

重庆邮电大学  
CHONGQING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

专业学位硕士论文  
MASTER THESIS FOR PROFESSIONAL DEGREE



论文题目 重庆邮电大学学位论文  
格式模板

学科专业 电子科学与技术

学号 S20202222

作者姓名 张三

指导教师 李四

学院 光电工程学院/国际半导体学院

学校代码 10617 UDC XXXXXX  
分 类 号 XXXXXX 密级

# 学 位 论 文

重庆邮电大学学位论文格式模板

某 某

指导教师 某某某 教 授  
某 某 副教授

申请学位级别 博士 学科专业 XXXX  
专业学位领域 XXXXXX  
答辩委员会主席 某某某 教授 论文答辩日期 2021 年 5 月 20 日  
学位授予单位和日期 重庆邮电大学 2021 年 6 月

**Dissertation Template for Master Degree of  
Engineering in CHONGQING UNIVERSITY OF  
POSTS AND TELECOMMUNICATIONS**

A Doctoral Dissertation Submitted to  
Chongqing University of Posts and Telecommunications

Discipline	XXXX
Student ID	XXXX
Author	XXXX
Supervisor	XXXX
School	XXXX

## 重庆邮电大学

### 学位论文独创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文中不包含其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在论文中以明确方式标明。本人完全知晓本声明的法律后果由本人承担。

学位论文作者签名：

日期： 年 月 日

## 重庆邮电大学

### 学位论文使用授权书

本人同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。

本学位论文属于：

公开论文

涉密论文，保密\_\_\_\_年，过保密期后适用本授权书。

(请在以上方框内选择打“√”)

作者签名：

导师签名：

日期： 年 月 日

## 摘要

学位论文是研究生从事科研工作的成果的主要表现，集中表明了作者在研究工作中获得的新发明、新理论或新见解，是研究生申请硕士或博士学位的重要依据，也是科研领域中的重要文献资料和社会的宝贵财富。

为进一步规范我校研究生学位论文撰写格式，提高研究生学位论文质量，参照国家标准《学位论文编写规则》(GB/T 7713.1-2006)，结合我校实际，制定本模板。

**关键词：**学位论文，撰写规范，论文模板，重庆邮电大学

## ABSTRACT

Dissertation /Thesis is postgraduate ' s main academic performance to display her/his works of scientific research, which shows the author ' s new invention, new theory or new opinion in her/his research. It is the crucial document for the graduate students to apply for degree, and it is also the important scientific research literature and the valuable wealth of society.

In order to further standardize the format of dissertation/thesis writing and improve graduate dissertation/thesis quality, this temolate is formulated with reference to the national standard "Rules for Dissertation Writing" (GB/T 7713.1-2006) and the reality of CQUPT.

**Keywords:** Dissertation/Thesis, Writing Specification, Thesis Template, Chongqing University of Posts and Telecommunications

## 目 录

摘 要 .....	I
ABSTRACT .....	II
图目录 .....	VI
表目录 .....	VII
主要符号表 .....	VIII
缩略词表 .....	IX
第 1 章 绪论 .....	1
1.1 研究背景及意义 .....	1
1.2 国内外研究现状 .....	2
1.2.1 基于反向传播的显著图解释技术 .....	2
1.2.2 基于类激活映射的显著图解释技术 .....	3
1.2.3 基于扰动的显著图解释方法 .....	3
1.3 论文研究的主要内容 .....	4
1.4 论文组织结构 .....	4
第 2 章 相关理论介绍 .....	5
2.1 引言 .....	5
2.2 卷积神经网络 .....	5
2.3 基于 Transformer 的图像分类神经网络 .....	7
2.3.1 多头自注意力机制 .....	7
2.3.2 编码器 .....	8
2.3.3 解码器 .....	10
2.3.4 应用于图像分类领域的 Transformer .....	11
2.4 常见的显著图解释方法 .....	12
2.4.1 Grad-CAM .....	12
2.4.2 Grad-CAM++ .....	14
2.4.3 Score-CAM .....	15
2.4.4 LRP .....	17
2.4.5 Transformer Attribution .....	20
2.5 本章小结 .....	23
第 3 章 基于输入图片多尺寸放大的卷积神经网络显著图解释方法 .....	24
3.1 本章引言 .....	24

3.2 问题描述和研究思路.....	24
3.3 基于输入图片多尺寸放大的卷积神经网络显著图解释方法.....	25
3.3.1 生成高分辨率掩膜.....	25
3.3.2 掩膜优化.....	28
3.3.3 生成显著图 .....	29
3.4 实验与分析.....	30
3.4.1 实验硬件配置和软件环境.....	30
3.4.2 数据集及其预处理和实验参数说明 .....	30
3.4.3 定性评估.....	32
3.4.4 插入 (insertion) 和删除 (deletion) 实验 .....	32
3.4.5 定点游戏实验.....	34
3.4.6 合理性检验 .....	36
3.5 本章小结 .....	37
<b>第 4 章 一种通用的基于二维滑动窗口和放大的图像分类神经网络显著图解释增强方法.....</b>	<b>38</b>
4.1 本章引言 .....	38
4.2 问题描述和研究思路.....	38
4.3 一种通用的基于二维滑动窗口和放大的图像分类神经网络显著图解释增强方法.....	39
4.3.1 获取窗口图片集合.....	39
4.3.2 获取窗口图片的显著图和权重 .....	40
4.3.3 融合窗口图片集合的显著图.....	41
4.3.4 平滑和归一化显著图 .....	42
4.4 实验与分析.....	43
4.4.1 实验硬件配置和软件环境 .....	43
4.4.2 数据集说明和数据集处理 .....	43
4.4.3 作为对照的显著图方法 .....	44
4.4.4 增强方法相关参数 .....	45
4.4.5 定性评估 .....	45
4.4.6 扰动实验.....	45
4.4.7 图像分割实验.....	47
4.5 本章小结 .....	47
<b>第 5 章 显著图解释方法对比评测系统.....</b>	<b>50</b>
5.1 系统需求分析 .....	50

## 目 录

---

5.2 开发环境及相关技术介绍 .....	52
5.3 系统设计 .....	52
5.3.1 系统架构设计 .....	52
5.3.2 模块设计 .....	52
5.3.3 功能设计 .....	53
5.4 系统展示 .....	55
5.5 本章小结 .....	56
参考文献 .....	60
附录 A 各学院中英文名称对照表 .....	65
附录 B 常见一级学科中英文名称对照表 .....	66
附录 C 常见专业学位类别中英文名称对照表 .....	67
作者简介 .....	68
1. 基本情况 .....	68
2. 教育和工作经历 .....	68
3. 攻读学位期间的研究成果 .....	68
3.1 发表的学术论文和著作 .....	68
3.2 申请（授权）专利 .....	68
3.3 参与的科研项目及获奖 .....	68
致 谢 .....	70

## 图目录

图 2-1 标准卷积网络结构示意图 .....	6
图 2-2 Transformer 结构示意图 .....	8
图 2-3 ViT 结构示意图 .....	12
图 2-4 Grad-CAM 流程示意图 .....	13
图 2-5 一个假设性的例子说明 Grad-CAM++ 的优势 .....	15
图 2-6 CAM, Grad-CAM, Grad-CAM++ 的各自计算方式 .....	16
图 2-7 特征图扰动示例 .....	16
图 2-8 像素相关性分解过程可视化 .....	18
图 2-9 多层神经网络前向传播计算示例 .....	19
图 2-10 多层神经网络下前向传播和层间相关性传播过程 .....	19
图 2-11 层间相关性传播分配相关性分数的例子 .....	20
图 2-12 Transformer attribution 计算过程 .....	23
图 3-1 MSG-CAM 算法流程示意图 .....	25
图 3-2 单张图片在不同方法下的插入删除实验曲线对比 .....	34
图 3-3 .....	35
图 3-4 MSG-CAM 合理性检验 .....	37
图 4-1 当前显著图解释方法分辨低的原因 .....	39
图 4-2 单张图片在不同方法下的插入删除实验曲线对比 .....	40
图 4-3 单张图片在不同方法下的插入删除实验曲线对比 .....	47
图 4-4 单张图片在不同方法下的插入删除实验曲线对比 .....	49
图 4-5 单张图片在不同方法下的插入删除实验曲线对比 .....	49
图 5-1 系统架构设计图 .....	53
图 5-2 系统模块设计图 .....	53
图 5-3 图像分类模型的查看和导入界面图 .....	56
图 5-4 显著图解释方法参数设置页面 .....	57
图 5-5 用户密码修改和查看页面 .....	57
图 5-6 显著图生成页面 .....	58
图 5-7 显著图分割页面 .....	58
图 5-8 实验评测页面 .....	59

## 表目录

表 3-1 实验环境和硬件配置 .....	31
表 3-2 电流类型对效率的影响 .....	33
表 3-3 电流类型对效率的影响 .....	36
表 4-1 实验环境和硬件配置 .....	43
表 4-2 电流类型对效率的影响 .....	46
表 4-4 显著图增强算法在图像分割的表现 .....	48
表 5-1 软件开发环境和硬件配置 .....	51

## 主要符号表

---

符号	说明	页码
c	电磁波的相平面速度	10

---

## 缩略词表

---

英文缩写	英文全称	中文全称
CQUPT	Chongqing University of Posts Telecommunications	重庆邮电大学

---

## 缩略词表

---

## 第1章 绪论

### 1.1 研究背景及意义

当今的社会正处于智能化趋势的浪潮中，由深度学习理论所衍生的相关技术被广泛的应用在人们所熟知的各个领域。在学术界和工业界的互相融合促进下，深度学习算法不断推陈出新，深度神经网络也不断进化并发展出适用与文字、图像、视频等信息介质的自动化识别和信息提取的高效的深度神经网络模型，许多相关领域深度神经网络模型已经是现代社会正常运转的不可或缺的一部分。

但是，当前主流的深度神经网络并不具备良好的可解释性，即便这些深度神经网络在各种测试任务下展现出了很高的准确率。例如在图像分类应用中，将待分类的图片送入训练好的深度神经网络会得到不同类别物体的置信概率分数，即便某一类别的置信概率分数是 99.99%，我们也无法得到深度神经网络做出这一决策的依据，即该深度神经网络的输出结果并不具备可解释性。并且随着应用场景的复杂化多样化，深度神经网络结构日趋复杂，参数数量日趋庞大，这使得深度神经网络的“黑盒”特点变得更为突出。这种难以解释的黑盒特性使得深度神经网络在可靠性要求极高的领域，诸如医疗影像、自动驾驶、航空航天等领域的应用就受到限制。

除此之外，上述的黑盒特性也会成为当前的深度神经网络的研究过程当中的“拦路虎”。研究者往往是将训练好的深度神经网络在既有的数据集上根据各种外部的量化指标评价训练效果，但是在实际应用过程当中，模型可能会对现实世界中某些特殊的数据或者人为恶意伪造的数据给出异常的结果。如果训练的深度神经网络不能对这些意外数据有着良好的鲁棒性，那么该神经网络的就不能得到人们的信任。因此这使得相关研究者必须从更加全面的角度判断深度神经网络的性能表现并且解释其训练的神经网络的结果输出，而不只是依赖单一的评价指标来判断神经网络的性能表现。

因此，从深度神经网络实际应用和可靠性的角度出发，都需要有适用于深度神经网络黑盒特性的可解释方法，所以许多研究人员将目光转移到深度神经网络的可解释性上，他们或试图从既有的深度神经网络内外寻找其的决策依据，或试图从原理上构建可解释的深度神经网络，这两类研究路径分别就是后解释的人工智能和自解释的人工智能。

通过利用针对深度神经网络的可解释方法，不仅可以使得研究者和用户知道深度神经网络的决策依据，理解其中的决策机理，增强人对神经网络输出结果的信任程度，还可以使得研究者对神经网络的可靠性进行针对性的验证和测试。加强

对深度神经网络可解释方法的研究有助于研究人员用更加全面且灵活双向的方式和神经网络进行交互，能做到“知其然并知其所以然”。可解释性的研究赋予了深度神经网络更多的可能性并加强了其可靠性。

本文主要聚焦于图像分类神经网络的可解释性研究，更加具体的是利用图像即显著图的方式来提供图像分类神经网络的可解释性，根据显著图中给出权值数据来解释图像分类神经网络的输出结果并判断其内部决策过程是否合理可靠。同时本文的研究侧重点在于不改变神经网络内部参数，仅利用既有训练好的深度神经网络模型来进行可解释性研究。既有的图像分类神经网络模型在工业界和学术界已经得到了大量的应用，并且拥有很好的性能表现，因此在不改变模型的前提下，本文的研究可以提供简单明了能直接使用的可解释方法，对已经训练完成模型的输出结果进行显著图分析解释，将单一的，不可解释的输出结果转变为直观明了的，可解释的图像结果，提高了模型的可靠性和可信任性。此外，显著图给出的分析结果可以帮助研究者有效分析模型是否学习到正确的特征，提供神经网络决策的关键依据。

## 1.2 国内外研究现状

### 1.2.1 基于反向传播的显著图解释技术

利用反向传播的机制来实现可视化解释是较早的一些工作所采用的方法。Zeiler 等人<sup>[1]</sup>提出了一种基于反卷积的可视化方法，反卷积将特征值逆映射回了输入图片的像素空间，借此说明图片中的哪些像素参与激活了该特征值。在这项工作的基础上，导向反向传播方法<sup>[2]</sup>提出在反向传播时通过抑制输入和梯度小于 0 的值，从而突出可视化目标的重要特征。DeepLIFT<sup>[3]</sup>，LRP(Layer-wise Relevance Propagation)<sup>[4]</sup>方法通过修改反向传播的规则，将输出层的贡献逐渐向下分配直到输入层，以此来获得输入图片中每个像素对输出相关性分数。Simonyan 等人<sup>[5]</sup>提出使用输入的梯度作为可视化解释的一种手段，这种方法认为输入的某些像素对网络的预测结果起到了主要作用，它直接计算网络输出的特定类别分数对输入的梯度，但是输入的梯度中包含明显的噪声，导致显著图可视化十分模糊。SmoothGrad 方法<sup>[6]</sup>和 VarGrad 方法<sup>[7]</sup>的原理都是共同的，它们向输入图片中多次添加噪声生成一组包含噪声的图片，通过平均化结果使得生成的显著性图更加平滑。为了解决梯度饱和问题，Sundararajan 等人提出了一种积分梯度方法<sup>[8]</sup>，该方法结合了直接计算梯度和基于反向传播的归因技术 DeepLIFT 和 LRP 的分而治之的设计思想，满足敏感性和实现不变性的公理。这些研究虽然有坚实的理论基础，但是它们的可视化结果对于人类来说不容易理解，而且噪声较多。此外这些方法中许多是和具

体类别不相干的，无法对指定类别给出显著图可视化解释结果。另外有研究<sup>[9]</sup>指出其中有些方法的可靠性是值得怀疑的，它们对深度神经网络的参数不敏感，即使网络没有经过训练也能得到相似的结果。

### 1.2.2 基于类激活映射的显著图解释技术

基于类激活映射的方法是目前被大量研究和应用的一种流行的方法。这类方法利用了卷积神经网络中靠近输出端的信息，这些信息中包含着和预测结果相关的丰富的类别信息，所以这类方法能够给出类别相关的显著图可视化解释。CAM<sup>[10]</sup>方法首先提出了将卷积神经网络的最后一层全局平均池化后得到权重和该层提取的特征图线性相乘后累加从而生成显著图。Grad-CAM<sup>[11]</sup>对 CAM 方法进行了改进，无需修改网络结构，利用反向传播的梯度取均值作为权重。Grad-CAM++<sup>[12]</sup>进一步改进了 Grad-CAM，它对不同像素的梯度进行加权，生成的显著图中能将同一类别物体出现多次的情况给较好的展示出来；XGrad-CAM<sup>[13]</sup>通过分析敏感性和实现不变性公理采用了另一种加权方法来获得特征图的权重，它引入了特征图中的像素权重为对应梯度进一步加权。为了减少噪声的影响，Smooth Grad-CAM++<sup>[14]</sup>，SS-CAM<sup>[15]</sup>也采用了 SmoothGrad<sup>[6]2</sup> 和 VarGrad<sup>[?2]</sup> 中的向输入图片中多次添加噪声的措施。Score-CAM<sup>[?]</sup> 和 Ablation-CAM<sup>[16]</sup> 没有使用反向传播中的梯度作为特征图的权重，它们将前向传播中从最后一层卷积层获得的特征图作为掩膜来扰动输入图片，利用网络输出值或者下降值作为特征图的权重，这种方法有效避免了使用梯度而产生的噪声，取得了良好的效果。Relevance-CAM<sup>[17]</sup>将卷积神经网络中提取的特征图再分别输入到卷积神经网络中，对每张特征图的预测结果进行层间相关性传播（Layer-wise Relevance Propagation），得到对应的相关性图，将相关性图全局平均池化后作为特征的权重，该方法有效解决了之前的 CAM 方法对卷积神经网络中间层可视化解释不足的问题。CAMERAS<sup>[18]</sup>提出了将输入图片进行多尺度放大再输入到网络当中，将提取到的不同分辨率的特征图和梯度全部放大到和原图分辨率一样。然而，基于类激活映射的方法只针对卷积神经网络进行设计，无法便捷的迁移到其他网络当中，比如基于 Transformer 的图片分类神经网络模型。

### 1.2.3 基于扰动的显著图解释方法

基于扰动的显著图可视化方法最突出的优点便是这种方法基本只关心网络的输入和输出，可以在不同结构网络轻松的应用，即使这些网络中的细节千差万别。Zeiler 等人<sup>[1]2</sup> 使用一个固定形状的方块来对输入图片进行扰动，观察输出变化从而找到对模型而已输入图片中最重要的部分。SHAP 方法<sup>[19]</sup> 使用了 Shapley 值，计算不同输入像素在是否存在的情况下对网络预测结果的影响，公平分配贡献度给

这些像素点。LIME<sup>[20]</sup>方法通过在较小的范围内扰动输入图片，得到输出结果，利用输入数据和输出数据重新训练一个可解释的代理模型去逼近黑盒模型在局部的决策边界，从而获得不同特征的重要性。RISE<sup>[21]</sup>利用蒙特卡洛方法随机生成的数量巨大的掩膜来扰动输入图片，将模型对特定类别输出概率作为的权重，再将多个所有掩膜加权相乘得到可视化结果。有研究提出了基于机器学习的方法，将掩膜作为优化对象，通过定义限制掩膜的损失函数，不断迭代从而找到最优的掩膜。在此基础上，Fong 等人<sup>[22]</sup>提出通过限制搜索区域，重新设计损失函数，达到了很好的效果。基于扰动的可视化算法因为能够对模型预测结果进行直接观察，所以可以较为真实的反映模型决策机制。但是，这种类型算法往往需要多次迭代，需要付出高额的计算成本

### 1.3 论文研究的主要内容

学位论文……

### 1.4 论文组织结构

本文……

## 第2章 相关理论介绍

### 2.1 引言

解释深度神经网络引起了越来越多的关注，因为它有助于理解网络的内部机制以及网络做出特定决策的原因。在计算机视觉领域中，可视化和理解深度神经网络最流行的方法之一是生成与网络决策相关的显著区域的显著性图。许多深度神经网络相关的可解释方面的研究和方法都可以在图像分类神经网络上生成显著图。显著图生成的质量可以直观反映不同可解释或者可视化算法的优劣，此外显著图还可以作为图像弱监督分割和目标的定位的一种手段，因其可以反映目标物体在图像中的空间位置而且其只需要训练好的图像分类神经网络即可完成任务。

深度神经网络的可解释方面的研究是在最近十年才逐渐兴起并收到关注的，在计算机视觉领域，基于深度卷积神经网络的图像分类模型是较早受到研究的，研究者试图从参数量庞大的深度卷积神经网络中找到输出结果和在输入图片中对应的依据。也有一些研究者将图像分类神经网络看作是一个黑盒，通过各种手段扰动输入图片观测输出结果变化来生成显著图。随着 Transformer 架构异军突起，基于 Transformer 架构的图像分类神经网络的可解释性也逐渐受到关注和研究，也已经由研究者设计了针对 Transformer 架构的反向传播归因机制，该机制在计算机视觉领域也能生成效果良好的显著图。上述的深度神经网络可解释研究生成的显著图较少关注显著图生成的质量和对关键特征的定位能力，本文的显著图解释研究专注于对显著图生成质量的改善和相关显著图解释算法的改善。

接下来本章首先介绍图像分类神经网络的主流架构概念包括基于卷积神经网络（CNN）的和基于 Transformer 架构的，然后介绍五种著名的深度学习可解释算法，这些算法在图像分类神经网络上也能生成显著图，后续实验当中这五种算法也会用来作为对比。

### 2.2 卷积神经网络

卷积神经网络（Convolutional Neural Networks, CNNs）一种特殊类型的神经网络，特别适合于计算机视觉应用。它的设计灵感来源于生物学上对动物视觉皮层的研究，旨在模拟人类视觉系统的工作原理。卷积神经网络的结构包括卷积层、池化层和全连接层，这些层的组合使得 CNNs 能够有效地处理图像识别、分类和分割等任务。有两个关键的设计思想推动了卷积架构在计算机视觉中的成功。首先，卷积神经网络利用图像的二维结构和一个邻域内的像素通常高度相关的事

因此, ConvNets 避免使用所有像素单元之间的一对一连接(即大多数神经网络的情况), 而倾向于使用分组的局部连接。此外, ConvNet 架构依赖于特征共享, 因此每个通道(或输出特征映射)都是由在所有位置使用相同滤波器进行卷积生成的, 如图2-1所示。与标准神经网络相比, 卷积神经网络的这一重要特征导致其架构依赖的参数要少得多。其次, 卷积神经网络还引入了一个池化步骤, 该步骤提供了一定程度的平移不变性, 使体系结构较少受到位置变化的影响。另外, 池化还允许网络逐渐看到更大的输入部分, 这要归功于网络接受视野的增加。随着网络深度的增加, 接受视野的大小增加(同时输入分辨率减小), 使得网络能够表示输入的更抽象特征。例如, 在目标识别任务中, 卷积神经网络的层从关注物体的边缘开始, 逐渐覆盖整个物体的各个部分, 最终从更高层次上覆盖整个物体。

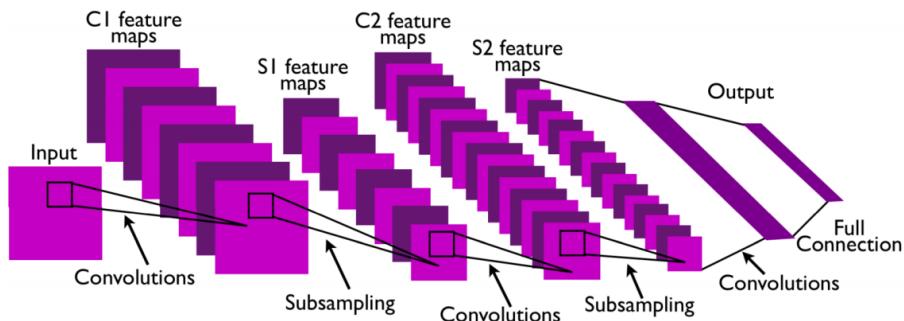


图 2-1 标准卷积网络结构示意图

Fig. 2-1 Illustration of the structure of a standard Convolutional Network

卷积神经网络的核心是卷积层, 它通过卷积操作提取输入图像的特征。卷积操作是一种线性、平移不变的操作, 它通过在输入信号上进行局部加权组合来捕获不同的特征。卷积操作的参数是卷积核, 不同的卷积核可以捕获图像的不同特征, 例如边缘、纹理和形状等。卷积层的输出被送入激活函数进行非线性变换, 以增加网络的表达能力。

除了卷积层, 池化层也是卷积神经网络中的重要组成部分。池化层通过对卷积层的输出进行下采样, 减少特征图的尺寸, 同时保留最显著的特征。这有助于使网络对输入的位置变化具有一定的不变性, 同时减少了参数数量, 提高了计算效率。

全连接层通常位于网络的末尾, 用于将卷积层和池化层提取的特征映射转换为最终的分类或回归结果。全连接层将所有特征进行组合, 最终输出网络对输入的预测结果。

## 2.3 基于 Transformer 的图像分类神经网络

Transformer<sup>[23]</sup>是一种基于自注意力机制的深度学习模型架构,最初由 Vaswani 等人于 2017 年提出,被广泛应用于自然语言处理任务,如机器翻译、文本生成和语言建模等。相较于传统的循环神经网络(RNN)和卷积神经网络(CNN),Transformer 在处理长距离依赖性和并行计算方面具有显著优势,成为了当今深度学习领域的研究热点。

Transformer 模型的核心思想是完全基于注意力机制来实现信息的传递和组合。在传统的神经网络结构中,信息的传递是通过固定的层次顺序进行的,而 Transformer 则引入了自注意力机制,使得模型可以在不同位置之间建立动态的注意力联系。这种注意力机制使得模型能够更好地捕捉输入序列中不同位置之间的依赖关系,从而提高了模型在处理长距离依赖性任务时的性能。

如图2-2所示,Transformer 模型由编码器和解码器两部分组成,每部分都包含多层的注意力机制模块。在编码器中,输入序列首先通过一个自注意力层,然后再通过一个全连接前馈神经网络层。在自注意力层中,每个输入位置都可以与其他位置建立注意力联系,从而使得模型能够同时考虑到整个输入序列的信息。在解码器中,除了编码器的结构外,还引入了一个额外的注意力层,用来对编码器的输出进行进一步处理。

除了自注意力机制,Transformer 还引入了位置编码来表示输入序列中不同位置的信息。由于 Transformer 没有显式的循环结构,无法像 RNN 一样自然地捕捉序列中的位置信息,因此位置编码的引入可以帮助模型更好地理解输入序列中不同位置的相对位置关系。

另一个 Transformer 模型的重要特点是其并行计算能力。由于自注意力机制的特性,Transformer 可以高效地进行并行计算,加速模型的训练和推理过程。这使得 Transformer 在处理大规模数据和长序列任务时具有明显的优势。

### 2.3.1 多头自注意力机制

多头注意力机制是一种用于增强神经网络模型对输入序列的表示能力的技术。它通过将注意力机制应用于多个头部(head)来并行处理不同的注意力表示,从而提高模型的泛化能力和性能。多头注意力机制最早由 Vaswani 等人在 Transformer 模型中提出,并在自然语言处理领域取得了巨大成功。

多头注意力机制的基本原理是将输入序列分别映射到多个不同的查询、键和值空间,并通过计算每个头部的注意力权重来融合不同的信息。具体来说,多头注

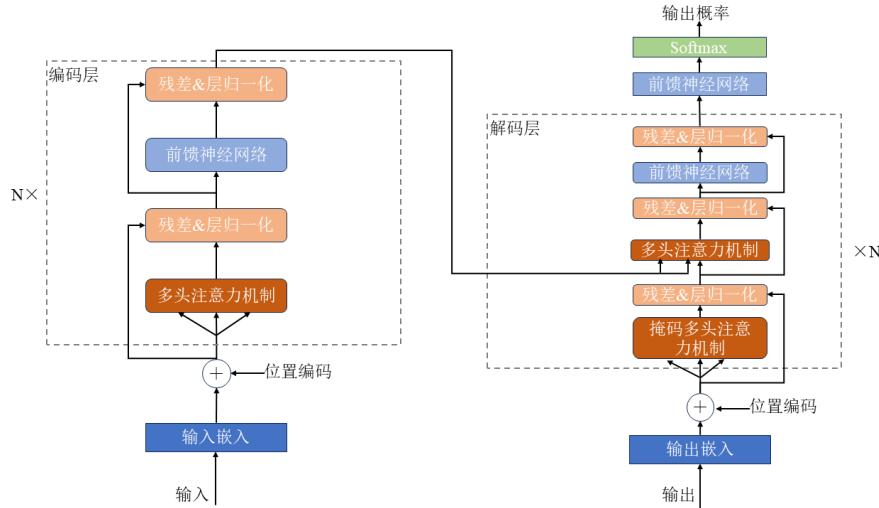


图 2-2 Transformer 结构示意图

Fig. 2-2 Illustration of the structure of Transformer

注意力机制可以表示为以下公式：

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O \quad (2-1)$$

其中， $Q, K, V$  分别表示输入序列的查询、键和值， $h$  表示头部的数量， $\text{head}_i$  表示第  $i$  个头部的注意力表示， $W^O$  是输出权重矩阵。

在多头注意力机制中，每个头部都有自己的查询、键和值权重矩阵，通过线性变换将输入序列映射到不同的空间。然后，每个头部分别计算注意力权重，并将不同头部的注意力表示进行拼接，最后通过输出权重矩阵  $W^O$  融合多个头部的信息。

多头注意力机制的优势在于可以提高模型对不同部分的关注度，从而更好地捕捉输入序列中的重要信息。通过并行计算多个头部的注意力表示，模型可以更好地处理长距离依赖关系和复杂的序列结构，提高模型的泛化能力和性能。除了在 Transformer 模型中应用多头注意力机制，它还被广泛应用于其他神经网络模型中，如 BERT、GPT 等。在自然语言处理、计算机视觉等领域，多头注意力机制已成为提高模型性能的重要技术之一。

### 2.3.2 编码器

Transformer 模型是一种基于注意力机制的神经网络模型，由编码器和解码器组成，广泛应用于自然语言处理领域。编码器是 Transformer 模型的核心组件之一，它负责将输入序列转换成隐藏表示，提供给解码器进行生成输出序列。编码器由多个编码器层堆叠而成，每个编码器层内部包含多头自注意力机制和前馈神经网络。

编码器的每一层都包含两个子层，分别是多头自注意力机制和前馈神经网络。

下面分别介绍编码器每一层的计算过程：

1. 多头自注意力机制 (Multi-Head Self-Attention):

多头自注意力机制是编码器每一层的第一个子层，用于计算输入序列的注意力表示。具体的计算过程可以表示为以下公式：

$$\text{Attention}(Q, K, V) = \text{softmax} \left( \frac{QK^T}{\sqrt{d_k}} \right) V \quad (2-2)$$

其中， $Q, K, V$  分别表示输入序列的查询、键和值， $d_k$  表示注意力头部的维度。在多头自注意力机制中，将输入序列分别映射到多个不同的查询、键和值空间，并通过计算每个头部的注意力权重来融合不同的信息。

2. 前馈神经网络 (Feed-Forward Neural Network):

前馈神经网络是编码器每一层的第二个子层，用于对注意力表示进行非线性变换。具体的计算过程可以表示为以下公式：

$$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2 \quad (2-3)$$

其中， $x$  表示输入的注意力表示， $W_1, b_1, W_2, b_2$  分别表示两个线性变换的权重和偏置。前馈神经网络通过两个线性变换和 ReLU 激活函数来对注意力表示进行非线性变换，从而提高模型的表征能力。

在编码器中，每个编码器层都包含多头自注意力机制和前馈神经网络这两个子层，通过堆叠多个编码器层来构建整个编码器。每个编码器层都可以表示为以下公式：

$$\text{EncoderLayer}(x) = \text{FFN}(\text{Attention}(x) + x) \quad (2-4)$$

其中， $x$  表示输入序列的隐藏表示， $\text{Attention}(x)$  表示多头自注意力机制的输出， $\text{FFN}(\cdot)$  表示前馈神经网络的输出。通过将多头自注意力机制的输出和原始输入序列相加，并通过前馈神经网络进行非线性变换，编码器可以逐层提取输入序列的特征表示。

在 Transformer 模型的编码器中，除了多头自注意力机制和前馈神经网络之外，还包括残差连接和层归一化这两个重要的组件。残差连接和层归一化在每个编码器层中都起着重要的作用，有助于提高模型的训练稳定性和加速收敛。下面分别介绍这两个组件的作用和公式表达：

3. 残差连接 (Residual Connection):

残差连接是指将输入序列的隐藏表示与经过子层处理后的表示进行相加，从而形成残差连接。这样的设计有助于减轻梯度消失和梯度爆炸的问题，提高模型的训练效果。残差连接的计算公式可以表示为：

$$\text{Residual}(x, \text{Sublayer}(x)) = x + \text{Sublayer}(x) \quad (2-5)$$

其中， $x$  表示输入序列的隐藏表示， $\text{Sublayer}(x)$  表示经过子层处理后的表示。通过残差连接，编码器每一层的输出可以保留原始输入序列的信息，同时加入子层处理后的新信息，从而提高模型的表征能力。

#### 4. 层归一化 (Layer Normalization):

层归一化是一种对神经网络中每一层输出进行归一化的技术，有助于加速模型的收敛和提高模型的泛化能力。在编码器的每个子层（包括多头自注意力机制和前馈神经网络）后都会应用层归一化操作。层归一化的计算公式可以表示为：

$$\text{LayerNorm}(x) = \gamma \odot \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta \quad (2-6)$$

其中， $x$  表示输入序列的隐藏表示， $\gamma$  和  $\beta$  分别表示学习到的缩放因子和偏置项， $\mu$  和  $\sigma$  分别表示输入的均值和标准差， $\epsilon$  是一个很小的常数，避免分母为零。层归一化通过对每一层的输出进行归一化，有助于减少内部协变量转移，提高模型的训练效果。

### 2.3.3 解码器

在 Transformer 模型中，解码器是用于生成目标序列的部分，与编码器相对应。解码器由多个解码器层组成，每个解码器层包含掩码多头自注意力机制、多头自注意力机制、层归一化、残差连接、前馈神经网络。此处重点介绍一下掩码多头自注意力机制 (Masked Multi-Head Self-Attention)。

在 Transformer 模型中，掩码多头自注意力机制是解码器中的关键组件之一，它允许解码器在生成目标序列时关注目标序列中的不同位置，并避免未来信息的干扰。掩码多头自注意力机制包含多个注意力头，每个头都学习不同的注意力权重，然后将它们合并起来以获得最终的注意力表示。

在掩码多头自注意力机制中，给定一个输入序列  $X = \{x_1, x_2, \dots, x_n\}$ ，我们首先通过线性变换得到查询 (Query)、键 (Key) 和值 (Value) 的表示：

$$Q = XW^Q, K = XW^K, V = XW^V \quad (2-7)$$

其中  $W^Q, W^K, W^V$  是学习的权重矩阵。接下来，计算每个注意力头的注意力分数  $A_i$ ：

$$A_i = \text{softmax}\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) \quad (2-8)$$

其中  $Q_i, K_i$  表示第  $i$  个注意力头的查询和键， $d_k$  是键的维度。然后，将每个注意力

头的注意力分数与值相乘并加权求和，得到最终的输出表示：

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O \quad (2-9)$$

其中  $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$  表示第  $i$  个注意力头的输出， $W^O$  是最终的输出权重矩阵。

在掩码多头自注意力机制中，为了避免未来信息的泄露，会对注意力分数进行掩码操作。具体来说，在计算注意力分数时，会将当前位置之后的信息屏蔽掉，以确保模型只能使用当前位置之前的信息。在实际操作中是将输入矩阵加上一个上三角矩阵，该上三角矩阵元素为无穷大。这种掩码操作有助于解码器在生成目标序列时保持因果关系，不受未来信息的影响。

通过掩码多头自注意力机制，解码器可以有效地关注目标序列中的不同部分，并根据输入序列和先前生成的部分序列调整生成的单词，从而实现对目标序列的准确生成。

### 2.3.4 应用于图像分类领域的 Transformer

Transformer 技术最初是为了处理自然语言处理任务而设计的，但随着其出色的性能和可扩展性，人们开始将 Transformer 技术应用于其他领域，包括图像分类。虽然传统的卷积神经网络（CNN）在图像分类任务中表现出色，但 Transformer 在图像分类领域也展示出了很好的潜力。

在图像分类领域应用 Transformer 技术的一个主要方法是 Vision Transformer (ViT)。ViT 将输入的图像像素视为序列化的数据，然后利用 Transformer 模型来对序列化的图像数据进行处理。ViT 的基本架构如下：

1. 图像块划分：将输入图像分成固定大小的图像块（patches），每个图像块将被视为一个 token。
2. 输入嵌入层：将每个图像块映射为一个向量表示。
3. 位置编码：为每个图像块的向量表示添加位置编码，以保留图像块之间的位置信息。
4. Transformer 编码器：使用 Transformer 模型对序列化的图像数据进行处理，包括多层的自注意力机制和前馈神经网络。
5. 分类头：将 Transformer 的输出通过一个全连接层进行分类，得到图像的类别预测。

ViT 在图像分类领域的应用有以下优点：- 可扩展性：Transformer 模型的自注意力机制使得 ViT 能够捕捉图像中的全局信息，而不受卷积核大小的限制。- 灵活性：ViT 可以处理不同尺寸的图像，而无需调整网络结构。- 泛化能力：ViT 在处理小样本数据集时表现较好，因为 Transformer 模型具有强大的表示学习能力。

尽管 ViT 在图像分类领域取得了一定的成功，但也存在一些挑战，例如对大规模数据集的训练需要较长的时间，以及对输入图像块的划分方式和位置编码的设

计需要一定的经验和调整。

总的来说，Transformer 技术在图像分类领域的应用为研究人员提供了一种新的思路和工具，有望进一步推动图像分类任务的发展和性能提升。

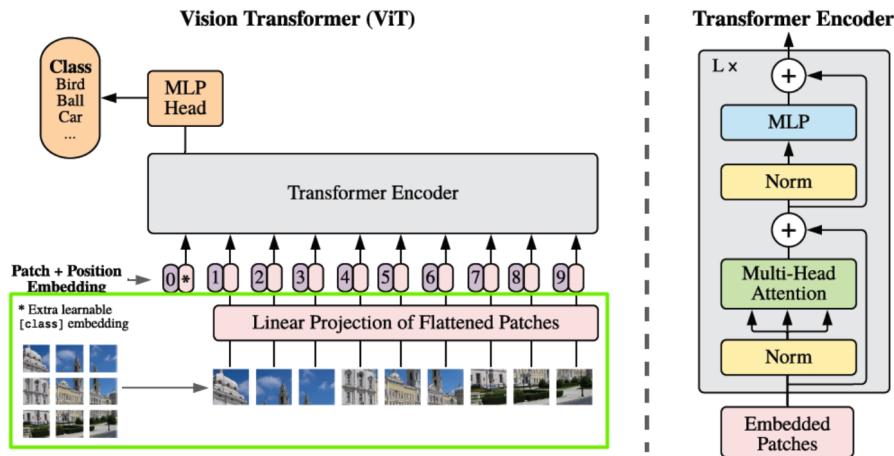


图 2-3 ViT 结构示意图

Fig. 2-3 Illustration of the structure of ViT

## 2.4 常见的显著图解释方法

本节将会介绍五种著名的显著图解释方法，包括 Grad-CAM、Grad-CAM++、Score-CAM、LRP 和 Transformer attribution。本节将会介绍这五种方法的具体理论依据，各自的应用条件和算法流程。

### 2.4.1 Grad-CAM

Grad-CAM (Gradient-weighted Class Activation Mapping) 是一种用于解释深度学习模型的方法，它能够生成图像级别的重要区域热力图，从而帮助理解模型的决策过程。Grad-CAM 的主要优点是它不需要对模型进行修改，可以应用于任何卷积神经网络模型，并且能够提供直观的可视化结果。

Grad-CAM 的核心思想是利用模型的梯度信息和特征图信息来推导出图像中哪些区域对于模型的分类决策最为关键。在深度学习模型中，每个卷积层都可以看作是对输入图像进行特征提取的过程，因此可以通过分析梯度信息来了解哪些特征对于模型的分类决策起到了关键作用。之前的一些研究已经断言，卷积神经网络的更深层表征捕获了更高层次的视觉结构<sup>[23][24]</sup>。此外，卷积层自然地保留了在全连接层中丢失的空间信息，因此 Grad-CAM 的作者认为最后一层卷积层在高级语义和详细空间信息之间有良好的表现。因此 Grad-CAM 使用流入卷积神经网

络最后一个卷积层的梯度信息，为每个神经元分配一个特定感兴趣的决策的重要性值。虽然 Grad-CAM 的技术相当通用，因为它可以用来解释深度网络任何层的激活，但是一般来说它用来可视化解释输出层做的决定。

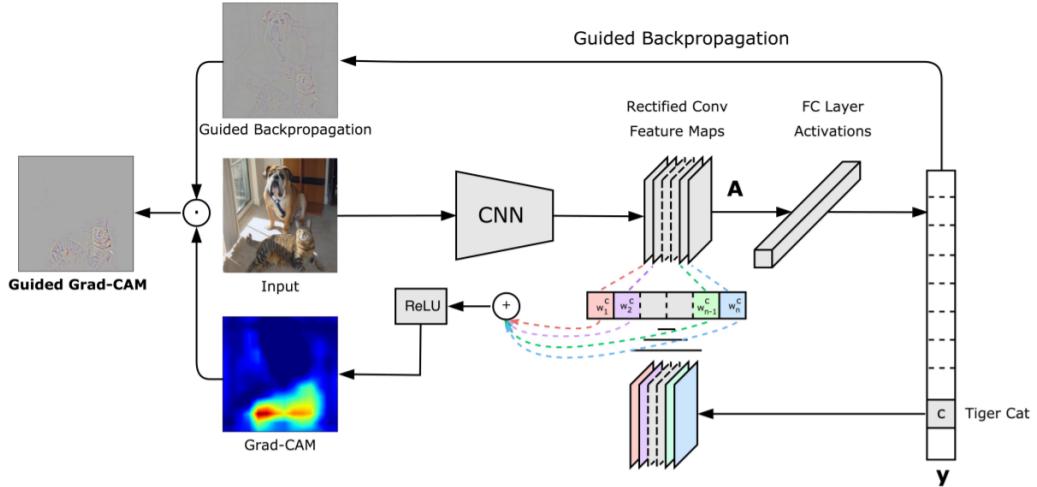


图 2-4 Grad-CAM 流程示意图

Fig. 2-4 Pipeline of Grad-CAM

下面详细解释一下 Grad-CAM 的计算过程。若假设  $I \in \mathbb{R}^{3 \times H \times W}$  是输入图片， $\mathcal{F}$  是预训练好的基于卷积神经网络的图像分类模型， $\mathcal{F}_c(I)$  表示输入图片是  $I$  的情况下模型  $\mathcal{F}$  输出的关于类别索引  $c$  的分数（该分数未经 Softmax 函数）。对于给定的模型  $\mathcal{F}$  卷积层  $l$ ，在图片  $I$  输入到模型  $\mathcal{F}$  前向传播过程中可以从卷积层  $l$  提取特征图集合  $\mathbf{A}^*$ 。为了获取具有类别区分性的显著图，Grad-CAM 首先需要计算关于类别  $c$  的梯度，为了得到  $\mathcal{F}_c(I)$  相对于第  $k$  张特征图  $A^k \in \mathbf{A}^*$  的梯度，我们可以对  $\mathcal{F}_c(I)$  进行反向传播。为了得到每个特征图的重要性权重，梯度会在高度和宽度两个维度上进行全局平均。

$$w_k^c = \frac{1}{m \times n} \sum_{i=1}^m \sum_{j=1}^n \frac{\partial \mathcal{F}_c(I)}{\partial A_{ij}^k(I)} \quad (2-10)$$

式子 2-10 表示了特征图集合中特征图  $A^k$  的权重计算方式，其中， $A_{ij}^k$  表示第  $k$  张特征图中第  $i$  行和第  $j$  列的值， $m$  和  $n$  分别是梯度矩阵的宽度和高度。 $w_k^c \in \mathbf{W}^*$  是第  $k$  张特征图的权重。 $\mathbf{W}^*$  是大小为  $K$  的一维向量， $K$  是  $\mathbf{A}^*$  的通道数。

$w_k^c$  的含义表示第  $k$  张特征图在后续计算中对输出分数  $\mathcal{F}_c(I)$  的贡献程度。最后将每张特征图和其权重相乘然后将相乘结果进行线性组合，最后通过 ReLU 函数

数即可得到 Grad-CAM 的显著图。具体计算方式如下所示：

$$L_{Grad-CAM} = \text{ReLU} \left( \sum_K (w_k^c A^k) \right) \quad (2-11)$$

需要注意的是注意，这会产生一个与特征图  $A^k$  大小相同的原始显著图，若需要将其展示重叠在输入图片上，应该将其进行上采样。还有就是 ReLU 函数应用于显著图的线性组合，是因为 Grad-CAM 只对感兴趣的类别有积极影响的特征感兴趣，即类别索引  $c$  所代表的类别。原始显著图中为负值的像素可能属于图像中的其他类别。若没有这个 ReLU 函数，显著图中会将其他不相干的类别进行凸显。图2-4简要展示了 Grad-CAM 的工作流程。

#### 2.4.2 Grad-CAM++

在原始的 CAM<sup>[10]</sup> 方法中每张特征图的权重  $w_k^c$  是通过最后一层卷积层生成的特征图来训练线性分类器来获得的，但是其局限是必须使用全局平均池化层，需要重新训练模型，Grad-CAM 解决了这一问题，通过最后的类别分数反向传播对最后一层卷积层的特征图求导，利用单个梯度矩阵的平均值来得到单张特征图权重，但是这会导致其对于某张特征图加权值过大从而将其他包含目标物体特征的特征图权值过小，从而在最后生成的显著图上忽略关键的特征。图2-5举例说明了这一现象，如图所示，图中黑色方块部分表示目标物体在图中二维空间上特征分布，最后经过卷积层生成 3 个特征图分布是  $A^1$ 、 $A^2$  和  $A^3$ ，其中  $A^1$  特征图的特征区域占比最多， $A^2$  和  $A^3$  的特征区域明显较少，如果使用 Grad-CAM 的加权方式则最后的生成的显著图会将  $A^2$  和  $A^3$  的特征区域弱化甚至忽略，这显然是不对的。这种情况一般会出现在有多个目标物体的图片当中。

Grad-CAM++ 为了解决这一现象则没有直接将梯度矩阵直接全剧平均池化取平均值来作为特征图权重，而且为梯度矩阵上每一个元素赋予了一个权重，之后将梯度矩阵上的元素值全部相加取和作为特征图权重。具体的权重表示公式如下：

$$w_k^c = \sum_i \sum_j \alpha_{ij}^{kc} \cdot \text{ReLU} \left( \frac{\partial Y^c}{\partial A_{ij}^k} \right) \quad (2-12)$$

式2-12中  $Y^c$  即和章节2.4.1  $\mathcal{F}_c(I)$  含义一致，均表示输入图片进入模型中，模型输出的关于类别索引  $c$  的分数。 $i$  和  $j$  表示特征图中的  $A^k$  的行列迭代器。式2-12中  $\alpha_{ij}^{kc}$

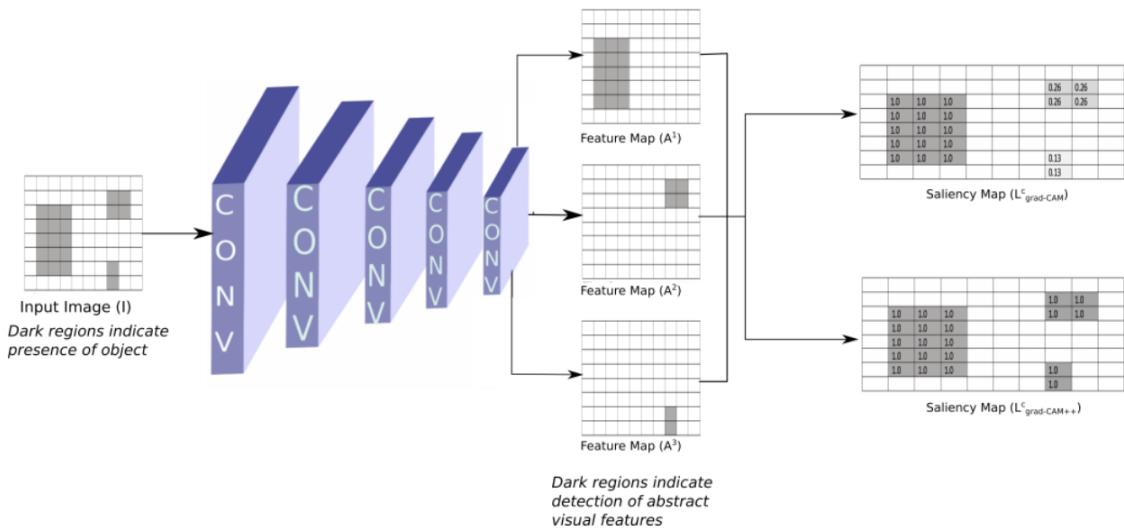


图 2-5 一个假设性的例子说明 Grad-CAM++ 的优势

Fig. 2-5 A hypothetical example elucidating the intuition behind Grad-CAM++

的具体计算方式如下：

$$\alpha_{ij}^{kc} = \frac{\frac{\partial^2 Y^c}{(\partial A_{ij}^k)^2}}{2 \frac{\partial^2 Y^c}{(\partial A_{ij}^k)^2} + \sum_a \sum_b A_{ab}^k \frac{\partial^3 Y^c}{(\partial A_{ij}^k)^3}} \quad (2-13)$$

将式2-13和式2-12结合即可得到如下完整的权重计算公式：

$$w_k^c = \sum_i \sum_j \left[ \frac{\frac{\partial^2 Y^c}{(\partial A_{ij}^k)^2}}{2 \frac{\partial^2 Y^c}{(\partial A_{ij}^k)^2} + \sum_a \sum_b A_{ab}^k \frac{\partial^3 Y^c}{(\partial A_{ij}^k)^3}} \right] \cdot \text{ReLU} \left( \frac{\partial Y^c}{\partial A_{ij}^k} \right) \quad (2-14)$$

得到权重后，将权重  $w_k^c$  和其对应的特征图  $A^k$  相乘，然后所有特征图线性相加即可得到 Grad-CAM++ 生成的最终显著图。计算过程和式2-11一致。图2-6中展示了 CAM、Grad-CAM、Grad-CAM++ 的权重计算方式异同。

### 2.4.3 Score-CAM

Grad-CAM 和 Grad-CAM++ 都依赖反向传播的梯度作为权重，但是这种基于梯度的方法也有缺点，若激活函数是 Sigmoid 函数或者 ReLU 函数，则会分别带来梯度饱和以及梯度消失的问题<sup>[25]</sup>，反映到显著图上就是会导致噪声问题。此外依靠梯度作为权重也存在不可靠的问题，如图2-7所示，在这幅图中 (2)(3)(4) 分别是对应的特征图上采样后叠加到原图上，其中没有遮盖的部分表示该特征图在原图上较为关注的特征区域，将 (2)(3)(4) 通过图像分类模型，得到对应类别的得分

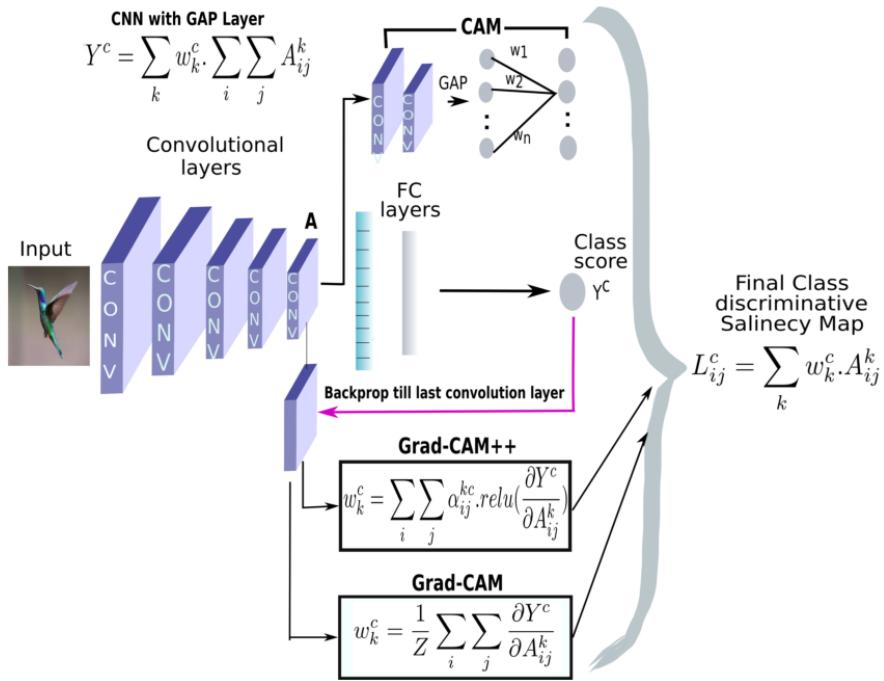


图 2-6 CAM, Grad-CAM, Grad-CAM++ 的各自计算方式

Fig. 2-6 Respective calculations for CAM, Grad-CAM, Grad-CAM++

分别是 0.003, 0.999, 0.997。从结果上看, (3)(4) 的特征图关注的区域对于该类别是比较重要的。然而在生成 Grad-CAM 的显著图时, 这三者计算出的权重分别是 0.035, 0.027, 0.021。反而是对于结果影响很小的 (2) 的特征图权重更高。这就显示出了基于梯度的类激活映射方法的不足。

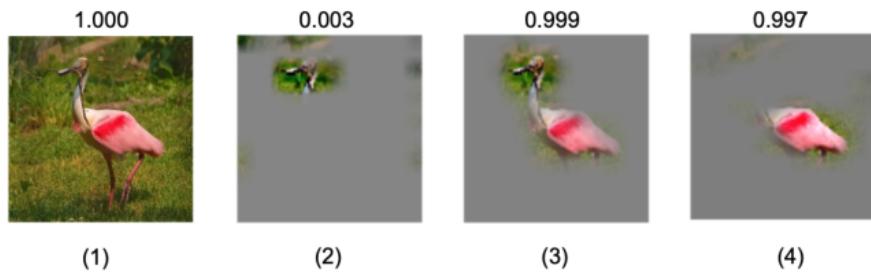


图 2-7 特征图扰动示例

Fig. 2-7 Example of feature map perturbation

因此 Score-CAM 不依赖于梯度来获得特征图线性加权的权重, 而是通过每张特征图在目标类上的前向传播的概率分数来获得每张特征图的权重。下面通过具体的定义来说明 Score-CAM 的计算方式。当输入图片为  $I \in \mathbb{R}^{3 \times H \times W}$  时, 图像分类模型为  $\mathcal{F}$  时, 经过前向传播从指定卷积层  $l$  提取的第  $k$  个通道的特征图为  $A_l^k$ , 那么该特征图对于指定类别  $c$  的概率分数  $\mathcal{F}_c(I)$  的贡献程度, 即  $A_l^k$  权重  $C(A_l^k)$  可以

由以下式子计算得出：

$$C(A_l^k) = \mathcal{F}_c(I \circ H(l^k)) - \mathcal{F}_c(I_b) \quad (2-15)$$

式子2-15中， $I_b$  表示一张基准图像，该图像要求在类别索引  $c$  上的概率值尽可能的小，即  $\mathcal{F}_c(I_b)$  的值趋近于 0，在算法实际实施的过程当中，该基准图片  $I_b$  一般是值均为 0 的全黑图像。 $\circ$  表示哈达玛积运算，即运算符两边的矩阵对应位置元素相乘， $H(l^k)$  是  $A_l^k$  上采样后的图片。其计算式子如下所示：

$$H(l^k) = s(Up(A_l^k)) \quad (2-16)$$

式2-16中， $Up$  表示上采样运算，此处是将特征图其  $A_l^k$  上采样至原始输入图片尺寸。 $s$  表示对图片进行归一化运算，将图片中的所有像素值控制在  $[0,1]$ 。最后可以得到 Score-CAM 的显著图计算式子：

$$L_{\text{Score-CAM}}^c = \text{ReLU} \left( \sum_k \alpha_k^c A_l^k \right) \quad (2-17)$$

式2-17中，权值  $\alpha_k^c$  即表示  $C(A_l^k)$ 。Score-CAM 和 Grad-CAM，Grad-CAM++ 最大的不同就是它创新性的利用特征图作为扰动输入图片的掩膜，以网络模型关于指定类别的概率分数来作为每张特征图的权重。

#### 2.4.4 LRP

LRP 全称为 Layer-wise relevance propagation，意思是层间相关性传播，是深度学习中用于理解和解释神经网络决策的一种技术。LRP 的主要思想是通过赋予输出结果一个相关性的分数，然后定义反向传播规则，将该相关分数逐层向后传播，通过计算给每层的每个神经元赋予一个相关性分数来表示它对输出结果的贡献程度，传播回输入层即可得到输入图片中每个像素的相关性分数，分数值的大小表示该像素与最终决策结果的相关性程度，也即贡献程度。利用获得的输入图片的每个像素的相关性分数也可以生成显著图。在传播过程中每层网络的神经元的相关性分数总和是一致的。通过这样做，LRP 有助于揭示每个输入特征对最终决策的贡献，为神经网络的内部运作提供宝贵的见解。LRP 的主要思想遵循以下公式：

$$f(x) \approx \sum_{d=1}^V R_d \quad (2-18)$$

式2-18中  $f(x)$  表示输入图片是  $x$  的情况下，其中某一个特定类别在图片  $x$  中的存在概率。 $R_d$  表示图片中单一像素点对该类别的相关性分数，可以看作是贡献值。图2-8大体展示了这种分解思想。图像  $x$  被转换为一串特征向量，并应用分类器将

图像归入给定类别，如“猫”或“无猫”，该图中将“猫”的分类结果概率分数作为总的相关性分数反向传播至输入图片中的每个像素，最终可以得到可视化单个像素对预测的贡献。

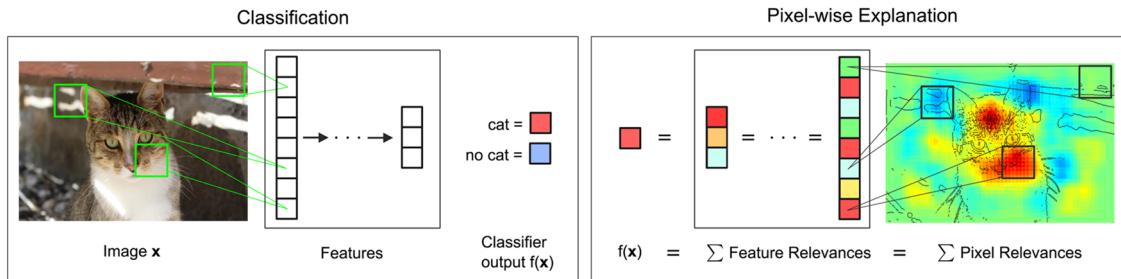


图 2-8 像素相关性分解过程可视化

Fig. 2-8 Visualization of the pixel relevance decomposition process

下面结合公式推导详细说明 LRP 的计算过程。对于图像分类神经网络关于某个类别的输出  $f(x)$  有以下规则，每层的所有神经元的相关性分数是相等的。

$$f(x) = \dots = \sum_{d \in l+1} R_d^{l+1} = \sum_{d \in l} R_d^l = \dots = \sum_d R_d^1 \quad (2-19)$$

式2-19中  $l$  表示神经网络的某一层， $l$  越大越靠近输出层。

对于普通的多层神经网络，有如图2-9所示的计算过程。其中：

$$z_{ij} = x_i w_{ij} \quad (2-20)$$

$$z_j = \sum_i z_{ij} + b_j \quad (2-21)$$

$$x_j = g(z_j) \quad (2-22)$$

对于式2-20，神经元  $i$  乘以其和神经元  $j$  之间的权重  $w_{ij}$  得到中间计算结果  $z_{ij}$ 。而式2-21的值是第  $i$  层所有神经元和权重  $w_{ij}$  相乘后得到的  $z_{ij}$  的和加上偏置项  $b_j$  得到的。最后  $z_j$  经过激活函数  $g$  得到第  $j$  层某个神经元的值  $x_j$ 。这三个式子的意思可以总结成所有上层神经元到下层的某个神经元  $j$  的值，也可以理解成贡献，决定了下层神经元  $j$  的中间值  $z_j$ ，只不过最后要加上偏置项和经过激活函数。

从以上前向传播的示例过程中可以推导出 LRP 相关性传播的基本过程，如图2-10所示，左半部分是前向传播的过程，右半部分是 LRP 的相关性反向传播的过程，其中  $R_i^{(l)}$  表示第  $l$  层神经元  $i$  的相关性分数， $R_{i \leftrightarrow j}^{(l,l+1)}$  表示  $l+1$  层的某个神经元  $j$  的相关性等于  $l+1$  层的神经元  $j$  给  $l$  层所有神经元的相关性之和。下面结合图2-11进行详细说明。

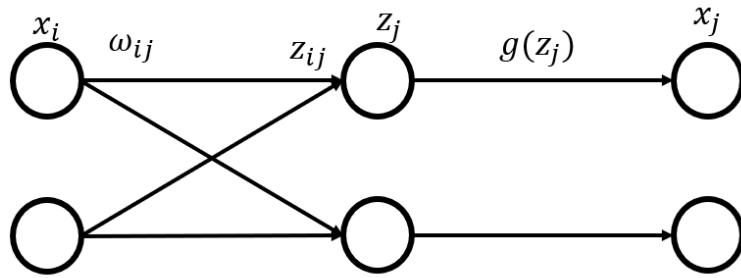


图 2-9 多层神经网络前向传播计算示例

Fig. 2-9 Example of a multilayer neural network forward propagation calculation

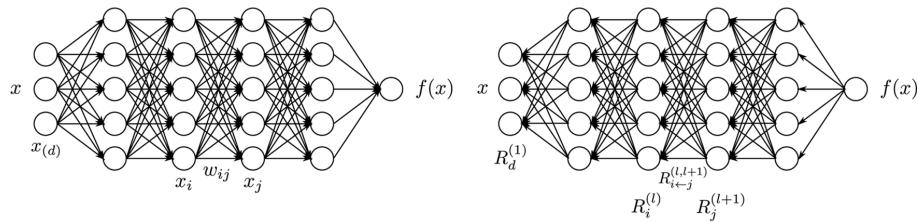


图 2-10 多层神经网络下前向传播和层间相关性传播过程

Fig. 2-10 Forward propagation and LRP processes under multilayer neural networks

在图2-11中的右半部分,第3层的7号神经元的相关性分数 $R_7^{(3)}$ 相关性反向传播时是按照如下式子分配的:

$$R_7^{(3)} = R_{4 \leftarrow 7}^{(2,3)} + R_{5 \leftarrow 7}^{(2,3)} + R_{6 \leftarrow 7}^{(2,3)} \quad (2-23)$$

其中 $R_{4 \leftarrow 7}^{(2,3)}$ 、 $R_{5 \leftarrow 7}^{(2,3)}$ 和 $R_{6 \leftarrow 7}^{(2,3)}$ 的值,也就是7号神经元分配给第二层的4,5和6号神经元的相关性分数在神经网络是线性的情况下(没有激活函数)可以结合公式2-20进行计算,具体计算公式如下:

$$R_{i \leftarrow j}^{(l,l+1)} = \frac{z_{ij}}{\sum_i z_{ij}} R_j^{(l+1)} \quad (2-24)$$

式2-24的含义就是就是下层的神经元 $j$ 的分配给上层神经元 $j$ 相关性分数是 $j$ 的总相关性分数乘以神经元 $i$ 输出给 $j$ 的值占 $j$ 收到上层所有值的比。

但是在实际应用当中,神经网络会存在激活函数和偏置,式2-24无法完美应用。考虑到激活函数是双曲正切  $Tanh(x)$  和  $ReLU$  函数的情况下,可以用以下公式取近似值:

$$R_{i \leftarrow j}^{(l,l+1)} = \frac{z_{ij}}{z_j} R_j^{(l+1)} \quad (2-25)$$

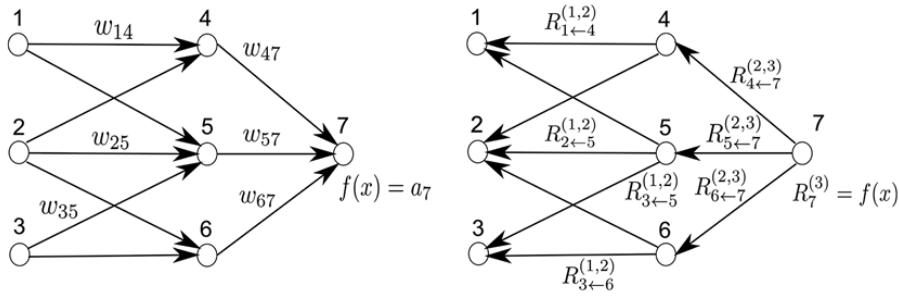


图 2-11 层间相关性传播分配相关性分数的例子

Fig. 2-11 Example of LRP assigning relevance scores

其他激活函数情况可以使用如下式子计算：

$$\sum_i R_{i \leftarrow j}^{(l, l+1)} = R_j^{(l+1)} \cdot \left( 1 - \frac{b_j}{z_j} \right) \quad (2-26)$$

#### 2.4.5 Transformer Attribution

Transformer Attribution 是 Chefer 等人<sup>[26]</sup>于近些年提出的针对基于 Transformer 架构的深度神经网络的可解释方法，其在图像分类神经网络上也能够生成质量较高的显著图。Transformer Attribution 针对 Transformer 的特殊结构改进了传统的 LRP 算法，针对相关性反向传播过程中的注意力层和残差连接的提出了特殊的 LRP 传播规则，同时将其应用到基于 transformer 的图片分类模型中，相比之前的可视化算法，Transformer Attribution 得到了较为可靠的显著图结果。

下面是 Transformer Attribution 的详细推导过程。对于 LRP 传播，总是遵守以下规则：

$$\sum_j R_j^{(n)} = \sum_i R_i^{(n-1)} \quad (2-27)$$

即每层的相关性分数总和总是相等的。

若用  $L^{(n)}(\mathbf{X}, \mathbf{Y})$  表示层对两个张量  $\mathbf{X}$  和  $\mathbf{Y}$  的操作。通常，这两个张量是第  $n$  层的输入特征映射和权重。则 LRP 遵循通用深度泰勒分解<sup>[27]</sup>的规则，具体如下表示：

$$R_j^{(n)} = \mathcal{G}(\mathbf{X}, \mathbf{Y}, R^{(n-1)}) = \sum_i \mathbf{X}_j \frac{\partial L_i^{(n)}(\mathbf{X}, \mathbf{Y})}{\partial \mathbf{X}_j} \frac{R_i^{(n-1)}}{L_i^{(n)}(\mathbf{X}, \mathbf{Y})} \quad (2-28)$$

对于激活函数是  $ReLU$  的情况，第  $n$  层的第  $j$  个神经元的相关性分数可以如下计算：

$$R_j^{(n)} = \mathcal{G}(x^+, w^+, R^{(n-1)}) = \sum_i \frac{x_j^+ w_{ji}^+}{\sum_{j'} x_{j'}^+ w_{j'i}^+} R_i^{(n-1)} \quad (2-29)$$

式2-28和式2-29中,  $j$  是和第  $n$  层的神经元对应的,  $i$  和第  $n-1$  层的神经元对应,  $j'$  是第  $n$  层神经元的取和过程中的迭代计数符号。加号表示只保留正值, 计算方式为  $\max(0, x)$ 。但是 Transformer 使用的是  $GELU$  函数, 有正值也有负值同时也是非线性函数。其相关性分数计算方式如下:

$$R_j^{(n)} = \mathcal{G}_q \left( x, w, q, R^{(n-1)} \right) = \sum_{\{i|(i,j) \in q\}} \frac{x_j w_{ji}}{\sum_{\{j'|(j',i) \in q\}} x_{j'} w_{j'i}} R_i^{(n-1)} \quad (2-30)$$

其中  $q = \{(i, j) | x_j w_{ji} \geq 0\}$ , 表示只考虑对输入结果有正向贡献的相关性值。

在注意力层中有两个特征图作为输入, 因此要分别考虑两者的相关性计算。假设有张量  $u$  和  $v$ , 那么它们各自的相关性分数计算表示如下:

$$R_j^{u(n)} = \mathcal{G} \left( u, v, R^{(n-1)} \right), \quad R_k^{v(n)} = \mathcal{G} \left( v, u, R^{(n-1)} \right) \quad (2-31)$$

同时它们满足相关性守恒的原则:

$$\sum_j R_j^{u(n)} + \sum_k R_k^{v(n)} = \sum_i R_i^{(n-1)} \quad (2-32)$$

式2-32中  $u$  和  $v$  两个向量做加法操作 (对应跳跃连接的情况) 时满足上述相关性原则, 但是当两者作乘法时 (对应注意力层矩阵乘法的情况) 则不适用。因此为了解决由于矩阵乘法导致的在注意机制下相关性不守恒以及跳跃连接时存在的数值膨胀问题, Transformer Attribution 使用如下公式2-33和式2-34分别计算  $u$  和  $v$  的相关性:

$$\bar{R}_j^{u(n)} = R_j^{u(n)} \frac{\left| \sum_j R_j^{u(n)} \right|}{\left| \sum_j R_j^{u(n)} \right| + \left| \sum_k R_k^{v(n)} \right|} \cdot \frac{\sum_i R_i^{(n-1)}}{\sum_j R_j^{u(n)}} \quad (2-33)$$

$$\bar{R}_k^{v(n)} = R_k^{v(n)} \frac{\left| \sum_k R_k^{v(n)} \right|}{\left| \sum_j R_j^{u(n)} \right| + \left| \sum_k R_k^{v(n)} \right|} \cdot \frac{\sum_i R_i^{(n-1)}}{\sum_k R_k^{v(n)}} \quad (2-34)$$

经过规范化的计算公式在各网络层满足  $\sum_i R_i^{(n)} = 1$ , 且完全满足各层相关性守恒的原则即式2-35所表示的含义。此外, 每个张量的相关性和也遵循式2-36的规则限制, 该规则意思即第  $n$  层的每个张量的相关性和都不超过第  $n-1$  层所有相关性和。

$$\sum_j \bar{R}_j^{u(n)} + \sum_k \bar{R}_k^{v(n)} = \sum_i R_i^{(n-1)} \quad (2-35)$$

$$0 \leq \sum_j \bar{R}_j^{u(n)}, \sum_k \bar{R}_k^{v(n)} \leq \sum_i R_i^{(n-1)} \quad (2-36)$$

假设  $M$  是一个包含  $B$  个模块的基于 Transformer 的模型, 每个模块包含自注意力层、跳跃连接 (残差连接)、线性层和规范化层。该模型将输入转为  $s$  个 token, 并且每个 token 的尺寸大小是  $d$ , 且包含一个特殊的 token 用来分类, 该 token 记为 [CLS]。模型  $M$  输出一个长度为  $C$  的分类概率向量  $y$ , 使用分类 token 计算。自注意力模块作用于嵌入维数  $d$  的一个子空间  $d_h$  上,  $h$  为“头”的个数, 令  $d_h = d$ 。自注意力模块定义如下:

$$\mathbf{A}^{(b)} = \text{softmax} \left( \frac{\mathbf{Q}^{(b)} \cdot \mathbf{K}^{(b)T}}{\sqrt{d_h}} \right) \quad (2-37)$$

$$\mathbf{O}^{(b)} = \mathbf{A}^{(b)} \cdot \mathbf{V}^{(b)} \quad (2-38)$$

式2-38和2-37中, 运算符  $(\cdot)$  表示矩阵乘法,  $\mathbf{O}^{(b)} \in \mathbb{R}^{h \times s \times d_h}$  是注意力模块  $b$  的输出。 $\mathbf{Q}^{(b)}, \mathbf{K}^{(b)}, \mathbf{V}^{(b)} \in \mathbb{R}^{h \times s \times d_h}$  分别表示 Query, Key, Value 向量。 $\mathbf{A}^{(b)} \in \mathbb{R}^{h \times s \times s}$  是模块  $b$  的注意力图, 其中第  $i$  行表示输入中每个 token 相对于第  $i$  个 token 的相关性系数, 由于式2-37中的归一化运算 (softmax),  $\mathbf{A}^{(b)}$  中每个注意力头的每一行的值的和为 1。

按照相关性传播和梯度传播的过程, 每个注意力图  $\mathbf{V}^{(b)}$  对于目标类别  $c$  都有其梯度矩阵  $\nabla \mathbf{A}^{(b)}$  和相关性矩阵  $R^{(n_b)}$ , 其中  $n_b$  是模块  $b$  中对应于式2-37中 softmax 操作的那一层。

经过上述介绍, Transfomer attribution 的输出  $\mathbf{C} \in \mathbb{R}^{h \times s \times s}$  则由以下式子定义并计算得出:

$$\bar{\mathbf{A}}^{(b)} = I + \mathbb{E}_h (\nabla \mathbf{A}^{(b)} \circ R^{(n_b)})^+ \quad (2-39)$$

$$\mathbf{C} = \bar{\mathbf{A}}^{(1)} \cdot \bar{\mathbf{A}}^{(2)} \cdot \dots \cdot \bar{\mathbf{A}}^{(B)} \quad (2-40)$$

式2-39表示单个模块输出计算。运算符  $\circ$  表示哈达玛积运算,  $\mathbb{E}_h$  表示取所有注意力头上的一个平均值,  $+$  表示只考虑正相关性的值, 即对结果有正向贡献的值, 并且考虑到跳跃连接的影响, 为了防止计算结果出现全 0, 最后加上单位矩阵。

如式2-40所示, Transfomer attribution 的最终解释输出是一个  $s \times s$  的矩阵  $\mathbf{C}$ ,  $s$  表示经过输入到 Transformer Encoder 中的序列长度。 $\mathbf{C}$  中的每行对应于一个 token 对于其他 token 的相关性权重序列。由于是考虑基于 Transformer 的图像分类神经网络, 因此只用考虑封装了分类信息的 [CLS] 的 token 的相关性序列, 即  $\mathbf{C}_{[CLS]} \in \mathbb{R}^s$

对应于  $\mathbf{C}$  中的关于 [CLS]token 的那一行，因此该行中每个分数即是对每个输入 token 对输出分类的相关性分数。

不考虑其他特殊 token，比如 [CLS] 等，在实际的基于 Transformer 的图像分类神经网络 ViT [?] 中，每一个输入 token 表示一个图像块，为了获得最终的显著图，则需要将获得相关性序列  $\mathbf{C}_{[CLS]} \in \mathbb{R}^s$  进行重新调整，调整为尺寸为  $\sqrt{s-1} \times \sqrt{s-1}$  的原始显著图，由于  $s-1$  的数值一般为图像块的个数，因此原始显著图分辨率较低，需要将其通过插值进行上采样为最终显著图。下图2-12展示了 Transformer attribution 计算过程，对应于式2-39和式2-40。

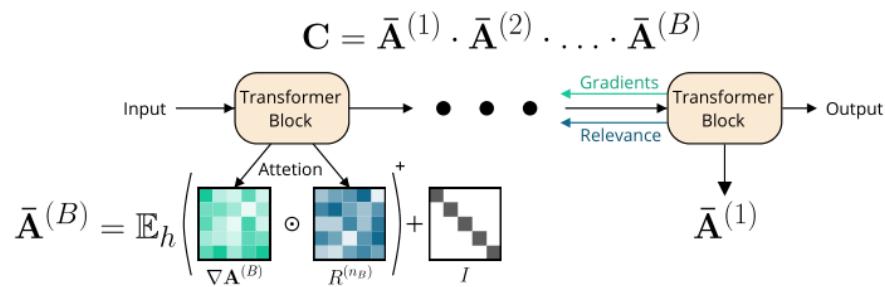


图 2-12 Transformer attribution 计算过程

Fig. 2-12 Transformer attribution calculation procedure

## 2.5 本章小结

本章介绍了……

## 第3章 基于输入图片多尺寸放大的卷积神经网络显著图解释方法

### 3.1 本章引言

### 3.2 问题描述和研究思路

当前大多数基于类激活映射的显著图解释方法诸如 CAM、Grad-CAM、Grad-CAM++、XGrad-CAM、Score-CAM 等都存在一个共有问题，即最终生成的显著图的分辨率较低，这导致其只能在显著图中呈现一个模糊的解释效果，无法聚焦更加精细的特征。究其原因是这些基于类激活映射的方法一般提取的都是卷积神经网络的最后一层卷积层的输出作为特征图的来源，因该层包含较为丰富的类别特征信息，之后通过各种加权方法组合这些特征图来得到最终的显著图。因为卷积神经网络的结构特性，其最后一层卷积层的会输出多个通道的低分辨率特征图，因此无论如何组合这些特征图，最终得到的也只是一张分辨率和最后一层的卷积层输出的单一通道的特征图相当的初级显著图。当然一般情况下为了得到显著图和原始输入图片的特征对应关系，都会将这张原始输入图片进行分辨率层面的放大，使用的一般也是双线性插值算法，但这并不意味着原始显著图的有效信息的增多。

这里以 VGG19 这个基于卷积神经网络的图片分类模型举例，若输入图片的尺寸是  $3 \times 224 \times 224$ ，其中 3 表示 RGB 颜色通道， $224 \times 224$  表示图片分辨率，那么该模型最后一层卷积层的输出的特征图尺寸是  $512 \times 14 \times 14$ ，其中数字 512 是通道数量， $14 \times 14$  是特征图的分辨率。若我们使用双线性插值函数  $\phi(s, H, W)$  将原始显著图分辨率提升到  $224 \times 224$ ，那么意味着我们插入了额外 99.9% 的信息，而最终的显著图中的这些额外的像素信息并不能反映图片分类模型对原始输入图片的对应像素的兴趣程度。即便有部分文献<sup>[9][11]</sup> 试图改进这种插值函数，但是它们仍然引入了外部不相关的信息。

因此为了解决上述的问题，本文从特征图的提取这一关键点着手。考虑到若使用原始输入图片，那么卷积神经网络最后一层卷积层输出的单一特征图分辨率较低且包含的特征信息有限，因此本文提出基于输入图片多尺寸放大的卷积神经网络显著图解释方法，其核心就是将输入图片进行逐级多尺寸的双线性插值放大，例如将输入图片放大至  $334 \times 334$ 、 $434 \times 434$ 、 $534 \times 534$  等分辨率，这时能得到一组分辨率不同的输入图片，接着将这组输入图片分别送入基于卷积神经网络的图像分类模型中，再从其最后一层卷积层中分别提取特征图和梯度矩阵数据，基于卷积神经网络的采样特性，可以得到不同分辨率的特征图，更高分辨率的输入图片即对

应更高分辨率的特征图，也即拥有更丰富的特征信息。因此若将这些特征图的信息进行融合即能得到分辨率更高的掩膜，再用这些更高分辨率的掩膜对原始输入图片进行扰动，即可得到相应的掩膜权重，用权重和高分辨率掩膜进行融合即能得到包含更多特征信息并且分辨率更高的显著图。

### 3.3 基于输入图片多尺寸放大的卷积神经网络显著图解释方法

本章提出基于输入图片多尺寸放大的卷积神经网络显著图解释方法来解决目前可视化算法结果分辨率低、无法给出更详细目标特征的问题。为了方便本章后文的描述，本文对该方法的简称是 MSG-CAM (Multi-Scale inputs of Group-CAM)。MSG-CAM 的算法流程描述参见算法1，算法流程图参见图3-1。

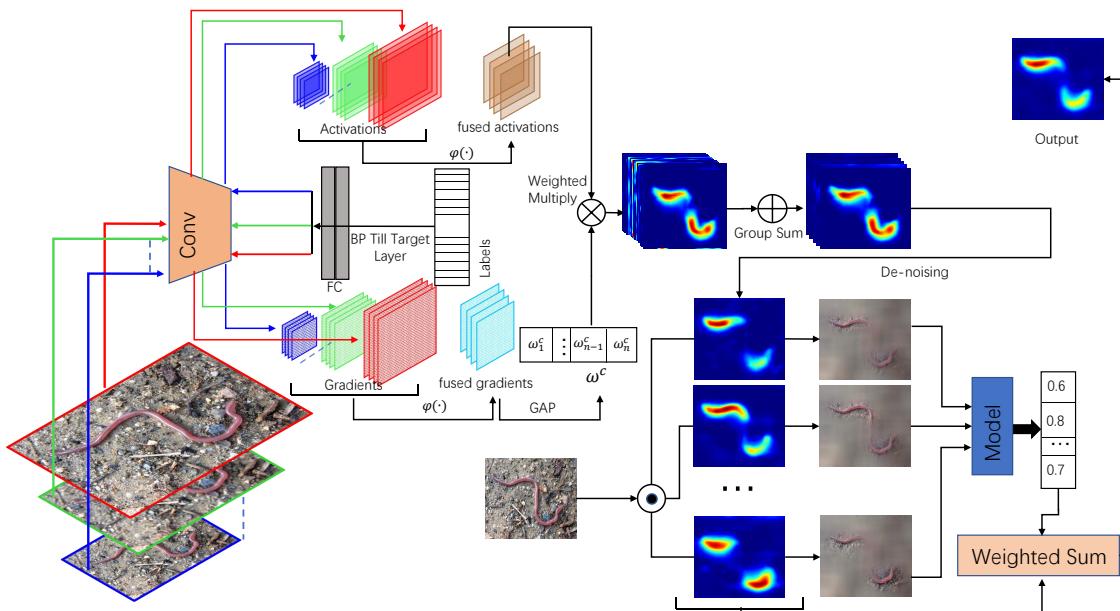


图 3-1 MSG-CAM 算法流程示意图

Fig. 3-1 Pipeline of MSG-CAM

#### 3.3.1 生成高分辨率掩膜

令原始输入图片为  $I_0 \in \mathbb{R}^{3 \times H_0 \times W_0}$ ，其中 3 表示 RGB 颜色通道数量， $H_0$  和  $W_0$  分别原始输入图片的长和宽的像素个数，即原始输入图片的分辨率可以表示为  $\zeta_0 = (H_0, W_0)$ 。此外，我们用  $\mathcal{F}$  表示一个预训练好的基于卷积神经网络的图像分类模型，那么  $\mathcal{F}_c(I_0)$  则表示在输入图片为  $I_0$  的情况下，图像分类模型  $\mathcal{F}$  对于类别  $c$  的输出分数，值得注意的是，该分数是未经归一化指数函数 (softmax 函数) 之前的分数。将原始输入图片送入图像分类模型  $\mathcal{F}$ ，并从其最后一层卷积层  $l$  中提取

**Algorithm 1** MSG-CAM

**输入:** 基于卷积神经网络的图像分类模型  $\mathcal{F}$ , 原始输入图片  $I_0 \in \mathbb{R}^{3 \times H_0 \times W_0}$ , 上采样分辨率上限  $\zeta_{max} = (H_{max}, W_{max})$ , 卷积层  $l$ , 最大迭代次数  $N$ , 类别  $c$ , 每个分组中掩膜数量  $B$ , 高斯模糊参数:  $kernel\_size, sigma, interpolation\ function\ \varphi(\cdot)$

**输出:** 显著图  $L_{MSG-CAM} \in \mathbb{R}^{3 \times H_0 \times W_0}$

```

1: 初始化  $L_{MSG-CAM} \leftarrow 0, \mathcal{A}_0 \leftarrow 0, \mathcal{G}_0 \leftarrow 0, r \leftarrow 0, t_{max} \leftarrow 0, t_m \leftarrow 0, kernel\_size =$ 
    $51, sigma = 50$ , 基准图片  $I'_0 = gaussian\_blur2d(I_0, kernel\_size, sigma)$ ,  $c = \mathcal{F}(I_0)$ 
2: while  $t \leq N$  do
3:    $t \leftarrow t + 1$ 
4:    $\zeta_t \leftarrow \zeta_0 + \lfloor \frac{\zeta_{max}}{N} \rfloor (t - 1)$ 
5:    $I_t \leftarrow \varphi(I_0, \zeta_t)$ 
6:   if  $\mathcal{F}(I_t) \rightarrow c$  then
7:      $t_{max} \leftarrow t_{max} + 1$ 
8:      $\mathbf{A}_t \leftarrow \mathcal{A}(I_t, l)$ 
9:      $\mathbf{G}_t \leftarrow \nabla \mathcal{J}(l, c)$ 
10:     $\mathcal{A}_t \leftarrow \mathcal{A}_{t-1} + \varphi(\mathbf{A}_t, \zeta_0)$ 
11:     $\mathcal{G}_t \leftarrow \mathcal{G}_{t-1} + \varphi(\mathbf{G}_t, \zeta_0)$ 
12:   end if
13: end while
14:  $\bar{\mathbf{A}} \leftarrow \mathcal{A}_t / t_{max}$ 
15:  $\bar{\mathbf{G}} \leftarrow \mathcal{G}_t / t_{max}$ 
16:  $\bar{\mathbf{W}} \leftarrow \frac{1}{m \times n} \sum_{i=1}^m \sum_{j=1}^n \bar{\mathbf{G}}$ 
17:  $\mathbf{M} = \bar{\mathbf{A}} \cdot \bar{\mathbf{W}}$ 
18:  $K \leftarrow \mathbf{M}$  的通道数量;
19:  $g \leftarrow$  每个分组中掩膜数量;
20: while  $r < B$  do
21:   在组内生成单一的掩膜
       $\mathbf{M}_r = ReLU(\sum_{k=r \times g}^{(r+1) \times g - 1} \mathbf{M}^k)$ 
22:   初始化  $\mathbf{M}'_r \leftarrow$  将  $\mathbf{M}^r$  进行去噪操作和归一化
23:   用掩膜扰动原始输入图片
       $I_r = I_0 \odot \mathbf{M}'_r + I'_0 \odot (1 - \mathbf{M}'_r)$ 
24:   计算每张掩膜的权重
       $\alpha_r = \mathcal{F}_c(I_r) - \mathcal{F}_c(I'_0)$ 
25:    $L_{MSG-CAM} \leftarrow L_{MSG-CAM} + \alpha_r \mathbf{M}_r$ 
26:    $r \leftarrow r + 1$ 
27: end while
28: return  $ReLU(L_{MSG-CAM}^c)$ 

```

所有通道的特征图集合，该特征图集合表示为  $\mathbf{A}_0^*$ 。通过获得的类别  $c$  的输出分数  $\mathcal{F}_c(I_0)$ ，对该分数反向传播至最后一层卷积层特征图，则可以获得对应特征图关于类别  $c$  的梯度矩阵  $\mathbf{G}_0^*$ ，计算公式如下：

$$\mathbf{G}_0^* = \frac{\partial \mathcal{F}_c(I_0)}{\partial \mathbf{A}_0^*} \quad (3-1)$$

3-1公式中，梯度矩阵集合  $\mathbf{G}_0^*$  一共有  $K$  个通道，通道数量和特征图集合  $\mathbf{A}_0^*$  一致，并且每个通道的梯度矩阵和特征图一一对应。

接着将原始输入图片  $I_0$  经由双线性插值函数  $\varphi(I_0, \zeta_t)$  上采样至图片  $I_t$ ， $t$  表示第  $t$  次上采样结果，经由上采样后的  $I_t$  的分辨率可表示为  $\zeta_t$ ， $\zeta_t$  是由于原始输入图片的分辨率  $\zeta_0$  逐步递增得来的。为了控制计算的时间成本，考虑将  $\zeta_{max} = (H_{max}, W_{max})$  作为上采样分辨率的上限，同时设  $N$  为最大迭代次数即上采样次数，那么第  $t$  次上采样得到的图片分辨率可由以下公式计算得出：

$$\zeta_t = \zeta_0 + \lfloor \frac{\zeta_{max}}{N} \rfloor (t - 1) \quad (3-2)$$

在上述迭代过程当中，每次上采样获得的输入图片  $I_t$  的分辨率都不同，则将其输入到图像分类模型当中  $\mathcal{F}$  从最后一层卷积层提取的特征图集合  $\mathbf{A}_t^*$  的分辨率也不同，而且随着输入图片分辨率的提高， $\mathbf{A}_t^*$  的分辨率也会相应提高，相对应的  $\mathbf{A}_t^*$  也能包含更多和类别  $c$  相关的细节特征信息。因此若将这些不同的分辨率的特征图集合进行融合则能够得到更多的特征信息。融合的方法可由以下的公式进行表示：

$$\bar{\mathbf{A}} = \frac{1}{t_{max}} \sum_{t=0}^{t_{max}} \varphi(\mathbf{A}_t^*, \zeta_0) \quad (3-3)$$

公式3-3中， $t_{max}$  表示最大的有效迭代次数，此处的“有效”指的是只有当图像分类模型的输出结果中概率分数最大的类别是  $c$  时才采用这次迭代的特征图集合  $\mathbf{A}_t^*$ ，其他情况下均舍弃，因此不难得出  $t_{max} \leq N$ 。同时  $\varphi(\mathbf{A}_t^*, \zeta_0)$  表示将特征图集合  $\mathbf{A}_t^*$  中的每一个通道上的特征图均上采样至原始输入图片的分辨率  $\zeta_0$ ，这样做的目的是方便将不同分辨率的特征图集合进行融合，也方便后续对原始输入图片进行扰动。

在每次迭代过程中，都会计算并提取保存图像分类模型关于类别  $c$  的输出分数  $\mathcal{F}_c(I_t)$  对于卷积层  $l$  的特征图集合  $\mathbf{A}_t^*$  的反向传播梯度  $\mathbf{G}_t^*$ ，和特征图集合一样，将不同迭代过程中保存的梯度矩阵集合  $\mathbf{G}_t^*$  也进行融合，可以得到  $\bar{\mathbf{A}}$  分辨率尺寸一

致梯度矩阵集合，融合的公式如下所示：

$$\bar{\mathbf{G}} = \frac{1}{t_{max}} \sum_{t=0}^{t_{max}} \varphi(\mathbf{G}_t^*, \zeta_0) \quad (3-4)$$

同样，公式3-4中，同时  $\varphi(\mathbf{G}_t^*, \zeta_0)$  表示将梯度矩阵集合  $\mathbf{G}_t^*$  中的每一个通道上的梯度矩阵均上采样至原始输入图片的分辨率  $\zeta_0$ 。由于融合后的特征图集合中不仅包含类别  $c$  特征信息，也包含其他类别的特征信息，因此为了将类别  $c$  的特征信息进行凸显，其他类别的特征信息进行削弱，我们将利用融合后的梯度矩阵  $\bar{\mathbf{G}}$ ，它在一定程度上反映了特征图上不同像素对类别分数  $\mathcal{F}_c(I_0)$  的敏感程度，或者说是重要程度<sup>[11]3</sup>。仿照 Grad-CAM<sup>[11]4</sup> 中的方法，我们将  $\bar{\mathbf{G}}$  中每张梯度矩阵进行全局平均池化，在每个通道上得到一个权重值，每个通道下的权重值都和  $\bar{\mathbf{A}}$  中的每个通道下的特征图一一对应，所有通道下权重值的集合的计算公式如下：

$$\bar{\mathbf{W}} = \frac{1}{H_0 \times W_0} \sum_{i=1}^{H_0} \sum_{j=1}^{W_0} \bar{\mathbf{G}} \quad (3-5)$$

额外说明的是，融合后的特征图集合  $\bar{\mathbf{A}}$  的尺寸是  $[1 \times K \times H_0 \times W_0]$ ，它对应的权重集合  $\bar{\mathbf{W}}$  的尺寸是  $[1 \times K \times 1 \times 1]$ ，其中  $K$  就是第  $l$  层卷积层输出的特征图通道数量。

本小节的最后，初始掩膜集合  $\mathbf{M}$  可以通过以下公式计算得出：

$$\mathbf{M} = \bar{\mathbf{A}} \cdot \bar{\mathbf{W}} \quad (3-6)$$

其中，运算符  $\cdot$  表示  $\bar{\mathbf{A}}$  中的每个通道下的特征图中的每个像素值都和  $\bar{\mathbf{W}}$  中对应每个权值相乘。

### 3.3.2 掩膜优化

$\mathbf{M}$  是经由输入图片多尺寸上采样放大后从卷积层中提取的特征图和梯度矩阵结合而得到的，相比单一原始输入图片得到掩膜（比如 Score-CAM<sup>[28]2</sup>）， $\mathbf{M}$  包含更丰富的类别特征信息，但是目前得到  $\mathbf{M}$  仍然不能直接当作掩膜来扰动输入图片，它还有两个明显的缺点。第一点是  $\mathbf{M}$  的尺寸是  $[1 \times K \times H_0 \times W_0]$ ，其中  $K$  是卷积层  $l$  输出的特征图的通道数量，一般我们取最后一层卷积层作为  $l$ ，因此  $K$  的值将会是数百至上千，显然这样掩膜的数量太多了，若像 Score-CAM<sup>[28]3</sup> 一样逐个将掩膜去扰动原始输入图片得到权值，那样将会极为耗时。第二个缺点  $\mathbf{M}$  就是使用全局平均池化后的梯度作为特征图的权重，由于 ReLU 函数的零梯度问题<sup>[29]</sup>，这意味着  $\mathbf{M}$  显然会有噪声。因此，为了使  $\mathbf{M}$  成为合格的掩膜，需要让它变得更加纯净。

对于第一个缺陷，本节的解决办法是将  $\mathbf{M}$  中的  $K$  张掩膜平分为  $B$  个组，分类

依据是它们的相邻关系，接着将每个组内的所有掩膜相加合并为一个掩膜，这个相加合并过程可由以下的公式计算得出：

$$M_r = \text{ReLU} \left( \sum_{k=r \times g}^{(r+1) \times g - 1} M^k \right) \quad (3-7)$$

公式3-7中  $g$  是每个组中的特征图的数量且  $g = K/B$ ,  $r$  的取值范围是  $\{0, 1, 2, \dots, B-1\}$ ，即合并完成后一共有  $B$  张掩膜， $M_r$  是第  $r$  组中合并完成后的掩膜， $M^k$  是  $\mathbf{M}$  中第  $k$  个通道的特征图。

对于第二个缺陷，我们设计了一个去噪函数  $f(m_{ij}, \theta)$ ，该函数中  $m_{ij}$  是掩膜  $M_r$  中第  $i$  行第  $j$  列的像素值， $\theta$  是一个百分比值。具体的去噪计算结果由以下公式计算得出：

$$f(m_{ij}, \theta) = \begin{cases} m_{ij}, & \text{if } m_{ij} > p(M_r, \theta); \\ 0, & \text{otherwise.} \end{cases} \quad (3-8)$$

公式3-8中  $p(M_r, \theta)$  表示  $M_r$  的所有像素值从大到小处于  $\theta$  百分比的值，比如有 100 个像素值，每个像素值都是 1 到 100 中的一个整数取值且取值唯一，此时  $\theta = 70$ ，那么  $p(M_r, \theta) = 70$ 。3-8公式的作用就是将掩膜  $M_r$  中百分比前  $\theta$  大的像素值保留，小于  $p(M_r, \theta)$  的像素值置为 0。经过去噪操作后可以得到更加纯净的掩膜。

到目前为止，还剩下最后一个操作即可得到最终的掩膜。将  $M_r$  中的所有像素值进行最大最小归一化，使其像素值区间位于  $[0, 1]$ ，这样方便将掩膜直接和输入图片相乘进行扰动操作。具体计算公式如下：

$$M'_r = \frac{M_r - \min(M_r)}{\max(M_r) - \min(M_r)} \quad (3-9)$$

最后，掩膜  $M'_r$  有这和原始输入图片  $I_0$  一致的分辨率且掩膜的像素值和  $I_0$  中的像素值一一对应。

### 3.3.3 生成显著图

如果显著性方法确实识别出了对模型预测有重要意义的像素，那么这一点就应该反映在重建图像的模型输出当中<sup>[30]</sup>。将  $M'_r$  作为掩膜来扰动原始输入图片  $I_0$ ，其背后的原理是从原始输入图像中保留掩膜中获得的关于类别  $c$  的特征信息，让图像分类模型来判断保留的这部分信息的重要性，判断依据就是图像分类模型对扰动后的图片关于类别  $c$  的输出概率分数。但是，如果直接将掩膜和原始输入图像相乘进行扰动，那么扰动后的图像中的被掩盖区域和被凸显的区域之间的边界会过于明显锐利，从而对图像分类神经网络造成对抗效果<sup>[31]</sup>。

为了遇到避免上述的问题，高斯模糊函数被引入到了接下来的改进方法中。具体方法是将扰动区域即被掩膜遮盖的区域用高斯模糊后的原始输入图片的对应区域进行替代，这样可以使得凸显的图片区域和被扰动的图片区域之间的边界更加平滑，从而像真实图片，不易让图像分类神经网络产生异常的输出。对于单一掩膜  $M'_r$ ，这个扰动模式可以由以下的公式计算得出：

$$I_r = I_0 \odot M'_r + I'_0 \odot (1 - M'_r) \quad (3-10)$$

公式3-10中， $I'_0$  是将原始输入图片  $I_0$  高斯模糊后的得到的， $I'_0$  作为一张基础图片送入图像分类模型  $\mathcal{F}$  当中，当然得到其关于类别  $c$  的输出结果  $\mathcal{F}_c(I'_0)$  是一个非常低的值且趋近于 0。具体的高斯模糊函数是 *gaussian\_blur2d(input, kernel\_size, sigma)*，在本章中，参照 Group-CAM<sup>[29]</sup> 的工作，设置高斯模糊的参数  $kernel\_size = 51$  和  $sigma = 50$ 。 $1 - M'_r$  表示  $M'_r$  中每个像素值都作为减数被 1 相减，作用是将掩膜中关于类别  $c$  的特征信息进行凸显。

参考 RISE<sup>[21]</sup> 中的方法，每张掩膜  $M_r$  的权重  $\alpha_r$  可以由扰动后的  $I_r$  输入到图像分类模型  $\mathcal{F}$  中得到。 $\mathcal{F}_c(I_r)$  则是该权重，其表示模型对显著图区域中类别  $c$  的特征信息的感兴趣程度。具体计算公式由以下式子给出：

$$\alpha_r = \mathcal{F}_c(I_r) - \mathcal{F}_c(I'_0) \quad (3-11)$$

最终的显著图即将所有  $M_r$  的权重  $\alpha_r$  和其本身相乘，并经过 ReLU 函数后得到，ReLU 函数的作用是保留显著图中的正值，即模型关于类别  $c$  感兴趣的像素区域。具体式子如下表示：

$$L_{MSG-CAM} = \text{ReLU}(\sum_r \alpha_r M_r) \quad (3-12)$$

## 3.4 实验与分析

### 3.4.1 实验硬件配置和软件环境

下面给出本章实验的硬件配置和实验环境相关信息，具体信息见表??。

### 3.4.2 数据集及其预处理和实验参数说明

在本章的实验当中，主要使用了三个数据集，下面是三个数据集的简要介绍：1. ILSVRC 2012 数据集 ILSVRC 2012 (ImageNet Large Scale Visual Recognition Challenge 2012) 是一个用于视觉对象识别和定位的大规模数据集和竞赛。该数据集包含超过 120 万张标注图片，涵盖 1000 个不同类别的物体和场景。ILSVRC 2012 竞

表 3-1 实验环境和硬件配置

Table 3-1 Experimental environment and hardware configuration

类型	配置信息
硬件	CPU: 10-core Intel® Xeon® W-2255 CPU
	内存: 128GB 64-bit DDR4 3700MHz
	显卡: NVIDIA RTX A5000 24GB
软件	操作系统: Ubuntu 20.04 LTS
	Python 版本: Python3.8
	深度学习框架: Pytorch 1.10.1
	计算架构: CUDA 11.4
	计算加速库: CUDNN 8.2.0
	AI 性能: 27.8 TFLOPS

赛旨在推动计算机视觉领域的发展，参与者需要开发能够识别图像中物体类别的算法，并对物体进行定位。该竞赛对于深度学习和卷积神经网络等技术的发展起到了重要推动作用，成为了评估图像识别算法性能的重要基准。ILSVRC 2012 数据集的发布和竞赛对于推动计算机视觉领域的发展产生了深远影响。

2. PASCAL VOC 数据集 PASCAL VOC (Visual Object Classes) 数据集是一个常用的计算机视觉数据集，用于目标检测、图像分割和场景分类等任务。该数据集最初由牛津大学的计算机视觉研究组创建，包含了 20 个常见的物体类别，如人、狗、猫、飞机等。数据集中的图像来自于自然场景和网络图像，每个图像都标注了包含的物体类别和位置信息。PASCAL VOC 数据集被广泛应用于评估目标检测和图像分割算法的性能，是计算机视觉领域中的重要基准数据集之一。同时，PASCAL VOC 数据集也被用于举办国际性的计算机视觉竞赛，吸引了全球的研究者和工程师参与。通过使用 PASCAL VOC 数据集，研究人员可以开发和评估各种视觉任务的算法，推动计算机视觉技术的发展。

3. COCO2014 数据集 COCO2014 数据集 (Common Objects in Context) 是一个用于计算机视觉任务的大型数据集，包含超过 200,000 张图像和相关的注释信息。这些图像涵盖了 80 个不同类别的物体，并且每张图像都有多个物体的标注，这使得数据集在目标检测、图像分割和物体识别等任务中非常有用。COCO2014 数据集的引入为计算机视觉领域的研究和发展提供了重要的资源，成为了许多视觉任务的标准基准。研究人员和开发者可以利用该数据集进行模型训练、算法评估和性能比较，从而推动计算机视觉技术的进步。COCO2014 数据集的广泛应用促进了

目标检测、图像分割和物体识别等领域的发展，为相关领域的研究和应用提供了有力支持。

在本章当中插入与删除实验中使用的是 ILSVRC 2012 验证集，包含 50000 张图片，并且将该验证集中的图片尺寸调整为  $(3 \times 224 \times 224)$ ，其中 3 是表示 RGB 颜色通道数量，像素值进行归一化调整，调整后的像素值范围是  $[0, 1]$ ，最后使用 Imagenet 数据集的均值  $[0.229, 0.224, 0.225]$  和方差  $[0.485, 0.456, 0.406]$  对所有图片像素值进行标准化处理。在定点游戏实验中，使用了两个数据集，分别是 PASCAL VOC 数据集中的测试集，包含 4952 张图片，和 COCO 2014 数据集的验证集，包含 50000 张图片。用于本章实验的图像分类神经网络模型是 torchvision 提供的预训练模型 VGG19<sup>[32]</sup>，它是基于卷积神经网络架构的。如果没有特别说明，本章提出的方法 MSG-CAM 的默认设置参数是  $B = 32$ ,  $\theta = 70$ 。所有方法生成的最终显著图默认都上采样至  $224 \times 224$  的分辨率。

### 3.4.3 定性评估

### 3.4.4 插入（insertion）和删除（deletion）实验

插入（insertion）和删除（deletion）实验首次在 RISE<sup>[21]</sup> 实验中提出。插入实验背后所反映的原理是当我们按照显著图给出的像素重要程度优先级开始在原图的对应位置逐渐插入重要的像素直至完全插入所有像素，在此过程中记录每次插入操作时模型对指定类别给出的可能性概率分数。插入实验可以衡量显著图对像素重要性的排序是否与模型实际决策时关注的像素重要性排序一致。与之相反的是，删除实验则是按照显著图中给出的像素重要程度优先级逐渐从原输入图片中抹去对应的像素信息。和插入实验一样，删除实验也要求记录每次删除操作后模型对感兴趣类别给出的可能性分数。删除实验可以直观的呈现出缺失重要特征的像素信息后模型对感兴趣类别的置信程度下降情况。

具体而言，对于插入实验，我们将原输入图片高斯模糊化后作为画布，随后每次迭代过程中，我们按照显著图给出的像素优先级逐渐向画布中对应的位置中引入原输入图片的像素，每次迭代过程中记录模型对引入像素后的的图片关于指定类别的可能性概率分数。为了更加精确的反映显著图的像素优先级，我们每次迭代只引入约 0.89%( $224 \times 2$ ) 的像素。和插入实验相对比，删除实验每次迭代将原输入图片中的像素逐渐替代为画布上的像素，直到输入图片被完全替换为画布为止。删除实验每次迭代仍然是变更约 0.89% 的在原图中的像素。需

要特别说明的是，我们引入高斯模糊后的原输入图片作为画布是为了避免引入像素或者删除像素时产生过于锐利的边界，从而更加接近真实图片，避免产生

表3-2 电流类型对效率的影响

Table 3-2 Current type impact on efficiency

AUC	GradCAM	GradCAM++	XGradCAM	ScoreCAM	GroupCAM	CAMERAS	MSGCAM
Insertion	53.19	51.57	52.57	<b>55.10</b>	54.61	44.10	54.52
Deletion	11.52	12.16	11.53	11.43	11.21	<b>8.01</b>	9.78
Over-all	41.67	39.41	41.04	43.67	43.40	36.09	<b>44.74</b>

对抗攻击的样本。此外，每次迭代记录的可能性分数都是经过 softmax 函数进行归一化后的数据。得到插入实验和删除实验每次迭代获得的分数后，我们使用概率分数随插入或者删除次数的曲线的曲线下面积 (Area Under Curve) 作为的作为量化指标。为了总体衡量插入实验和删除实验的优劣，我们使用总体 (over-all) 分数。按照上面关于插入实验和删除实验的描述，AUC(insertion) 越高表明显著图越准确，同理，AUC(deletion) 越低越好。因此，overall 分数计算方式是 AUC(insertion)-AUC(deletion)。图3-2展示了 Grad-CAM、Score-CAM、Group-CAM 和 MSG-CAM 为所选的 4 幅代表性图像生成了显著图以及相应的插入和删除曲线。对于删除曲线，更好的显著图解释方法给出的显著性图应尽可能快地下降，插入曲线与删除曲线正好相反。从图中可以看出本文提出的 MSG-CAM 在视觉解释效果以及插入删除曲线的表现来看均优于其他三种基于类激活映射的方法。

表??列出了在 50000 张图像上进行插入和删除的实验结果。虽然 MSG-CAM 在单项指标上并不突出，但在 AUC(overll) 这项指标上却是第一名，比第二名的 Score-CAM 高出 1.07%。此外，为了探究图像分类神经网络对目标类别的置信程度与三个指标之间的关系。我们根据每幅图像的分类类别的的最大得分将其分为 11 组，并统计每组的 AUC(insertion), AUC(insertion) 和 AUC(overall)，统计数据绘制成了表3-3。从图中可以看出，当图像的分类类别的概率分数较低时，各种显著图解释方法之间很难拉开差距。随着图像的分类类别的概率分数的上升，MSG-CAM 逐渐显示出优势。在图像的分类类别的概率分数  $\geq 99.99$  时，即当图像分类神经模型对这组图像中的分类类别有很高的置信度时，MSG-CAM 在 AUC(overall) 有着明显领先。这表明，当图像分类神经网络接近完美的学习到相关类别信息时，MSG-CAM 能准确反映这种情况。

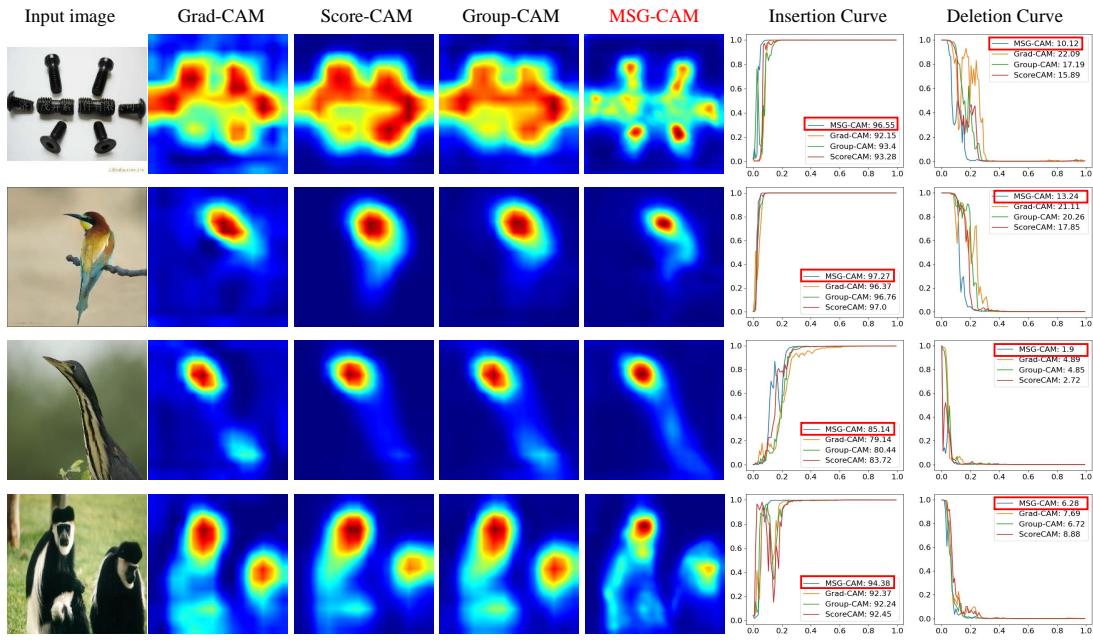


图 3-2 不同方法下的插入删除实验曲线对比图

Fig. 3-2 Comparison of experimental curves of insertion and deletion under different methods

### 3.4.5 定点游戏实验

定点游戏 (pointing game) [33] 实验对于给定显著图解释方法的空间选择性能能够给出比较客观的评估结果。所谓空间选择性就是定点游戏要求图像分类神经网络对给定类别在输入图片中指出其所在位置, 这对显著图解释方法提出了挑战, 方法生成的显著图必须要准确反映给定类别在输入图片的具体位置。定点游戏的具体操作是从针对一个指定类别生成的显著图中找到值最大的那个点并记录它的坐标, 如果该坐标位于该类别对象的预先人工标注的边界框内部, 则记录一次击中 (hit), 否则就是没有击中 (miss)。定点游戏的量化评价指标即是显著图解释方法在给定数据集上对输入图片中所有类别的击中准确率, 具体计算公式如下:

$$Acc = \frac{\#hits}{\#hits + \#misses} \quad (3-13)$$

在式3-13中,  $\#hits$  表示数据集中所有显著图的最大值点落在标注框中的个数,  $\#misses$  则表示数据集中所有显著图的最大值点落在标注框外的个数。在只计算最大值空间选择性的定点游戏当中,  $\#hits + \#misses$  就等于数据集中所有图片标注框的个数。

为了更加精确的反映显著图的空间选择性, 本节实验当中摈弃了只计算最大值点所在坐标的做法, 转而计算显著图中里面从大到小的前 100 个点的坐标, 这可以排除偶发性的噪声影响。最终的性能指标是对一个数据集中每个类别的  $Acc$  取平均。在本次实验中, 我们使用的是 VGG19 模型, 且分别在 PASCAL VOC 测试

图 3-3 不同方法下的插入删除实验曲线对比图

Fig. 3-3 Comparison of experimental curves of insertion and deletion under different methods

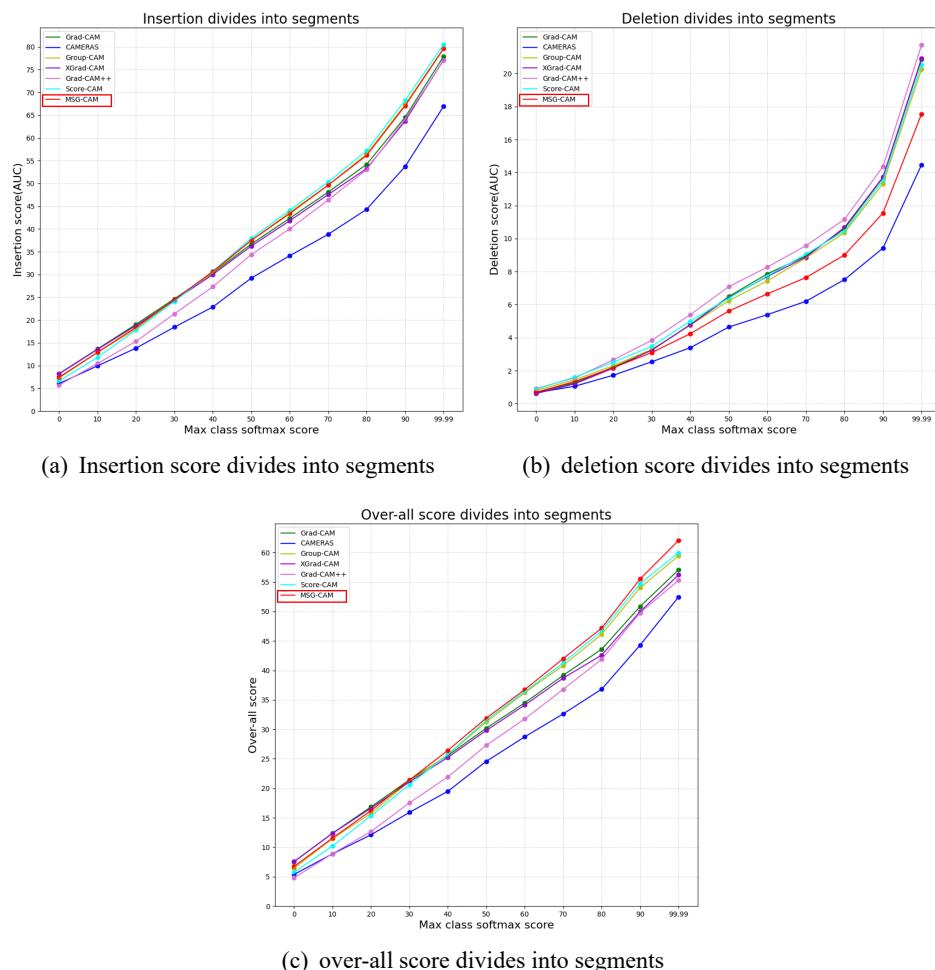


表 3-3 电流类型对效率的影响

Table 3-3 Current type impact on efficiency

Method	PASCAL VOC test	COCO 2014 validation
	Mean Accuracy(%)	Mean Accuracy(%)
Grad-CAM	83.04	55.50
Grad-CAM++	83.21	52.91
XGrad-CAM	86.70	55.93
Score-CAM	73.92	51.20
Group-CAM	82.41	54.14
CAMERAS	87.16	55.40
<b>MSG-CAM</b>	<b>87.24</b>	<b>56.62</b>

集和 COCO 2014 验证集上进行了微调训练。如表3-3所示，我们的方法在两个数据集上都处于领先地位，这表明我们的方法比其他基于类激活映射图的方法更能反映图像分类神经网络的空间选择性。

### 3.4.6 合理性检验

部分基于反向传播的显著图方法对模型参数并不敏感，这意味着不管图像分类模型有没有经过训练，它们都能够给出相似的结果，这显然是违背显著图解释的初衷。所以，能否通过合理性检验是衡量显著图解释方法是否具备可解释性的标准。具体而言，本节 [33] 根据中的实验方法对经过预训练的 VGG19 采用级联随机化 (cascade randomization) 和独立随机化 (independent randomization)。级联随机化是从靠近模型的输出端开始，逐渐将卷积层的参数随机化，直至将所有卷积层的参数随机化。独立随机化也是从靠近模型的输出端开始，每次仅将单独的一层卷积层参数随机化，其他卷积层不变。

图3-4中展示了 MSG-CAM 在单一图片下分别进行级联随机化和独立随机化的结果示意图，图中 Conv34 表示的是将 VGG19 的第 34 层卷积层经过独立随机化和级联随机化后的显著图生成结果，其他的 Conv32 等以此类推。可以看出无论是级

联随机化还是独立随机化，本章提出的 MSG-CAM 生成的显著图均受到了明显影响，表面 MSG-CAM 是受到图像分类神经网络训练参数影响的，能够通过合理性检验。

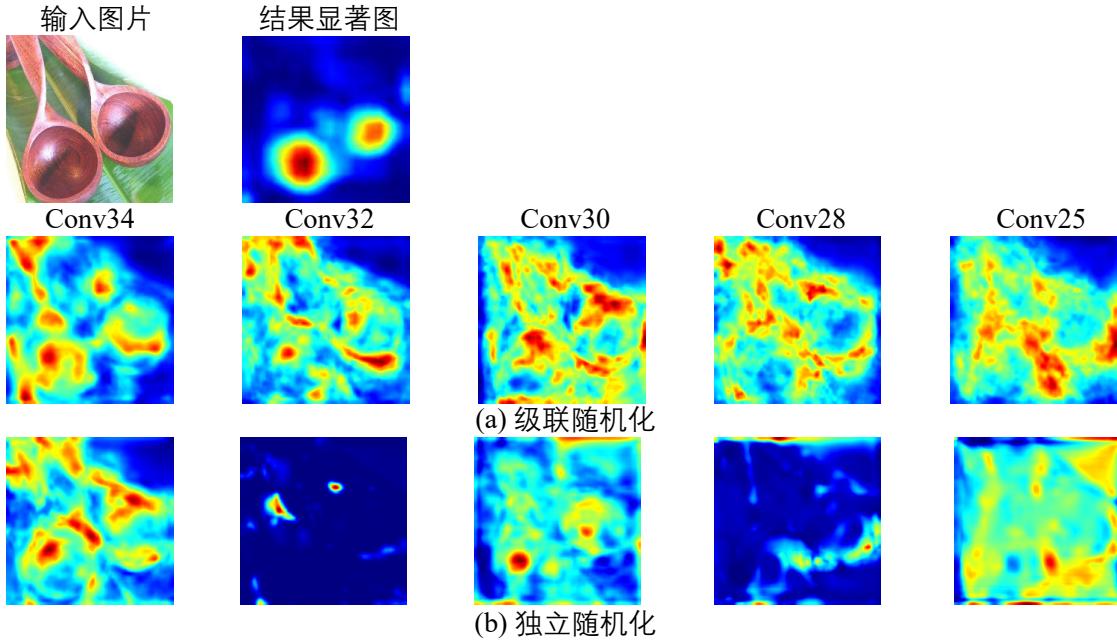


图 3-4 MSG-CAM 合理性检验示意图

Fig. 3-4 Sanity check of MSG-CAM

### 3.5 本章小结

部分基于反向传播的显著图方法对模型参数并不敏感，这意味着不管图像分类模型有没有经过训练，它们都能够给出相似的结果，这显然是违背显著图解释的初衷。所以，能否通过合理性检验是衡量显著图解释方法是否具备可解释性的标准。具体而言，本节 [33] 根据中的实验方法对经过预训练的 VGG19 采用级联随机化 (cascade randomization) 和独立随机化 (independent randomization)。级联随机化是从靠近模型的输出端开始，逐渐将卷积层的参数随机化，直至将所有卷积层的参数随机化。独立随机化也是从靠近模型的输出端开始，每次仅将单独的一层卷积层参数随机化，其他卷积层不变。

图3-4中展示了 MSG-CAM 在单一图片下分别进行级联随机化和独立随机化的结果示意图，图中 Conv34 表示的是将 VGG19 的第 34 层卷积层经过独立随机化和级联随机化后的显著图生成结果，其他的 Conv32 等以此类推。可以看出无论是级联随机化还是独立随机化，本章提出的 MSG-CAM 生成的显著图均受到了明显影响，表面 MSG-CAM 是受到图像分类神经网络训练参数影响的，能够通过合理性检验。

检验。

## 第4章 一种通用的基于二维滑动窗口和放大的图像分类神经网络显著图解释增强方法

### 4.1 本章引言

### 4.2 问题描述和研究思路

显著图分辨率低，包含特征细节信息少仍然是当前绝大多数显著图解释方法的一个弊病。从基于卷积神经网络的图像分类神经网络来看，针对这一网络的显著图解释方法通常是从卷积神经网络的最后一层卷积层提取特征图。这一层包含了丰富的类别特征信息，然后通过加权组合这些特征图来生成最终的显著图。然而，由于卷积神经网络的结构特性，最后一层卷积层会输出多个通道的低分辨率特征图。因此，无论怎样组合这些特征图，最终得到的显著图分辨率仅与最后一层卷积层的单一通道特征图相当。当然，为了建立显著图与原始输入图片的特征对应关系，通常会对原始输入图片进行分辨率放大。这一过程通常采用双线性插值算法，但这并不意味着显著图中有效信息的增加。另外，当前的一些研究者也在积极探索对基于 Transformer 架构的图像分类模型进行显著图解释，Chefer 等人<sup>[26]</sup>在近年发布了针对这一方面的研究，他们针对 Transformer 的结构的制定了新的层间相关性反向传播规则，解决了传播过程中遇到的注意层和跳跃连接时的挑战。他们将每一个 Transformer 块的梯度和其对应的相关性分数结合解决了 Transformer 在图像分类领域的显著图可视化问题。但是即便如此他们针对 Transformer 提出的方法仍然会面临低分辨率的困扰，究其原因是 Transformer 结构的图像分类神经网络反向传播过程中计算并生成显著图时也只能受限于 token 的尺寸，无法生成包含信息量更多的显著图。图4-1描述了这一问题，无论针对卷积神经网络的基于类激活映射图的方法还是针对 Transformer 架构提出的 Transformer attribution 方法（即 Chefer 等人提出的方法）都只能生成分辨率较低的原始显著图，之后通过上采样获得最终的显著图。

因此为了解决上述的问题，本章提出了一种通用的显著图增强方法，可以直接应用在多数可视化算法上。该方法使用固定尺寸的滑动窗口对输入图片中的所有局部区域上采样到输入图片尺寸，然后将结果输入到选定的可视化算法中得到所有图片的针对特定类别的显著图和概率分数，最后将显著图下采样到输入图片对应位置上的窗口中，并乘以概率分数，即可得到具备更多细节的显著图。将该方法应用在不同的可视化算法上，这些算法基于不同架构的网络，无论是量化指标还是直观评测都显示出我们的方法使这些可视化算法得到了明显提升，从而证明本章

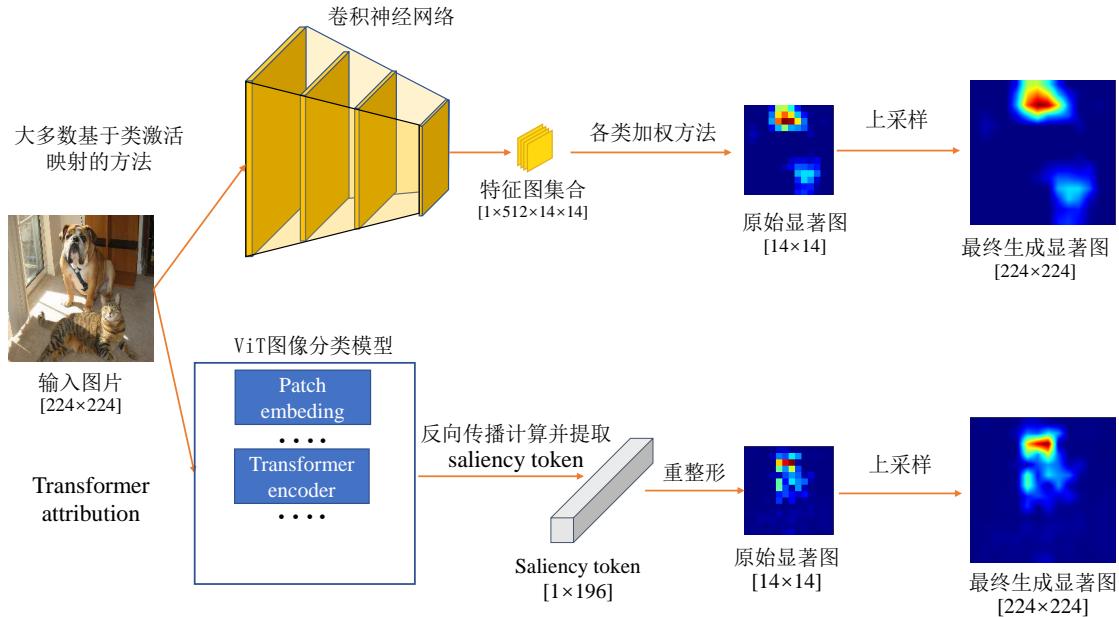


图 4-1 当前显著图解释方法分辨率低的原因

Fig. 4-1 Reasons for the low resolution of current saliency map interpretation methods

提出的方法的有效性和可靠性。

### 4.3 一种通用的基于二维滑动窗口和放大的图像分类神经网络显著图解释增强方法

为了解决当前可视化算法存在的分辨率低，特征模糊，噪声多的问题，我们的方法在输入图片上应用了滑动窗口，通过采集并融合不同窗口图片的可视化信息，让可视化算法加强对输入图片中的局部信息的感知。

#### 4.3.1 获取窗口图片集合

设定原始输入图片  $I_0 \in \mathbb{R}^{3 \times H_0 \times W_0}$ ，使用一个二维滑动窗口函数  $\psi(I, start, h, w, stride)$  来从输入图片  $I_0$  中提取窗口图片，函数  $\psi(I, start, h, w, stride)$  中  $I$  就是函数所应用的输入图片， $start$  表示滑动窗口在二维坐标系统的起点，一般设定在输入图片的左上角位置， $h$  和  $w$  表示窗口像素尺寸且窗口尺寸应该小于输入图片尺寸， $stride$  表示每次滑动窗口移动的像素个数，窗口从左上角起点开始移动，只能在图片区域内移动，即滑动窗口应该向右或者向左移动。每次移动则将窗口内的图片区域进行复制并保存，移动完毕后即可得到关于窗口图片的集合，单张窗口图片的获取的公式如下：

$$p_{k_n} = \psi(I_0, start, h, w, stride) \quad (4-1)$$

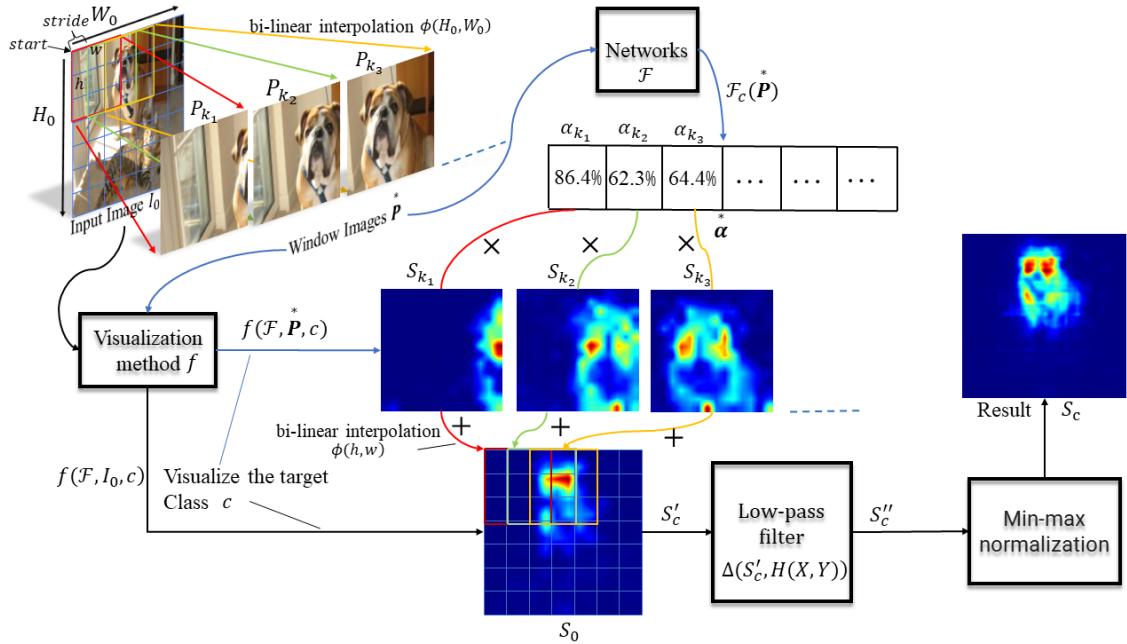


图 4-2 不同方法下的插入删除实验曲线对比图

Fig. 4-2 Comparison of experimental curves of insertion and deletion under different methods

式4-1中  $p_{k_n}$  表示一张窗口图片,  $k_n$  表示这张图片在输入图片  $I_0$  中的二维坐标标记。所有窗口图片的集合可以如下表示:

$$\mathbf{P}^* = \{p_{k_1}, p_{k_2}, \dots, p_{k_n}\} \quad (4-2)$$

式4-2中,  $\mathbf{P}^*$  即表示滑动窗口所有窗口图片集合。获取的窗口图片尺寸是和窗口一致的, 因此是无法直接输入图像分类模型的, 所以需要将其上采样到原始输入图片  $I_0$  的尺寸, 本方法上采样所用的函数是双线性插值函数  $\phi$ , 具体上采样过程描述见如下公式:

$$\mathbf{P}^* = \phi(\mathbf{P}^*, H_0, W_0) \quad (4-3)$$

式4-3中,  $H_0$  和  $W_0$  是原始输入图片  $I_0$  的长和宽的像素个数,  $\mathbf{P}^*$  表示窗口图片上采样后的图片集合, 因此  $\mathbf{P}^* = \{P_{k_1}, P_{k_2}, \dots, P_{k_n}\}$ ,  $P_{k_n}$  即表示窗口图片  $p_{k_n}$  上采样到原始输入图片尺寸后的图片。

### 4.3.2 获取窗口图片的显著图和权重

设定  $\mathcal{F}$  是经过预训练的图像分类神经网络,  $\mathcal{F}_c(I)$  表示当输入图片是  $I$  的情况下, 图像分类神经网络  $\mathcal{F}$  关于类别  $c$  的输出概率分数。对于当前主流的显著图解释算法来说, 将某一显著图解释算法考虑为一个函数  $f$ , 该函数的输入参数是图像分类神经网络  $\mathcal{F}$ , 输入图片  $I$  和指定类别  $c$ 。当输入图片是  $I_0$ , 可以获得图像分类

神经网络  $\mathcal{F}$  在采用显著图解释算法  $f$  时生成的关于类别  $c$  的显著图:

$$S_0 = f(\mathcal{F}, I_0, c) \quad (4-4)$$

式4-4是生成的显著图, 且  $S_0 \in \mathbb{R}^{1 \times H_0 \times W_0}$ 。显著图  $S_0$  的分辨率和原始输入图片是一致的, 因其已经经过上采样处理了。 $S_0$  中的像素和原始输入图片的像素是一一对应的, 且其像素的值表示图像分类神经网络对输入图片  $I_0$  关于类别  $c$  的判断权值。现在已经获得了原始输入图片  $I_0$  的显著图, 在上一小节当中, 获取的窗口图片集合  $\overset{*}{\mathbf{P}}$  也需要经过同样的步骤分别获取每张窗口图片关于类别  $c$  的显著图:

$$\begin{aligned} \overset{*}{\mathbf{S}} &= f(\mathcal{F}, \overset{*}{\mathbf{P}}, c) \\ &= \{S_{k_1}, S_{k_2}, \dots, S_{k_n}\} \end{aligned} \quad (4-5)$$

式4-5中  $\overset{*}{\mathbf{S}}$  表示图像分类神经网络  $\mathcal{F}$  对窗口图片集合生成的关于类别  $c$  的显著图集合,  $\overset{*}{\mathbf{S}}$  中所有显著图的分辨率都和原始输入图片  $I_0$  一致。

在所有基于类激活映射的显著图解释算法中, 都采用了各种形式的特征图加权, 而这一步骤对于衡量每个特征图关于类别  $c$  的贡献至关重要。为了衡量由不同窗口图片生成的显著图的重要性, 可以获得每张窗口图片相对于类别  $c$  的概率分数。这个概率分数反映了图像分类神经网络关于窗口图片中类别  $c$  的特征信息的贡献权重。

$$\begin{aligned} \overset{*}{\boldsymbol{\alpha}} &= \mathcal{F}_c(\overset{*}{\mathbf{P}}) \\ &= \{\alpha_{k_1}, \alpha_{k_2}, \dots, \alpha_{k_n}\} \end{aligned} \quad (4-6)$$

式4-6中  $\overset{*}{\boldsymbol{\alpha}}$  就是  $\overset{*}{\mathbf{P}}$  所有窗口图片的关于类别  $c$  的概率分数, 以此作为  $\overset{*}{\mathbf{S}}$  的权重。

### 4.3.3 融合窗口图片集合的显著图

由于不同窗口图片生成的显著图只包含了原始输入图片部分区域的特征信息, 因此需要将所有窗口图片进行拼接融合从而得到包含丰富特征信息的完整显著图, 这一将窗口图片显著图拼接融合的操作也变相提高了最终呈现特征图对细节特征的解释能力, 能够给出更精确的特征信息。由于式4-7中窗口图片显著图集合  $\overset{*}{\mathbf{S}}$  中的显著图尺寸和原始输入图片  $I_0$  的分辨率是一致的, 因此要将其下采样至窗口图片的尺寸才方便根据其携带的在原始输入图片中的坐标信息进行拼接融合, 具体

的过程用公式表示如下：

$$\begin{aligned} \mathbf{s}^* &= \phi(\mathbf{S}^*, h, w) \\ &= \{s_{k_1}, s_{k_2}, \dots, s_{k_n}\} \end{aligned} \quad (4-7)$$

式4-7中， $\mathbf{s}^*$  是窗口图片集合  $\mathbf{P}^*$  的显著图集合  $\mathbf{S}^*$  下采样至滑动窗口尺寸的显著图集合， $s_{k_n}$  中的  $k_n$  表示该张显著图在原始输入图片中的坐标位置，这个坐标位置是用来方便后续显著图进行拼接融合的。

显著图拼接融合的具体方法是将所有  $\mathbf{s}^*$  中窗口尺寸的显著图乘以该显著图在  $\mathbf{a}^*$  对应的权重，再根据记录的坐标和原始输入图片的显著图  $S_0$  中对应位置区域的像素值相加，窗口显著图拼接融合的具体公式如下：

$$\begin{aligned} S'_c &= \sum_{k_1}^{k_n} (\mathbf{s}^* \mathbf{a}^* + S_0) \\ &= \sum_{k_1}^{k_n} (\alpha_{k_n} s_{k_n} + S_0^{k_n}) \end{aligned} \quad (4-8)$$

在式4-8中， $S_0^{k_n}$  中  $k_n$  表示原始输入图片的显著图  $S_0$  中左上角坐标为  $k_n$  长宽为  $h$  和  $w$  的窗口区域。 $S'_c$  表示拼接融合后的显著图。

#### 4.3.4 平滑和归一化显著图

因为滑动窗口是以一定步长在输入图片上截取窗口图片，这会导致窗口图片的显著图在按照以上方法进行融合的过程中不可避免的会在拼接融合后的显著图上留下网格状的痕迹，使得显著图视觉效果不够平滑。因此为了尽可能降低滑动窗口带来的不利影响，本方法使用了一个理想低通滤波器对其进行优化，将显著图中网格状的痕迹进行减弱或者消除。定义这个理想低通滤波器函数为  $\delta$ ， $S'_c$  经过其处理的过程由以下式子定义：

$$S''_c = \Delta(S'_c, H(X, Y)) \quad (4-9)$$

式4-9中， $H(X, Y)$  为该理想低通滤波器的传递函数，它的具体定义如下：

$$H(X, Y) = \begin{cases} 1 & D(X, Y) \leq D_0 \\ 0 & D(X, Y) > D_0 \end{cases} \quad (4-10)$$

在式4-10中， $D_0$  是截止频率到频率域中心的距离，它决定了低通滤波器的频率响应范围。当  $D(X, Y)$  小于  $D_0$  时， $H(X, Y)$  接近于 1，表示对低频分量的保留；当  $D(X, Y)$  大于等于  $D_0$  时， $H(X, Y)$  接近于 0，表示对高频分量的抑制。因此， $D_0$  决

定了滤波器的频率截止位置, 即在该位置之前的频率成分会被保留, 之后的频率成分会被抑制。 $D(X, Y) = \sqrt{X^2 + Y^2}$  是显著图  $S'_c$  的频率域里的点  $(X, Y)$  到频率域中心的距离, 频率域的中心通常指的是频率域图像的中心点, 也就是频率为零的点。

最后, 经过低通滤波器处理过后的显著图  $S''_c$  再经过最大最小归一化即可得到增强的且拥有更多特征细节的显著图  $S_c$ :

$$S_c = \frac{S''_c - \min(S''_c)}{\max(S''_c) - \min(S''_c)} \quad (4-11)$$

## 4.4 实验与分析

### 4.4.1 实验硬件配置和软件环境

下面给出本章实验的硬件配置和实验环境相关信息, 具体信息见表4-1。

表 4-1 实验环境和硬件配置

Table 4-1 Experimental environment and hardware configuration

CPU	GPU	操作系统	Python 版本	PyTorch 版本
Intel® Xeon® W-2255 CPU @3.70GHz	NVIDIA RTX A5000	Ubuntu20.04	Python3.8	1.10.1+cu113

### 4.4.2 数据集说明和数据集处理

在本章的实验当中, 主要使用了两个数据集, 下面是三个数据集的简要介绍: 1. ILSVRC 2012 数据集 ILSVRC 2012 (ImageNet Large Scale Visual Recognition Challenge 2012) 是一个用于视觉对象识别和定位的大规模数据集和竞赛。该数据集包含超过 120 万张标注图片, 涵盖 1000 个不同类别的物体和场景。ILSVRC 2012 竞赛旨在推动计算机视觉领域的发展, 参与者需要开发能够识别图像中物体类别的算法, 并对物体进行定位。该竞赛对于深度学习和卷积神经网络等技术的发展起到了重要推动作用, 成为了评估图像识别算法性能的重要基准。ILSVRC 2012 数据集的发布和竞赛对于推动计算机视觉领域的发展产生了深远影响。

2. ImageNet-Segmentation 数据集 ImageNet-Segmentation 数据集是由论文<sup>[34]</sup>发布的, 该论文提出了一种自动填充该数据集的方法, 通过像素级的对象-背景分割来丰富图像的注释信息。研究者们通过逐步利用已经分割的图像来引导新图像的分割过程, 同时结合类别级别的分割, 以实现高质量的分割结果。他们还通过实验证明了该方法在 iCoseg 数据集上取得了最先进的结果, 并且成功地将该方法应用于 ImageNet 数据集, 共包含约 50 万张图像。最终, 他们通过人工标注的方

式对 445 个类别的 4276 张图像进行了分割，并将这些分割结果整理为 ImageNet-Segmentation 数据集并发布。

在本章扰动实验中使用的是 ILSVRC 2012 验证集，包含 50000 张图片，并且将该验证集中的图片尺寸调整为  $(3 \times 224 \times 224)$ ，其中 3 是表示 RGB 颜色通道数量，像素值进行归一化调整，调整后的像素值范围是  $[0, 1]$ ，最后使用 Imagenet 数据集的均值  $[0.229, 0.224, 0.225]$  和方差  $[0.485, 0.456, 0.406]$  对所有图片像素值进行标准化处理。在图像分割实验当中使用了 ImageNet-Segmentation 数据集，包含 4276 张图片，总共 445 个类别。

#### 4.4.3 作为对照的显著图方法

为了验证本章提出的增强算法，一共选用了 5 种显著图解释方法来作为对照，本章提出的增强算法将应用在这 5 种显著图解释方法上来验证增强算法的有效性和可靠性。下面是这 5 种显著图解释方法的简短介绍和在实验当中的应用说明。

1. Grad-CAM Grad-CAM (Gradient-weighted Class Activation Mapping) 是一种用于解释深度学习模型预测的技术，它可以生成显著图帮助理解模型是如何做出特定的预测的。Grad-CAM 是由 Selvaraju 等人在 2017 年提出的，它结合了梯度和全局平均池化的方法，能够生成图像中与模型预测结果相关的区域。在深度学习模型中，我们往往无法直观地理解模型是如何做出预测的，特别是在图像分类任务中。Grad-CAM 技术可以帮助我们理解模型是如何关注图像中的哪些区域来做出预测的。它可以生成热力图，显示出模型对于不同区域的关注程度，从而帮助我们理解模型的决策过程。

Grad-CAM 技术的核心思想是结合梯度和全局平均池化来生成热力图。在传统的深度学习模型中，我们可以通过计算输入图像对于输出类别的梯度来理解模型的关注点。然而，这种方法只能提供整体的梯度信息，无法给出具体的区域信息。而全局平均池化可以将特征图的每个通道的特征值求平均，得到一个全局的特征向量。Grad-CAM 将这两种方法结合起来，通过对特征图的每个通道的梯度和全局平均池化的特征向量进行加权求和，生成与模型预测结果相关的区域。

具体来说，Grad-CAM 技术的公式如下：

$$L_{Grad-CAM}^c = ReLU \left( \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}} A_{ij} \right) \quad (4-12)$$

其中， $L_{Grad-CAM}^c$  表示生成的显著图， $c$  表示输出类别的索引， $y^c$  表示模型对于类别  $c$  的得分， $A_{ij}$  表示特征图的第  $i$  行第  $j$  列的特征值， $\frac{\partial y^c}{\partial A_{ij}}$  表示模型对于类别  $c$  的得分对于特征图的第  $i$  行第  $j$  列的梯度。

在本章的实验中，当 Grad-CAM 是应用在基于卷积神经网络的 VGG19 模型上时，生成特征图的卷积层是其最后一层卷积层，当 Grad-CAM 应用于基于 Transformer 结构的图像分类神经网络 ViT 时，是将最后一层注意力层的【CLS】标记视作指定的特征图，在反向传播过程中也是考虑该【CLS】标记维度上的梯度。

2. LRP
3. Partial LRP
4. Transformer attribution
5. Grad-CAM++
5. Score-CAM

#### 4.4.4 增强方法相关参数

在本章实验当中，如无特别说明，所有输入图片的尺寸都是  $3 \times 224 \times 224$ ，滑动窗口相关参数是： $start = (0, 0), h = 96, w = 96, stride = 32$ ， $start$  是输入图片的左上角的像素作为起点，也是整个二维坐标系统的起点。低通滤波器的参数设置： $D_0 = 35$ 。

#### 4.4.5 定性评估

#### 4.4.6 扰动实验

优秀的显著图解释算法要能准确找到神经网络做出决策的关键特征，并且在生成的显著图中对每个像素赋予它与其对决策贡献相匹配的数值。扰动实验就是在这一标准上比较不同显著图解释算法的性能。我们分别进行正负扰动实验，正负实验测试采用两阶段设置。首先，使用预先训练好的网络提取 ImageNet 验证集的可视化图像。其次，逐渐屏蔽掉输入图像的像素，并测量网络的平均 top-1 准确率。在正扰动实验中，像素从相关度最高的像素被屏蔽到最低的像素，而在负扰动中，像素从最低的像素被屏蔽到最高的像素。在正向扰动中，我们会看到性能急剧下降，这表明被遮挡的像素对分类得分很重要。在负扰动实验情况下，一个好的解释可以保持模型的准确性，同时去除与类别无关的像素。在这两种情况下，我们都测量了去除 10%-90% 像素时的曲线下面积 (AUC)。这两项实验都可应用于图像分类模型预测结果类 (Predicted) 和真实标注类 (Target)。在后一种情况下，特定类别的方法有望获得更好的性能，而与类别无关的方法则会在两种测试中表现出相似的性能。

正负扰动实验和插入删除实验本质是一致的，都是根据显著图给出的像素权值将输入图片中的像素删除来记录得到相关类别索引的概率分数的变化。在正负扰动实验当中，也可以用  $AUC()$  来综合评价扰动实验结果，它的计算方式是  $AUC() =$

表 4-2 电流类型对效率的影响

Table 4-2 Current type impact on efficiency

模型	显著图解释方法	增强算法	Predicted			Target		
			负扰动	正扰动	总体	负扰动	正扰动	总体
GradCAM		✓	41.52	34.06	7.46	42.02	33.56	8.46
			<b>47.54</b>	<b>26.99</b>	<b>20.55</b>	<b>48.46</b>	<b>26.49</b>	<b>21.97</b>
ViT	LRP	✓	43.49	41.94	1.55	43.49	41.94	1.56
			<b>62.69</b>	<b>27.51</b>	<b>35.18</b>	<b>64.76</b>	<b>26.45</b>	<b>38.31</b>
partial LRP		✓	50.49	19.64	30.85	50.49	19.64	30.85
			<b>55.57</b>	<b>18.45</b>	<b>37.12</b>	<b>56.13</b>	<b>18.14</b>	<b>37.99</b>
Transformer attribution		✓	54.14	17.03	37.11	55.04	<b>16.04</b>	39.00
			<b>57.57</b>	<b>16.93</b>	<b>40.64</b>	<b>58.83</b>	16.25	<b>42.58</b>
VGG19	GradCAM	✓	38.08	12.15	25.93	39.04	11.71	27.33
			<b>39.07</b>	<b>10.20</b>	<b>28.87</b>	<b>40.09</b>	<b>9.78</b>	<b>30.31</b>
	GradCAM++	✓	40.50	11.79	28.71	40.81	11.61	29.20
			<b>40.95</b>	<b>10.09</b>	<b>30.86</b>	<b>41.49</b>	<b>9.81</b>	<b>31.68</b>

$AUC() - AUC()$ 。从表 1 中可以看出，我们的增强算法在 overall 指标上都取得明显提升，其中提升最为显著的是 ViT 模型的可视化算法 GradCAM 和 LRP，这是因为他们对 ViT 的可视化能力很差，无法定位关键特征，经过我们的方法增强后，初步具有定位关键特征的能力。这表明我们的方法可以帮助这些可视化算法找到神经网络感兴趣的关键特征，这在不同的模型上都是成立的。

从表4-3中可以看出，我们的增强方法显著提高了所有”总体”得分。应用于 ViT 的 Grad-CAM 算法和 LRP 算法的改进最为明显，这两种算法以前缺乏准确可视化和定位重要特征的能力。使用增强方法后，这些方法能够识别重要特征。图4-3就是一个例子，它表明无论底层网络结构如何，我们的方法都能帮助显著性方法识别神经网络感兴趣的重要特征。

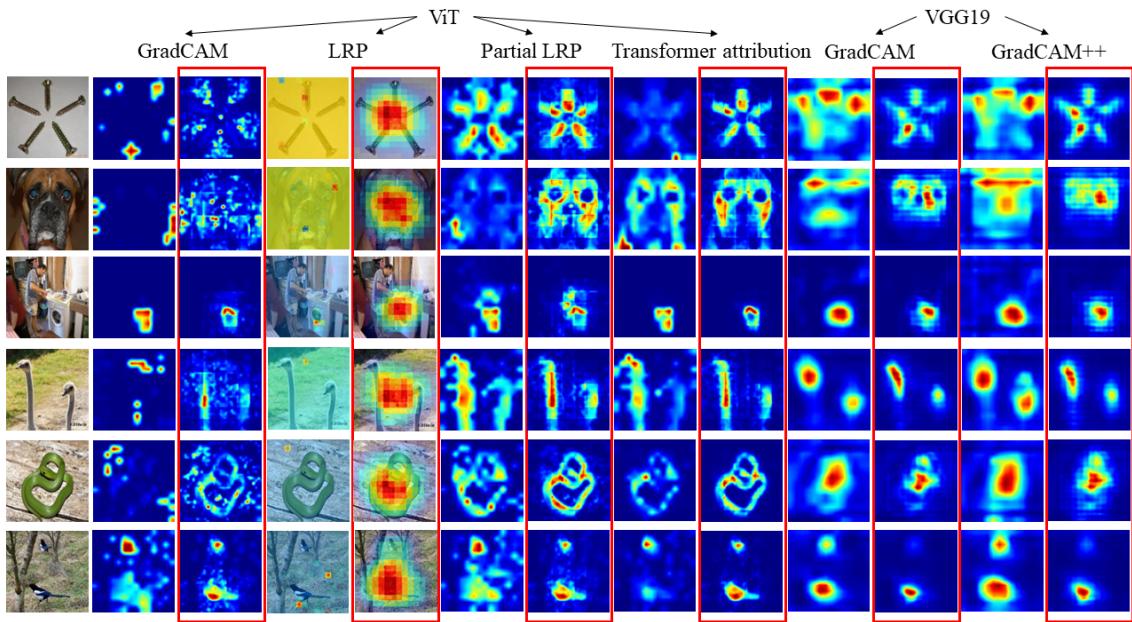


图 4-3 不同方法下的插入删除实验曲线对比图

Fig. 4-3 Comparison of experimental curves of insertion and deletion under different methods

#### 4.4.7 图像分割实验

图像分割实验将每张显著图视为图像的软分割，并将其与 ImageNet-Segmentation 数据集的真实标注的分割进行比较。在分割实验中，计算每个突出图中像素的平均值，并将突出图中高于平均值的像素设为 1，其余像素设为 0。本节使用语义分割中常用的三个指标来衡量性能：像素精确度（Pixel Accuracy）、平均交叉重叠率（mIoU）和平均精确度（mAP），其中 mIoU 即对每张显著图进行平均值阈值化处理后获得的准确度，mAP 是使用不含阈值的显著图计算得出的。

分割结果如表4-4所示。从中可以看出，我们的增强方法在所列的像素精确度（Pixel Accuracy）、平均交叉重叠率（mIoU）和平均精确度（mAP）关键指标，上都有显著的改进，甚至 Transformer attribution [26] 也能获得不可忽略的改进，这是目前在基于 Transformer 架构上的图像分类神经网络 ViT 模型上表现最好的方法。这表明，本章的增强方法可以帮助提高当前显著图解释方法在分割领域的性能。

#### 4.5 本章小结

本章介绍了……

表 4-4 不同的显著图解释方法应用增强算法前后图像分割的表现

Table 4-4 Performance of saliency map enhancement algorithms for image segmentation

模型	显著图解释方法	增强算法	Pixel Acc(%)	mAP(%)	mIoU(%)
GradCAM			64.44	71.60	40.82
		✓	<b>70.33</b>	<b>77.02</b>	<b>47.81</b>
LRP			51.09	55.68	32.89
		✓	<b>69.34</b>	<b>80.88</b>	<b>50.37</b>
ViT			76.31	84.67	57.94
	partial LRP		✓	<b>80.42</b>	<b>85.83</b>
Transformer attribution			79.74	86.03	62.01
		✓	<b>81.90</b>	<b>86.56</b>	<b>64.56</b>
GradCAM			69.03	76.76	48.99
		✓	<b>73.78</b>	<b>79.99</b>	<b>53.82</b>
VGG19			76.77	<b>85.48</b>	58.89
	GradCAM++		✓	<b>78.60</b>	85.42

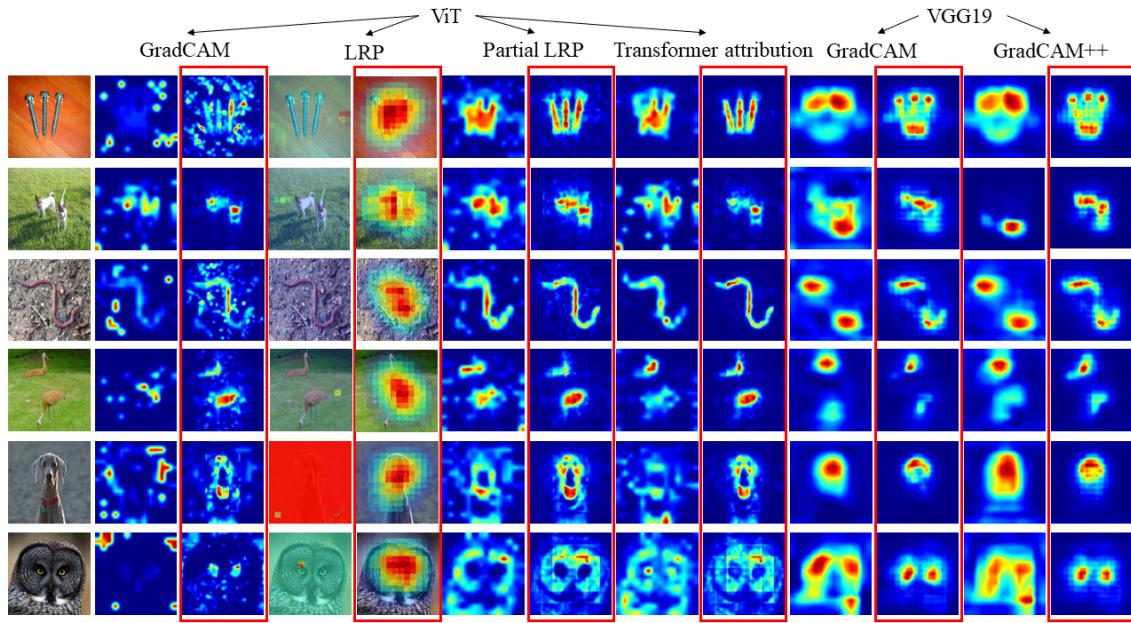


图 4-4 不同方法下的插入删除实验曲线对比图

Fig. 4-4 Comparison of experimental curves of insertion and deletion under different methods

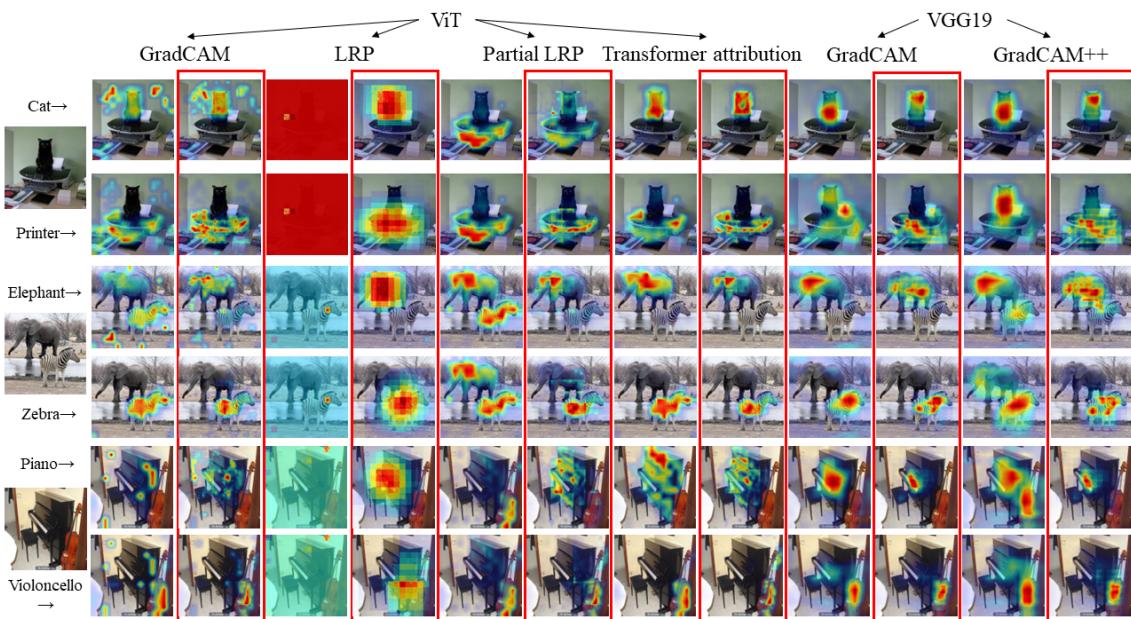


图 4-5 不同方法下的插入删除实验曲线对比图

Fig. 4-5 Comparison of experimental curves of insertion and deletion under different methods

## 第5章 显著图解释方法对比评测系统

当前存在众多的显著图解释方法，但是许多方法由于代码编写复杂，或者方法的作者提供的接口不统一，难以直观的对比不同的显著图解释方法在相同条件下的性能。因此本章将许多已有的显著图解释方法统一程序接口，并结合软件可视化开发技术开发了一套显著图解释方法对比评测系统，该系统可以方便研究人员对当前显著图解释方法进行对比评测，直观展现不同显著图解释方法在相同条件下的视觉差别和评测指标差异。

### 5.1 系统需求分析

本系统的功能目标主要是方便对比不同的显著图解释方法。通过导入模型设置显著图解释方法的参数可以针对性地适应不同显著图解释方法在不同模型下的表现要求；通过选择特定的图片生成显著图可以直观对比不同显著图解释方法的显著图；通过利用生成的显著图可以对原始输入图片进行图像分割直观对比不同显著图解释方法对目标物体的特征捕获能力；通过选定实验类型和数据集可以比较不同显著图解释方法在相同评价指标的性能表现；可以分别导出备份显著图解释方法的性能评测指标相关数据。

本系统主要包含以下几个功能模块，每个功能模块的需求信息如下：

1. 用户登录注册模块：用户可以注册账号，之后可以利用账号密码登录系统。
2. 信息设置模块：在模型设置当中用户可以将本地的模型文件上传导入到系统当中，并可以修改模型名称；在显著图解释方法设置当中用户可以根据不同的模型对应的显著图解释方法设置相关必须参数，例如选择卷积层等；在账号设置当中用户可以查看密码更改密码。
3. 显著图生成模块：在该模块中，用户可以从下拉框中选定要使用的图像分类模型，之后可以从本地选择要生成显著图的输入图片，点击生成预测类别的按钮即可得到当前模型对该图片的前几名类别和对应的概率，用户可以从下拉框中选择要生成显著图的类别，之后从下拉框中选择要使用的显著图解释方法，此处选择的显著图解释方法会自动应用显著图解释方法设置中的对应的参数，最后点击生成显著图即可在页面中生成当前显著图解释方法在选定类别索引下的显著图。
4. 显著图切割模块：在这个模块中，用户可以选择图像分类模型，并从本地选择输入图片，然后点击生成预测类别的按钮，得到当前模型对该图片的前几个类别和对应的概率。用户可以选择类别并生成对应的显著图，然后选择要使用的显著图解释方法。所选的显著图解释方法将自动应用设置中的相应参数。页面会展示

表 5-1 软件开发环境和硬件配置

Table 5-1 Software development environment and hardware configuration

配置	详情
	CPU: 10-core Intel® Xeon® W-2255 CPU
	内存: 128GB 64-bit DDR4 3700MHz
	显卡: NVIDIA RTX A5000 24GB
	操作系统: Ubuntu 20.04 LTS
服务端配置	Python 版本: Python3.8 深度学习框架: Pytorch 1.10.1 计算架构: CUDA 11.4 计算加速库: CUDNN 8.2.0 AI 性能: 27.8 TFLOPS
开发端配置	CPU: AMD Ryzen 5 4600U 内存: 16GB 64-bit LPDDR4 3200MHz 操作系统: Windows11 家庭中文版 Python 版本: Python3.8 PyQt 版本: PyQt5 数据库: MySQL 5.5 集成开发环境: PyCharm 社区版 2021.2.3

已经生成的显著图，此时用户可以选择将显著图中对应原始输入图片的重要区域单独分割出来，分割的比例可以从下拉框中选择。用户还可以选择不同的显著图解释方法直观对比不同显著图解释方法的分割效果。

5. 实验测试模块：在该模块中用户可以选择数据库中既有的图像分类模型其中也包括用户自行上传的模型，之后选择要进行的实验类型，其中包括几种主流的实验有插入删除实验，分割测试实验等，最后选择显著图解释方法即可进行实验，实验结束后会将实验数据结果显示给用户，并存入用户对应的数据库中。

6. 备份保存模块：用户可以把在实验测试模块中的数据进行备份保存，保存方式是纯文本形式，包括实验时间，实验结果，实验相关参数等。

## 5.2 开发环境及相关技术介绍

本章的系统由于采用客户端/服务器架构，所以软件硬件环境也分为服务端和客户端。在服务端主要是执行计算量较大的关于深度神经网络的计算，客户端主要是负责页面展示和功能选择。本系统的开发端即可看作客户端，开发的客户端即在开发端运行。具体配置要求如表5-1所示。

本章的系统使用了 PyQt 作为前端图形开发框架，在此对其进行简要的介绍。PyQt 是一个用于创建图形用户界面（GUI）应用程序的开发框架，它基于 Qt 库和 Python 语言。PyQt 提供了丰富的工具和组件，使开发者能够快速、灵活地构建跨平台的应用程序。它包括了用于创建窗口、按钮、菜单、对话框等各种 GUI 元素的类和方法，同时也支持事件处理、信号与槽机制，以及与数据库、网络等其他模块的集成。PyQt 还提供了对 Qt Designer 的支持，可以通过可视化界面设计工具来快速布局和设计界面。由于 PyQt 基于 Qt 库，因此能够充分利用 Qt 的强大功能和跨平台特性，使开发者能够轻松地将应用程序部署到不同的操作系统上。总之，PyQt 是一个功能强大、灵活且易于使用的图形开发框架，适用于各种规模和类型的应用程序开发。

## 5.3 系统设计

### 5.3.1 系统架构设计

本系统采用经典的 C/S 架构设计，即客户端/服务器架构。由于显著图解释方法所需的图像分类模型较多较杂，需要依赖的外围软件较多，计算量较大，因此直接在用户的本地客户端运行显著图解释方法实现难度较大。所以该系统架构将主要的逻辑功能实现放在后台服务器端，用户在客户端操作将相关操作指令通过网络上传后台服务器，由服务器处理完成后传输数据给客户端展示。客户端/服务器架构模式有利于后期维护和功能升级。

本系统的整体系统架构可以从下到上可以分为四个层次，分别是：基础平台、软件平台、功能逻辑和前端页面。每个层次包含的模块内容如图5-1所示：

### 5.3.2 模块设计

显著图解释方法对比评测系统的主要功能可以分为用户功能模块和主要系统功能模块。用户功能模块包括用户登录注册、用户修改账号密码、模型查看和导入、显著图解释方法参数设置、用户评测数据备份。主要系统功能模块包括显著图方法生成对比、显著图方法分割对比、显著图方法实验评测。

显著图解释方法对比评测系统的模块设计图如图5-2所示，用户需要注册登录

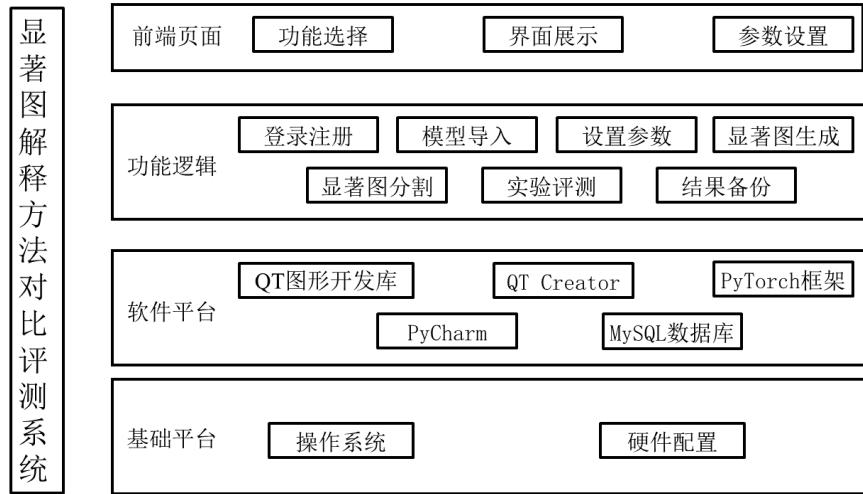


图 5-1 系统架构设计图

Fig. 5-1 System Architecture Design Diagram

才能使用系统主要功能，用户的相关个人信息保存在服务端的数据库中，服务端和客户端通过 http 协议进行通信。

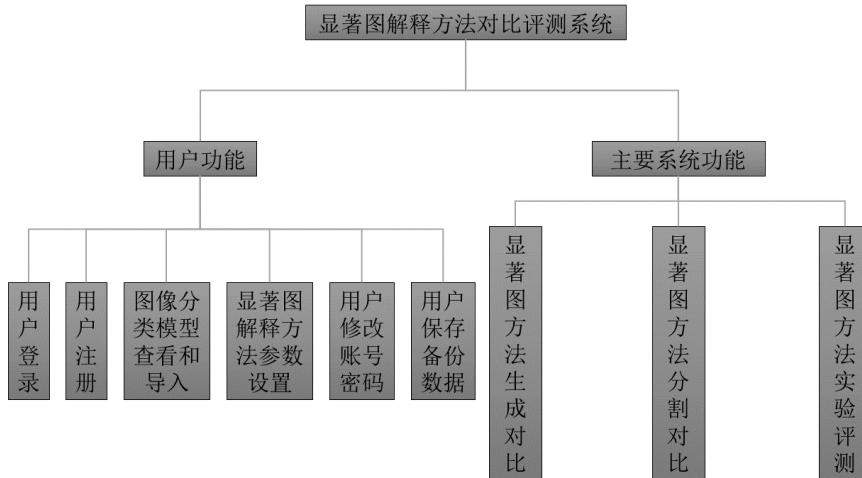


图 5-2 系统模块设计图

Fig. 5-2 System Module Design Diagram

### 5.3.3 功能设计

本系统包含众多的分支功能，下面是对每个功能的详细描述：

#### 1. 用户注册登录功能

为了记录每个用户的实验数据和显著图解释方法的配置信息，每个用户都需要注册并登录，注册后每个用户有唯一的用户账号用来在服务端数据库中保存相

关实验数据。注册用户需要自定义唯一的用户名和设置登录密码，注册成功后系统会生成唯一的数字账号。后续登录功能时用户可以用用户名和数字账号进行登录。

## 2. 图像分类模型的查看和导入

在用户登录后，用户需要在信息设置中查看当前服务端可用的图像分类神经网络模型，用户也可自行上传合适的图像分类神经网络模型到服务端。目前可用的图像分类模型包括：VGG16、VGG19、ResNet18、ViT。

## 3. 显著图解释方法的参数设置

在显著图解释方法的参数设置当中，用户可以看到当前可用的显著图解释方法，对于每一种显著图解释方法，用户可以自行设置参数，可以针对每一种图像分类神经网络的参数设置都会被保存到服务端，后续显著图生成选择显著图解释方法时会自动应用用户的参数设置信息，并且会根据用户选择的图像分类模型进行参数调整。目前可用的显著图解释方法包括：Grad-CAM、Grad-CAM++、Score-CAM、XGrad-CAM、RISE、Transformer Attribution、MSG-CAM。除此之外还包括本文第四章提出的增强方法，该显著图增强方法可以直接应用于上述显著图解释方法。

## 4. 显著图生成功能

显著图生成功能主要是方便用户在单一图片情况下直接对比不同显著图解释方法的生成效果。用户首先需要选定图像分类模型，再选择输入图片，输入图片可以从用户本地上传，选定输入图片后页面会显示用户的输入图片。然后用户需要点击按钮将该图片输入到选定的图像分类模型当中获取该图片的前5名的类别和对应概率。系统会将获取的类别和对应概率通过下拉框展示出来，用户需要选择生成显著图的类别。最后用户选择要应用的显著图解释方法，然后点击生成即可获取服务端返回的显著图。通过选择不同的显著图解释方法即可在页面上展示不同显著图解释方法的显著图，用户可以直观进行对比。

## 5. 显著图分割功能

显著图分割功能主要是利用显著图给出的像素优先级将原始输入图片进行分割，用户首先需要选择图像分类模型，然后选择输入图片。输入图片可以来自用户的本地上传。选定输入图片后，页面会显示用户的输入图片。接下来，用户需要点击按钮，将该图片输入到选定的图像分类模型中，以获取该图片的前5个类别及其对应概率。系统会通过下拉框展示获取的类别和对应概率，用户需要从中选择生成显著图的类别。用户需要选择要应用的显著图解释方法获取服务端返回的显著图，最后用户选择设置需要切割像素比例，此处默认将权重值排名靠前的像素进行保留，其余像素置为0。通过显著图分割可以直观对比不同显著图解释方法对物体特征的感知能力。

## 6. 实验测试功能

当前存在的多种实验来评测显著图解释方法，这些实验代码复杂，对于不同的显著图解释方法需要单独配置，因此该功能旨在让用户轻松进行实验。在该功能模块中用户需要选择图像分类模型，选择评测数据集，之后选择要评测的显著图解释方法，最后选择要进行的实验类型。全部选择完成后点击开始实验，客户端会将相关实验参数提交给后台服务器，后台服务器会根据实验参数自动执行脚本程序，并返回实验大致所需时间。实验完成后数据会存入用户数据库，用户之后可以查看实验结果。

### 7. 备份导出功能

该功能可以将用户的实验数据和相关实验参数设置保存备份。服务器将要保存的数据按规则写入纯文本文件当中，包括实验时间，实验结果，实验相关参数等，写入完成后服务端将保存备份文件传给客户端本地。

## 5.4 系统展示

根据本章前文的系统功能分析，本节将以图片的方式对系统主要的功能进行进一步的展示和说明。

### 1. 图像分类模型的查看和导入

图像分类模型的查看和导入界面如图5-3所示，用户可以点击图中的导入模型按钮导入图像分类模型，此外用户还可以修改模型名称，通过输入模型编号删除个人的图像分类模型。

### 2. 显著图解释方法的参数设置

在图5-4中展示了本文第3章提出显著图解释方法MSG-CAM的参数设置页面，用户通过选定图像分类模型，可以对该模型进行针对性的参数设置，包括选择要使用的卷积层等，每个显著图解释方法都有不同的设置页面。

### 3. 用户密码修改和查看功能

在图5-5中展示了当前用户的数字账号和用户名等信息，用户密码默认以密文方式展示，用户需要点击查看密码按钮进行查看，同时在该页面上用户也可以修改密码。

### 4. 显著图生成功能

在页面截图5-6中，展示了不同显著图解释方法生成的显著图对比情况，可以看到在选择的输入图片当中，生成了前5名预测类别和相应的概率，用户还可以选择是否应用增强算法来增强该显著图解释方法生成的显著图。

### 5. 显著图分割功能

在图5-7中展示了根据显著图进行分割情况，在该页面中用户可以设置分割使

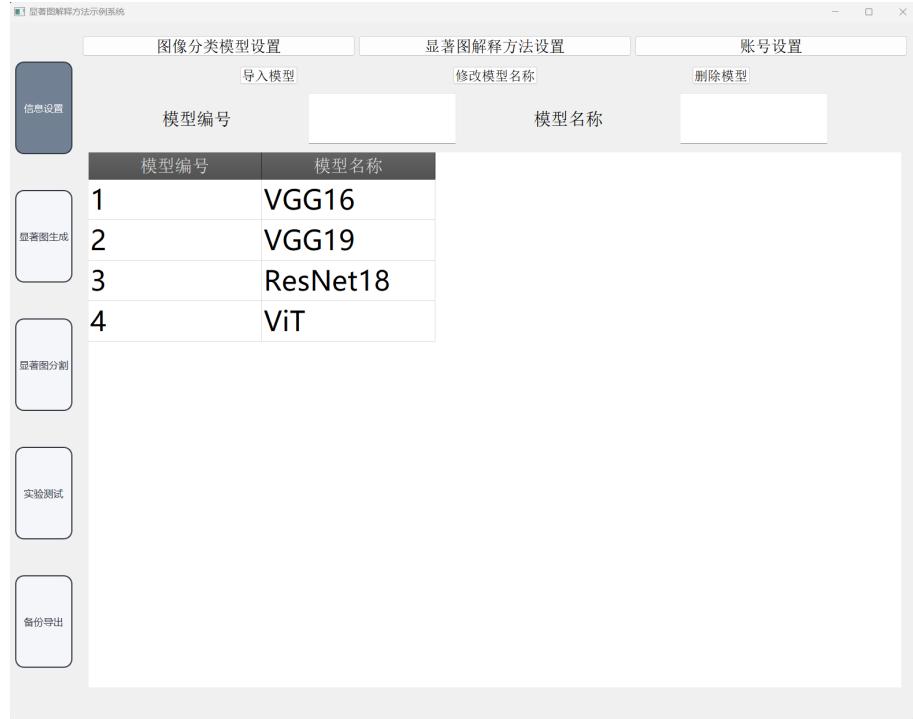


图 5-3 图像分类模型的查看和导入界面图

Fig. 5-3 View and import interface for image classification models

用的像素比例，选定后生成分割图界面上可以展示分割结果，用户可以直观对比不同方法分割效果，正如该图中展示的那样，在相同分割比例情况下，本文提出的MSG-CAM分割效果更佳，展示了完成的物体，不相关区域更少。

## 6. 实验评测功能

在图5-8中展示了实验评测的页面，实验结果通过表格的方式展示在下面页面当中。

## 5.5 本章小结

本章将多种现有的显著图解释方法整合到一个统一的程序接口中，并利用软件可视化开发技术开发了一套显著图解释方法对比评测系统。该系统可以方便研究人员对当前的显著图解释方法进行对比评测，直观展现不同显著图解释方法在相同条件下的视觉差别和评测指标差异。本章通过分析需求和功能设计逐步细化系统功能，并且通过系统页面展示了原型系统的页面功能。



图 5-4 图像分类模型的查看和导入界面图

Fig. 5-4 Saliency map method interpretation methods parameter setting page

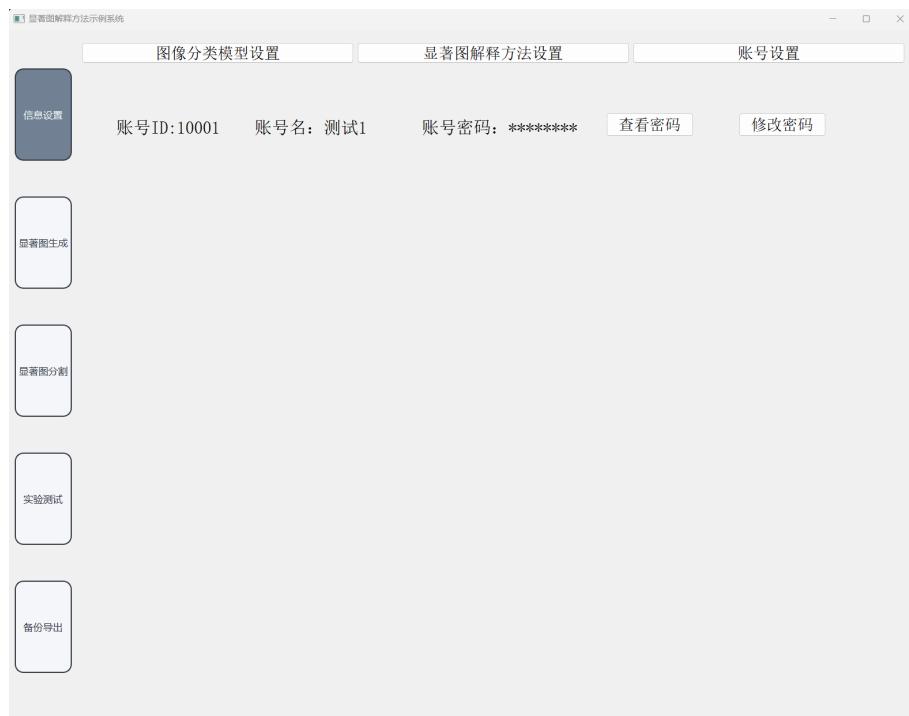


图 5-5 用户密码修改和查看页面

Fig. 5-5 User password change and view page

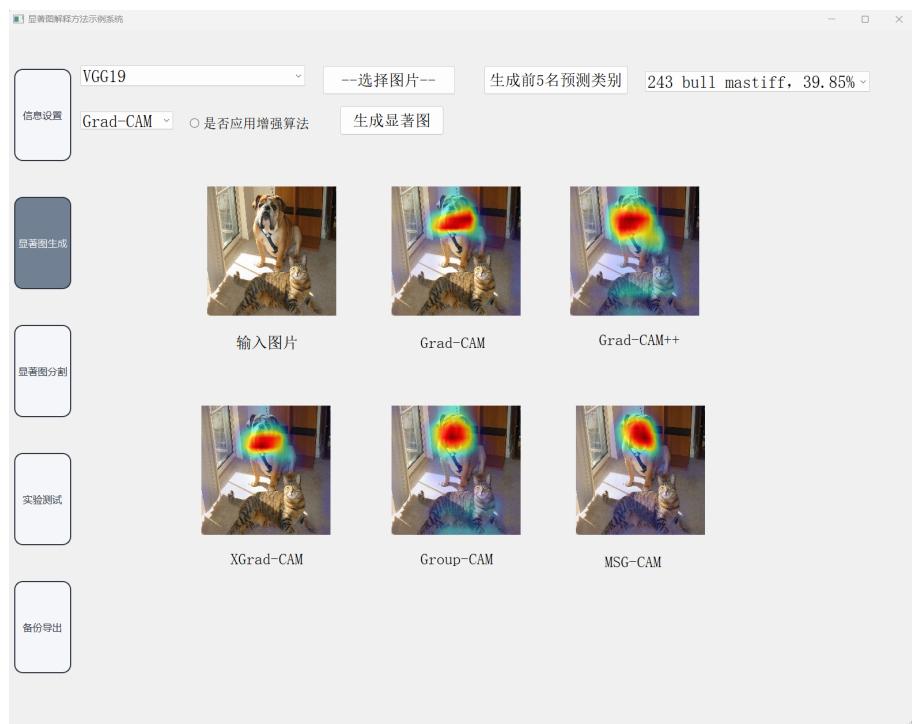


图 5-6 显著图生成页面

Fig. 5-6 Saliency map generation page

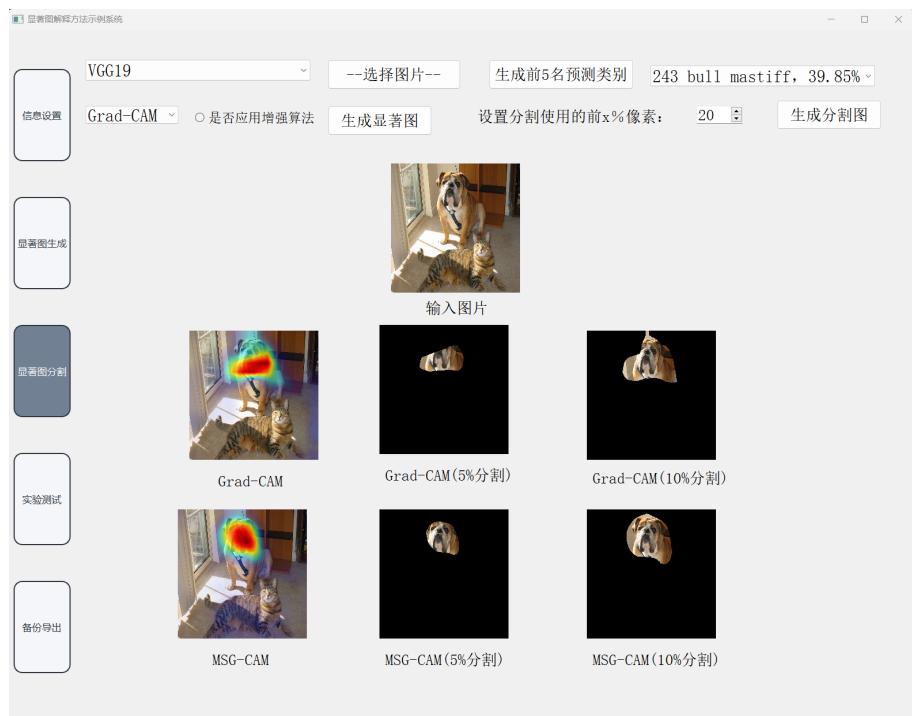


图 5-7 显著图分割页面

Fig. 5-7 Saliency map segmentation page



图 5-8 实验评测页面

Fig. 5-8 Experimental review page

## 参考文献

- [1] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.
- [2] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.
- [3] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International conference on machine learning*, pages 3145–3153. PMLR, 2017.
- [4] Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for neural networks with local renormalization layers. In *International Conference on Artificial Neural Networks*, pages 63–71. Springer, 2016.
- [5] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *In Workshop at International Conference on Learning Representations*. Citeseer, 2014.
- [6] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *ArXiv preprint*, abs/1706.03825, 2017.
- [7] Julius Adebayo, Justin Gilmer, Ian Goodfellow, and Been Kim. Local explanation methods for deep neural networks lack sensitivity to parameter values. *ArXiv preprint*, abs/1810.03307, 2018.
- [8] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.
- [9] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian J. Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In Samy Bengio, Hanna M. Wallach,

- Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 9525–9536, 2018.
- [10] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2921–2929. IEEE Computer Society, 2016.
  - [11] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 618–626. IEEE Computer Society, 2017.
  - [12] Aditya Chattopadhyay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 839–847. IEEE, 2018.
  - [13] Ruigang Fu, Qingyong Hu, Xiaohu Dong, Yulan Guo, Yinghui Gao, and Biao Li. Axiom-based grad-cam: Towards accurate visualization and explanation of cnns. In *31st British Machine Vision Conference 2020, BMVC 2020, Virtual Event, UK, September 7-10, 2020*. BMVA Press, 2020.
  - [14] Daniel Omeiza, Skyler Speakman, Celia Cintas, and Komminist Weldermariam. Smooth grad-cam++: An enhanced inference level visualization technique for deep convolutional neural network models. *ArXiv preprint*, abs/1908.01224, 2019.
  - [15] Haofan Wang, Rakshit Naidu, Joy Michael, and Soumya Snigdha Kundu. Sscam: Smoothed score-cam for sharper visual feature localization. *ArXiv preprint*, abs/2006.14255, 2020.
  - [16] Harish Guruprasad Ramaswamy et al. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 983–991, 2020.

- 
- [17] Jeong Ryong Lee, Sewon Kim, Inyong Park, Taejoon Eo, and Dosik Hwang. Relevance-cam: Your model already knows where to look. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14944–14953, 2021.
  - [18] Mohammad AAK Jalwana, Naveed Akhtar, Mohammed Bennamoun, and Ajmal Mian. Cameras: Enhanced resolution and sanity preserving class activation mapping for image saliency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16327–16336, 2021.
  - [19] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4765–4774, 2017.
  - [20] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi, editors, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1135–1144. ACM, 2016.
  - [21] Vitali Petsiuk, Abir Das, and Kate Saenko. RISE: randomized input sampling for explanation of black-box models. In *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*, page 151. BMVA Press, 2018.
  - [22] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 2950–2958. IEEE, 2019.
  - [23] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

- [24] Aravindh Mahendran and Andrea Vedaldi. Visualizing deep convolutional neural networks using natural pre-images. *International Journal of Computer Vision*, 120:233–255, 2016.
- [25] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Visualising image classification models and saliency maps. *Deep Inside Convolutional Networks*, 2, 2014.
- [26] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 782–791, 2021.
- [27] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern recognition*, 65:211–222, 2017.
- [28] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 24–25, 2020.
- [29] Qinglong Zhang, Lu Rao, and Yubin Yang. A novel visual interpretability for deep neural networks by optimizing activation maps with perturbation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3377–3384, 2021.
- [30] Andrei Kapishnikov, Tolga Bolukbasi, Fernanda Viégas, and Michael Terry. Xrai: Better attributions through regions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4948–4957, 2019.
- [31] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6967–6976, 2017.
- [32] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

- [33] Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102, 2018.
- [34] Matthieu Guillaumin, Daniel Kütte, and Vittorio Ferrari. Imagenet auto-annotation with segmentation propagation. *International Journal of Computer Vision*, 110(3):328–348, 2014.

## 附录 A 各学院中英文名称对照表

序号	中文名称	英文名称
01	通信工程学院	School of Communications and Information Engineering

## 附录 B 常见一级学科中英文名称对照表

代码	中文名称	英文名称
0810	信息与通信工程	Information and Communication Engineering

## 附录 C 常见专业学位类别中英文名称对照表

代码	中文名称	英文名称
1256	工程管理	Engineering Management

## 作者简介

### 1. 基本情况

张某某，男，重庆人，1993年8月出生，重庆邮电大学XX学院XX专业2018级博士研究生。

### 2. 教育和工作经历

#### 3. 攻读学位期间的研究成果

##### 3.1 发表的学术论文和著作

##### 3.2 申请（授权）专利

##### 3.3 参与的科研项目及获奖

以下文字用于测试。



感谢老师、同学们的关心、支持和帮助！

感谢老师、同学们的关心、支持和帮助！