华中科技大学

2024

计算机视觉　　课程设计报告

题　　目：	
专　　业：	计算机科学与技术
班　　级：	CS2209
学　　号：	U202214056
姓　　名：	赵子昕
电　　话：	15755382147
邮　　件：	2171023436@qq.com

目 录

1 实验概述

1.1 实验背景

卷积神经网络在进行图像分类任务时的可解释性分析是一个重要研究课题。经典的可解释性方法有 LIME, RISE, Grad-CAM, Grad-CAM++, ScoreCAM, LayerCAM 等。

1.2 设计要求

请阅读上述 CNN 可解释性方法所对应的原始论文，并要求：

1. 简述每种可解释性方法的基本原理，并对各类可解释性方法进行分析、比较
2. 复现（也可运行开源）LIME 代码，得到可解释性分析结果
3. 复现（也可运行开源）Grad-CAM++代码，得到可解释性分析结果
4. 复现（也可运行开源）ScoreCAM 代码，得到可解释性分析结果
5. 对比三种方法的可解释性分析结果，分析各自的优缺点

鼓励在报告中展现自己对每种方法的理解或对方法存在的缺陷的思考、改进等。

2 实验原理

2.1 引言

随着深度学习在图像分类领域的广泛应用，卷积神经网络（CNN）因其卓越的性能得到了广泛认可。然而，CNN 作为一种"黑盒"模型，其决策过程的不可解释性引发了人们对其透明性和可信度的担忧。为此，许多研究致力于开发 CNN 的可解释性分析方法，以提高模型的可解释性和可视化能力。本文将简要介绍几种经典的可解释性方法，包括LIME、RISE、Grad-CAM、Grad-CAM++、ScoreCAM 和 LayerCAM，并对其进行分析与比较。此外，本文将复现 LIME、Grad-CAM++和 ScoreCAM 的可解释性分析结果，并进行对比分析。

2.2 可解释性方法简介

2.2.1 LIME（Local Interpretable Model-agnostic Explanations）

LIME[1]是一种模型无关的解释方法，主要通过局部线性模型对复杂模型的预测进行解释。其核心思想是在输入样本的局部区域生成一系列扰动样本，并基于这些样本构建一个简单的线性模型来近似原始模型的决策行为。

原始论文中给出了一个基于文本分类的例子：判断无神论和基督教徒，在第二个例子中，Post 被认为是判断文本意向是无神论的重要的单词——然而事实并非如此，因为 Post 在这里是发帖的抬头。这意味着分类器虽然预测正确但原因错误。

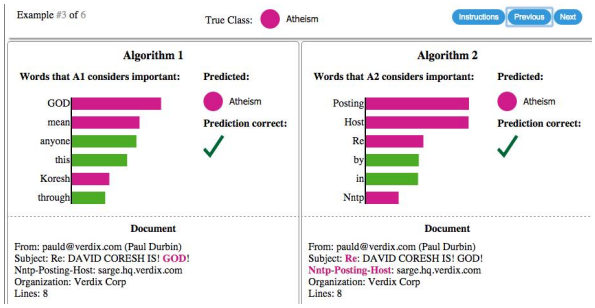

图 2.1 无神论分类原因错误的例子

LIME 能够在可解释表示上识别出一个局部真实反映分类器的可解释模型。具体

[1] " Why should i trust you?" Explaining the predictions of any classifier

操作步骤如下：

1. 选择一个要预测的样本。
2. 在样本周围选择一些邻近样本。
3. 使用这些邻近样本来训练一个简单的解释模型，例如线性模型。
4. 使用这个解释模型来解释原始模型的预测。

2.2.2 RISE（Randomized Input Sampling for Explanation）

RISE[1]方法通过随机遮挡输入图像的一部分并观察模型预测的变化来生成重要性图。其主要思想是通过对不同遮挡掩码的采样，分析输入图像不同区域对模型预测的贡献。它的最大特点是完全黑箱，不需要访问神经网络的任何部分，不需要考虑模型本身的 accessibility。它和 LIME 方法比较像的一点是通过样本扰动来作为评价手段。

RISE 的基本方法也非常简单：

1. 是通过蒙特卡洛采样生成大量随机遮挡掩码；
2. 使用遮挡后的输入图像计算模型的预测分数；
3. 根据预测分数对每个像素的重要性进行加权平均。

此外，RISE 还给出了两个可以自动计算的指标：deletion 和 insertion。deletion 意味着随着越来越多的像素被删除，预测类的概率的减少；insertion 提供了一个互补的测量指标。它测量了随着越来越多的引入概率的增加。在删除、引入像素值的时候要注意随机，比如如果删除一个椭圆形的区域，分类器还是有很大可能将其分类成气球。

相比较其他方法而言，RISE 最大的缺点是运行效率较低，毕竟要生成多张被遮蔽的图片，还要将每张图片在模型上识别一次。

2.2.3 Grad-CAM（Gradient-weighted Class Activation Mapping）

在介绍 Grad-CAM[2]之前，先简单介绍一下它的前身——CAM[3]。

CAM 是通过模型的最后一层卷积层生成的，主要用于理解卷积神经网络（CNN）如何在图像上进行分类。它通过加权卷积层的特征图生成一个类激活图（activation

[1] Rise: Randomized Input Sampling for Explanation of black-box models
[2] Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization
[3] Learning Deep Features for Discriminative Localization

map)，这些特征图是 CNN 在做出最终预测时的中间层特征。假设我们有一个 CNN，最后一层是全连接层（FC）。为了得到类激活图，我们首先计算每个卷积层的特征图。对于每个特定的类（例如，图像分类中的“狗”类），通过对最后一层卷积层的特征图进行加权求和来生成激活图。权重是通过训练好的模型在输出层的类别概率分配的权重。最终得到的热图反映了哪些区域对该类的分类决策贡献最大[1]。

$$\mathrm{CAM}_c = \sum_i w_i^c A^i$$

其中：w_i^c是最后一层全连接层权重（即每个特征图对某类的贡献），A^i是卷积层的第 i 个特征图。详情见图 2.2。

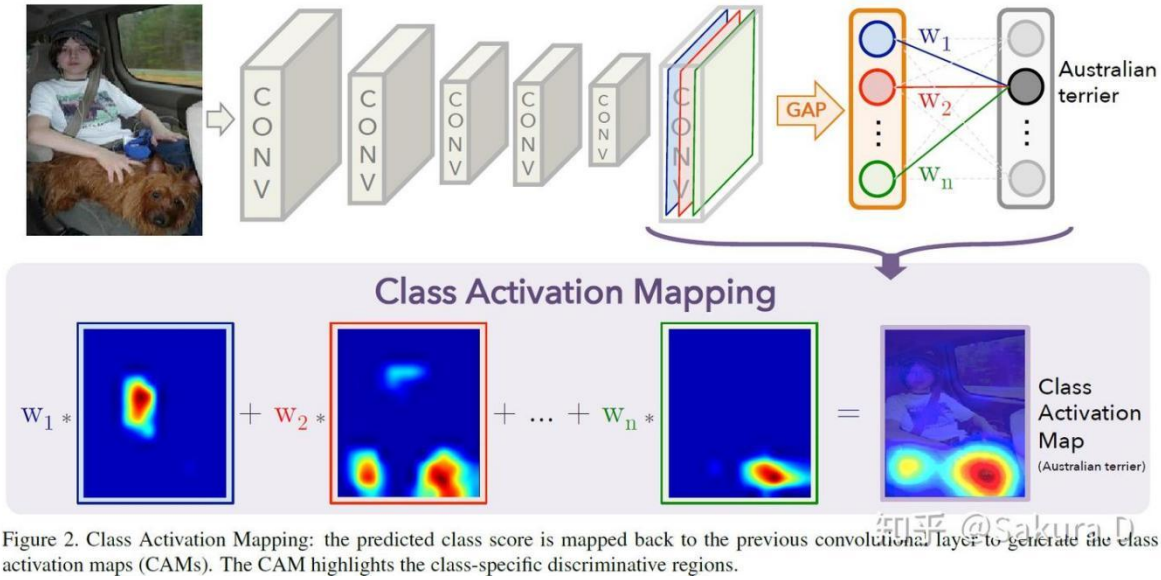

Figure 2. Class Activation Mapping: the predicted class score is mapped back to the previous convolutional layer to generate the class activation maps (CAMs). The CAM highlights the class-specific discriminative regions.

图 2.2 CAM 利用特征图进行加权求和的过程

CAM 虽然简单，但是它要求网络架构里必须有 GAP 层，并且需要修改原模型的结构，导致需要重新训练该模型，这大大限制了它的使用场景。如果模型已经上线了，或着训练的成本非常高，几乎是不可能为了它重新训练的。

回到 Grad-CAM，Grad-CAM 解决了这个问题，基本思路和 CAM 是一致的，也是通过得到每对特征图对应的权重，最后求一个加权和。区别是求解权重的过程，CAM 通过替换全连接层为 GAP 层，重新训练得到权重，而 Grad-CAM 另辟蹊径，用梯度的全局平均来计算权重。事实上，经过严格的数学推导，Grad-CAM 与 CAM

[1] 神经网络的可解释性（可视化篇） https://zhuanlan.zhihu.com/p/479485138

计算出来的权重是等价的[1]。

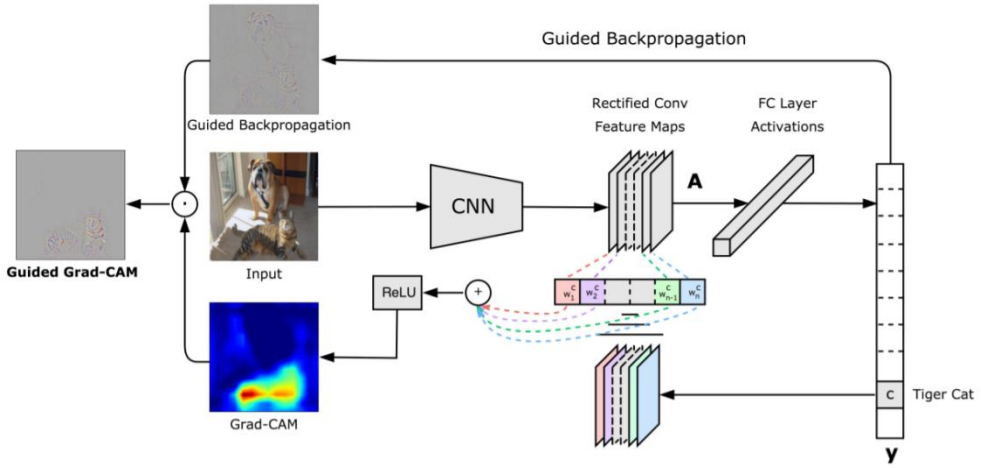

图 2.3 获得类别判别图 Grad-CAM 的过程

如图，为了获得类别判别图 Grad-CAM，首先用 softmax 之前的 logits 计算 c 这个类的梯度，定义特征图的激活值为A^k。这些回流的梯度在宽度和高度维度（分别由 i 和 j 索引）上全局平均池化，以获得神经元重要性权重α_k^c。

$$\alpha_k^c = \overbrace{\frac{1}{Z}\sum_i\sum_j}^{\text{global average pooling}} \underbrace{\frac{\partial y^c}{\partial A_{ij}^k}}_{\text{gradients via backprop}} \qquad (3)$$

Grad-CAM 的工作原理大致如下：

1. 计算类别概率的梯度： Grad-CAM 的核心是使用模型的梯度信息，具体来说，是通过计算目标类别的输出相对于某一卷积层特征图的梯度。这个梯度反映了该层的特征图对最终类别输出的影响程度。

2. 权特征图： 然后，将这些梯度进行全局平均池化（Global Average Pooling），得到每个特征图的权重系数，表示每个特征图对分类决策的影响大小。

3. 生成加权的类激活图： 使用这些权重对卷积层的特征图进行加权求和，从而得到类激活图。最后，通过 ReLU 操作，将负值抑制（变成 0），使得热图只关注对分类有正面影响的区域。

再来讲讲 Grad-CAM++。Grad-CAM++是对 Grad-CAM 的优化，它的定位更精准，也更适用于目标类别物体在图像中不止一个的情况。Grad-CAM 是利用目标特

[1] CNN 可视化 Grad-CAM https://zhuanlan.zhihu.com/p/105373864

征图的梯度求平均获取特征图权重，因此梯度 map 上每一个元素的贡献是一样。Grad-CAM++认为梯度 map 上的每一个元素的贡献不同，因此增加了一个额外的权重对梯度 map 上的元素进行加权。

简而言之，Grad-CAM++提供了更加精细的算法去计算权重，能够处理地更加平滑，能够胜任多目标的计算任务。

2.2.4 ScoreCAM

ScoreCAM[1]是一种无梯度的可解释性方法，相比较 Grad-CAM 使用梯度信息来计算权重，它不需要原始网络提供梯度信息，通过直接使用特征图激活区域的贡献分数来生成热力图。ScoreCAM 的主要步骤如下：

1. 特征图提取：将输入图像输入未经修改的 CNN，获取指定卷积层的特征图。

2. 特征图归一化和上采样：将每个特征图通过归一化处理，使其值在 0, 1 之间，一般采用 Min-Max 归一化。将归一化后的特征图上采样（通常通过双线性插值）到与输入图像相同的尺寸。

3. 生成遮蔽图像并计算权重：遮蔽图像：将上采样的特征图作为掩码，与原始输入图像逐元素相乘，得到一系列遮蔽后的图像。

4. 计算权重：将每个遮蔽后的图像输入 CNN，获取目标类别的得分（例如 Softmax 输出或 logit 值）。这些得分反映了对应特征图对目标类别的贡献，作为特征图的权重。

5. 生成热力图：将原始特征图按照计算得到的权重进行加权求和，得到最终的热力图。

由于无需梯度信息，不同于 Grad-CAM，ScoreCAM 完全基于前向传播，不需要计算梯度，并且不需要对已有的 CNN 模型进行任何修改，具有很强的适用性。尤其适合处理在深层网络梯度消失和梯度爆炸的情况。

2.2.5 LayerCAM

LayerCAM[2]基于 Grad-CAM 的框架，是一种改进的类激活映射方法，旨在提

[1] Wang H, Wang Z, Du M, et al. Score-CAM: Score-weighted visual explanations for convolutional neural networks[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. 2020: 24-25.

[2] Jiang P T, Zhang C B, Hou Q, et al. Layercam: Exploring hierarchical class activation maps for localization[J]. IEEE Transactions on Image Processing, 2021, 30: 5875-5888.

供更细粒度、更准确的模型可视化。它对传统的 CAM 方法进行了优化，使得在不同的卷积层级别上都能生成类激活映射，从而更好地理解深度神经网络的决策过程。

与 Grad-CAM 不同，Layer-CAM 在计算权重时，不是对整个特征图计算一个全局的权重 ，而是对特征图中的每一个像素位置 (i, j) 计算对应的权重$\alpha_k^{i,j}$。具体来说，针对目标类别计算输出对特征图 k 中每个位置 (i, j) 的梯度。

对于每一个卷积层的特征图 ，结合逐像素的梯度权重，计算激活图：

$$L^c_{Layer\_CAM} = \mathrm{ReLU}\left(\sum_k \alpha_k^{i,j} A_{i,j}^k\right)$$

为了利用不同层次的特征，Layer-CAM 对多层的激活图进行融合。可以简单地对各层的激活图进行求和或加权平均。这种方式结合了浅层的细节特征和深层的语义特征，生成了更全面、更准确的可视化结果。

由于需要计算多个卷积层的逐像素梯度，Layer-CAM 比传统方法计算成本更高。并且在非常深的网络中，浅层梯度可能非常微弱，需要注意数值稳定性。

3 实验内容

本次实验按照实验要求主要对 LIME[1]、Grad-CAM++[2]、ScoreCAM、LayerCAM 算法进行了对比测试。

3.1 LIME 算法复现

由于 LIME 的相关代码进行了开源，因此笔者直接运行了开源代码 lime。

LIME 开源算法使用的模型是 inception_v3，笔者直接使用该模型进行测试。测试非常顺利。笔者简单更换了几种模型对不同任务进行测试，发现 LIME 效果一般。

图 3.1 LIME 提供的参考图片

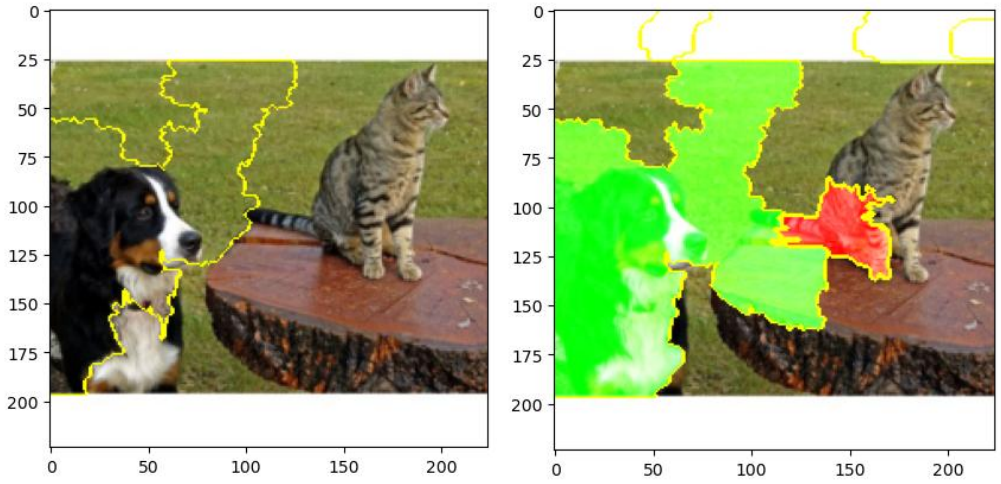

图 3.2 LIME 测试结果

[1] https://github.com/marcotcr/lime
[2] https://github.com/frgfm/torch-cam

3.1.1 单目标任务

LIME 在单目标任务上表现一般。这里挑选了两张图片进行测试。见图 3.3 图 3.4。

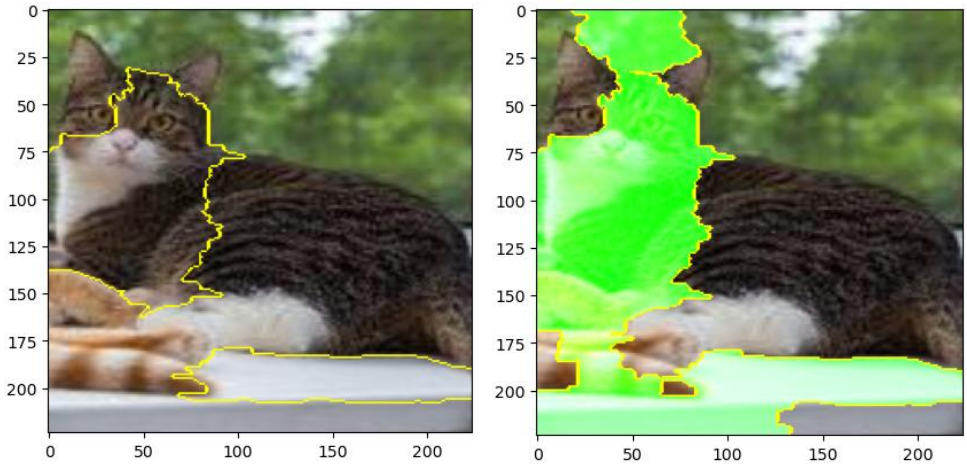

图 3.3 一只猫的图片在 inception_v3 上 LIME 的处理结果

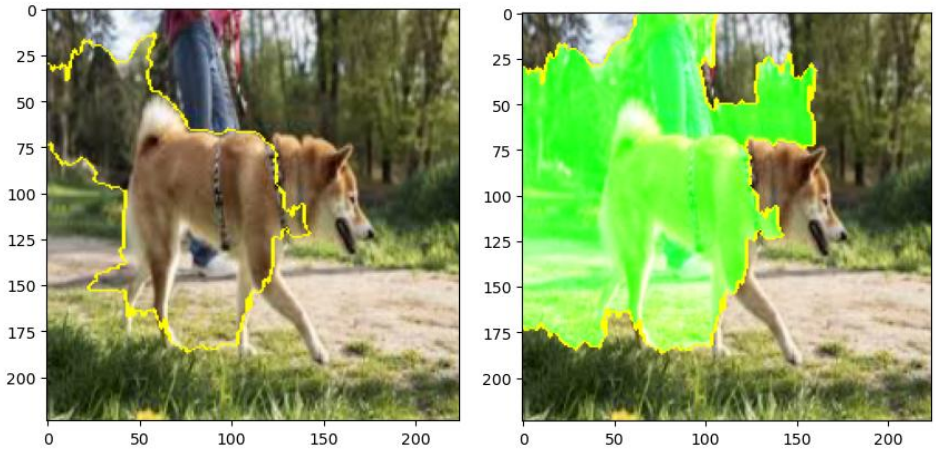

图 3.4 一条狗在 inception_v3 上 LIME 的处理结果

可以看到，猫只覆盖到了上半身，狗只覆盖了下半身，并且大部分覆盖区域在旁边的空地上。并且在狗头部分，LIME 完美从边缘避开了狗头，这可能与 LIME 局部算法表现不佳有关。

3.1.2 多目标任务

首先笔者先测试了多个相同目标的情况，效果一般。但是注意到，LIME 在边缘分割方面做的非常好。这可能与它局部采样与临近像素进行对比这一特性有关系。

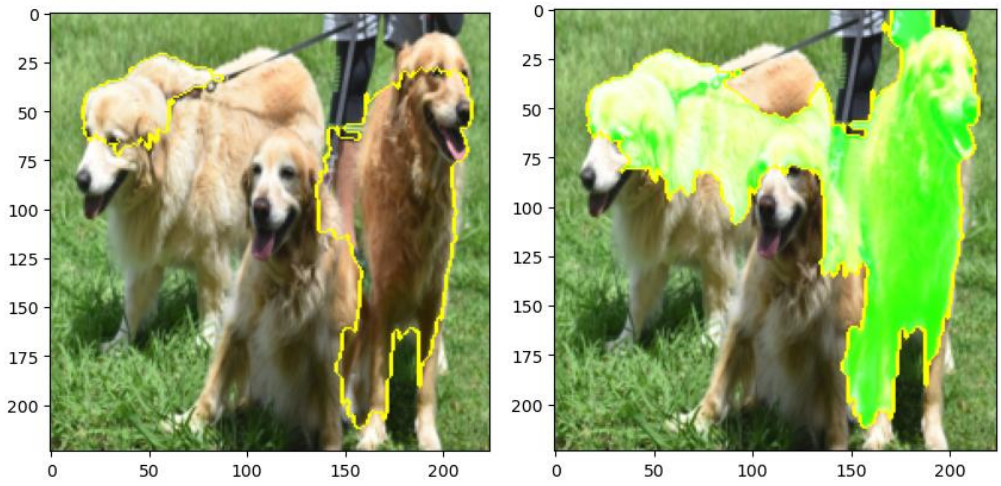

图 3.5 多条狗 LIME 测试

笔者在 Google 图片库里面随机挑选了一张狗和猫同时出现的照片，在原始模型 inception_v3 下进行了测试。如。

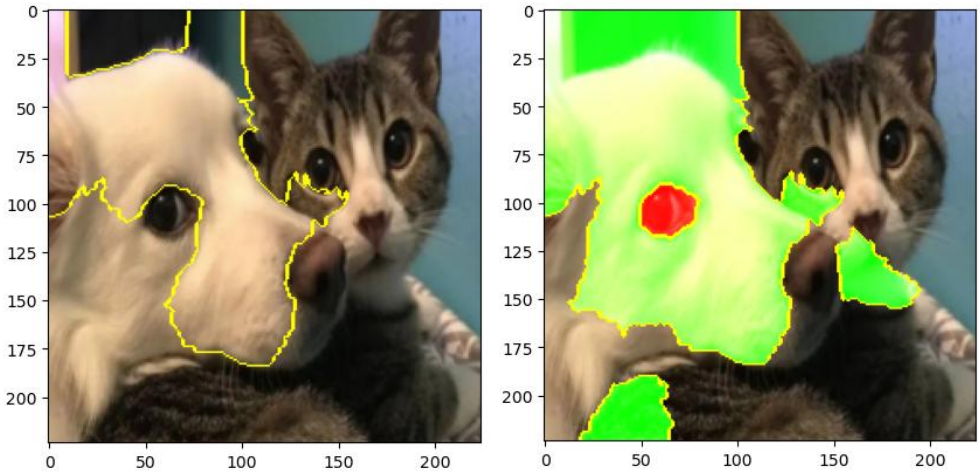

图 3.6 LIME 自选图片 dac1（dog and cat）测试效果

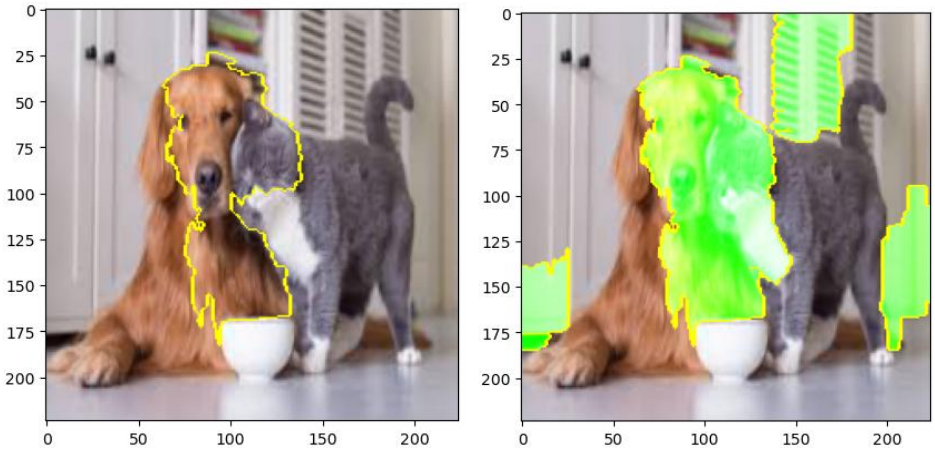

图 3.7 LIME 自选图片 dac2 测试效果

在 dac1（dog and cat）图片上面，LIME 取得了较好的效果，非常清晰地分开了狗和猫，但是莫名其妙除去了狗的眼睛，这与上文的局部算法表现不佳可能有关；在 dac2 上面表现较差，没有很好区分。

这有可能是模型的问题，笔者又更换了经典模型 resnet34，同样测试了 dac1，dac2。这次测试在 dac2 上效果较好，但是质量仍然比较一般。

从这方面来看，LIME 方法在多混杂目标处理任务上可能比较差。

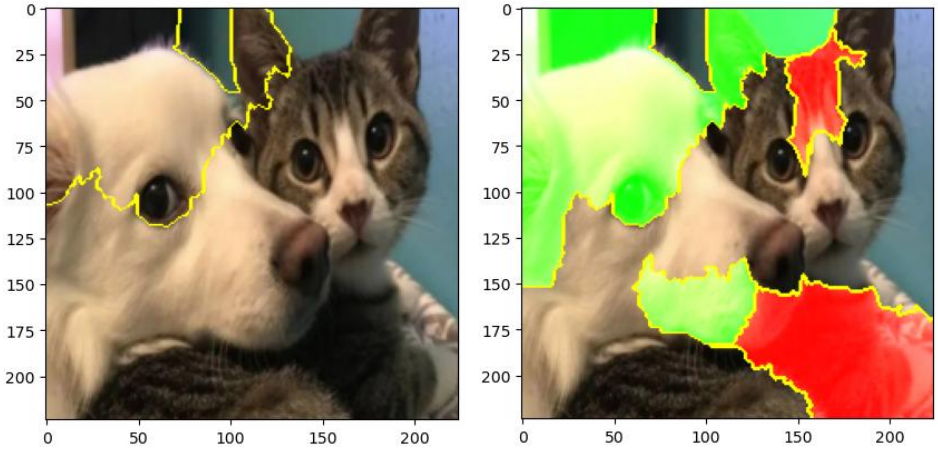

图 3.8 使用 resnet34 模型 dac1 的结果

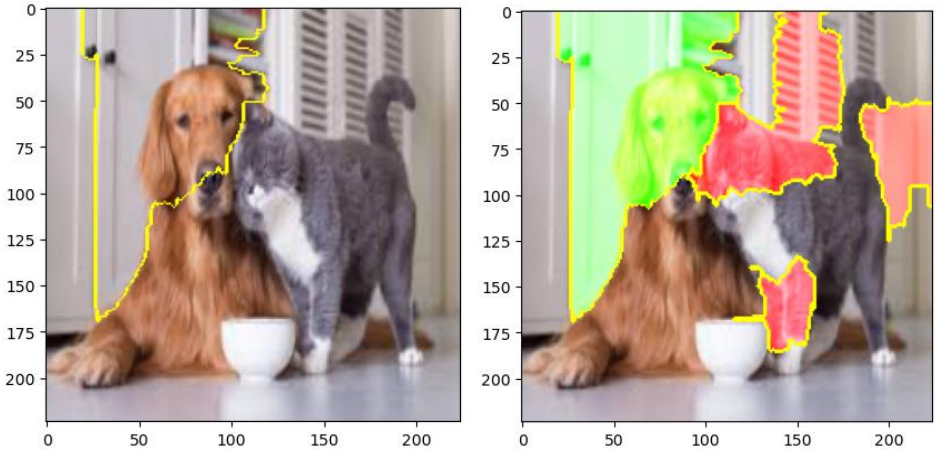

图 3.9 使用 resnet34 模型 dac2 的结果

3.2 CAM 类算法实现

笔者主要测试了 Grad-CAM++ 和 ScoreCAM，简单测试了 LayerCAM。

对于每种方法，笔者主要测试了 4 种模型：densenet121、resnet18、resnet34、efficientnet_b0，为了进行对比，辅助测试了模型 inception_v3。

3.2.1 单目标任务

两种模型在单目标任务上表现都非常优异。此处给出部分测试结果。

图 3.10 Grad-CAM++（左）和 ScoreCAM（右）在一条狗的图片上的可解释性效果

3.2.2 多目标任务

在同种类多目标任务上，对于同一种模型，不同模型效果不同。

首先简单测试了 Grad-CAM++在 densenet121、resnet18、resnet34、efficientnet_b0 模型上的效果，可以看出 Grad-CAM++综合效果都还不错，efficientnet_b0 效果最好，总体而言，Grad-CAM++在同种类多目标任务表现优异。

图 3.11 Grad-CAM 在 densenet121、resnet18、resnet34、efficientnet_b0 模型（从左到右）上的效果

再简单测试了 ScoreCAM 在 densenet121、resnet18、resnet34、efficientnet_b0 模型上的效果，可以看出，resnet34 效果最好，efficientnet_b0 可能因为没有进行分割处理，虽然在狗头部分有识别，但是识别不集中。总体而言，ScoreCAM 效果较好。

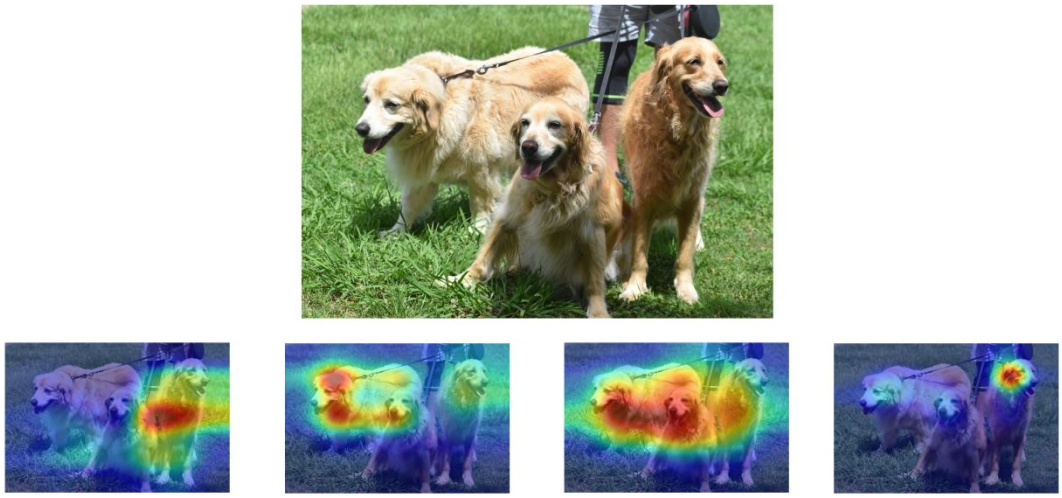

图 3.12 ScoreCAM 在 densenet121、resnet18、resnet34、efficientnet_b0 模型（从左到右）上的效果

笔者再次尝试了 dac1 在不同 CAM 方法和模型下的测试来评估多混杂目标效果，限于篇幅，这里只放每种方法效果最好的模型。见，从左到右分别是 Gard-CAM++、ScoreCAM 和 LayerCAM 处理。

可以看出来，三种方法的效果都比较好，清晰地分出了猫和狗。

图 3.13 Gard-CAM++、ScoreCAM 和 LayerCAM 处理后 dac1 检测狗的效果

图 3.14Gard-CAM++、ScoreCAM 和 LayerCAM 处理后 dac1 检测猫的效果

3.3 额外尝试

一个比较实际的问题是，一张图片中的相似元素可能有不同的 index，对于一个

固定的 CAM，它一般只接受一个 index。比如上面两张图 dac1 中猫和狗，狗的 index 是 207，猫的 index 是 281/282。

接下来笔者希望在 dac1 和 dac2 两张图片中，同时把猫和狗标注出来。这是一个比较普遍的需求，同一种类的对象在模型中极有可能有不同的索引，比如一些车在同一张图片中，有些车是跑车，有些车是面包车，但是它们在分类器中极有可能有不同的 index。但是我们的目标是把得到所有车的索引的可解释性。对多个 index 混合的可解释性进行混合在上述情景下就有一定的价值。

在测试过程中，笔者给前五大概率的 index 分别在最终热力图上给予了 3，2,1.5,1,1 的权重，随后测试了 dac1 和 dac2。

然而效果不是非常好，这里给出效果最好的两张图片进行解释。

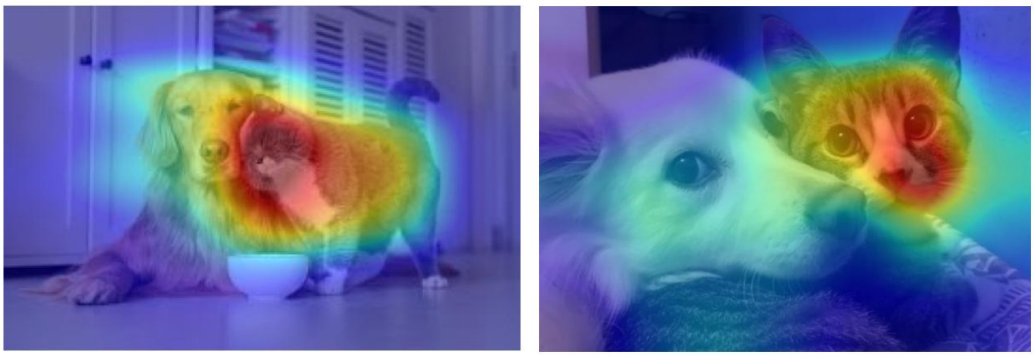

图 3.15 效果最好的两张左为 Gard-CAM++在 resnet18 模型下得到的效果，右为在 LayerCAMresnet34 模型下得到的效果

一个可能的解释是，模型本身素质比较一般，除了几率最大的标签（预测结果）有可能排名前几的标签对应的是图片中比较复杂混乱的情况，这种情况下使用 Top5 效果不一定好。可能需要指定标签，比如指定 281（猫）和 207（狗）。

笔者输出了 dac1 和 dac2 过程中 Top5 的 index 具体的值，只有出现 281（猫）而没有出现狗，说明狗的比重并不大，可能需要直接指定。

3.4 几种方法综合对比

3.4.1 最终效果

从最终效果来看，Grad-CAM++ 和 LayerCAM 表现较好，ScoreCAM 表现适中，LIME 综合较差。但是 LIME 能够比较清晰地提取出目标对象和周围的轮廓（虽然有几率会标反目标对象和周围），这可能与 LIME 算法特点有关。

这可能与两种方法的主要实现思路有关。前者是直接从训练网络中提取有关信

息，后者是偏向对网络当做一个黑盒，通过样本扰动进行测试。有可能前者的效率更高更准确。

3.4.2 运行效率

在实际测试中，笔者主要到几种算法运行所需的时间不同。总体来说：

Grad-CAM ++ ≈ LayerCAM > LIME >> ScoreCAM。

ScoreCAM 最慢，需要计算很长时间。LIME 由于要进行多次样本扰动测试也比较慢。Grad-CAM ++ 和 LayerCAM 运行效率都非常高，几乎立即得出结果。

4 实验总结

在本次实验中，笔者对多种可解释性分析方法进行了测试与比较，包括 LIME、Grad-CAM++、ScoreCAM 和 LayerCAM。这些方法在可解释性和适用性方面各有优势，适用于不同的任务和场景。以下是实验的整体总结：

不同方法对模型预测的可解释性提供了多角度的分析，能够有效揭示模型关注的区域和特征。

LIME 作为模型无关的方法，具有一定的通用性，但在高维数据上的稳定性和效率需要进一步优化。

Grad-CAM++、ScoreCAM 和 LayerCAM 等基于特征图的可视化方法在图像分类任务中表现较好，能够直观地展示模型的关注区域。

笔者对不同的任务（单目标任务、多同类目标、多混杂目标任务）进行了测试，得到了不同方法在不同模型上的效果，有一定参考价值。

不同方法的适用性和计算效率差异较大，在实际应用中需要根据任务需求选择合适的解释方法。总体而言，这些方法为分析深度学习模型提供了有力的支持，能够帮助我们更好地理解模型的预测机制。

参考文献

[1] Ribeiro M T, Singh S, Guestrin C. " Why should i trust you?" Explaining the predictions of any classifier[C]//Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. 2016: 1135-1144.

[2] Petsiuk V. Rise: Randomized Input Sampling for Explanation of black-box models[J]. arXiv preprint arXiv:1806.07421, 2018.

[3] Zhou B, Khosla A, Lapedriza A, et al. Learning deep features for discriminative localization[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2921-2929.

[4] Selvaraju R R, Cogswell M, Das A, et al. Grad-cam: Visual explanations from deep networks via gradient-based localization[C]//Proceedings of the IEEE international conference on computer vision. 2017: 618-626.

[5] Chattopadhay A, Sarkar A, Howlader P, et al. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks[C]//2018 IEEE winter conference on applications of computer vision (WACV). IEEE, 2018: 839-847.

[6] Wang H, Wang Z, Du M, et al. Score-CAM: Score-weighted visual explanations for convolutional neural networks[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. 2020: 24-25.

[7] Jiang P T, Zhang C B, Hou Q, et al. Layercam: Exploring hierarchical class activation maps for localization[J]. IEEE Transactions on Image Processing, 2021, 30: 5875-5888.

一、原创性声明

本人郑重声明本报告内容，是由作者本人独立完成的。有关观点、方法、数据和文献等的引用已在文中指出。除文中已注明引用的内容外，本报告不包含任何其他个人或集体已经公开发表的作品成果，不存在剽窃、抄袭行为。

特此声明！

作者：赵子昕