进行打包。

4 Noise2Noise 模型设计与实现

4.1 整体架构

Noise2Noise 模型采用模块化设计，主要由编码器和解码器两部分组成，并包含跳跃连接。其整体架构如下：

4.1.1 网络结构

模型采用模块化设计方法, 主要包含以下组件:

1. 基础卷积块 (ConvLeakyReLU):

- 卷积层
- LeakyReLU 激活函数

2. 编码器模块:

- 多个卷积块串联
- 最大池化层用于下采样
- 跳跃连接的特征保存

3. 解码器模块:

- 上采样层
- 与编码器对应层的跳跃连接
- 特征融合与重建

4.1.2 跳跃连接设计

采用堆栈系统实现跳跃连接:

- 编码阶段将特征图压入堆栈
- 解码阶段依次弹出特征图进行特征融合
- 确保特征的有效传递和重建

4.2 训练策略

4.2.1 超参数调优

为了获得更好的 PSNR 分数, 我们尝试了不同的修改和超参数调整, 如不同的激活函数、引入批量归一化和优化器的学习率。

我们使用 Ray 和 HyperOpt 库在搜索空间上进行并行优化。结果如表 1所示。我们得出结论:ReLU 激活并不比 LeakyReLU 好; 使用批量归一化使训练速度慢 1.5 倍, 同时结果略差; 高学习率和小动量的效果与小学习率和大动量的效果相当。

4.2.2 数据增强方法

实验了多种数据增强策略:

- 水平和垂直翻转
- 源图像和目标图像随机交换
- 色调和亮度调整 (效果不理想, 未采用)

表 1: 不同参数配置下的模型性能对比

损失	激活函数	BN	学习率	动量	时间 (秒)
0.00375	LeakyRelu	FALSE	0.05521	0.18	3780
0.00378	ReLu	FALSE	0.00801	0.86	3796
0.00383	LeakyRelu	TRUE	0.03571	0.18	5306
0.00448	ReLu	FALSE	0.00130	0.24	2612
0.00458	ReLu	FALSE	0.00081	0.43	3774
0.00462	LeakyRelu	FALSE	0.00059	0.44	3779
0.00480	LeakyRelu	TRUE	0.00178	0.21	5345
0.00480	ReLu	TRUE	0.00404	0.31	5031
0.00494	LeakyRelu	FALSE	0.00018	0.66	3274
0.00605	ReLu	TRUE	0.00024	0.81	5227

5 实验结果分析

5.1 训练过程分析

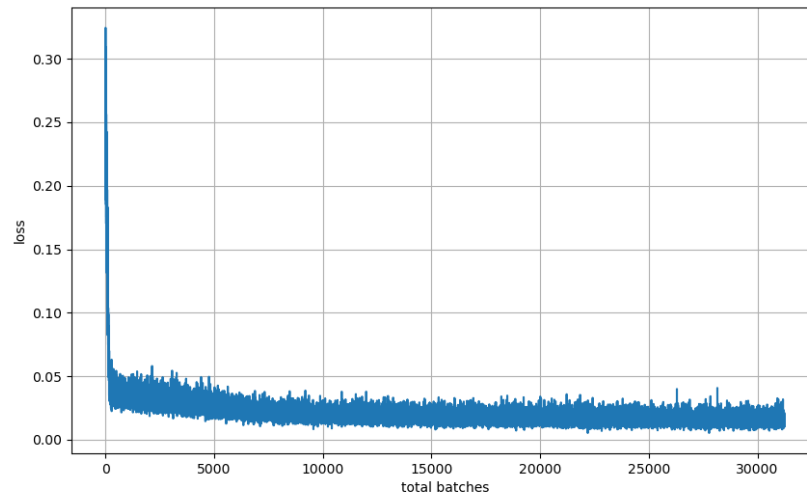


图 1: Noise2Noise 模型训练过程中的损失变化曲线

- 损失变化:

- 训练初期, 损失值急剧下降, 模型能够迅速捕捉含噪图像间的映射关系。
- 随后进入平稳下降阶段, 模型在此期间进行参数微调, 持续优化其去噪能力。
- 最终趋于稳定, 表明模型去噪能力得到了良好训练。

5.2 训练效率分析

- 硬件环境: RTX 4060 Laptop

- 单轮训练时间：约 60 秒/epoch
- 最终 PSNR：达到 25.51dB

5.3 模型推理效果展示

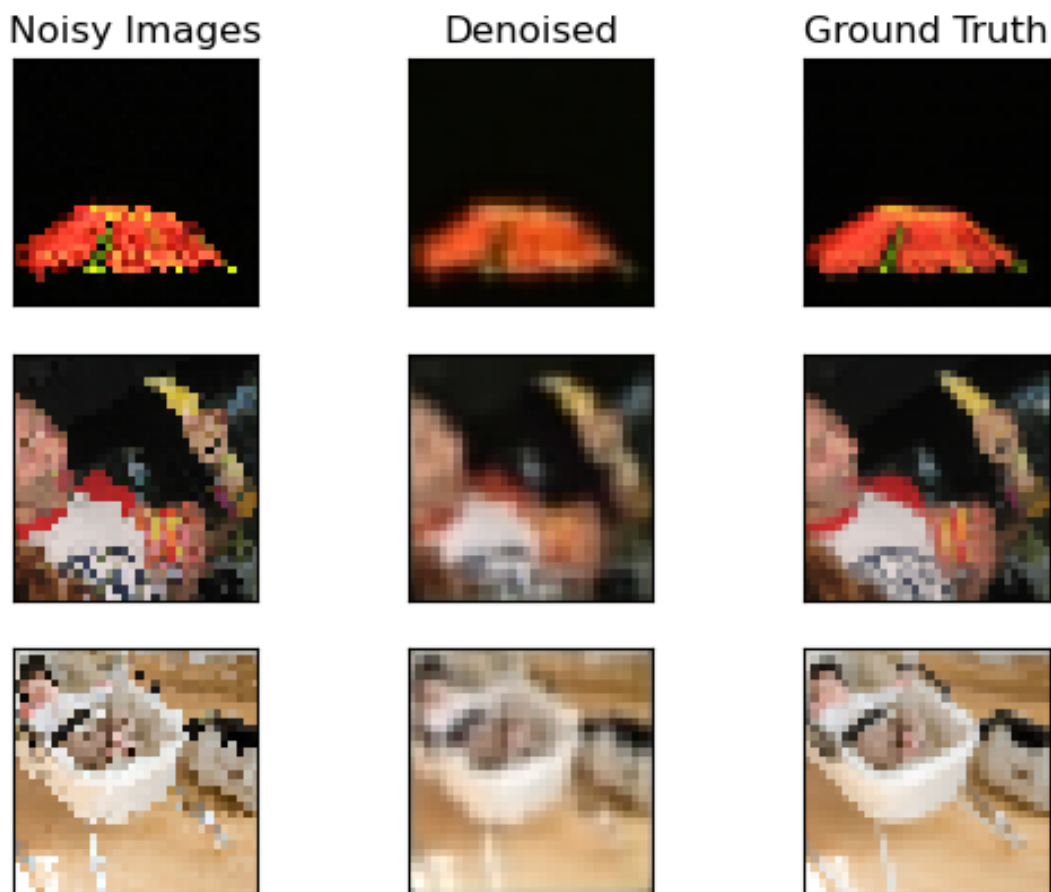


图 2: Noise2Noise 模型在不同噪声类型下的推理结果

从图 2 的检测结果可以观察到以下特点：

- 去噪成功案例：
 - 模型在大多数图像中成功去除了噪声，恢复了图像的细节
- 存在的问题：
 - 在某些高强度噪声下，去噪效果不理想

6 实验结论

本次实验通过对 Noise2Noise 模型的实现和测试，验证了其在无需干净图像作为训练目标的情况下，能够有效去除图像噪声。实验结果表明，Noise2Noise 在大多数情况下能够恢复图像的细节，但在高强度噪声下仍有改进空间。

7 总结与思考

7.1 Noise2Noise 的特点

7.1.1 模型结构特点

- **无需干净图像**：通过学习噪声图像之间的映射关系实现去噪
- **轻量级设计**：简化的网络结构，计算效率高

7.1.2 优势与局限性

- **优势**：
 - 适用于无干净图像的场景
 - 计算效率高，适合资源受限场景
- **局限性**：
 - 对高强度噪声的去噪效果有限

7.2 改进方向

7.2.1 可能的优化方向

1. 网络结构优化：

- 引入更强大的特征提取骨干网络
- 添加注意力机制，提升关键特征的提取能力

2. 训练策略改进：

- 采用更先进的数据增强技术
- 改进损失函数设计，平衡去噪和细节保留