中图分类号: 学校代码: 10055
UDC: 密级: 公开

南開大學

博 士 学 位 论 文

显著性目标检测中的特征高效融合研究

Research on Efficient Feature Fusion in Salient Object Detection

论文作者 刘姜江 指导教师 程明明教授
申请学位 工学博士 培养单位 计算机学院
学科专业 计算机科学与技术 研究方向 计算机视觉
答辩委员会主席 龚怡宏教授 评 阅 人 匿名

南开大学研究生院

二〇二二年五月

南开大学学位论文使用授权书

本人完全了解《南开大学关于研究生学位论文收藏和利用管理办法》关于南开大学 (简称“学校”) 研究生学位论文收藏和利用的管理规定，同意向南开大学提交本人的学位论文电子版及相应的纸质本。

本人了解南开大学拥有在《中华人民共和国著作权法》规定范围内的学位论文使用权，同意在以下几方面向学校授权。即：

1. 学校将学位论文编入《南开大学博硕士学位论文全文数据库》，并作为资料在学校图书馆等场所提供阅览，在校园网上提供论文目录检索、文摘及前 16 页的浏览等信息服务；

2. 学校可以采用影印、缩印或其他复制手段保存学位论文；学校根据规定向教育部指定的收藏和存档单位提交学位论文；

3. 非公开学位论文在解密后的使用权同公开论文。

本人承诺：本人的学位论文是在南开大学学习期间创作完成的作品，并已通过论文答辩；提交的学位论文电子版与纸质本论文的内容一致，如因不同造成不良后果由本人自负。

本人签署本授权书一份（此授权书为论文中一页），交图书馆留存。

学位论文作者暨授权人 (亲笔) 签字：_____

20　　年　　月　　日

南开大学研究生学位论文作者信息

论 文 题 目	显著性目标检测中的特征高效融合研究				
姓　　名	刘姜江	学号	1120190167	答辩日期	2022 年 5 月 13 日
论 文 类 别	博士 ☑　学历硕士 ☐　专业学位硕士 ☐　同等学力硕士 ☐　划☑选择				
学院（单位）	计算机学院		学科/专业 (专业学位) 名称		计算机科学与技术
联 系 电 话	15302185181		电子邮箱		j04.liu@gmail.com
通讯地址 (邮编)：湖北省武汉市新洲区邾城街城南华府 2 栋 2 单元 1004 号（邮编 430400）					
非公开论文编号			备注		

注：本授权书适用我校授予的所有博士、硕士的学位论文。如已批准为非公开学位论文，须向图书馆提供批准通过的《南开大学研究生申请非公开学位论文审批表》复印件和“非公开学位论文标注说明”页原件。

南开大学学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下进行研究工作所取得的研究成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签名：_____ 年 月 日

非公开学位论文标注说明

(本页表中填写内容须打印)

根据南开大学有关规定，非公开学位论文须经指导教师同意、作者本人申请和相关部门批准方能标注。未经批准的均为公开学位论文，公开学位论文本说明为空白。

论文题目			
申请密级	☐ 限制 (⩽2 年) ☐ 秘密 (⩽10 年) ☐ 机密 (⩽20 年)		
保密期限	20 年 月 日至 20 年 月 日		
审批表编号		**批准日期**	20 年 月 日

南开大学学位评定委员会办公室盖章 (有效)

注：限制 ★2 年 (可少于 2 年); 秘密 ★10 年 (可少于 10 年); 机密 ★20 年 (可少于 20 年)

摘要

显著性目标检测任务的目标在于：通过分析图像的内容来准确定位并分割出其中最吸引人类视觉注意的对象或者区域。作为一种通用的图像预处理方法，显著性目标检测不依赖于待检测对象的语义类别，已被广泛应用于多个计算机视觉任务中，以帮助它们高效地捕获图像中最重要的部分。目前，深度全卷积神经网络因其强大的多尺度特征提取能力，已在显著性目标检测领域占据了主流地位。然而现有的显著性目标检测算法在多尺度特征融合方面通常存在着设计复杂、计算复杂度高、泛化性差等缺点，当被部署到实际应用场景时，常常面临效率的瓶颈，并且它们也无法满足实际场景中常见的多目标任务同时预测需求。

具体而言，当前的显著性多尺度特征融合算法主要存在以下不足：1）特征融合没有充分考虑显著性目标检测的特性，导致结构设计冗余；2）在同等计算复杂度下，依靠基础 U 型结构融合得到的特征表征力有限；3）多任务协同的方法对所有任务采取预设、固定的特征融合策略，导致模型泛化能力差。通常而言，一个更加先进的多尺度特征融合方式往往意味着更高的效率、更优的性能和更广的适应性。因此，设计更加先进的特征融合方式成为了一个亟待解决的问题。

为此，本文提出通过对提取自骨干网络的金字塔状的多尺度特征设计更加高效的融合方式，来同时提升显著性目标检测算法的性能和效率。针对上述具体的不足，本文在现有研究的基础上从神经网络中基础的池化算子和注意力机制出发，分别在三个不同的方面提出了对应的改进方案。本文的具体贡献如下：

1. 提出通过利用高效、无参数的池化操作来解决基于 U 型结构的显著性目标检测模型中实际感受野不足，以及高层语义信息会逐渐被浅层细节所稀释从而导致的主体缺失问题。实验结果表明所提出的方法可以更准确地定位显著性目标，并具有更清晰的细节，较现有领先方法有着明显的性能提升。所提出的方法同时也在 RGB-D 显著性目标检测、边缘检测和伪装对象检测任务上获得了良好的跨任务泛化效果。
2. 提出通过跨尺度共享可学习的滤波器来促进基于 U 型结构的显著性目标检测方法中多尺度特征之间的信息交互，以极少的参数和计算量代价换

来更具表征力的特征和模型性能提升。在五个广泛使用的评测数据集上的实验结果表明，所提出的方法相比于现有的领先方法有着更优的表现，并且计算复杂度更低。

3. 提出利用注意力机制来动态选择多尺度特征并平衡不同任务分支，以实现显著性目标检测、边缘检测和骨架提取三个任务在一个模型下高效地协同学习。在多个具有代表性的测评数据集上的实验结果表明，所提出的方法可以同时且高效地完成上述三个不同任务，并取得了比当前领先的单一目标方法更好的性能。

关键词： 显著性目标检测；池化操作；信息交互；特征选择；多任务学习；卷积神经网络

Abstract

Salient object detection aims to accurately identify and segment the objects or regions that most attract human visual attention by analyzing the content of an image. Benefiting from its semantic category-agnostic character, salient object detection has been widely applied in various computer vision tasks as a general pre-processing method to help them capture the most important parts of an image efficiently. Currently, deep fully convolutional neural networks (FCNs) have occupied the mainstream position in salient object detection due to their strong multi-scale feature extraction capability. However, existing salient object detection methods usually suffer from the problems of complex design, high computational complexity, and poor generalization in multi-scale feature integration. When applied to practical scenarios, they often have efficiency issues and cannot meet the requirements of simultaneous prediction on multiple tasks.

Specifically, the current multi-scale feature integration methods in salient object detection mainly have the following shortcomings: 1) The way of feature integration does not fully consider the characteristics of salient object detection, resulting in redundant structure design; 2) With similar computational complexity, the feature representation capability obtained by the basic U-shape structure is insufficient; 3) Methods based on multi-task learning usually adopt a preset and fixed feature integration strategy for all tasks, resulting in poor generalization ability. Generally speaking, a more advanced feature integration method usually brings higher efficiency, better performance, and wider adaptability. Therefore, designing more advanced feature integration methods has become an urgent problem to be solved.

To this end, this dissertation proposes to design more efficient integration methods for the multi-scale feature pyramid extracted from the backbone network to improve the performance and efficiency of salient object detection simultaneously. Aiming at the problems mentioned above, this dissertation proposes improvement schemes from three different aspects on the basis of existing research, starting from the basic pooling

operator and attention mechanism in neural networks. The specific contributions of this dissertation are as follows:

1. Proposing to solve the problem of incomplete segmentation of subjects caused by insufficient receptive fields and gradual dilution of high-level semantic information by shallow details in the U-shape-based salient object detection models using efficient and parameter-free pooling operations. Experimental results show that the proposed approach can locate the salient objects more accurately with sharpened details and substantially improve the performance compared with the existing state-of-the-art methods. The proposed approach also generalizes well to the RGB-D salient object detection, edge detection, and camouflaged object detection tasks.
2. Proposing to promote the information interaction among multi-scale features in the U-shape-based salient object detection methods by sharing the learnable filters across scales. The proposed approach produces more representative features and achieves better performance at the cost of very few additional parameters and computation. Experimental results demonstrate that the proposed approach performs favorably against the previous state-of-the-arts on five widely used benchmarks with less computational complexity.
3. Proposing to dynamically select multi-scale features and balance different task branches to better jointly learn three different tasks, including salient object detection, edge detection and skeleton extraction within a single model by utilizing attention mechanism. Experimental results on multiple representative datasets show that though these three tasks are naturally quite different, the proposed approach can work well on all of them and even perform better than current single-purpose state-of-the-art methods.

Key Words: Salient object detection; pooling operation; information interaction; feature selection; multi-task learning; convolutional neural network

目录

主要缩略词列表

缩略词	代表意义
ASPP	Atrous Spatial Pyramid Pooling
BCE	Binary Cross Entropy
BN	Batch Normalization
CII	Centralized Information Interaction
CNN	Convolutional Neural Network
DFIM	Dynamic Feature Integration Module
FAM(+)	(Advanced) Feature Aggregation Module
FC	Fully-Connected
FCN	Fully Convolutional Network
F_β	F-measure
FPN	Feature Pyramid Network
FPS	Frames Per Second
GAP	Global Average Pooling
GGF	Global Guidance Flow
GGM	Global Guidance Module
GMP	Global Max Pooling
GN	Group Normalization
HED	Holisitcally-Nested Edge Detector
IoU	Intersection over Union
MAdds	Multiply-Adds
MAE	Mean Absolute Error
MLP	Multi-Layer Perceptron
MTL	Multi-Task Learning
NMS	Non-Maximal Suppression
ODS	Optimal Dataset Scale
OIS	Optimal Image Scale
PPM	Pyramid Pooling Module
PR	Precision-Recall
ReLU	Rectified Linear Unit
RGB-D	Red, Green, Blue, Depth
RGC	Relative Global Calibration
SE(Net)	Squeeze-and-Excitation (Network)
SGD	Stochastic Gradient Descant
S_α	S-measure
TAM	Task-adaptive Attention Module
VAM	Visual Attention Mechanism
VSD	Visual Saliency Detection

第一章 绪论

第一节 研究背景和意义

随着互联网技术的快速发展，基于互联网的社交和多媒体应用愈发成熟和普及。以图像和视频等数字信息为载体的视觉数据呈现出爆发式增长的趋势。尤其是近几年，移动互联网蓬勃发展，高度依赖图像和视频等视觉数据的新兴的、便捷化的工作和生活方式，已经深深地融入到了现代人的日常习惯之中：如手机拍照、远程会议、在线购物、网络直播等各式各样的应用场景。在这些应用场景中，常常需要对其中的主体部分进行快速而准确的聚焦，例如手机拍照中的大光圈效果、视频通话或者会议中的背景模糊功能、以及在线视频中的弹幕穿透特性等。然而，上述场景中的主体是个具体而相对的概念，没有确切和固定的语义类别，即无法脱离实际的场景内容而定义。现实中也无法针对各种各样可能出现的场景而枚举出所有的目标主体类别。这使得例如语义分割等依赖于绝对语义类别信息的技术无法适用，而亟需一种能够支持各种不确定类别目标识别的通用能力的技术。

美国心理学之父 William James 发现人类眼睛可以自动聚焦到心理感兴趣的物体上[1]。这一现象后来被著名神经科学家、诺贝尔奖得主 Torsten N. Wiesel 和 David H. Hubel 通过对视觉系统和视觉皮层的研究所证实：人类在通过视觉系统获取和处理外界视觉信息时，存在着明显的自适应性和选择性[2, 3]。我们在日常生活中都会有这种感受：当进入一个陌生场景时，我们往往会首先注意到其中最具吸引力或者最有趣的部分。即人类的视觉系统会自发地、优先地集中于那些最能刺激视觉神经的点或者区域，英国皇家学会院士 Anne Treisman 将这一认知能力归纳为视觉注意力机制（Visual Attention Mechanism, VAM），并因此获得美国国家科学勋章。即使面对各种复杂场景，人类的视觉系统仍然可以快速地定位至其中最有趣的部分。在我们反应过来之前，视觉注意力机制已经对整个视野进行了初步处理和分析，并筛选出其中最为重要或者有趣的物体或者区域进行聚焦，以使得我们能更加有效地集中于关键信息，从而加速大脑对于场景的理解和分析进程[4, 5]。

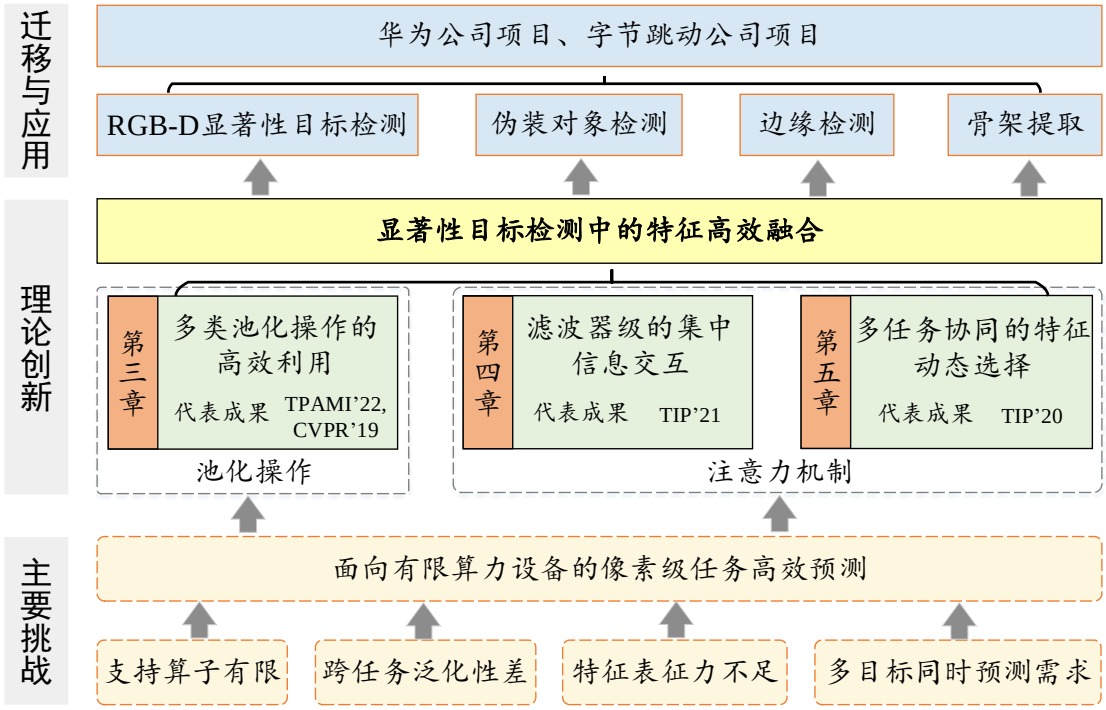


图 1.1 本论文主要研究的问题和相应工作的关系分析图。

为了让计算机系统也具有这一不依赖语义类别、高效过滤干扰、快速定位关键信息的能力，计算机视觉研究者们立足于视觉注意力机制，提出了视觉显著性检测（Visual Saliency Detection, VSD）任务。视觉显著性作为一种通用的图像属性，通常被定义为：从观察者角度出发，当其直视一给定图像时，认定的突出于背景或周围环境的某个或某些目标对象或者区域[6]。视觉显著性目标检测任务的目的在于计算出给定图像中的视觉显著性对象或者区域，及其显著的概率，而这一过程不依赖于任何目标对象或者区域的语义类别信息[7]。显著性目标检测具备有对各种不确定类别目标的通用识别能力，除了被直接应用于文章开头所列举的一些实际生活场景之外，其还被视为一种高效的图像预处理技术，应用于大量下游计算机视觉任务中，例如图像识别、图像压缩、图像检索、图像分割、图像增强、图像渲染、图像评价、目标检测与定位、目标跟踪和智能交通等。高效的视觉显著性技术可以大幅降低上述应用的计算复杂度，并达到更优的效果，对于提升人们生活和生产的效率有着重要意义。

第二节 研究难点

作为一项基础而重要的计算机视觉任务，视觉显著性目标检测自被提出至今已吸引众多学者和技术人员的广泛关注和研究。虽然近年来基于深度学习的显著性目标检测算法依靠卷积神经网络（Convolutional Neural Network，CNN）提取浅层和深层等多尺度特征的能力而得以快速发展，但当它们被部署于实际生活场景时，却因为模型复杂、计算复杂度高而面临不少限制。在硬件层面，现实场景中经常需要算法在一些终端设备（例如智能手机、机器人、自动驾驶汽车等）上实时运行，而这些设备通常运算能力较低且底层支持的深度学习算子有限。在算法层面，当前的显著性特征融合算法在同等计算复杂度下融合得到的特征的表征力有限。除此之外，现在的显著性目标检测算法通常只能预测单个任务，并且很难被迁移应用到其他任务上，导致算法设计的投入产出比较低。并且现在多数现实场景中需要算法具有多目标任务同时预测的能力，如果一个一个地去运行各任务的模型，设备的存储和计算资源会立即枯竭。不仅限于显著性目标检测，效率瓶颈目前正成为各类像素级预测任务的卡脖子问题。而一个先进的多尺度特征融合方式往往意味着能以更简单的模型设计和更少的计算代价，实现更优的预测性能和更鲁棒的跨任务泛化性。因此如何设计更加强大和高效的多尺度特征融合方式一直是各像素级预测领域一个热门的研究方向。

如图 1.1所示，本文针对显著性目标检测等像素级预测任务当前面临的问题，从池化操作和注意力机制两个神经网络中最基础的组成单元出发，对提升多尺度特征的融合效果和效率进行了研究。池化操作作为一种无参数、对输入具有平移不变等优势的操作，是图灵奖得主 Yann LeCun 于上世纪八十年代发明 CNN 开山之作 LeNet[8] 时便有的几个基本原子操作中效率最高的一种。它也是后续的各类成功的 CNN 基础网络（例如 AlexNet[9]，VGGNet[10]，ResNet[11]，GoogLeNet[12]，DenseNet[13] 等）中不可缺少的基本组件之一。注意力机制最早起源于美国国家科学勋章获得者 Anne Triesman 的视觉注意理论，由于其能够帮助神经网络过滤无关信息的干扰，自动关注目标任务中最有利的信息而有着广泛的应用。近两年来席卷计算机视觉和自然语言处理等领域中各任务的 Transformer 结构[14] 和去年计算机视觉论文最高奖马尔奖的工作[15] 本质上都属于注意力的一种。

本文将利用基础的池化操作并结合任务特性来高效融合已有特征，以解决终端设备支持的底层算子有限，以及现有模型跨任务泛化性差的问题。同时通

过对注意力机制从两个不同角度的研究：滤波器级的多层级信息高效集中交互和多任务协同的多尺度特征高效动态选择，来提升融合特征的表征力和满足现实场景中多目标同时预测的需求。除了显著性目标检测任务之外，本文还进一步将所设计的特征高效融合技术迁移应用到了其他具有通用图像属性的像素级预测任务中，包括：RGB-D 模态的显著性目标检测、伪装对象检测、边缘检测和骨架提取，以验证所提出方法广泛的跨任务适用性。为了帮助读者更好理解本文中的工作，本文将当前在多尺度特征高效融合领域仍然存在的三个具体的研究难点从理论研究的角度总结归纳如下：

- **对任务特性考虑不足：**卷积神经网络一个重要的特性便是其金字塔式的结构[9, 10]，即其深层特征包含丰富的语义信息，但是空间尺寸相对较小从而使得生成的显著性图像比较模糊而且缺少必要的细节信息，例如边缘等；而其浅层特征则通常有着较大的空间尺寸，但是仅包含低层级的局部细节信息。现有的基于卷积神经网络的视觉显著性目标检测方法通常只对骨干卷积神经网络提取到的多尺度特征进行笼统的组合，而没有充分结合视觉显著性目标检测这个目标任务本身所具有的特别属性。这类没有针对性的特征融合方式通常意味着冗余和低效。因此，如何面向视觉显著性目标检测这一特定任务而设计与其匹配的高效多尺度特征融合方式是一大研究难点。
- **融合特征的表征力有限：**在众多现有基于卷积神经网络的视觉显著性目标检测方法中，U 型网络以其简洁有效的多尺度特征融合方式而颇受青睐。一个常见的 U 型网络通常包含两个分路：一个作为特征提取器的自底向上分路和一个作为特征融合器的自顶向下分路。两个分路均包括多个不同的特征层级，而对应层级之间通常被直连。已有的视觉显著性方法大多只是简单地将 U 型网络整体地当作一个多尺度特征提取器，着重于研究如何在其基础上添加额外的结构或者模块来提升网络的整体性能，而忽略了修改和提升 U 型网络本身这一选项。因而，如何在引入尽量少的参数和计算量的情况下，修改和提升 U 型网络本身的特征融合能力，以生成更具表征力且更适应视觉显著性目标检测任务的特征，并最终提升模型性能为一大研究难点。
- **融合策略预设且固定：**为了检测出显著性对象或者区域，我们首先需要对显著性对象或者区域进行准确的定位，然后需要对定位后的目标对象或者

区域进行精确的分割。我们知道卷积神经网络需要大量的标注数据来进行训练，而大多数现有的视觉显著性目标检测算法仅集中于研究视觉显著性目标检测这一单一任务。因此，能否利用其他类型任务的互补特点和它们已有的不同类型的标注数据，来辅助视觉显著性目标检测任务以获得更好的定位和分割结果？以及如何在一个网络中进行多个不同任务的联合训练时，尽量缓和不同任务目标对于特征融合需求的冲突，并充分发挥不同任务在特点和数据之间的互补性，以获得所有联合训练任务的综合性能最大化，均为该方向的研究难点。

第三节　研究目标与主要贡献

本文主要研究显著性目标检测中的高效特征融合技术，将立足于神经网络中的基础算子和机制，从三个不同的角度和方向来探索研究如何利用它们来有机结合显著性目标检测任务的特性，并融入到高效特征融合策略的设计之中，以期获得更加准确和精细的显著性图。具体而言，本文分别基于已有显著性目标检测算法在特征融合方面存在的三个不足和挑战：对任务特性考虑不足、融合特征的表征力有限、以及融合策略预设并且固定，提出从多类型池化操作的利用、多层级信息交互和多任务协同学习三个方面进行了相应的高效特征融合方案的研究，弥补了已有算法的局限和不足。下面将对本文的主要研究内容和贡献进行简要的介绍。

- 第二章对与本文工作相关的视觉显著性目标检测文献、常用的数据集和评价指标，相关的二元预测任务所属领域的文献以及一些相关的基础操作、结构和机制所属领域的文献进行了介绍。
- 第三章提出了基于高效特征池化和融合的显著性目标检测算法。该算法利用池化这一不需要可学参数的高效像素级操作，结合显著性目标检测中先定位后分割的特性，实现了对卷积神经网络提取得到的多尺度特征进行高效融合的目的。现有的模型在融合卷积神经网络提取得到的多层级特征时，通常忽略了显著性目标检测这个目标任务本身的特点，在效果和效率方面均有所欠缺。本章首先提出了一个基于适应性平均池化操作的全局信息指导模块。该模块利用全局的定位信息来高效地指导其余层级间的信息融合过程，有效缓解了基线模型在多尺度特征融合过程中的定位信息损失问题。为了改善在跨尺度特征融合过程中，低分辨率、细节模糊的高层级

特征与高分辨率、定位缺失的低层级特征之间的信息难以对齐的问题，本章进一步提出了一种基于平均池化操作的多尺度特征融合模块。该模块利用平均池化来桥接不同尺度特征之间的感受野差距，通过在不同下采样倍率下进行信息对齐来让模型能够更高效地捕获跨尺度信息，实现了更有效的特征融合。实验结果表明，本章算法在多个显著性目标检测测评数据集上的表现均优于现有的各类型方法，并且达到了更快的速度。本章还进一步将提出的算法迁移应用到了 RGB-D 显著性目标检测、边缘检测以及伪装对象检测三个任务上，相较于这些任务的领先方法，所提出的方法取得了相当甚至更好的结果。

- 第四章提出了基于高效信息集中交互与融合的显著性目标检测算法。该算法以显著性目标检测模型中常见的 U 型结构为落脚点，重新设计了 U 型结构的自底向上通路和自顶向下通路之间的连接。现有的基于 U 型结构的显著性目标检测模型通常只是简单地将其整体性地视为一个特征提取器来使用，而忽视了可以通过改进 U 型结构本身来提升融合特征的表征能力，进而促进模型的整体性能这一途径。本章提出在 U 型网络的自底向上和自顶向下两个通路之间增加额外的多层级信息交互渠道，从而使得从自底向上支路提取得到的多尺度特征在被送入到自顶向下通路进行融合之前，能够高效利用各尺度特征之间的信息来进行动态调整，以获得更好的整体融合效果。在上述改进的基础上，本章进一步提出了一种能够发挥新引入的多层级交互渠道优势的相对全局信息矫正模块。该模块能够在多尺度特征进行信息交互时，充分利用相邻特征层级之间有着天然的感受野差距这一特点，以获得更具表征力的融合特征。得益于所提出的多层级交互策略主要依赖于共享的、可学习的滤波器作为信息载体，本章提出的算法在相对于骨干网络只增加极少量的参数量和计算复杂度的情况下，在多个公开数据集上取得了明显优于已有方法的速度和效果。
- 第五章提出了基于高效特征动态选择与融合的多任务协同学习算法。随着智能手机等移动设备的迅速普及，许多浅层计算机视觉任务被广泛地植入进移动设备中作为基础系统的一部分。受限于移动设备有限的存储和计算资源，如何将多个不同的任务集中进一个算法模型中变得至关重要。本章提出的算法以显著性目标检测任务为基点，以边缘检测任务和骨架提取任务为支点，以高效的多任务协同的方式来动态融合来自于其他任务中的，

相对于显著性目标检测任务的互补特征，最终提升显著性目标检测任务的性能。为了有效平衡不同任务在数据分布和特征偏好之间的差异性，本章进一步设计了一种面向多任务的自适应注意力模块。该模块以极小的计算代价构建出不同任务之间必要的信息交互，能够有效避免不同任务之间因为可能存在的互斥特性而导致的网络收敛困难问题。本章所提出的算法能够有效发挥多任务协同学习的优势，通过利用骨架提取任务中对于主体的准确定位能力和边缘检测任务中对于细节的精细分割能力，本算法在一个模型中同时且高效地完成了三个不同的任务，并且各任务均达到了相对于自身领域已有领先算法更优的结果。

- 第六章对本文中所提出的算法和模型进行了总结，并从研究和应用两个方面指出了显著性目标检测领域当前所面临的问题和挑战。同时本章也提出了本文中所涵盖的算法和模型的一些可改进的方向，探讨了显著性目标检测领域未来可能的发展趋势，并对未来工作进行了展望。

第二章　相关工作综述

本章对本文主要研究内容的相关工作进行介绍，具体分为如下三个部分：第一节主要介绍了显著性目标检测任务相关的工作、数据集和评价指标；第二节主要针对本文相关的一些二元任务及其文献进行了介绍；第三节主要介绍了本文相关的一些基础操作、结构和机制及其分别对应的工作。

第一节　显著性目标检测相关的工作、数据集和评价指标

近年来，在学术和工业界同时涌现出了大量不同类型的显著性目标检测算法。这些算法根据其所使用的特征的来源的不同，可以被大致归纳为以下两类：基于领域专家根据经验和观察而手工设计的各类特征算子和基于深度学习相关技术而自动学习特征的显著性目标检测算法。在深度学习出现之前，早期的基于手工设计特征的算法主要聚焦于对各类浅层视觉特征的利用，例如前背景中颜色或形状等先验、图像超像素、频谱分析、以及颜色对比度等。这类方法虽然具有较强的可解释性和较低的计算复杂度，也能够在一些背景成分和结构相对简单、前背景差异明显的场景中取得不错的效果，但当它们被应用于真实世界下的复杂场景时，却常常表现不佳。研究者将上述缺点归因为手工设计的特征无法有效建模隐藏在图像场景中的语义信息，以至于在那些背景复杂、前背景对比度低、以及显著性对象形态结构复杂等场景下不能有效定位显著性对象或者完整地将其检测出。而基于深度学习得到特征的算法，得益于其能够自动地从标注图像集合中学习到隐藏在其中的语义信息，并能够自适应地提取多个空间和语义尺度下的特征的优点，可以有效弥补手工特征算法的缺点，同时也在精度和速度两个方面取得了长足的进步。本节接下来的两个子小节将分别对上述两类算法的研究历史和现状进行简要的介绍。本节最后的两个子小节将分别对显著性目标检测任务中常用的数据集和评价指标进行介绍。

2.1.1　传统的显著性目标检测方法

本小节根据各传统方法中手工设计的特征的出发角度的不同，将近些年来具有代表性的传统方法大致分为下述四个类别[16, 17]。

基于频域转换的方法：此类方法将输入图像原属于空间域的特征映射到适当的频域空间上，然后利用显著性对象或区域在频域空间的某些特性来完成对其检测的目的。例如基于傅立叶变换理论的谱残差（Spectral Residual）[18] 算法，该算法首先使用傅立叶变换将图像转换到频域空间，并得到其对应的振幅谱和相位谱；然后利用自然图像在频域空间的对数谱存在统计相似性这一特点来过滤掉背景信息，即将对数振幅谱减去其均值滤波后的结果以得到剩余谱；再将剩余谱和相位谱相加并求自然指数后通过傅里叶反变换转换到空间域，最后经过简单的高斯模糊滤波就可以得到显著性结果。之后的部分工作[19, 20] 也尝试从频域的相位谱角度来进行改进。Achanta 等人[21] 发现显著性对象的边界区域通常有着较大的颜色或亮度变化，而这一空间域的变化一般也对应着频域的变化，于是提出了一种基于图像颜色和亮度信息的频域分析算法。该算法利用高斯函数的差分（Difference of Gaussian）将图像转换到频域，然后通过设定不同的高斯方差来尽可能多地保留原始图像中的频率内容，以获得覆盖更加全面的显著性检测结果。一般而言，基于频域转换的方法只能获得较为粗糙的结果，主要表现为显著性对象或区域的大致位置和轮廓，而难以得到完整而均匀的内部区域。

基于视觉先验的方法：视觉先验信息主要借鉴于人类视觉系统中注意力机制的相关理论，在一些早期的显著性目标检测工作中发挥了重要的设计指导作用。部分颇具代表性的先验信息包括：全局和局部对比度先验、前景和背景先验、以及颜色和形状先验等。例如 Goferman 等人[22] 提出了一种基于上下文信息感知的显著性检测算法，该算法通过一系列具有不同空间尺寸的窗口来获得图像的全局和不同层级的局部颜色对比信息，并基于此计算出初步的显著性结果，然后再结合高层级的语义信息对显著性区域进行扩张以得到更精细的最终检测结果。而 Wei 等人[23] 从图像背景的角度提出了边缘和连接性两个先验，即在基本的摄影构图中图像的边界一般都是背景区域，以及背景中的大多数图像块都相互连接，然后将显著性图的预测问题转化为了一个计算对象到边界距离的问题。Cheng 等人[24] 提出像素点之间的颜色差异可以被用来预测所对应的显著性值，并通过颜色直方图的形式进行建模以获得图像的初步区域分割结果，然后再依据区域的局部颜色对比度信息对显著性值进行加权以得到最终的结果。

基于稀疏表示的方法：以经典的稀疏理论为依据的方法，主要参考了人类视觉系统对于目标场景中稀疏区域的物体有着更高关注度的特点。这类方法首先将图像转换为高效的稀疏表示形式，然后以迭代的方式完成对显著性区域的

稀疏系数的优化，从而完成显著性图的计算。稀疏表示作为一种常用的信号处理技术，能够有效降低数据的维度并简化计算模型。Han 等人[25] 通过稀疏编码将输入图像分解为编码和残差两部分，编码长度和局部复杂度之间通常有着密切的关联，而残差则主要代表着不确定性。他们提出以稀疏编码的长度作为显著性值，并以其 L_0 范数作为残差的加权以获得最终的显著性预测图。Li 等人[26] 通过超像素分割来得到图像的边界，并以此为模版来提取背景特征并获得背景模版，然后在此基础上构建稠密和稀疏的外观表示，再通过对多尺度的重构误差进行融合以得到显著性得分，最后利用贝叶斯公式对融合得到的显著性值进行积分以获得最终的显著性图。

基于图论的方法：基于图论的方法通常会对输入图像进行不同形式的降维处理，并在此基础上构建图模型。例如 Zhang 等人[27] 首先在图像的颜色通道进行随机的阈值分割以获得对应的布尔图，然后以此作为原始图像的特征描述，最后基于布尔图的拓扑结构计算显著性结果。Jiang 等人[28] 提出将显著性目标检测建模为图像图模型的吸收马尔可夫链，并协同考虑显著性对象和背景的外观差异和空间分布特点。他们将虚拟边界节点作为马尔可夫链中的吸收节点，并通过计算每个转移节点到所有边界吸收节点的吸收时间来衡量它与它们的全局相似性，然后利用吸收时间作为度量便可以将显著性对象和背景分离出来。Yang 等人[29] 提出将图像表示为以超像素为节点的闭环图，然后利用图形的流形排序算法来对图像元素与前景或背景线索的相似性进行排序，最后各超像素的显著性可以通过其与给定种子或查询点的相关性而得到。

总体而言，基于手工设计特征的传统显著性目标检测算法通常都是根据某个特定的观察或者是经验而设计，因此具有较强的针对性和较大的局限性，一般只适用于背景的颜色、纹理和结构相对简单，前背景间有着明显差异的简单场景。传统算法本身缺乏高层的语义信息指导，因而不能在跨分布、跨场景等情况下有着鲁棒的表现。但由于其对标注数据的依赖较小，并且有着较快的计算速度，也在部分实际场景中有着应用。

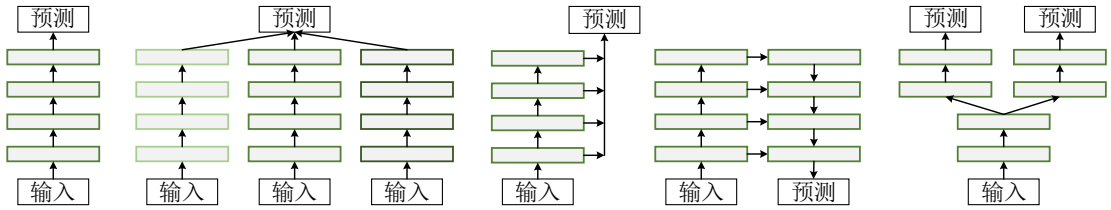
2.1.2 基于深度学习技术的显著性目标检测方法

近年来随着消费级高性能并行计算设备的普及，需要强大算力作为支撑的深度学习技术在众多领域取得了长足进步。显著性目标检测也不例外，越来越多基于深度学习的模型和算法相继涌现，并逐渐成为了主流。不同于传统算法需要手工去设计各类特征，深度神经网络能够自动地从标注数据中学习并提取

出任务适用的特征。深度神经网络通常具有层次化的结构，并能够提取从浅到深多个层级的、具有强大表征能力的特征。基于深度学习的方法相较于传统算法表现出了更强的泛化性，并在各种复杂场景下达到了更好的效果。根据所采用的骨干深度神经网络的不同，这些方法可以被大致分为两类：基于多层感知器（Multi-layer Perceptron，MLP）的方法和基于全卷积网络（Fully Convolutional Network，FCN）的方法。

2.1.2.1 基于多层感知器的方法

基于多层感知器的方法一般不会在输入图像的原始像素上直接进行操作，而是通过一些单独的预处理方法首先将其划分为适量数目的独立子区域（例如超像素、规则图像块、对象候选框等），然后利用深度神经网络提取各个子区域上的特征，再通过一个多层感知器为每个子区域分别预测一个显著性得分，最后将所有子区域的得分进行融合以得到最终的显著性结果图。例如 He 等人[30]提出将显著性目标检测任务建模为一个二元标注问题并利用深度学习技术来提取层次化的对比特征。作者提出了一个包含多个不同超像素分割尺度分支的超像素卷积神经网络。输入图像首先被分割为含有不同数目的超像素集合，然后每个超像素会提取出对应的颜色独特性序列和分布序列，并被送入到对应尺度的网络分支中获得该超像素的显著性值，最后所有尺度分支预测的显著性图被融合以得到最终的显著性图。Wang 等人[31] 提出了一个基于局部估测和全局搜索的显著性目标检测算法，在局部估测阶段，作者使用一个深度卷积网络来学习每个局部图像切块中的特征，并预测每个像素所对应的显著性值，然后再结合高层物体概念来改善局部显著性图；在全局搜索阶段，作者将前一步得到的局部显著性图和图像的全局对比以及几何信息进行融合作为全局特征，并通过其他候选框生成算法生成一系列物体候选区域框，之后另外一个深度卷积网络将全局特征作为输入并预测每个物体候选区域所对应的显著性值；最后通过对显著性目标区域进行加权求和以得到最终的显著性图。Li 等人[32] 提出了一个基于多尺度深度特征的显著性目标检测算法，作者首先将输入图像划分为若干个独立的图像区域，然后在每个图像区域的周围裁剪三个窗口尺寸逐渐增大的图像块，并分别送入三个独立的深度卷积网络以提取特征，然后这些特征通过一个全连接网络进行融合并回归得到相应图像区域的显著性值，最后通过聚合不同尺度下得到的显著性值得到最终的显著性图。Zhao 等人[33] 提出了一个基于空间多尺度上下文的深度显著性检测框架，其包括一个关注局部上下文的通路



(a) 单路 (b) 多路 (c) 侧融合 (d) U型 (e) 多分支

图 2.1 常见的全卷积显著性目标检测方法中五种典型的结构。

和一个关注全局上下文的通路。作者首先将输入图像划分为超像素表示，然后通过上述两个通路分别提取每个超像素周围的局部和全局特征，最后将两路特征进行融合并预测得到各超像素对应的显著性值。

虽然基于多层感知器的方法相较于传统的手工特征算法已经有了明显进步，但是它们依然有着许多传统算法的影子。例如输入图像首先需要被分割为超像素或者图像块之类的子区域，这一分割过程通常通过传统算法实现，分割效果的好坏便成了最终预测精细程度的上限。再者就是这类算法通常利用全连接层来获得最终的预测，这一设定意味着网络在最终预测的阶段缺乏图像的空间结构信息，这对于图像级预测任务是不合理的。

2.1.2.2 基于全卷积网络的方法

为了解决上述基于多层感知器的显著性目标检测方法中所存在的缺点和局限，受到全卷积网络[34] 在语义分割任务上成功应用的启发，近几年来主流的显著性目标检测算法几乎都是基于全卷积网络结构。全卷积网络一般只包含卷积层、池化层、非线性激活层等输入输出均保留有空间位置信息的网络层，而不包括全连接层。全卷积网络的优势在于可以端到端地处理任意尺寸的图像输入，并可以通过简单的插值操作获得和输入图像相同空间尺寸的像素级显著性预测图。除了推理阶段的优势之外，全卷积网络通常也可以很容易地被端到端地训练，同时得益于卷积操作的平移不变性，其需要的参数量和计算复杂度也更少。如图 2.1所示，常见的全卷积显著性目标检测方法可以被大致归类为如下五种典型结构[35]：单路结构、多路结构、侧融合结构、U 型结构、和多分支结构。

单路结构：单路结构是最简单的一种网络结构。它一般由一系列级联的卷积层，非线性层，以及几个用于空间尺寸下采样目的的池化层组成，而显著性预测图直接在网络的末尾生成。由于单路结构的模型可以在结构层面进行的修改有限，因此大多数基于单路结构的方法主要关注于其他层面的改进。例如 Zhang

等人[36] 为了提升单路结构模型的鲁棒性和预测精度，在一个标准的编码器-解码器模型的基础上分别提出了一个修正随机失活机制来在卷积神经网络中提供有不确定性的特征，以及一个混合上采样方法来缓解反卷积操作导致的棋盘伪影现象。Wang 等人[37] 则提出了一种循环式的全卷积网络，输入图像首先通过传统的启发式算法得到初始的显著性图，然后被当作先验与原始的输入图像合并后输入全卷积网络中得到细化后的显著性图；细化后的显著性图与原始输入图像被合并后再次输入到全卷积网络中以得到进一步细化后的显著性图；通过多次循环上述步骤，显著性预测图可以逐渐被矫正和完善。Hu 等人[38] 提出了一个深度水平集全卷积网络以获得边界更加准确、主体更加均匀的显著性图，该网络首先通过一个卷积网络从输入图像生成一个粗糙的显著性水平集图并被上采样到输入图片分辨率，之后被一个引导式超像素滤波层通过原始图像的超像素分割结果进行优化，再通过一个单位阶跃函数得到最终的显著性结果图。Kuen 等人[39] 提出了一个循环注意力卷积网络来更好地检测多尺度的显著性对象，该网络首先通过一个基于编码器-解码器结构的卷积网络来生成粗糙的显著性图，再利用循环注意力机制以迭代的方式在原始输入图像中裁切出与显著性对象相关的区域，通过渐进的优化以得到最终的显著性结果图。

多路结构：多路结构的模型通常有着多个不同分辨率或者结构的输入，并利用多个卷积神经网络支流来显式地学习得到多尺度的显著性特征。例如 Li 等人[40] 提出了深度对比学习模型，该模型包括两个互补的分支：一个像素级全卷积分支和一个块级空间池化分支。第一个分支直接预测出输入图像对应的像素级显著性图，第二个分支则提取能够更好表征物体边缘信息的块级特征，然后通过一个全连接的条件随机场来适应性地融合两个分支的预测，以得到空间连贯性和边缘定位更好的结果。Wang 等人[41] 提出了一个分段微调模型，该模型包括一个使用了金字塔池化模块的负责预测粗糙显著性图的主分支，和一个利用局部上下文信息来微调预测结果的辅助分支，辅助分支以多阶段微调的形式逐步增强主分支预测结果中的局部细节信息，最终生成更加精细的显著性结果图。Chen 等人[42] 受眼动追踪实验中，复杂图像中的显著性对象可以通过选择获得最高注视密度的语义预分割对象来得到的启发，提出了一个双路模型：一个在眼动追踪数据上预训练过的视觉注视支路，和一个在语义分割数据集上预训练过的分割支路，注视支路的输出被融合到分割支路上来指导其获得关于显著性对象的预测。Zeng 等人[43] 为了得到高分辨率的显著性目标检测结果，提出了

一个包含三个子分路的模型。作者首先使用一个全局语义子模型提取下采样后的输入图像中的全局语义信息，然后使用这些信息来指导另一个专注于局部区域的局部微调子模型并生成高分辨率的显著性预测，最后通过一个全局-局部融合子模型来融合前述两个子模型的预测以得到最终的全尺寸高分辨预测结果。

侧融合结构：基于侧融合结构的模型主要利用了 CNN 天然的空间金字塔型结构和其产生的多尺度特征，通过将来自骨干网络的多层级特征融合后再进行显著性目标预测。这类模型中一个典型特点为侧边输出的各尺度预测通常也会由真值标签进行监督优化，达到深度监督学习的目的。例如 Hou 等人[44] 提出在整体嵌套边缘检测网络（Holisitcally-Nested Edge Detector，HED）[45] 的跳跃层结构的基础上引入一系列短连接，以实现高层语义信息向低层细节信息之间的传递。短连接的加入使得高层语义信息能够给低层特征提供更多显著性区域的定位信息，而同时低层的细节信息可以辅助高层特征恢复显著性区域的细节，最终获得整体定位更加准确以及细节更加丰富的显著性结果图。Luo 等人[46] 受传统显著性目标检测方法中对比度先验的启发，提出了一个基于非局部深度特征的显著性目标检测方法，该方法构建了一个 4×5 的多分辨率网格结构，首先以自顶向下的方式来融合各层级的特征以及它们之间的对比度特征来得到局部的显著性图，再通过将局部显著性图和模型最顶部提取到的全局信息相融合以得到最终的预测结果。除此之外，作者还提出了一个贝叶斯损失来惩罚边界上的预测错误以实现空间一致性。Wu 等人[47] 认为卷积神经网络中高分辨率的浅层特征相较于深层次的特征对性能的提升很小却带来了大量的计算负担，于是提出了一个部分级联的解码器网络，该网络舍弃了浅层特征以获得更高的计算效率和速度，并利用生成的显著性图作为注意力来改善高层特征的学习并提高预测性能。Zhao 等人[48] 提出了一个金字塔特征注意力网络来增强高层特征中的上下文信息和浅层特征中的空间结构性信息，该网络首先通过一个上下文感知的金字塔特征提取模块和一个逐通道的注意力模块来捕捉高层特征中的多尺度上下文信息，然后利用一个空间注意力模块来去除浅层特征中背景细节的影响，最后将上述特征进行融合并结合一个边缘保留损失来获得更加精确的显著性图。

U 型结构：基于 U 型结构的模型通常包括一个自底向上的特征提取通路和一个自顶向下的特征融合通路。模型通过这个额外的自顶向下通路来聚合较浅层中更精细的特征以逐渐改善由前向网络顶端所生成的粗糙显著性预测。模型的最终预测结果直接在自顶向下通路的末尾生成。Chen 等人[49] 提出在 HED[45]

网络的基础上通过残差学习的方式在侧向输出中学习残差特征以改善前向网络顶端的粗糙预测，然后通过在自顶向下通路中引入反向注意力模块来促进显著性残差的学习，该模块通过擦除掉较深层次输出中的显著性区域来鼓励网络去学习被漏掉的目标区域，以得到更加完整的预测结果。Zhang 等人[50] 提出了基于渐进注意力引导的循环式显著性目标检测模型，该模型利用注意力机制来选择性地融合与显著性目标相关的多尺度上下文特征，从而减轻来自背景的干扰信息。该模型还提出了一个多通路循环反馈机制来将网络最高卷积层的全局语义信息传递到较浅的网络层中，以提升模型的整体性能。Liu 等人[51] 提出了一个基于像素级上下文注意力的显著性目标检测模型，该模型通过将全局像素级上下文注意力模块嵌入到自底向上通路的顶端，以及局部像素级上下文注意力模块嵌入到自顶向下通路中，并利用多层监督来促进全局上下文信息和多尺度局部上下文信息之间的交互，达到提升模型性能的目的。除了注意力机制之外，Wang 等人[52] 提出了一个全局定位和局部修正的模型，该模型包括一个利用上下文信息来抑制杂乱的浅层特征的全局循环定位网络，该网络以迭代的方式不断改善 CNN 中的隐式特征；以及一个专门的边界改善网络来自适应地学习对应于每个空间像素的局部上下文信息，并获得更清晰的显著性边界结果。Feng 等人[53] 则提出在 U 型结构的两个通路对应的层之间嵌入注意力反馈模块来促进多层级特征之间的信息交互，并逐尺度地预测显著性图；同时作者还提出了一个边缘增强损失来辅助模型更好地学习显著性对象的边缘区域。类似的，Qin 等人[54] 提出了一个边界感知的显著性目标检测模型，该模型包括一个深度监督的编码器-解码器网络用于显著性预测，和一个残差改善模块用于边界区域的修正。作者同时还利用一个结合了二元交叉熵损失、结构性相似度损失、和交并比损失的混合损失函数来分别监督模型对于像素、区域、和全图三个层次的显著性预测。Zhang 等人[55] 提出了一个门控的双向信息传递模型，该模型首先利用一个多尺度上下文特征提取模块来捕获输入图像中的上下文信息，然后通过一个双向结构来实现多尺度特征之间的信息交换并通过门控函数来控制信息的交换率，最终将多层次的预测结果融合以得到显著性预测结果图。

多分支结构：基于多分支结构的模型通常包括多个分别对应于不同的预测任务或者属性的输出分支。它们利用多任务协同学习的概念来提升模型的特征提取能力。例如 Kruthiventi 等人[56] 认为人类眼注视点和显著性对象的位置之间通常有着关联，因此提出将显著性目标检测任务和眼注视预测任务在一个模

型中进行联合学习，该模型包括一个共享的骨干网络用于多尺度特征的提取，以及两个独立的预测分支分别用于显著性目标检测任务和眼注视预测。Wu 等人[57] 提出通过多任务交织监督的形式来利用前景轮廓检测和边缘检测任务的监督信息以提升显著性目标检测结果的完整性和区域边缘的准确性，前景轮廓检测可以帮助显著性目标检测得到更精确的结果，而边缘检测可以辅助前景轮廓任务得到更准确的前景轮廓，反之前景轮廓可以减少边缘检测任务中的噪声干扰。Zhang 等人[58] 认为具有辨识度语义信息的特征对于凌乱场景中的显著性目标检测有着重要意义，并提出利用输入图像的描述文字作为一个辅助语义任务来提升显著性目标检测任务在复杂场中的性能，作者设计了一个包含两个子网络的模型：一个图像文字描述网络用于捕获编码在文字中的关于主要对象的语义信息，和一个局部-全局感知网络来结合文字描述和视觉上下文信息以预测显著性结果图。Zeng 等人[59] 提出将显著性目标检测任务和弱监督语义分割任务在一个模型中进行端到端的联合学习，该模型包括一个分割网络和一个显著性聚合模块，首先通过分割网络生成输入图像的分割结果，然后用显著性聚合模块得到每个语义类别的显著性结果，最后将所有类别的显著性结果聚合得到最终的显著性预测图。

2.1.3 显著性目标检测常用评测数据集

随着视觉显著性目标检测技术的发展，一系列相关数据集被逐渐提出。早期的数据集所涵盖的场景一般比较简单，且每张样本图像中只包含一个显著性目标，而标注形式仅为简单的边界框。但由于边界框形式的标注在指示显著性目标的同时也会框入一定量的背景像素，很快便被像素级别的标注形式所取代。而之后出现的显著性目标检测数据集几乎都是采用了像素级别的精确标注。此标注类型的数据集也经历了场景由单一到丰富、背景由简单到复杂和单图显著性目标数目由单个到多个的演变过程。除此之外，最近的一些显著性目标检测数据集除了显著性目标的像素级掩膜标注之外，还提供了诸如：场景特点、目标边界框、目标数目、各目标细节和特性等实例级别的信息。这些额外的标注信息为本领域的新方向探索和其它相关的领域的类似研究提供了基础和帮助。表格 2.1为六个颇具代表性的数据集的统计数据对比，图 2.2展示了从这六个数据集中挑选出来的部分困难样例，这些数据集的详细介绍如下：

1. **SOD 数据集**[60] 包含来自 Berkeley 分割数据集[64] 的 300 张图像。每张图

表 2.1 部分常用的视觉显著性目标检测数据集的统计数据。这些数据集均采用像素级别标注，每张样本图片中均包含一个或多个显著性目标。

数据集	发表年份	发表刊物	图像数量	长宽最大/最小分辨率	注视点标注
SOD[60]	2010	CVPR-W	300	481/321	✗
DUT-OMRON[29]	2013	CVPR	5,168	401/139	✓
PASCAL-S[61]	2014	CVPR	850	500/139	✓
ECSSD[62]	2015	TPAMI	1,000	400/139	✗
HKU-IS[32]	2015	CVPR	4,447	500/100	✗
DUTS[63]	2017	CVPR	15,572	500/100	✓

像均是由七个参与者同时进行的标注，并通过少数服从多数的机制来获得最终的像素级标注结果。该数据集中许多图像中包含有多个颜色和纹理容易与背景混淆，或者触及到图像边缘的显著性目标。

2. **ECSSD 数据集**[62] 包含 1,000 张从互联网上收集得到的自然图像，其中每张图像均包含有语义意义，但是场景相对复杂。该数据集中的所有图像均为测试图像，其真值结果由五位参与者标注得到。
3. **PASCAL-S 数据集**[61] 包含从 PASCAL VOC 2010 分割数据集[65] 的验证集中仔细挑选的 850 张图像，均作为测试用途。每张图像的标注结果是根据 PASCAL VOC 2010 数据集中的人眼注视点标注信息，然后将对应的显著性目标重新进行标注得到。该数据集中的显著性标注为像素级别的非二值化数值，其值表示包含该像素的区域被选择为显著性对象的比率。
4. **HKU-IS 数据集**[32] 包含 4,447 张像素级别标注信息的图像。其中有 3,000 张作为训练用途，其余的 1,447 张作为测试用途。该数据集涵盖的场景比较复杂，通常包括多个分布在不同空间位置，有着相似的前景或者背景表征，且互不相连的物体。并且每张图像中至少有一个显著性对象位于图像的边界部分。
5. **DUT-OMRON 数据集**[29] 包含 5,168 张具有相对复杂的背景和多样化的显著性目标的图像。该数据集中的每张图像均提供了高质量的像素级别标注、五名观测者的眼动数据和边界框标注。
6. **DUTS 数据集**[63] 包含 10,553 张挑选自 ImageNet[66] 的训练和验证子集的训练用图像，5,019 张挑选自 ImageNet[66] 的测试子集和 SUN 数据集[67]

图 2.2 选自不同显著性目标检测数据集中较难的样例及其对应的真值标注结果。

的测试用图像，是目前视觉显著性目标检测领域最大的数据集。由于该数据集包含的训练样本涵盖大量丰富且多样的场景和物体类别，其自提出以来被许多显著性目标检测算法作为标准的训练数据集使用。

2.1.4 显著性目标检测常用评价指标

为了有效评价各显著性算法的性能，学术界和工业界提出了多个不同类型的评价指标，以从不同角度、更加综合地衡量算法预测的结果和人类标注之间的一致性。本文主要采用四种被业界广泛认可并使用的显著性目标检测算法评价指标，它们详细的介绍如下：

1. **准确率-召回率曲线（Precision-Recall Curve, PR 曲线）**是视觉显著性目标检测领域最为常用的评价指标之一。通过预测每个图像像素是否显著，我们可以将图像显著性目标检测任务视为一个像素级别的二分类问题。给定算法预测的显著性结果图像和相应的真值标注图像，通过对两者的重叠面积采取不同的计算方式，我们便可以分别得到准确率和召回率。准确率主要用来衡量算法准确检测的能力，计算为算法预测出的真实显著的区域占所有预测出的显著区域的比值。召回率主要用来衡量算法完整检测的能力，计算为算法预测出的真实显著的区域占所有的真值显著区域的比值。给定一幅图像，假设其对应的某算法的显著性预测结果图为 P, 相应的显著性真值标注图为 G。通过一个给定的阈值 t，对显著性预测图 P 进行二值化操作，可以得到相应的二值化显著性预测图 $P(t)$,

则对应的准确率和召回率可以通过以下公式计算得到：

$$Precision(t) = \frac{|P(t) \cap G|}{|P(t)|}, Recall(t) = \frac{|P(t) \cap G|}{|G|}. \tag{2.1}$$

通常而言，阈值的选取范围为 $t \in [0,255]$，包括共 256 个不同的阈值。而依据不同的阈值，可以计算得到 256 组准确率和召回率对。再通过将召回率作为横轴，准确率作为纵轴，按顺序绘制出这些点对便可以得到 PR 曲线。一般而言，我们期望算法在有着更高的准确率的同时也有着更优的召回率，即所得到的 PR 曲线更加接近绘图坐标系的右上方。

2. **特征相似度（F-measure，F_β）**由于 PR 曲线是由一系列的准确率和召回率对所组成，反映的是算法性能随着不同阈值而动态变化的过程。并且由于 PR 曲线中包含了两个相互制约的评估指标，难以直观地反映出算法的优劣。基于上述不足，Achanta 等人[21] 提出了一种基于准确率和召回率的加权调和平均的综合性评价指标，即特征相似度 F-measure：

$$F_\beta = \frac{(1+\beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recall}. \tag{2.2}$$

β 是一个用来控制准确率和召回率对 F-measure 影响的可调系数。根据大量研究者的实践[68]，通常设定 $\beta^2 = 0.3$ 来强调准确率相对于召回率对预测性能的重要性。F-measure 的值越接近于 1，代表算法的性能越好。由于准确率和召回率会随着阈值 t 的变化而变化，所以完整的 F-measure 由 256 个值组成。一些文献采用最大的 F-measure 值作为最终的衡量指标，也有文献采用平均 F-measure 值，即根据不同的显著性预测图自适应确定阈值（一般为显著性预测图内的像素均值的两倍）而得到的 F-measure 值。由于最大 F-measure 值有着更好的代表性和鲁棒性，本文中的 F-measure 值默认均为最大 F-measure 值。

3. **平均绝对误差（Mean Absolute Error, MAE）**通过上述介绍，可以看出 PR 曲线和 F-measure 两个指标的计算都只考虑了显著性目标，而忽略了预测错误的像素或者背景区域。而在实际应用中，有时需要从准确预测背景区域的角度来衡量一个算法的性能。基于此，将所有像素的预测精度均纳入考虑的评测指标——平均绝对误差被提出。MAE 通过平均预测的显著性图像中所有像素和真值图像中对应像素之间的绝对误差而得

到：

$$MAE = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |P(x,y) - G(x,y)|, \tag{2.3}$$

W 和 H 分别代表显著性预测图 P 的宽和高。MAE 的取值范围为 $[0,1]$。当 MAE 的值越接近于 0，代表预测图和真值图越接近，算法的效果越好。

4. **结构相似度（S-measure, S_α）**通过公式可以看出，上述三个评价指标都是由逐像素计算得到，而没有考虑到显著性目标中局部和整体上的结构信息。作为一项视觉任务，准确预测物体结构细节的能力在很多应用中有着举足轻重的作用。受图像质量评估 (Image Quality Assessment) 领域的启发，Fan 等人[69] 提出了一个可以量化算法的显著性预测图和真值图之间结构相似性的评价指标 S-measure。S-measure 主要反映了预测图和真值图之间关于物体相关（S_o）和区域相关（S_r）的两种结构相似性：

$$S_\alpha = \gamma S_o + (1-\gamma) S_r, \tag{2.4}$$

其中，平衡常数 $\gamma \in [0,1]$，一般默认设置为 $\gamma = 0.5$ 来同时兼顾区域和物体两种维度的结构相似性。S-meansure 分数越高表示算法所预测的显著性图与真值图之间结构相似性越大，相应的性能也越好。区域相关的结构相似性度量可以表示为：

$$S_r = \sum_{k=1}^{K} w_k * SSIM(k), \tag{2.5}$$

其中，K 表示图像被划分成的总块数，w_k 表示第 k 块的权重，$SSIM(\cdot)$ 表示图像的结构化相似性指数（Structural Similarity Index）。通过将显著性预测图划分为多个块，S_r 可以有效描述它们与真值图之间在物体部分的结构相似性，但是不能很好描述物体整体层面的结构相似性。而对于显著性目标检测这一任务，完整的物体结构能够有效帮助场景级别的语义信息理解。物体相关的结构相似性度量包括整体的前景和背景区域两个部分，可以分别表示为：

$$O_{FG} = \frac{2\bar{x}_{FG}}{(\bar{x}_{FG})^2 + 1 + 2\lambda * \sigma_{xFG}}, \tag{2.6}$$

$$O_{BG} = \frac{2\bar{x}_{BG}}{(\bar{x}_{BG})^2 + 1 + 2\lambda * \sigma_{xBG}}, \tag{2.7}$$

其中，$\bar{x}_{FG}$ 和 $\bar{x}_{BG}$ 分别表示显著性预测图的前景和背景区域的平均概率值。综上我们可以得到物体相关的结构相似性度量：

$$S_o = \mu * O_{FG} + (1-\mu) * O_{BG}, \tag{2.8}$$

其中 μ 表示真值图中前景区域占全图像所有区域的比值。

第二节 相关的二元任务

2.2.1 RGB-D 显著性目标检测

RGB-D 显著性目标检测任务的目的在于预测输入 RGB-D 模态图像中视觉上最为突出和吸引人类注意的对象或区域。传统的 RGB-D 显著性目标检测方法主要依赖于相关领域专家根据 RGB 和深度图的显著性先验所手工设计一些算子[70, 71]。近年来，随着 CNN 的蓬勃发展，基于深度学习的方法逐渐兴起。早期的深度方法[72, 73] 主要集中于将手动设计的对比特征线索与 CNN 提取的多尺度特征相结合。最近的主流深度方法则均以端到端的形式直接利用原始的输入 RGB-D 图像。Liu 等人[74] 提出通过结合骨干特征提取网络的侧向输出来提升预测性能。Chen 等人[75] 提出以逐步渐进的方式来融合由相互独立的子网络所分别提取的 RGB 和深度特征中的互补信息。Han 等人[76] 通过迁移调整原先专门为 RGB 图像设计的模型结构来融合所提取得到的深度特征表示，以更好地利用深度图中的信息。Zhao 等人[77] 基于对比度先验提出了一个金字塔型的互补信息融合网络，以将增强的深度特征结合进 RGB 特征提取器中。Chen 等人[78] 提出利用跨模态交互的思想来丰富多模态特征的融合方式，以解决 RGB 和深度特征之间融合不充分的问题。他们进一步设计了一种基于注意力感知的三通路网络[79] 来高效地选择 RGB 和深度信息之间互补的表示。Piao 等人[80] 提出利用残差连接来从 RGB 和深度输入流中提取并融合多层次的成对互补线索。Liu 等人[81] 通过集成多类型的选择性自相互注意力策略来更精确地传递上下文信息。Fan 等人[82] 则为 RGB-D 显著性目标检测中的跨模态特征学习扩展了一个基础的通用网络结构和数据集。

2.2.2 边缘检测

边缘检测是计算机视觉中最基础的任务之一。传统方法，如 Canny[83] 等，主要关注于利用图像亮度和颜色梯度。这些早期的工作由于它们的检测结果不准确且嘈杂，基本不能直接被用于解决现实生活中的问题。后来，许多基于信

息论的特征学习方法[84–87] 被提出，它们试图利用各种手工特征从局部和全局两个方面来捕获上下文信息。具有代表性的方法有 Pb[84]、gPb[85]、BEL[86] 和 Structured Edges[87] 等，它们主要利用图像亮度、梯度、纹理等特征来预测边缘强度。尽管这些方法在一些简单的情况下显示出尚可的结果，但由于人工设计的特征的泛化能力有限，它们仍然不能被广泛应用于实际应用之中。得益于 CNN 强大的特征提取能力，基于深度学习的方法近年来在精度和速度上大幅超越传统方法。Ganin 等人[88] 提出了 N^4 域的概念来将 CNN 与最近邻搜索相结合。Shen 等人[89] 将边缘数据划分为多个子类，并通过不同的模型参数来分别拟合和预测各子类。Xie 等人[45] 提出了深度监督的概念，并通过在特征提取网络的每个阶段的最后一个卷积层添加额外的监督信息来将其与 CNN 相结合。Xu 等人[90] 提出了一种分层模型，以鲁棒地处理从不同尺度学习得到的边缘特征的融合过程。Yu 等人[91] 将语义分割的概念扩展到了边缘检测任务中，并提出了一种可以同时检测和识别边缘像素及其语义类别的模型。Wang 等人[92] 提出通过引入一条自顶而下的反向细化通路来学习得到更加细化和清晰的边界。Liu 等人[93] 在 HED[45] 方法的基础上提出了改进，即在每个阶段的所有卷积层的输出求和后再添加侧监督而不仅在最后一个卷积层的输出后面添加监督。

2.2.3 伪装对象检测

与显著性目标检测任务关注于图像中最吸引人类视觉注意的物体或区域不同，伪装对象检测旨在发现隐藏在周围环境中的伪装对象。伪装对象检测在艺术、生物学和动物学等多个领域有着深远的影响。早期伪装对象检测研究[94–97] 可以追溯至数十年前。Cuthill 等人[98] 指出伪装对象的主要特点在于其主体颜色与纹理与背景及其相似，并且它们之间的界限很模糊，视觉上难以辨认。Pike[99] 通过结合伪装对象中最值得关注的特征来模拟捕食者的视角以发现隐藏的目标。Fan 等人[100] 针对本任务构建了一个大规模的基准数据集，以及一个深度基准模型来促进本领域的发展。Li 等人[101] 提出了不确定性感知和矛盾特征的概念并设计了一个相似特征模块以联合学习显著性目标检测和伪装对象检测两个对立的任务。

2.2.4 骨架提取

较早的骨架提取方法[102–104] 主要依靠输入图像的梯度强度图来提取骨架。后来，基于学习的方法一般将骨架提取视为像素分类问题或超像素聚类问题。

例如，Tsogkas 和 Kokkinos[105] 通过在每个像素位置提取手工设计的特征，并在此基础上训练了一个分类器来进行骨架检测。Sironi 等人[106] 通过训练一个回归器来学习尺度空间中最近骨架的距离，然后通过找到局部最大值来识别骨架。Levinshtein[107] 等人使用多尺度超像素和相邻超像素之间学习得到的匹配度来对近似中间点进行分组，然后利用图聚类算法来得到完整的骨架结果。最近的方法主要利用了深度网络的多尺度特征提取能力。例如，Shen 等人[108, 109] 认为在不同阶段可以捕获不同尺度的骨架，提出在 HED[45] 结构的基础上融合与尺度相关的深度侧向输出，并使用尺度相关的真值标注来作为监督。Ke 等人[110] 提出了一个侧输出残差模型，该模型通过由深到浅的方式级联残差单元来拟合真实值和侧输出之间的误差。Zhao 等人[111] 提出了一种分层特征集成机制，该机制通过双向引导将多尺度特征进行分层融合以得到更鲁棒的混合特征。Wang 等人[112] 提出利用 CNN 以一个二维向量场的形式来预测输入图像的通量表达，其中每个像素点被映射到一个最近的候选骨架像素，然后通过定位净向内通量高的终点来恢复骨架预测结果。

第三节 相关的基础操作、结构和机制

2.3.1 池化操作

作为现代 CNN 模型中的关键组成部分，池化操作主要有着两个功能。第一个功能是减少特征图的空间尺寸，从而降低计算代价。第二个功能是增强模型的平移不变性，并改善优化过程中的过拟合问题。一个合适的池化操作应该尽量保留有用的信息，同时滤除掉不相关的细节。大部分现有的池化操作可以被分为基础操作和改进版操作两类。

基础操作：作为最常用的两种池化操作类型，平均池化[113, 114] 和最大值池化[115] 旨在分别选择目标池化窗口内的平均值和最大值作为其输出。对于平均值池化，梯度在反向传播中被均匀地分散到池化窗口中的每个像素。而对于最大值池化，在反向传播中只有池化窗口中值最大的对应像素被更新，其余像素的梯度则被设置为零。Lee 等人[116] 提出了混合和门控两个策略来组合最大值和平均值池化，并进一步引入了更复杂的树形自学习池化策略。

改进操作：除了基于基础的平均值和最大值池化，Toutounchi 等人[117] 为图像超分辨任务提出了一种无损池化操作，可以将单通道特征图下采样为具有较低空间分辨率的多通道特征图，而不会丢失其中的信息。Gao 等人[118] 设计了一

个基于局部重要性的池化操作，它可以学习得到自适应和易判别的重要性图来聚合特征以进行下采样，而不是基于手工设计的特征。Hou 等人[119] 探究了不同池化窗口形状的影响，并提出了一种轻量级的条形池化策略，该策略在垂直和水平方向都采用了长而窄的池化窗口。

2.3.2 U 型结构

如何高效结合从骨干网络中提取的多尺度特征一直是个热门的研究方向。作为先驱工作，U-Net[120] 和 FPNs[121] 最先提出在由分类网络构成的自底向上的特征提取通路上添加额外的自顶向下的特征融合通路，以按照从高层级到低层级的顺序组合所提取到的多尺度特征。作为后续的工作，PANet[122] 提出在 FPN 结构的基础之上再添加一条额外的自底向上通路以获得更充分的多尺度特征融合。而 ASFF[123] 则提出在 FPN 结构的自顶向下通路中通过融合更多阶段的特征以得到更丰富的特征表达。EfficientDet[124] 提出了一个包含自底向上和自顶向下两个通路的双向 FPN（BiFPN）层，并通过将其级联使用多次来提升物体检测的性能。RFP[125] 提出通过将特征以递归的形式重复地通过自底向上的骨干网络来增强 FPN 的表征能力。最近，NAS-FPN[126] 和 Auto-FPN[127] 利用神经网络架构搜索[128] 以数据驱动的方式来自动搜索发现对于目标任务最优的 FPN 结构。

2.3.3 多尺度和注意力模块

PSPNet[129] 提出了一个金字塔池化模块（PPM），该模块包含多条具有不同大小池化核和步长的平均池化操作以高效地聚合多个尺度的上下文信息。Deeplabv2[130] 提出了一个空洞空间金字塔池化（ASPP）模块，该模块利用多个并行的具有不同空洞率的空洞卷积层来捕捉不同感受野下的上下文信息。DenseASPP[131] 提出在 ASPP 的基础通过将较小空洞率支路的输出作为所有比其空洞率更大的支路的输入，以获得更加密集的跨支路连接来获得更丰富的多尺度特征。最近，Auto-Deeplab[132] 证明了可以通过神经架构搜索的方式来自动获得适用于目标任务和数据集的最优的多尺度模块设计。除了利用多尺度信息之外，CBAM[133] 提出了一个串联使用通道注意力和空间注意力的模块来增强模型对于输入特征中不同通道和空间位置之间的关联。OCNet[134] 提出利用自注意力机制来增强 ASPP 模块对于上下文信息的提取能力。DANet[135] 提出了一个双注意力网络，该网络利用两个并列的注意力模块来分别提取输入特征中的位置注意力信息和通道注意力信息，以更好地建模特征中的远程依赖关系。

CCNet[136] 提出了一个十字形的注意力模块来高效捕获输入特征在水平和垂直两个方向上的全局上下文信息。

2.3.4 多任务学习

多任务学习（MTL）在机器学习领域有着悠久的历史[137, 138]，它与迁移学习[139] 和持续学习[140] 也有一些相似之处。随着深度学习的快速发展，最近出现了许多基于 CNN 的 MTL 方法，其中大部分集中在网络架构的设计，或者以平衡不同任务的重要性为目的的损失函数的设计上，或两者兼而有之。就网络架构设计而言，Misra 等人[141] 提出了一种 Cross-Stitch 架构，其中对于每个任务都有一个标准的前馈网络，并使用多个 Cross-Stitch 单元来实现跨任务的特征共享。Doersch 等人[142] 提出在单个共享网络中使用来自不同层级特征的套索正则化组合来联合学习多个自监督任务。Rusu 等人[143] 通过学习一个序列网络模型，以增量的方式在任务之间传输知识信息。Kokkinos[144] 提出了一个 UberNet 以输入图像金字塔的方式，利用多个网络分支分别处理各分辨率的图像，并且每个分辨率分支中均包括多个任务相关的特定层。Liu 等人[145] 提出通过注意力机制来帮助任务相关特征的学习。除了网络结构，Kendall 等人[146] 提出可以根据不同任务预定义的不确定性来修改损失函数以平衡各任务的权重。Chen 等人[147] 提出利用梯度归一化来平衡不同任务随着时间的推移对共享网络的影响。不同的工作也解决了不同的任务组合，例如：多领域图像分类[148]；目标识别、定位和检测[149–151]；姿势估计和动作识别[152–154]；语义类别、表面法线和深度估计[141, 145, 146, 155–157]；人脸和人脸标志检测[158, 159] 等。

2.3.5 门控机制

门控机制最开始是在自然语言和语音处理领域中被引入。最近的一些工作尝试将其应用于各种计算机视觉任务并证明了其有效性。在语义分割任务中，Qi 等人[160] 提出利用模型不同网络层之间的记忆门来学习每个像素在自定义尺度下的特征表示。Takikawa 等人[161] 提出通过模型中像素预测支路的较高层的激活来门控形状支路中较低层的激活，从而有效去除其中的噪声信息。Ding 等人[162] 提出了一种门控求和方案，以在每个空间位置选择性地聚合多尺度特征。Cheng 等人[163] 提出利用 RGB-D 模态的输入信息并设计了一个门控融合层来结合 RGB 和深度特征。Zhu 等人[164] 和 Li 等人[165] 通过门控技术来提升目标检测问题中锚点特征的选择效果。在图像分类任务中，Chen 等人[166] 提出了一个门

控网络以从骨干网络中选择更合适的滤波器，而 Li 等人[167] 设计了一个软门控机制来允许每个神经元自适应地调整其感受野尺寸。Hua 等人[168] 在模型剪枝任务中利用门控机制来去除掉那些不太重要的通道。

第三章 基于高效特征池化和融合的显著性目标检测算法

第一节 引言

显著性目标检测旨在检测得到给定图片中视觉上最明显的物体或区域，并由于其在许多计算机视觉任务中有着重要的应用而广受关注，例如视觉追踪[169]，内容感知的图像裁剪和编辑[170, 171]，图像检索[172]，视频分割[173]，机器人导航[174]，和弱监督语义分割[175, 176]。作为一项基础的视觉任务，显著性目标检测已经逐渐成为了计算机视觉领域不可或缺的一部分，并在更深层次的视觉研究中有着重要意义。近年来，卷积神经网络的出现极大促进了显著性目标检测的发展，这主要得益于其相比传统方法中手工设计的特征算子在提取深层语义和浅层细节等多尺度信息方面更强大的能力。现代 CNN 网络的一个典型特点便是金字塔型的结构，即浅层网络层输出的特征图通常具有更大的空间尺寸，同时包含有精密且精细的细节特征。相比之下，深层网络层更侧重于编码深层语义信息和显著性物体或区域的具体位置信息。各种基于上述观察的新网络结构[44, 52, 177] 在近年来被逐渐提出。在这些方法中，U 型结构[120, 121] 因其巧妙地利用一个单独的自顶向下通路来增强原有的自底向上分类网络，从而简洁有效地生成更加丰富的特征而最受吸引。

尽管上述方法已经达到了不错的效果，但它们其实仍有很大的提升空间。U 型结构一个明显的缺点为，由网络最顶层捕获的全局语义信息在经过自顶向下通路时可能会逐渐被大量浅层的局部细节信息所干扰并稀释，如图 3.1第一行所示。这一缺点削弱了这些方法精准定位并分割出显著性物体各个部分的能力（更多细节见图 3.3）。另一个缺点为，CNN 模型的实际感受野并不随着它层数的加深而成比例增加[129]。这会导致网络的输出层缺少足够的深层语义信息来确定显著性物体的具体位置。为了弥补上述缺点，现有方法或是提出向 U 型结构引入注意力机制[50, 51] ，或是以循环迭代的方式来不断改善特征图的质量[37, 50, 178]，或是利用多尺度特征信息[44, 46, 179]，亦或是向显著图增加额外的监督信息作为约束[46]。

不同于上面提到的方法，本章提出通过探究高效的池化操作在 U 型结构中

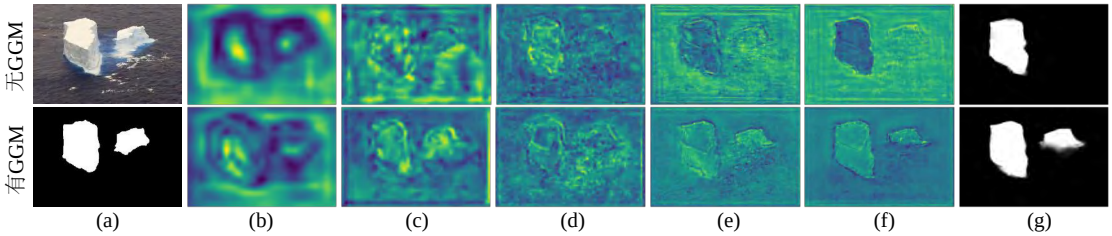


图 3.1 FPN 金字塔结构中不同层级提取得到的特征图。(a) 输入图像和它对应的真值标注；(b-f) FPN 从深到浅各层级获取的特征图；(g) 模型预测结果。从对比中可以观察到深层网络层提取的位置信息在原始的 FPN 中建立金字塔结构时逐渐被稀释（第一行）。然而，当向金字塔结构每一层添加额外的全局引导信息时（第二行），显著性物体的位置信息可以被更好地传达。这一现象在显著性物体相对不那么明显时更为明显（如图像右侧的冰山）。

的潜力来弥补这些缺点。结合上述分析，本章基于两个主要原则来进行网络结构的设计。一方面，包含显著性物体或区域位置信息的深层特征应被传播到 U 型结构所有的金字塔层级，这样深层语义信息才不会被稀释。另一方面，由于 U 型结构中不同金字塔层级的特征图通常有着不同的分辨率，如何无缝融合这些特征来保留检测到的显著性物体或区域的原始形态也很重要。

依据上述设计标准，本章设计的模型在特征金字塔网络（FPN）[121] 的基础上由两个基本模块构成：全局引导模块（Global Guidance Module，GGM）和特征聚合模块（Feature Aggregation Module，FAM）。[1] 如图 3.2所示，GGM 由一个改进版本的金字塔池化模块（Pyramid Pooling Module，PPM）和一系列全局引导流（Global Guiding Flows，GGFs）构成。GGFs 通过向特征金字塔所有层级传播由 PPM 收集的深层语义信息，以弥补 FPN 里自顶向下通路中语义信号会被逐渐稀释的缺点。考虑到 GGFs 中的低分辨率特征图与特征金字塔中各更高分辨率特征图之间的融合问题，本章进一步提出了将融合后的特征图作为输入的 FAM 模块。FAM 首先将融合后的特征图转化到多个尺度特征空间来捕获不同尺度下的局部上下文信息。接着 FAM 将结合这些信息以更好地分配被融合的各尺度输入特征的权重。

虽然本章所设计的 FAMs 在帮助模型获得更加丰富的局部信息方面已经有着不错的表现，但在本章中，作者进一步指出这一优点可以通过简单的结构调整被进一步扩大。不同于原始 FAM 中不同尺寸空间的特征变换被分别且并行地处理，受[180] 的启发，改进版的 FAM+ 显式地在这些并行分支间建立内在联系，

[1] 本章中的 U 型结构默认指特征金字塔网络（FPN）[121]。

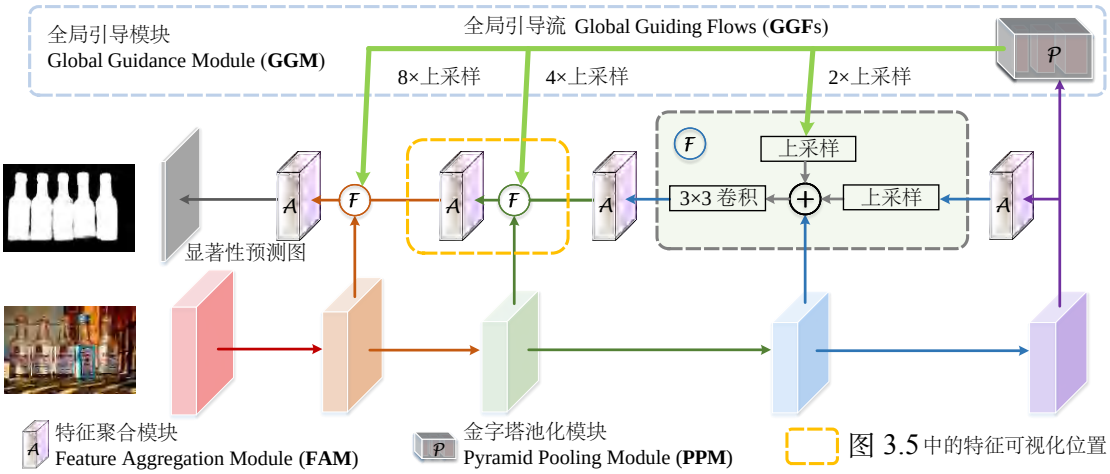
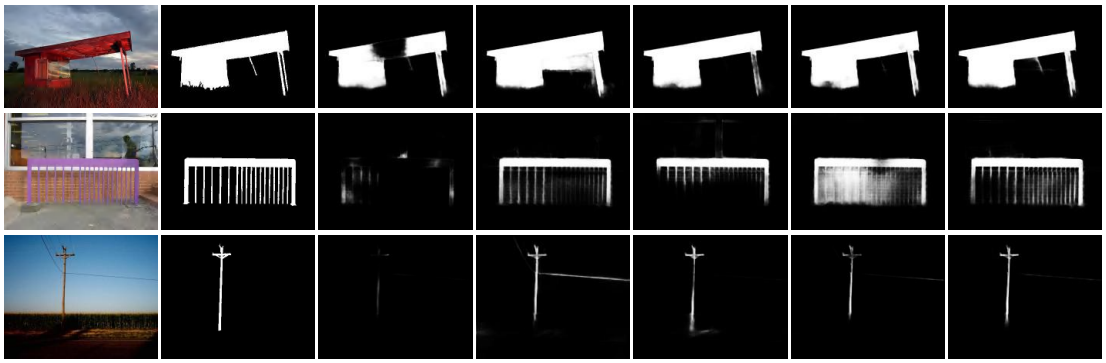


图 3.2 本章所提出模型的整体流程图，包含有位置信息的深层语义特征可被传播到特征金字塔的自顶向下通路中的每个层级。本章利用金字塔池化模块（PPM）来更好地定位显著性物体或区域，并进一步引入全局引导流（GGFs）以将所捕获的位置信息同特征金字塔中的各层级的特征相融合。在每次特征融合之后，一个特征聚合模块（FAM）被紧接着使用以减少混叠效应并丰富细节信息。

使得输出特征的表征能力可以被进一步增强。相比于原始的 FAM，FAM+ 没有引入任何可学习参数，但却极大提升了模型性能。本章将在实验部分给出更多实验数据和相应分析。

由于本章新设计的模块均主要基于池化操作，作者将所设计的基于原始 FAM 和 GGM 的模型命名为 PoolNet。除了在 FAM+ 模块上的改进之外，本章进一步探究了如何在不影响最终预测精度的前提下优化掉 PoolNet 中的冗余部分，进而得到了一个性能更优同时参数更少、速度更快的 PoolNet+（‘+’代表改进版）。

据作者所知，这是第一个研究如何利用高效的池化操作来改进显著性目标检测任务性能的工作。为检验 PoolNet+ 的性能，本章在五个流行的显著性目标检测数据集上进行了详细的对比分析。实验结果表明，PoolNet+ 大幅超越了现有的领先方法。本章也做了一系列消融实验以帮助读者更好地理解 PoolNet+ 中各部分对整体性能的影响。除了良好的检测效果，PoolNet+ 在计算效率上也很有优势。在 NVIDIA RTX-2080Ti GPU 上，PoolNet+ 可以以 53FPS 的速度处理 300×400 分辨率的输入图片。在拥有 10,533 张图片的训练集上训练 PoolNet+ 只需不到 7 小时，大幅快于先前大多数的方法[36, 44, 46, 51, 55, 179]。考虑到移动端设备的应用，本章进一步提供了 PoolNet+ 的一个轻量化版本，名为 PoolNet-M+，其可以以 66FPS 的速度运行，虽然在性能上有着轻微的下降（F-measure 指标上



(a) (b) (c) (d) (e) (f) (g)

图 3.3 本模型关于显著性目标检测任务在不同设置下的视觉效果对比。(a) 输入图像；(b) 真值标注；(c) FPN 基线方法[121]；(d) FPN + FAMs；(e) FPN + PPM；(f) FPN + GGM；(g) FPN + GGM + FAMs。可以看出 GGM 的引入提升了模型准确识别显著性物体位置的能力。而 FAMs 的使用可以进一步提升显著性预测图中细节部分的精细程度。

不到 1%），但参数量（约 3M）和 MAdds（约 1.2G）都大幅减少。这主要得益于只需少量计算资源的各类高效池化操作在网络中的合理利用。PoolNet+ 也展示出了强大的迁移泛化能力，并在边缘检测、RGB-D 显著性目标检测和伪装对象检测等任务上都达到了领先的效果。综上所述，PoolNet+ 可以被当作一个有助于简化未来显著性目标检测研究的基准方法。

本章其余部分组织如下：第二节介绍了 PoolNet 和 PoolNet+，分析了池化操作在它们结构中不同的利用方式，并将 PoolNet+ 拓展为 PoolNet-M+ 这一更轻量的版本。第三节在显著性目标检测任务上对比了本模型和现有领先方法的实验结果。第四节分析和讨论了本模型冗余设计的优化，运行效率和预测错误的例子。第五节将本章方法迁移应用到边缘检测、RGB-D 显著性目标检测、以及伪装对象检测三个任务。最后，第六节总结了本章内容。

第二节 多类型池化网络

文献[41, 44, 52, 178] 指出深层语义特征有助于发现显著性物体或区域的具体位置。与此同时，浅层和中层特征对于将网络深层提取的特征从粗略逐渐提升到精细的过程中也必不可少。基于上述知识，本小节设计了一系列基于高效的池化操作的模块，用于精确获取显著性物体或区域的准确位置，并同时增强融合得到的特征在细节处的精度。

3.2.1 整体流程

本章所提出的模型是基于特征金字塔网络（FPN）[121] 建立。FPN 属于一种经典的 U 型结构，包括一个自顶向下通路和一个自底向上通路。由于其强大的对提取自骨干网络[10, 11] 的多尺度特征的融合能力，这类结构被广泛应用于包括显著性目标检测在内的许多计算机视觉任务。尽管如此，对于显著性目标检测而言，FPN 的一个致命缺点在于其中的深层语义信息是被逐级传播到浅层的，这使得网络深层获取的物体位置信息会逐渐被其他更浅层级中的局部信息稀释并干扰。为了弥补这个缺点，本章提出在 FPN 的自底向上通路上引入了一个全局引导模块（GGM），如图 3.2所示。GGM 提取的深层信息会被显式地聚合进特征金字塔中的每个层级，以达到在不同金字塔层级精确识别显著性对象位置的目的。在 GGM 的引导信息被融合进不同金字塔特征层级后，一个个特征聚合模块（FAM）被分别紧连以确保不同尺度的特征图之间的信息可以被良好地结合。在原始 FAM 结构的基础上，本章进一步发现通过简单的结构调整可以得到一个性能更好、获取局部细节信息能力能强的改进版的 FAM+，而不需要引入任何额外的参数和计算量。下文将详细介绍上述所有模块的具体结构并解释其功能。

3.2.2 全局引导模块

FPN 提供了一种经典的结合从分类骨干网络提取得到的多尺度特征的方式。然而由于 FPN 中的自顶向下通路是构建在自底向上的骨干网络之上，这一结构的主要缺点是深层特征会在传播到浅层时逐渐被稀释。文献[129, 181] 指出 CNNs 的实际感受野远小于其对应的理论设计值，尤其是对于更深的网络层而言。因此，一个依靠理论设计的 CNN 模型的整体感受野并没有大到足以有效获取输入图片的完整全局信息。这一效应对于显著性目标检测模型的直接影响体现在只有部分显著性物体被检测到，如图 3.3c 所示。为缓解 FPN 的自顶向下通路中融合得到的精细特征里深层语义信息的匮乏问题，本章设计了一个全局引导模块（GGM）。如图 3.2所示，该模块包含了一个改进版本的金字塔池化模块[41, 129]，以及一系列全局引导流来显式地使每个金字塔层级的特征图都感知到显著性物体或区域的位置。

具体而言，GGM 中的金字塔池化模块由 4 个并行子分支构成，以获取输入图像不同尺度下的上下文信息。其中第一个和最后一个子分支分别是恒等映射

层和全局平均池化层。对于中间的两个分支，本模型采用自适应平均池化层 [2] 来确保输出特征的空间尺寸分别为 3×3 和 5×5。本模型接下来需要做的是确保金字塔池化模块捕获到的引导信息可以被恰当地融入进 FPN 特征金字塔的自顶向下通路中的不同层级之中。

与现有工作[41] 简单地将金字塔池化模块视作 FPN 的一部分不同，本模型中的金字塔池化模块独立于 FPN。通过增加一系列全局引导流（恒等映射），深层语义信息可以很容易地被传播到不同层级的特征图之中（见图 3.2中的绿色箭头）。上述方式将全局引导信息显式地传播到 FPN 的自顶向下通路的每一特征层级中，确保了位置信息不会在 FPN 的构建过程中被稀释。为更好地说明 GGM 的作用，图 3.3c 中展示了一些由 VGGNet 版本的 FPN 基线模型 [3] 生成的结果的视觉对比。从图中容易看到，只使用 FPN 的基线模型在一些复杂场景中很难准确定位出显著性物体。其中也包含一些只有部分显著性物体被检测出来的例子。但当 GGM 被引入后，显著性预测图的整体质量有了大幅提升，除了边缘部分的细节有少部分缺失。如图 3.3f 所示，显著性物体的准确检测和定位证明了 GGM 的重要性。

3.2.3 特征聚合模块

GGM 的存在使得全局引导信息能够被充分传播到特征金字塔的不同层级。然而，一个随之产生的问题便是如何将 GGM 生成的低分辨率、高语义层级的特征与特征金字塔不同层级的各更高分辨率和更低语义层级的特征进行无缝融合。以 VGGNet 版本的 FPN 为例，其提取得到的特征金字塔 $C = \{C_2, C_3, C_4, C_5\}$ 中各特征图的分辨率分别相对于输入图像有着 $\{2, 4, 8, 16\}$ 的下采样倍率。而在原始的 FPN 的自顶向下通路里，相邻层级之间的特征图之间的分辨率差异均为两倍。在这种情况下，在特征合并操作后添加一个 3×3 的卷积层便可以大幅缓解由上采样操作带来的混叠效应。然而，全局引导流和部分特征层级之间的分辨率差异则需要更大的上采样倍率来对齐（例如 8 倍）。因此，如何高效地弥合全局引导流与各尺度特征图之间的巨大空间和感受野差距至关重要。

基础版本： 本章引入了一系列特征聚合模块（FAMs），每个模块均包含四个子分支。如图 3.4a 所示，输出特征图首先并行经过多个不同下采样倍率的平均

[2] `https://pytorch.org/docs/stable/nn.html#adaptiveavgpool2d`

[3] 同[121] 类似，本章使用 conv2, conv3, conv4, conv5 网络层的输出特征图，分别用 $\{C_2, C_3, C_4, C_5\}$ 表示，来构建基于 VGGNet[10] 的 FPN 网络。$\{C_2, C_3, C_4, C_5\}$ 所对应的通道数分别被设定为 $\{128, 256, 512, 512\}$

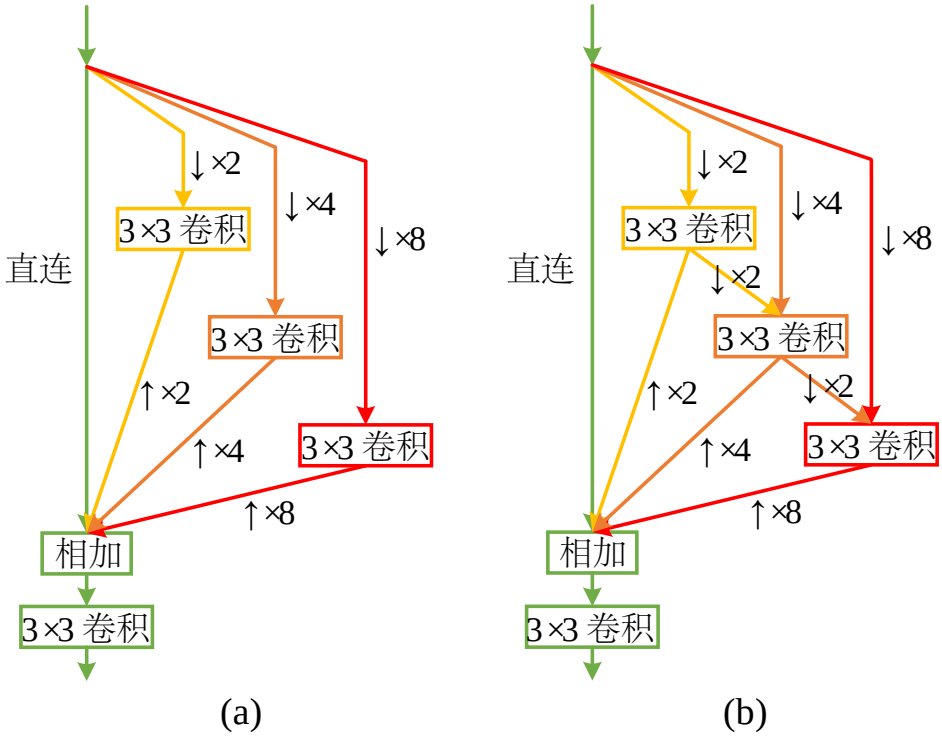


图 3.4 FAM 和它的改进版（FAM+）的具体结构对比。(a) 原始 FAM，它包括四个平行的子分支，每个子分支都在单独的尺度空间中工作。在上采样到同一分辨率后，所有这些子分支通过逐像素相加进行合并，并紧接着传给一个卷积层。(b) 改进版的 FAM+，在 FAM 的基础上引入一系列子分支间的跳跃连接，以显式建立分支之间的信息交流。

值池化层以变换进不同的尺度空间。不同子分支中经过卷积层后的特征再被上采样回原始尺寸后作为残差，再与输入特征图进行融合。融合后的特征会再经过一个 3×3 卷积层作为缓冲。总体而言，FAM 具有两个优势。首先，它能帮助模型缓解上采样操作带来的混叠效应，尤其是对于上采样倍率很大（如 8 倍）的情形。其次，它使得每个空间位置能够从不同尺度空间来获取局部上下文信息，进而增大模型的整体感受野。据作者所知，这是首个指出合理利用池化技术能够有助于减少上采样带来的混叠效应的工作，尤其是在上采样率较大的情况下。

为证明 FAM 的有效性，本小节在图 3.5中对 FAM 附近部分的特征图进行了可视化对比。通过对比左边部分（包含 FAM）和右边部分（不包含 FAM），可以观察到 FAM 之后的特征图（列 a）相比没有 FAM 的情形（列 c）能更好捕获显著性物体。除了中间特征图的可视化，本小节也在图 3.3 中对比了一些不同设置下模型最终预测的显著性结果图。通过对比列 f（不包含 FAM）和列 g（包含 FAM）的结果，容易看出 FAM 的多次利用能使网络预测得到边缘更清晰的显著

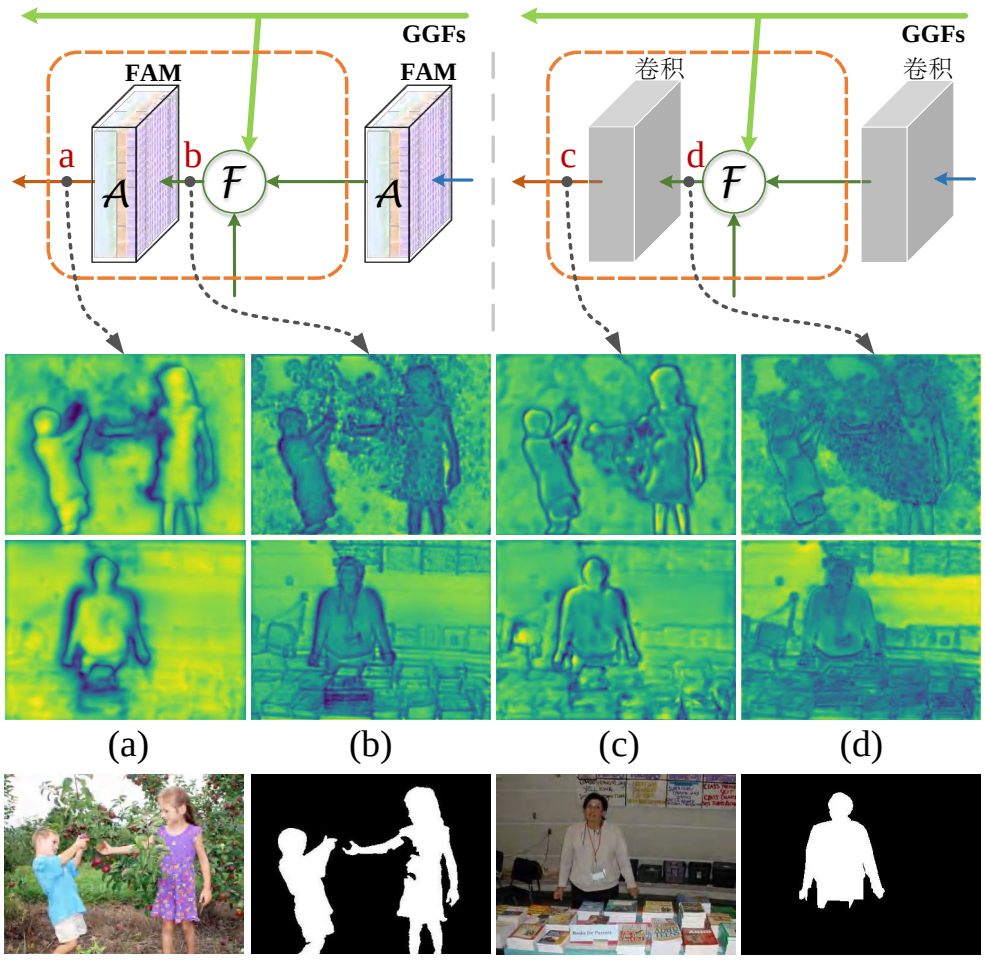


图 3.5 FAM 模块周围特征图的可视化比较。左边为 FAM 生成的特征图，右边是将 FAM 模块替换为两个串联的卷积层所得到的特征图。最后一行是输入图像和对应的真值标注。(a-d) 分别是在不同位置的特征图的可视化结果。可以看到，相比于将 FAM 换成两个卷积层的例子（列 c），使用 FAM 之后的特征图可以更准确获取显著性物体的位置和细节（列 a）。以彩图观看时可以观察到更明显和更清晰的效果。

性物体。这一现象在图 3.3的第二行中尤为明显。上述讨论和分析都体现了 FAM 在更好融合不同尺度特征图的重要能力。在本章的实验部分，将有更多基于具体数值结果的比较和分析。

改进版本： 前部分探究了 FAM 对提升结果细节和模型性能的影响。本部分将证明对 FAM 中子分支之间的依赖关系进行合理建模也有助于显著性图的预测。为同前面提到的 FAM 相区别，本章将改进版的特征聚合模块记作 FAM+。FAM+ 的具体结构见图 3.4（b）。相比原来的 FAM 分别在每个尺度空间中单独进行特征转换，FAM+ 在相邻子分支间新增了一系列跳跃连接（下采样操作）。

更具体地，卷积操作后的更精细层级特征图不仅仅会被上采样然后直接进行特征融合，同时也会被送往较粗糙层级的子分支以进行新的卷积变换。这一设计显式地在相邻子分支间建立了内部交流，使得输出特征具有更丰富的表征力。相比于原来的 FAM，FAM+ 无需任何额外可学习参数，但却达到了更好的性能。本章将在实验部分（第三节）给出更多数值比较结果。

优势分析：本章的特征聚合模块（FAM）提出利用高效的池化操作来弥合局部上下文信息和全局指导信息之间的巨大空间差距。现有的特征聚合模块通常借助于增大卷积操作中卷积核的尺寸或空洞率来解决这一问题。然而，大尺寸的卷积核意味着更多参数和计算量，大的空洞率则需要更多显存并会降低运行速度。与之相反，FAM 中的池化操作，不但不需要额外可学习参数，还能降低特征图的空间尺寸，使后续卷积操作消耗更少的内存和计算资源。除了高效这一优点，池化操作同时引入了更多平移不变性，进而有效避免了过拟合问题的发生。总体而言，如图 3.5所示，FAM 可以减少上采样操作带来的混叠效应，。它也能增大网络的整体感受野，以获取更精确的位置信息与并达到更好的性能，如图 3.3和表格 3.1所示。本章还通过显式构建相邻子分支间的内部连接以改进 FAM，并提出了能生成更丰富的特征表示的 FAM+。相比于 FAM，FAM+ 不需要额外可学习参数，却大幅提升了性能。

3.2.4 面向移动设备的轻量化 PoolNet-M

由于骨干网络需要大量的参数和计算量（相比于轻量模型[182]），因此想要直接将 PoolNet 应用于移动端会比较困难。在许多真实的应用场景中，如移动手机和机器人等，设计在算力受限的平台上计算性能高效的检测和分割算法十分重要。作为一个可选项，本小节通过重新权衡效率和精度提出了 PoolNet 的一个轻量化版本，简称为 PoolNet-M。

本小节以著名且成功的 MobileNetV2[182] 网络作为一个轻量骨干网络的例子来重新设计所提出 PoolNet-M。还值得一提的是，任何其他轻量级的分类网络也可以被拿来使用。MobileNetV2 包含一个具有 32 个滤波器的初始标准卷积层，随后为被分为七个层级的 19 个反向残差模块，和一个用于最终分类的全连接层。为了使原始的 MobileNetV2 更适合于显著性目标检测，本小节首先移除了其最后的全连接分类层，并将第六层级中的 3×3 卷积层的步进改为 1，最后两个层级中的 3×3 卷积层的空洞率增加为 2 以扩大感受野。PoolNet-M 通过使用第 $\{1, 2, 3, 5, 7\}$ 层级中的最后一层的输出特征图来建立 FPN，它们分别具有 {2,

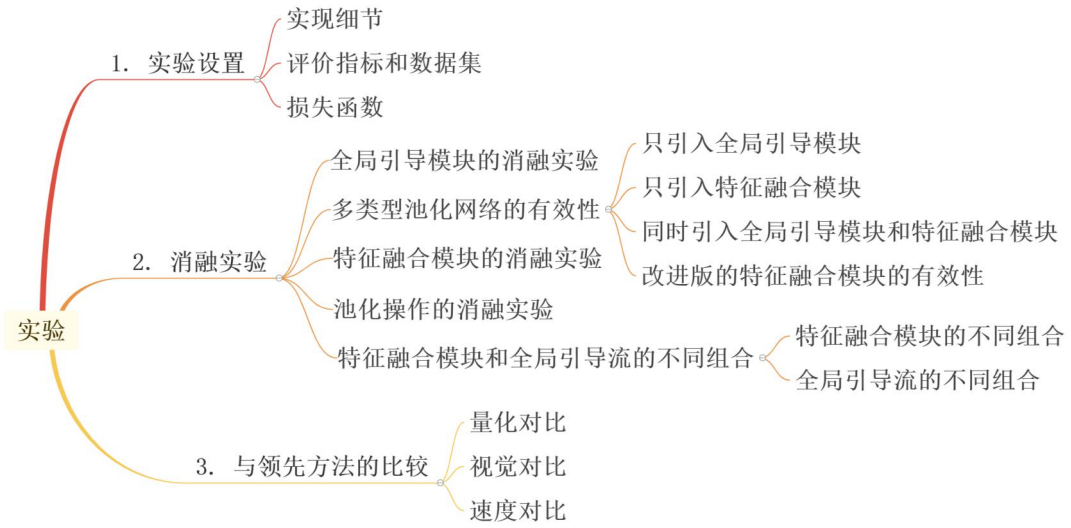


图 3.6 本章实验部分的总体实验方案导图。

4, 8, 16, 16} 倍的下采样率。考虑到计算消耗，五个输出通道数分别为 {12, 16, 24, 36, 72} 的 1×1 卷积层被分别与上述输出特征相连以达到通道削减的目的。

如果直接在 FAM，FAM+，或着金字塔池化模块中使用常规的 3×3 卷积会引入很多可学习参数。而深度可分离卷积[182] 通过巧妙结合深度卷积和逐点卷积来利用它们分别建立空间和通道间关联的能力缓解了这一问题。为进一步减少计算量并同时保持良好的性能，MobileNetV2 提出了逆残差模块，该模块减少了 1×1 逐点卷积的通道数，但是增加了 3×3 深度瓶颈模块的通道数。相似地，为使模型更轻量，本小节将金字塔池化模块和 FAM （或 FAM+）中的 3×3 卷积层均替换为逆残差模块。每个特征层级中逆残差模块的输入和输出通道数被分别设定为 {12, 16, 24, 36, 72}，所有模块的膨胀率均设为 3。本章将在第三节展示新设计的 PoolNet-M （PoolNet-M+）可以达到同现有领先方法相当的性能的同时拥有更少的可学习参数和计算量。

第三节 实验

本小节的总体实验方案设计如图 3.6所示。本小节首先介绍了实验设置，包括实现细节，使用的数据集，以及评价指标。接着进行了一系列消融实验来比较所提出模型的每个部分对性能的影响。最后展示了所提出方法在不同设置下的性能，并与现有的领先方法作了对比。

3.3.1 实验设置

实现细节：本模型主要基于 PyTorch 开源库[4]实现。所有实验均在一台装有 Intel Xeon 12-core CPU （3.6GHz）, 64GB 内存, 和一块 NVIDIA RTX-2080Ti 显卡的工作站上进行。本模型中骨干网络的参数（例如 VGG-16[10]，ResNet-50[11], MobileNetV2[182] 等）均使用了在 ImageNet[9] 上预训练过的对应模型的参数进行初始化，而其余参数均使用随机初始化。本章所有基于重型骨干网络的模型总共训练 36 个周期，初始学习率设为 5e-5，并在 27 个周期后除以 10。与之相比，轻量网络共训练 60 个周期，初始学习率为 1e-4，并在 50 个周期后除以 10。所有实验均使用权重衰减为 5e-4 的 Adam[183] 优化器，而训练的批大小设为 10。随机旋转和水平翻转被用作为数据增强。在训练和测试阶段，输入图像均被放缩为 384×384 分辨率大小。

评价指标和数据集：参考本领域通常的做法，本章的所有实验均使用 DUTS-TR[63] 数据集进行训练。对于性能评估，本章使用四个广泛使用的评价指标在五个流行数据集上进行性能评估和对比。评价指标包括：准确率-召回率（PR）曲线，特征相似度（F-measure，F_β）、结构相似度（S-measure，S_α）和平均绝对误差（MAE），它们的具体概念和计算方式可以在章节 2.1.4中找到。测评数据集包括：ECSSD[62], PASCAL-S[61], DUT-OMRON[29], HKU-IS[32] 和 DUTS-TE[63]，它们的具体信息可以在章节 2.1.3中找到。

损失函数：本模型使用了二元交叉熵损失（BCE）作为损失函数，其公式如下：

$$loss(S,G) = -\frac{1}{N}\sum_{k=1}^{N}[G_k log(S_k) + (1-G_k) log(1-S_k)], \tag{3.1}$$

S 和 G 分别表示预测的显著性图和对应的真值标注。k 表示像素下标，N 表示 S 中的总像素数。

3.3.2 消融实验

本小节对所提出的各模块的不同设计选项和模型配置进行了对比实验，以更好说明本模型中各组成部分的作用和有效性。除了特殊说明之外，本小节的消融实验默认均基于 VGG-16 骨干网络。本小节在四个颇具挑战性的数据集上对比测试了不同设定下的模型：PASCAL-S，DUT-OMRON，HKU-IS，和 DUTS-TE。

[4]https://pytorch.org

表 3.1 关于 GGM，FAM，和 FAM+ 的消融实验。可以看到，所提出的模型中每一部分都有着重要的作用并影响着最终的精度。尤其是 FAM+ 在多数情形下比原始 FAM 表现更好。每列最佳性能以**粗体**突出显示。

序号	FPN	GGM		FAM/	PASCAL-S[61]			DUT-OMRON[29]			HKU-IS[32]			DUTS-TE[63]		
		PPM	GGFs	FAM+	$F_\beta\uparrow$	MAE↓	$S_m\uparrow$	$F_\beta\uparrow$	MAE↓	$S_m\uparrow$	$F_\beta\uparrow$	MAE↓	$S_m\uparrow$	$F_\beta\uparrow$	MAE↓	$S_m\uparrow$
1	✓			-	0.825	0.090	0.816	0.760	0.071	0.779	0.910	0.041	0.889	0.830	0.054	0.835
2	✓	✓		-	0.848	0.081	0.835	0.796	0.062	0.814	0.917	0.038	0.898	0.856	0.048	0.858
3	✓		✓	-	0.833	0.087	0.823	0.773	0.068	0.791	0.913	0.040	0.892	0.839	0.051	0.844
4	✓	✓	✓	-	0.856	0.075	0.848	0.799	0.062	0.810	0.925	0.038	0.907	0.860	0.047	0.865
5	✓			FAM	0.855	0.080	0.840	0.808	0.061	0.821	0.923	0.039	0.903	0.863	0.049	0.865
6	✓	✓	✓	FAM	0.866	0.075	0.849	0.813	0.060	0.830	0.927	0.036	0.908	0.870	0.045	0.871
7	✓			FAM+	0.856	0.080	0.841	**0.817**	**0.059**	0.826	0.925	0.036	0.907	0.872	0.045	0.869
8	✓	✓	✓	FAM+	**0.872**	**0.070**	**0.857**	**0.817**	**0.059**	**0.832**	**0.931**	**0.035**	**0.914**	**0.878**	**0.043**	**0.880**

3.3.2.1 多类型池化网络的有效性

本小节中除了 GGM 和 FAM（或 FAM+s）模块在不同组合形式上的差异之外，其余网络配置均保持一致。

只引入全局引导模块：如表格 3.1中第四行所示，在 FPN 基线模型上引入 GGM 可以在四个数据集上获得 F-measure, MAE, 以及 S-measure 所有三个指标上的性能提升。GGM 生成的全局引导信息使得模型能够更好地关注显著性物体的整体性，大幅提升显著性预测图的质量。如表格 3.1所示，仅通过向 FPN 添加 GGM 就可以在 DUT-OMRON 数据集的 F-measure 指标上有大约 4% 的提升（0.799 对比 0.760），在 S-measure 指标上也有超过 3% 的提升。而在其余三个数据集上也有类似的提升效果。视觉上，在图 3.3（列 f 对比列 c）中可以看出 GGM 的使用有助于更准确地发现显著性物体的位置。因此，显著性物体在被准确检测的前提下其细节信息可以被进一步锐化，而对于感受野受限的模型（例如图 3.3末行的列 c）则可能将细节部分错误地预测为背景。

只引入特征融合模块：单纯将 FAM 嵌入到 FPN 基线模型中（表格 3.1第五行对比第一行）便能够在几乎所有四个数据集上获得效果提升。例如与没有 FAM 的模型相比，添加 FAM 可以在 DUT-OMRON 和 DUTS-TE 数据集上的 F-measure 指标分别提升 4.8% 和 3.3%。这是由于相比于没有 FAM 的情形，FAM 中的池化操作也在一定程度上也增大了模型的整体感受野。视觉上，从图 3.3（列 d 对比列 c）中可以看到 FAM 能帮助模型获得更完整的分割结果。FPN 基线模型在增加了 GGM 之后，其对来自不同层级的特征图的融合需求更加迫切。而在上述模型上添加 FAM 后可以获得进一步的性能提升（表格 3.1第六行对比第五行），这表明 FAM 能有效缓解上采样操作所导致的混叠效应。图 3.3中的视觉

结果对比（列 g 对比列 f）也证明了这点。从图中可以看到，FAM 能够增强模型提取显著性物体细节信息的能力。然而在列 f 中，没有 FAM 的模型并未展示出能够清楚定位物体边缘的能力。在缺少 FAM 的情况下，模型深层产生的低分辨率特征经上采样后不能同高分辨率的浅层特征很好地融合，从而导致其中包含不想要的混叠效应和欠佳的边缘质量。

同时引入全局引导模块和特征融合模块：通过向 FPN 基线模型中同时引入 GGM 和 FAM（表格 3.1第六行）所得到的模型在 F-measure, MAE, 以及 S-measure 三个指标上，相比于只包含 GGM 或只包含 FAMs 的模型有进一步的提升。这一现象表明所提出的 GGM 和 FAM 之间是互补的关系。一方面，GGM 使得模型拥有更强的准确定位显著性物体并保持其完整性的能力。另一方面，FAM 能帮助模型改善检测到的显著性物体的细节信息。对比图 3.3中列 g 和列 d，可以看到添加 GGM 能使模型更准确定位显著性区域。对比列 g 和列 f，可以看到同时使用 GGM 和 FAM 的模型能捕获更多显著性物体边缘部分的细节信息。更多视觉结果的比较可以在图 3.11 和图 3.12中找到。

改进版的特征融合模块的有效性：如表格 3.5所示，当同时添加 GGM 和 FAM 时，本章基于 VGG-16 骨干网络的模型便已经超过了现有的领先方法。本小节通过对比实验展示了可以仅仅通过调整原始 FAM 的结构，以在其子分支内引入更多内部交互（如章节 3.2.3），便可以使模型性能进一步提升。表格 3.1展示了将原始 FAM 替换为 FAM+ 的具体数值结果比较。无论是对比第七行和第五行，或者第八行对比第六行，都容易看出无论在有无 GGM 的情况下，FAM+ 在四个数据集上都带来了稳步的性能提升。特别地，在颇具挑战性的 PASCAL-S 和 DUTS-TE 数据集上，对应的 F-measure 和 S-measure 指标都提升了约 1%。这反映出在原始 FAM 各分支间构建信息交互通道对于提升性能很有帮助.

3.3.2.2 全局引导模块的消融实验

本小节做了两个消融实验来帮助更好地理解 GGM 的构造和作用，分别对应于表格 3.1的第二和第三行。本小节首先移除了 GGM 中的金字塔池化模块，并将特征图 C_5 直接连接到 FPN 自顶向下通路里的每个金字塔层级（即只保留全局引导流）。这一改动导致模型在 F-measure 指标上有超过 2% 的性能下降（第三行对比第四行）。进一步地，本小节尝试把金字塔池化模块仅仅当作一个额外的特征提取部分尾随在 C_5 之后，并同时去除掉从金字塔池化模块中除了连接到 C_5 之外的其他所有全局引导流。这一改动后的模型相对于使用完整 GGM 的模

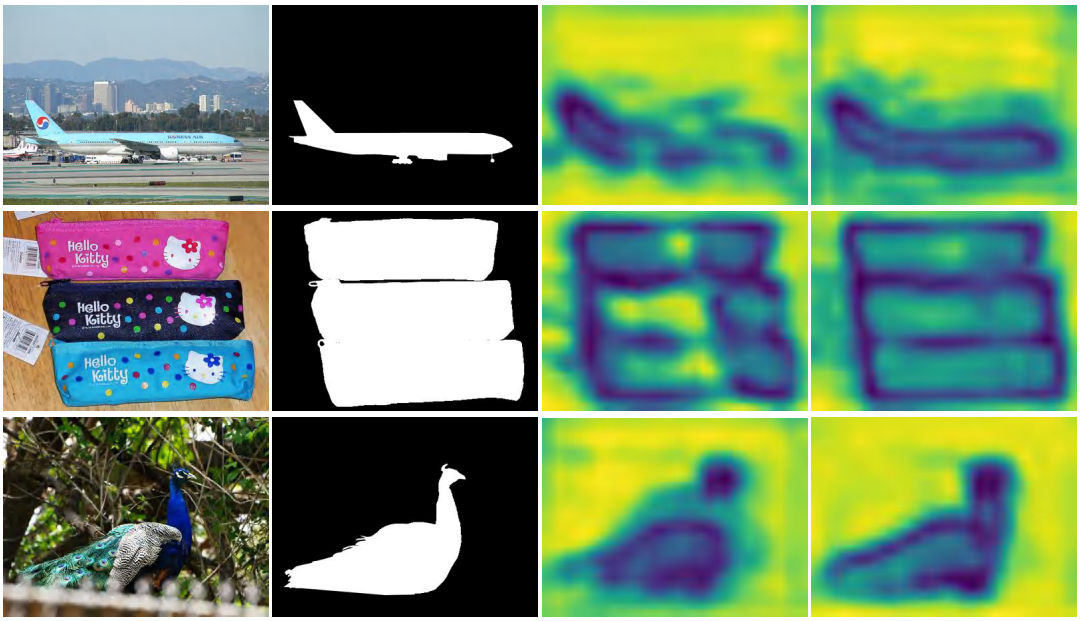


图 3.7 自底向上通路最后一个网络层输出的特征图对比。可以看到，PPM 使模型能够更准确地定位显著性物体，甚至是它们的边缘部分。与之相反，去除 PPM 后模型预测的显著性物体丢失了很多位置信息。这表明 PPM 对于提升模型感受野有着重要作用，并对完整分割出显著性物体有着明显帮助。

型也有性能降低（第二行对比于第四行）。图 3.7中也展示了一些由自底向上通路最末层所输出的特征图的可视化结果。从中可以看出，金字塔池化模块的加入能够明显提升模型整体获取显著性物体位置的能力，并保证其完整性。上述实验表明在 GGM 中，金字塔池化模块和全局引导流都扮演了重要的角色，缺少二者任一都会降低模型的性能。

3.3.2.3 特征融合模块的消融实验

如上所述，FAM+ 能有效缓解上采样操作所导致的混叠效应，尤其是在上采样率很大的情况（例如 4 倍或 8 倍）。同时 FAM+ 也增大了模型的感受野。如图 3.4所示，FAM+ 中默认使用三个不同下采样倍率（即 2，4 和 8 倍）。考虑到 C_5 的分辨率和骨干网络最末层特征图的尺寸，池化层中池化核的最大尺寸被设为 8×8。在上文中已经解释过，FAM+ 旨在融合具有不同分辨率的特征图时平滑它们之间的步进差距。为说明本章在 FAM+ 中所采用的三个下采样尺度的必要性，本小节也进行了一系列消融实验。如表格 3.2 的前三行所示，当逐渐增加 FAM+ 中下采样率较大的池化子分支的数量（恒等映射子分支不变）时，对应的

表 3.2 FAM 各子分支重要性的消融分析。$\mathcal{P}^i$ 代表池化核大小为 $i \times i$，步进为 i 的平均池化，$\mathcal{P}^1$ 代表恒等映射的子分支。每列最佳性能以**粗体**突出显示。

序号	FAM+				PASCAL-S[61]			DUT-OMRON[29]			HKU-IS[32]			DUTS-TE[63]		
	$\mathcal{P}^1$	$\mathcal{P}^2$	$\mathcal{P}^4$	$\mathcal{P}^8$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$
1	✓				0.856	0.075	0.848	0.799	0.062	0.810	0.925	0.038	0.907	0.860	0.047	0.865
2	✓	✓			0.866	0.071	0.853	0.806	0.061	0.819	**0.931**	**0.035**	0.910	0.873	**0.043**	0.874
3	✓	✓	✓		**0.872**	**0.070**	0.855	0.816	**0.059**	0.824	**0.931**	**0.035**	0.913	0.873	**0.043**	0.874
4	✓	✓	✓	✓	**0.872**	**0.070**	**0.857**	**0.817**	**0.059**	**0.832**	**0.931**	**0.035**	**0.914**	**0.878**	**0.043**	**0.880**

表 3.3 在 GGM 和 FAM+ 中使用不同类型的池化操作的影响的消融分析。可以看到，在 GGM 和 FAM+ 均使用平均池化能取得更好的整体性能。每列最佳性能以**粗体**突出显示。

序号	池化类型		PASCAL-S[61]			DUT-OMRON[29]			HKU-IS[32]			DUTS-TE[63]		
	GGM	FAM+	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$
1	最大	最大	0.868	0.069	0.852	0.812	0.061	0.823	**0.931**	**0.035**	**0.915**	0.869	0.046	0.865
2	最大	平均	0.868	0.070	0.857	0.812	0.063	0.818	0.930	**0.035**	**0.915**	0.872	0.045	0.871
3	平均	最大	**0.874**	**0.068**	**0.859**	0.815	0.061	0.821	**0.931**	**0.035**	0.914	0.874	0.044	0.873
4	平均	平均	0.872	0.070	0.857	**0.817**	**0.059**	**0.832**	**0.931**	**0.035**	0.914	**0.878**	**0.043**	**0.880**

模型在四个数据集上的整体性能呈现逐渐提升的趋势。当所有池化子分支均被利用时（表格 3.2 中的最后一行），模型性能可以被最大化。从上述分析中可以得出一个结论，即 FAM+ 中更丰富的下采样率组合通常会带来更好的整体性能和跨数据集的鲁棒性。它也说明了显式建模跨尺度特征表示之间联系的有效性。

3.3.2.4 池化操作的消融实验

池化技术在所提出的方法中发挥着重要作用。本小节探究了不同池化操作及其组合对性能的影响。本小节首先关注两种基本但也是最常见的池化操作类型：平均池化和最大池化。实验中通过将 GGM 中的所有自适应平均池算子替换为自适应最大池算子，以及（或）将 FAM+ 中的所有平均池算子替换为最大池算子，并观察相应的性能变化。表格 3.3中展示了在保持其余模型和实验配置均不变的情况下的量化结果。从第一行和第四行的对比可见，在 GGM 和 FAM+ 中均使用最大池化的模型在四个数据集上的 S-measure 指标上平均下降了 0.7%。而在 DUT-OMRON 和 DUTS-TE 两个数据集上的性能下降尤为明显。仅替换 GGM 或者 FAM+ 中的平均值池化为最大值池化也导致了不同程度的性能下降。通过对比第三行和第二行，可以看到在 GGM 中将最大值池化替换为平均值池化相比于在 FAM+ 中进行类似调整能够带来更多的收益。整体而言，在 GGM 和 FAM+ 中均使用平均值池化获得了最优的整体性能。作者认为上述现象可能是因为与最大池化不同，平均池化在整个池化窗口内的所有位置之间建立联系，从而可以更好地捕获局部上下文信息。

表 3.4 现有更精巧的池化操作在本模型中的效果的消融分析。从表中可以看出，基本的平均池化操作在性能和精度达到了较好的平衡。每列最佳性能以**粗体**突出显示。

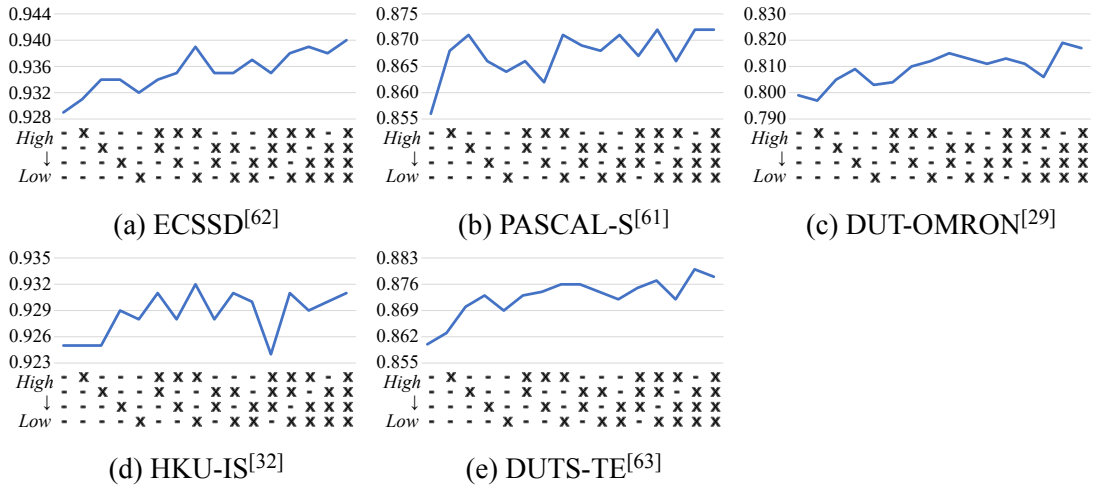
序号	池化类型	速度 (FPS)	PASCAL-S[61]			DUT-OMRON[29]			HKU-IS[32]			DUTS-TE[63]		
			$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$
1	平均	48	0.872	0.070	0.857	0.817	0.059	0.832	0.931	0.035	0.914	0.878	0.043	0.880
2	混合[116]	29	0.872	0.070	0.859	0.816	0.060	0.835	0.932	0.034	0.915	0.873	0.044	0.878
3	门控[116]	31	0.876	**0.068**	**0.862**	0.821	0.060	0.835	0.934	0.034	0.915	0.875	0.045	0.879
4	树状[116]	7	**0.880**	**0.068**	**0.862**	0.818	0.060	0.834	0.934	0.034	0.916	0.878	0.043	0.880
5	Lossless[117]	30	0.852	0.079	0.841	0.792	0.069	0.801	0.922	0.040	0.904	0.857	0.050	0.859
6	LIP[118]	32	0.873	0.070	0.859	0.825	0.059	**0.836**	**0.935**	**0.033**	**0.918**	0.878	0.043	0.881
7	带状[119]	27	0.879	0.069	0.858	**0.830**	**0.056**	0.833	**0.935**	0.034	0.916	**0.885**	**0.041**	**0.882**

在表格 3.4中，本小节也尝试了将现有的各种更为精巧的池化操作结合到所提出的模型当中。表中第二到第四行为基于不同策略来组合平均池化和最大池化的方法。通过将这些方法与第一行对比，可以看到更复杂的结构设计并不一定会带来更优的性能。尽管最复杂的树状池化操作有着略好的性能，却导致了 84% 的计算速度损失。相似现象也发生在采用适应性像素级池化策略的方法中，即表中第五和第六行。同时可以发现，前六个方法中的池化操作中的池化窗口形状和大小一致，因此它们所提供的感受野尺寸也相同。与这些方法不同的是，带状池化[119]（表中最后一行）借助长条形状的池化窗口来进行平均池化，以牺牲部分计算性能的方式来增大感受野。根据上述分析，作者认为在具有相同池化窗口大小的情况下，普通的平均池化便能够胜任并且效率更高。然而，如何选择和调整池化窗口大小以取得精度与计算性能之间更好的平衡仍然有待进一步的探究。

3.3.2.5 特征融合模块和全局引导流的不同组合

如图 3.2 所示，PoolNet+ 中利用 FAM+ 在自顶向下通路中的每个阶段来聚合具有不同感受野的特征图。除此之外，PPM 收集的全局信息通过一系列 GGFs 被引导到上述特征聚合过程中。到目前为止的章节均将所有放置在不同位置的 FAM+ 视为一个整体，对于 GGFs 也是如此。为了了解每个 FAM+（GGF）以及 FAM+s（GGFs）的不同组合如何影响模型性能并分析它们在不同数据集上的一致性，本小节对 FAM+s（GGFs）进行了解耦并进行了一系列消融实验。

特征融合模块的不同组合：当基于 VGG-16 骨干网络时，里面有四个合适的位置来放置 FAM+，这样共会产生 16 种不同组合。为了更好地说明，本小节在图 3.8 中绘制了 FAM+s 的不同组合与相应的模型在五个数据集上的 F-measure 分数之间的关系。大多数数据集上的总体趋势是更多的 FAM+s 具有更好的平均



(a) ECSSD[62] (b) PASCAL-S[61] (c) DUT-OMRON[29]

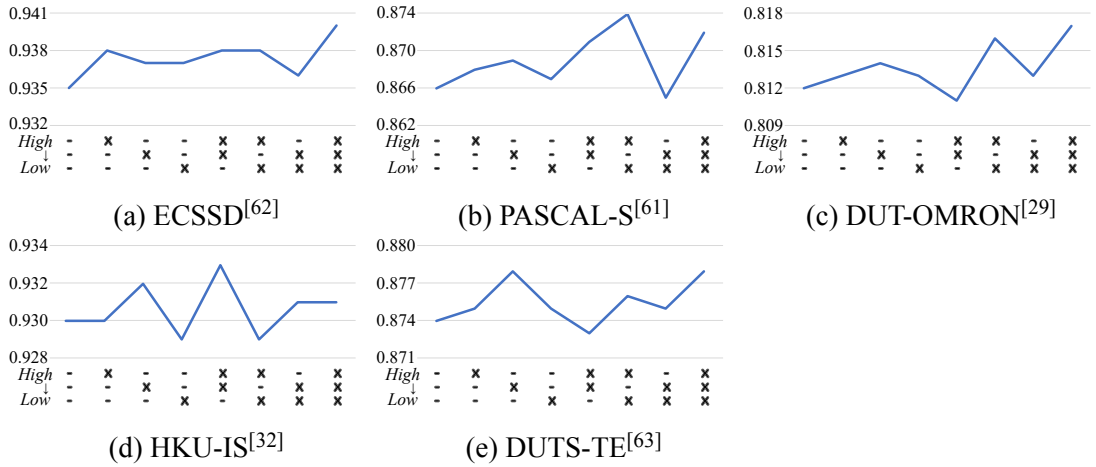
(d) HKU-IS[32] (e) DUTS-TE[63]

图 3.8 FAM+s 的不同组合对性能影响的消融分析。在每个子图中：竖轴表示 F-measure 值，横轴表示位于不同模型位置的 FAM+s 的组合。横轴中每个'x' 代表一个 FAM+，每个'-' 代表两个串联的 3×3 卷积层。模型中一共有从深层到浅层四个不同位置。

性能。同时还可以观察到，当只有一个 FAM+ 时，最好的选择是将其放在两个中间阶段中的任何一个上，而不是放在最高或最低阶段。有趣的是，如果有两个 FAM+，更合适的解决方案却是将它们分别放在最高和最低阶段。

如果在数据集之间横向对比，可以观察到 ECSSD 和 HKU-IS 数据集对最低阶段 FAM+ 的缺失更敏感，而 PASCAL-S、DUT-OMRON 和 DUTS-TE 数据集则对第二高的阶段更敏感。作者将其归因于 ECSSD 和 HKU-IS 数据集中的大多数样本只有一个小的显著性对象。在这种情况下，缩小局部上下文信息和全局引导信息之间的差距至关重要，因为后者可能会由于大尺度下采样而丢失显著性对象的位置信息。相反，PASCAL-S、DUT-OMRON 和 DUTS-TE 数据集的分布更贴近现实世界，因而包含更多大尺寸的样本。在这种情况下，更高阶段的 FAM+ 可以更有效地扩大模型的整体感受野，帮助其更好地定位显著性对象。

全局引导流的不同组合：对于 GGFs 而言，模型中一共有三个合适的位置用作连接，总共对应着八个可能的情况。本小节在图 3.9中绘制了 GGFs 的不同组合与相应的模型在五个数据集上的 F-measure 分数之间的关系。从曲线中可以看出，当不引入或只引入一个 GGF 时，相应模型在 ECSSD、PASCAL-S 和 DUT-OMRON 数据集上的性能大致相同。在大多数情况下，如果从模型中删除三个 GGF 中的任何一个，则性能或多或少会下降。尤其是当移除最高阶段的 GGF 时，模型在四个数据集的性能急剧下降。如果在数据集之间横向对比，可以观察到 ECSSD 和 PASCAL-S 数据集对最高阶段 GGF 的缺失最敏感，而



(a) ECSSD[62] (b) PASCAL-S[61] (c) DUT-OMRON[29]

(d) HKU-IS[32] (e) DUTS-TE[63]

图 3.9 GGFs 的不同组合对性能影响的消融分析。在每个子图中：竖轴表示 F-measure 值，横轴展示连接至模型不同层级位置的 GGFs 的组合。横轴中每个'x' 表示一个 GGF 连接，每个'-' 表示没有相应连接。模型中一共有从深层到浅层三个层级位置的连接。

DUT-OMRON 和 DUTS-TE 数据集则对最低阶段 GGF 的缺失更敏感。而 HKU-IS 数据集更容易受到中间阶段 GGF 的影响。上述现象表明，全局信息的消退问题在不同数据集上有着不同的表现形式。

总体而言，在模型中保留所有的 FAM+s 和 GGFs 能在所有数据集上获得最稳定和最鲁棒的性能表现。作者希望上述分析能帮助研究人员在针对多样分布的数据集设计网络结构时带来更多经验。

3.3.3 与领先方法的比较

本小节将 PoolNet+ 同 19 个现有领先的，和最近提出的实时的显著性目标检测方法进行了对比。为了确保比较的公平，其他方法用来作为对比的显著性图均由其作者所开源的原始代码和设置所生成，或者直接由其作者所提供。此外，所有结果都是直接从单模型测试中获得，而不依赖于任何其他预处理或后处理过程。所有结果的测评方法、代码和环境均是一样的。

量化对比：本段将本方法同现有领先方法进行了定量对比。详细结果可以在表格 3.5中找到。该表同时展示了本模型基于 VGG-16[10]、ResNet-50[11]、和 MobileNetV2[182] 三种骨干网络的结果。从表格 3.5中可以观察到在使用相同骨干网络时，基于原始版本 FAM 的模型（PoolNet-V 和 PoolNet-R）已经在大部分数据集上超过了几乎所有现有的领先方法。具体而言，基于 ResNet-50 骨干网络的 PoolNet-R 在 DUT-OMRON, HKU-IS, 和 DUTS-TE 三个数据集上相比于

表 3.5 本模型与 19 个现有方法在五个流行的显著性目标检测数据集上的量化对比结果。基于不同骨干网络的最优结果分别以**粗体**突出显示。可以看到，本模型在几乎所有数据集和指标上都达到了最好的结果。

方法年份	参数量 (M)	乘加量 (G)	ECSSD[62]			PASCAL-S[61]			DUT-OMRON[29]			HKU-IS[32]			DUTS-TE[63]		
			$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE $\downarrow$	$S_m\uparrow$
VGG-16 骨干网络																	
DCL_{16}[40]	66.25	-	0.896	0.080	0.869	0.805	0.115	0.800	0.733	0.094	0.762	0.893	0.063	0.871	0.786	0.081	0.803
SBF_{17}[184]	93.90	15.90	0.855	0.092	0.830	0.763	0.133	0.758	0.687	0.109	0.748	-	-	-	-	-	-
WSS_{17}[63]	14.70	15.40	0.855	0.106	0.806	0.771	0.140	0.740	0.694	0.110	0.726	0.862	0.079	0.819	0.740	0.099	0.743
DSS_{17}[44]	62.23	52.20	0.906	0.064	0.880	0.821	0.101	0.804	0.760	0.074	0.789	0.900	0.050	0.881	0.813	0.065	0.826
$NLDF_{17}$[46]	35.48	-	0.903	0.065	0.870	0.822	0.098	0.805	0.753	0.079	0.770	0.902	0.048	0.878	0.816	0.065	0.816
$Amulet_{17}$[179]	33.16	20.70	0.911	0.062	0.876	0.826	0.092	0.816	0.737	0.083	0.784	0.889	0.052	0.866	0.773	0.075	0.800
$C2SNet_{18}$[185]	137.05	20.50	0.910	0.055	0.894	0.842	0.082	0.836	0.757	0.072	0.798	0.896	0.048	0.883	0.807	0.062	0.828
$PAGR_{18}$[50]	-	-	0.924	0.064	0.883	0.847	0.089	0.822	0.771	0.071	0.775	0.919	0.047	0.889	0.854	0.055	0.839
RAS_{18}[49]	20.23	15.90	0.918	0.059	0.888	0.829	0.101	0.799	0.786	0.062	0.814	0.913	0.045	0.887	0.831	0.059	0.839
$BMPM_{18}$[55]	-	-	0.926	0.048	0.905	0.854	0.074	0.845	0.793	0.063	0.809	0.922	0.039	0.907	0.854	0.048	0.862
$JDFPR_{19}$[186]	87.61	-	0.925	0.052	0.902	0.854	0.082	0.841	0.802	**0.057**	0.821	0.920	0.039	0.903	0.833	0.058	0.836
$PAGE_{19}$[187]	-	-	0.928	0.046	0.906	0.848	0.076	0.842	0.791	0.062	0.825	0.920	0.036	0.904	0.838	0.051	0.855
$AFNet_{19}$[53]	25.78	-	0.932	0.045	0.907	0.861	**0.070**	0.849	**0.817**	0.058	0.825	0.926	0.036	0.906	0.867	0.045	0.867
PoolNet-V	52.51	48.81	0.935	0.046	0.909	0.866	0.075	0.849	0.813	0.060	0.830	0.927	0.036	0.908	0.870	0.045	0.871
PoolNet-V+	26.31	27.51	**0.940**	**0.044**	**0.914**	**0.872**	**0.070**	**0.857**	**0.817**	0.059	**0.832**	**0.931**	**0.035**	**0.914**	**0.878**	**0.043**	**0.880**
ResNet-50 骨干网络																	
SRM_{17}[41]	53.14	-	0.916	0.056	0.891	0.838	0.084	0.834	0.769	0.069	0.798	0.906	0.046	0.887	0.826	0.058	0.836
$DGRL_{18}$[52]	-	-	0.921	0.043	0.899	0.844	0.072	0.836	0.774	0.062	0.806	0.910	0.036	0.895	0.828	0.049	0.842
$PiCANet_{18}$[51]	47.22	54.06	0.932	0.048	0.912	0.864	0.075	0.854	0.820	0.064	0.830	0.920	0.044	0.904	0.863	0.050	0.868
$ICTB_{19}$[188]	-	-	0.935	0.045	0.912	0.855	0.071	0.850	0.811	0.060	0.837	0.925	0.037	0.909	0.855	0.043	0.865
CPD_{19}[47]	47.85	-	0.936	0.042	0.913	0.859	0.071	0.848	0.796	0.056	0.825	0.925	0.034	0.907	0.865	0.043	0.869
$CSNet_{20}$[189]	36.37	11.75	0.940	0.041	0.914	0.866	0.073	0.851	0.821	**0.055**	0.831	0.930	0.033	0.911	0.881	0.040	0.879
PoolNet-R	68.26	38.19	0.940	0.042	0.914	0.863	0.075	0.849	0.830	**0.055**	0.834	0.934	**0.032**	0.917	0.886	0.040	0.883
PoolNet-R+	34.12	14.03	**0.949**	**0.040**	**0.925**	**0.879**	**0.068**	**0.864**	**0.831**	0.056	**0.842**	**0.941**	0.034	**0.921**	**0.894**	**0.039**	**0.890**
MobileNetV2 骨干网络																	
PoolNet-M	3.00	1.20	0.932	**0.048**	0.902	0.847	0.083	0.835	0.818	0.058	0.821	0.924	0.038	0.902	0.866	**0.046**	0.862
PoolNet-M+	3.00	1.20	**0.938**	**0.048**	**0.909**	**0.864**	**0.078**	**0.844**	**0.830**	**0.056**	**0.830**	**0.930**	**0.037**	**0.909**	**0.872**	**0.046**	**0.868**

领先方法 CSNet[189] 在 F-measure 和 S-measure 两个指标上均获得了提升。基于 VGG-16 骨干网络的 PoolNet-V 也有着优于基于相同骨干网络的 AFNet 的性能。当结合新提出的 FAM+ 时，改进版本的 PoolNet+ 在基于 VGG-16 和 ResNet-50 两种骨干网络时都取得了进一步的性能提升，几乎在所有数据集上都取得了新的领先结果。相比于基于 ResNet-50 的 PoolNet-R+（34.1M 参数量和 14.0G 乘加量），轻量版本的 PoolNet-M+ 仅仅包含 3.0M 参数和 1.2G 乘加量（不到前者的 10%），却仍旧获得了良好的性能表现。此外，如表格 3.5 所示，PoolNet-M+ 在五个数据集上的结果优于如 ICTB[188] 和 CPD[47] 等依赖于 ResNet-50 骨干网络的重型模型的结果。量化结果表明，减少了大量参数和复杂度的 PoolNet-M+ 不仅运行速度非常快，而且比大多数利用更强大的分类网络作为特征提取器的重型模型取得了更好的结果。

除了数值化的结果比较，本段在图 3.10中展示了 PoolNet-V+，PoolNet-R+ 和七个领先的显著性目标检测方法在五个数据集上的 PR 和 F-measure 曲线对比。可以看到，PoolNet-R+ 的 PR 和 F-measure 曲线（实心红线）相比其他所有方法更加突出。即使是使用性能更弱骨干网络的 PoolNet-V+（实心青色）的表现仍同其他方法相当。进一步看，如图 3.10左侧一列所示，随着召回率接近 1，本章方法的准确率远高于其他方法。这种现象表明本章方法预测的显著性图中的假阳率更低，并且检测到的显著性对象更加完整，而这对于显著性目标检测任务至关重要。这一结论也可从图 3.10的右侧一列得出，本章的方法对应的 F-measure 曲线更凸显。

视觉对比：为进一步解释本方法的优势，在图 3.11和图 3.12中展示了一些 PoolNet-R+ 和其他领先方法的视觉结果对比。每个样例图像都与不同的属性相关联，包括透明物体、多个显著性物体、小物体、大物体、复杂场景和低对比度等。这样做的目的是证明本方法可以在不同情况下更好和更鲁棒地工作。可以很容易地看出，本方法不仅检测出了正确的显著性对象，而且在几乎所有情况下都保持了它们的清晰边界。然而，其他方法在处理复杂场景时有时会失败，特别是当显著性对象具有复杂几何结构时（如图 3.12第二行）。这主要是因为所提出的 GGM 可以更精确地定位显著性对象，而 FAM+ 可以更好地融合不同尺度的特征，从而可以很好地捕捉到显著性对象的主体部分和细节。

速度对比：表格 3.6中对比了所提出方法和现有领先的、以及最近实时的显著性目标检测方法的平均运行速度（FPS）对比（在相同环境测试下）。现有最

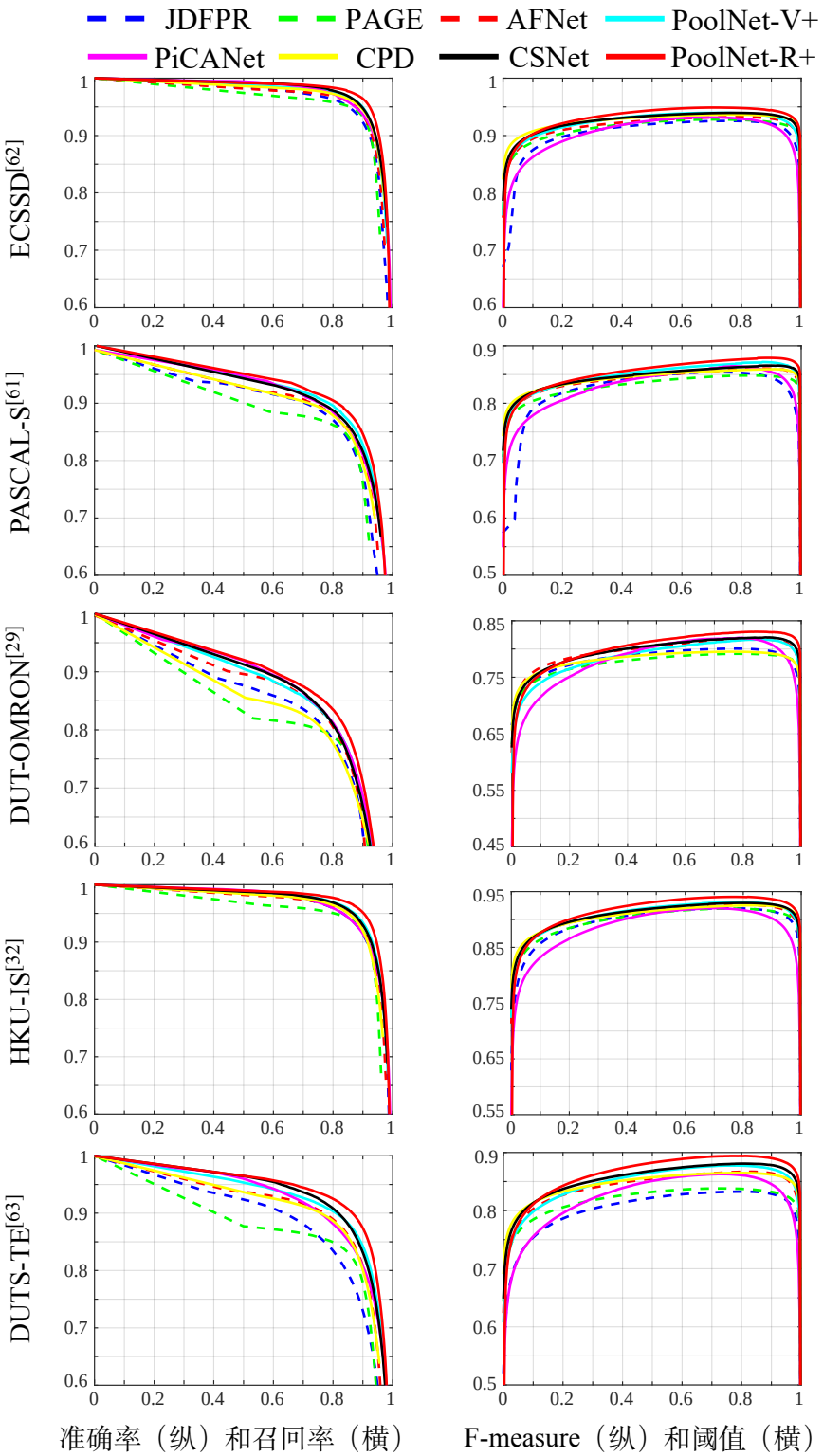


图 3.10 从左至右，每列分别为所提出方法和其他方法在五个流行显著性目标检测数据集上的：准确率和召回率曲线、F-measure（F_β）值和阈值曲线的对比。本表展示了本方法基于 VGG-16 和 ResNet-50 两个骨干网络的结果，分别对应 PoolNet-V+ 和 PoolNet-R+。

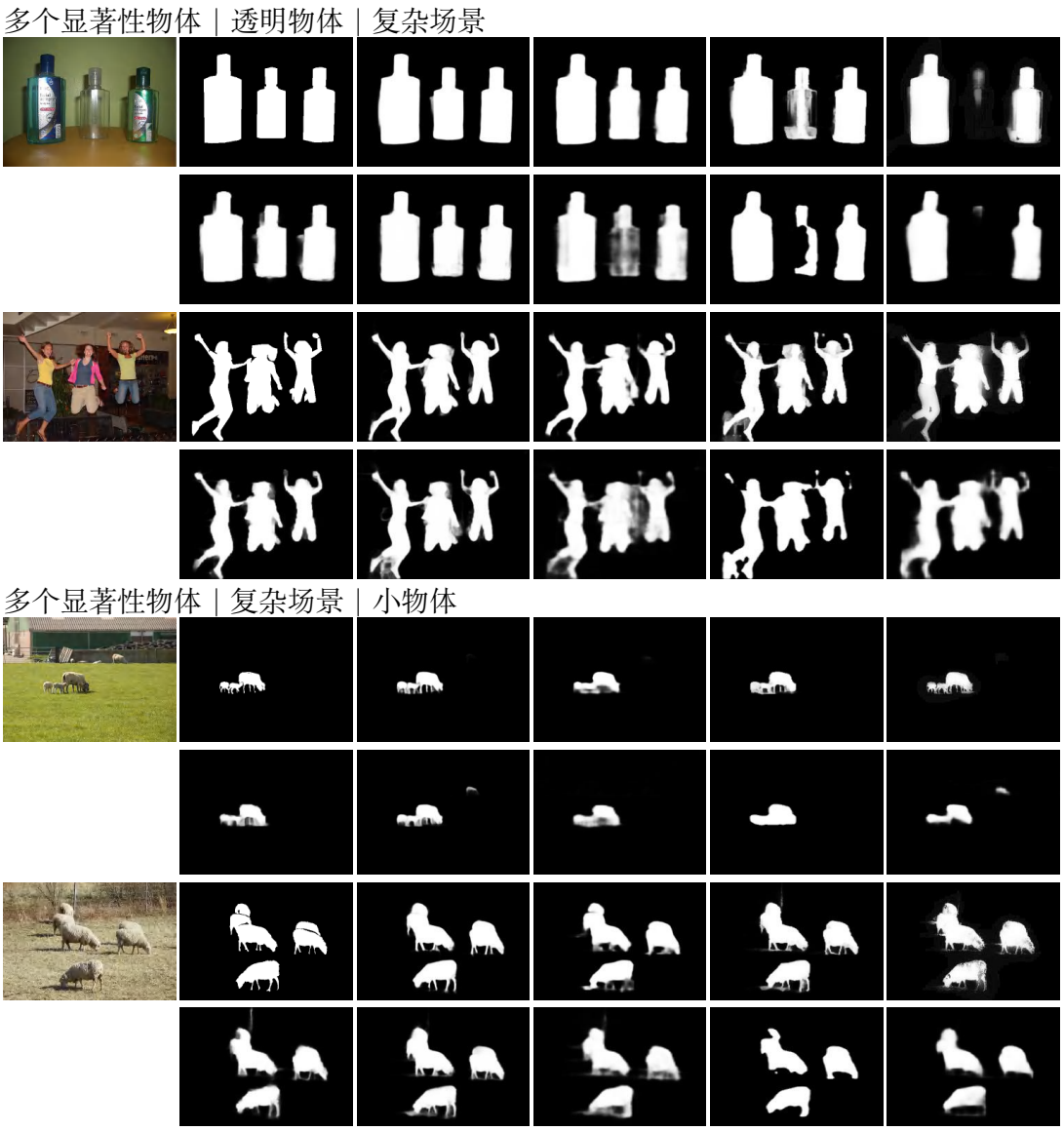


图 3.11 来自不同数据集的显著性图（一）。每组图像对应的标签分别为: 输入图像、标注图像、PoolNet-R+、CPD[47]、AFNet[53]、JDFPR[186]、PAGE[187]、BMPM[55]、PiCA[51]、DGRL[52] 和 SRM[41]。相比于其他方法，所提出的方法不仅能定位出更完整的显著性物体，同时还能获得显著性物体的细节。这使得本方法最终预测的显著性图十分接近真值标注。

快的方法 WSS[63] 在输入图像为 300 × 400 分辨率时，其运行速度为 52 FPS。相比之下，基于重型骨干网络的 PoolNet-R+ 在相同分辨率的图像上进行测试时达到了相当的运行速度（53 FPS）。但是 PoolNet-R+ 在所有五个数据集上的表现都明显优于 WSS。如表格 3.5所示，即使与现有的最佳性能方法（例如 CPD[47]）相比，PoolNet-R+ 在性能和速度方面均获得了更好的结果。表格 3.6 中的数据

图 3.12 来自不同数据集的显著性图（二）。每组图像对应的标签分别为: 输入图像、标注图像、PoolNet-R+、CPD[47]、AFNet[53]、JDFPR[186]、PAGE[187]、BMPM[55]、PiCA[51]、DGRL[52] 和 SRM[41]。相比于其他方法，所提出的方法不仅能定位出更完整的显著性物体，同时还能获得显著性物体的细节。这使得本方法最终预测的显著性图十分接近真值标注。

表明，PoolNetR+ 虽然同部分其他方法相比具有更多的参数，但所需要的计算资源（MAdds）更少，同时运行速度也更快。此外，轻量版本 PoolNet-M+ 的参数量不到 WSS 的 20%，MAdds 不到 RAS 的 10%，在更高效的同时有着明显更好的性能。PoolNet-M+ 在处理分辨率为 300×400 的图像时可以以 66 FPS 的速度运行。这些事实验证了所提出的方法在显著性目标检测上取得了最优结果的同

表 3.6 PoolNet-M+、PoolNet-R+ 与现有领先方法和最近的实时方法的平均运行速度（FPS）对比。得益于高效的池化技术，当具有相似数量的参数和乘加数时，本方法比所有其他方法有着更快的实际运行速度。值得注意的是，PoolNet-M+ 在与其他领先方法取得相当结果的同时只需要不到 10% 的计算资源。

	PoolNet-M+	**PoolNet-R+**	CPD[47]	AFNet[53]	PAGE[187]
输入分辨率	300 × 400	300 × 400	352 × 352	224 × 224	224 × 224
速度（FPS）	66	53	27	31	25
	PiCANet[51]	SRM[41]	Amulet[179]	DGRL[52]	NLDF[46]
输入分辨率	224 × 224	353 × 453	256 × 256	384 × 384	300 × 400
速度（FPS）	8	16	20	8	12
	DSS[44]	RAS[49]	C2SNet[185]	WSS[63]	SBF[184]
输入分辨率	300 × 400	300 × 400	300 × 400	300 × 400	300 × 400
速度（FPS）	12	39	32	52	36

表 3.7 模型之间的参数和乘加量（MAdds）组成的比较。表中以 ResNet-50 骨干网络为例。比较的两个模型包括初始版本的 PoolNet-R 和其对应的计算精简版。

	版本	总计	骨干网络	PPM	GGFs	FAMs	其他部分
参数量 (M)	优化前	68.26	23.51	11.27	5.31	18.14	10.04
	优化后	34.12	23.51	1.31	0.20	6.20	2.90
乘加量 (G)	优化前	37.58	6.24	2.37	12.76	1.65	14.56
	优化后	14.03	6.24	0.22	0.59	0.98	6.00

时运行速度非常快。这主要得益于本章基于池化的设计使得模型后续的操作比以前的方法占用更少的计算成本，因为特征图在空间上被大尺度下采样，从而带来了实质性的改进。

第四节 讨论

3.4.1 优化参数量和乘加量

基于原始 FAM 的 PoolNet 模型（PoolNet-V 和 PoolNet-R）的一个不可忽略的缺点是它们巨大的计算负担。如表格 3.5中所示，PoolNet-V 和 PoolNet-R 的参数量和乘加量（MAdds）都比较大。本小节展示了可以在不牺牲性能的情况下无缝去掉一半以上的计算负担。这一过程主要基于两个观察，1）显著性目标检测是一个浅层的视觉问题，因此不需要极其多样化的特征空间，尤其是在更深的网络层级；2) 过多 3×3 卷积层有时可能会带来冗余[11]。本小节以基于

ResNet-50 骨干网络的 PoolNet-R 作为优化例子。具体而言，计算负担的优化主要包括以下三个部分：

- 在构建特征金字塔时，ResNet-50 的中间阶段输出的特征图的通道数被分别从 $\{128,256,256,512,512\}$ 减少到 $\{128,128,256,256,256\}$。
- PPM 中各池化子分支之后的特征聚合操作从拼接（concat）改成逐像素相加（element-wise summation）。
- FAM+ 最后的卷积层和 GGFs 中所有卷积层的卷积核尺寸从 3×3 缩小为 1×1。

在表格 3.7中列出了 PoolNet-R 在计算负担优化过程之前和之后的参数和 MAdds 的组成。可以看到，优化后的模型分别在参数量和 MAdds 两个指标上减少了 50% 和 63%。其中 PPM 和 GGFs 部分所对应的减少率尤为明显。值得注意的是，表格 3.7 中列出的两个模型的性能基本相同（在 F_β 指标上平均波动 $\sim 0.2\%$）。上述结果表明，通过根据目标任务的特点而精心剪裁网络结构，可以有效降低冗余的计算消耗。

3.4.2 效率分析

本小节通过分解 PoolNet-R+ 的主要组件，并将它们分别与 FPN 基线模型进行比较来分析其效率。不失一般性地，PoolNet-R+ 可以被分解为五个部分：骨干网络、PPM、GGFs、FAM+s 以及其他组成网络的必要组件。本小节构建的 FPN 基线模型包括了骨干网络和其他组成网络的必要组件部分。而 PPM 和 GGFs 模块被排除在外，并且 FAM+s 模块被分别替换为两个串联的 3×3 卷积层。表格 3.8中的第一行代表 FPN 基线模型。通过将第二行和第三行分别和第一行对比，可以看到 PPM 和 GGFs 模块略微增加了参数量，乘加量和推理时延。然而，采用 FAM+s 在引入更多参数时减少了乘加量（第四行对比第一行），表明更多参数并不一定意味着更多的计算复杂度。

与此同时，一个不可避免的问题是：为什么更少的乘加量会导致更大的推理时延（表中第二行对比第三行，第四行对比第一行）？作者认为这与算法在不同平台上的优化和底层实现有关。例如，PPM 和 FAM+ 模块中有多个并行支路，但是这些分支在 PyTorch 上却是串行执行的。在 FAM+ 中，输入特征图首先在空间上进行下采样，然后再进行进一步处理。FAM+ 的核心部分（最后一个 3×3 卷积层之前的操作）甚至比单个 3×3 卷积层需要更少的乘加量。总体而言，PoolNet-R+（最后一行）在引入更多模块的情况下，需要的计算资源理论上

表 3.8 模型各主要部件对整体效率的影响的解耦分析。本表以 ResNet-50 骨干网络为例。不包含 FAM+s（第一至三行）的模型使用了两个串联的 3×3 卷积层作为替代。最后三列中的下标表示与第一行相比的相对变化。所有模型的乘加量和推理时延均是在一块 RTX 2080Ti 显卡上使用分辨率为 $1 \times 3 \times 224 \times 224$ 的输入张量测量得到。

序号	FPN	PPM	GGFs	FAM+s	参数量（M）	乘加量（G）	推理时延（ms）
1	✓				28.47	15.26	15.45
2	✓	✓			$29.79_{+1.32}$	$15.48_{+0.22}$	$16.11_{+0.66}$
3	✓		✓		$28.67_{+0.20}$	$15.84_{+0.58}$	$15.84_{+0.39}$
4	✓			✓	$32.60_{+4.13}$	$13.23_{-2.03}$	$15.96_{+0.51}$
5	✓	✓	✓	✓	$34.12_{+5.65}$	$14.03_{-1.23}$	$16.93_{+1.48}$

相比 FPN 基线模型少 8.1%。PoolNet-R+ 也取得了明显更好的预测性能。综合上述分析，可以合理预期在更多的工程优化工作的帮助下，所提出的 PoolNet+ 在未来能够达到更快的运行速度。

3.4.3 预测错误案例分析

图 3.13展示了一些本方法预测错误的案例。这些预测错误案例可大体被归类为以下四种情形。第一种是由于复杂的背景，如第一组的两行所示。第二种是由于前景和背景之间较低的对比度，如第二组的两行所示。上述两种情况的一个共同缺陷是显著性对象不能被完整地分割出来，即显著性对象的一些细小部分被遗漏了。另一个缺陷是无法检测到显著性对象的主体部分，或者某些非显著性区域被错误地预测为显著性区域。第三种是由于遮挡，如第三组的两行所示。在这类情形中，显著性对象的部分，尤其是被遮挡区域的周围部分不能被完整地提取出来。最后一种预测错误是由于透明物体导致，如最后一组的两行。尽管所提出的方法在大多数情况下可以部分检测出目标透明物体，但仍然难以完整地分割出显著性对象。在上述大多数情况下，即使是人类也很难准确区分出前景和背景之间的界限。

作者认为有三种可能的方法来解决上述问题。首先，一个简单直接且大概率有效的解决方案是扩大训练数据集的规模，因为基于 CNN 的模型从数据集中学习所有知识。如果模型在训练阶段看到足够多的样本，它在测试时面对相似场景则会表现得更好。一个足够庞大，包含尽可能多的场景并具有更接近现实世界的数据分布的训练集总是会有所帮助。其次，引入更多在分割层面的先验知识，以便可以将具有相似颜色或纹理的像素一起检测为一个区域。CNN 的特

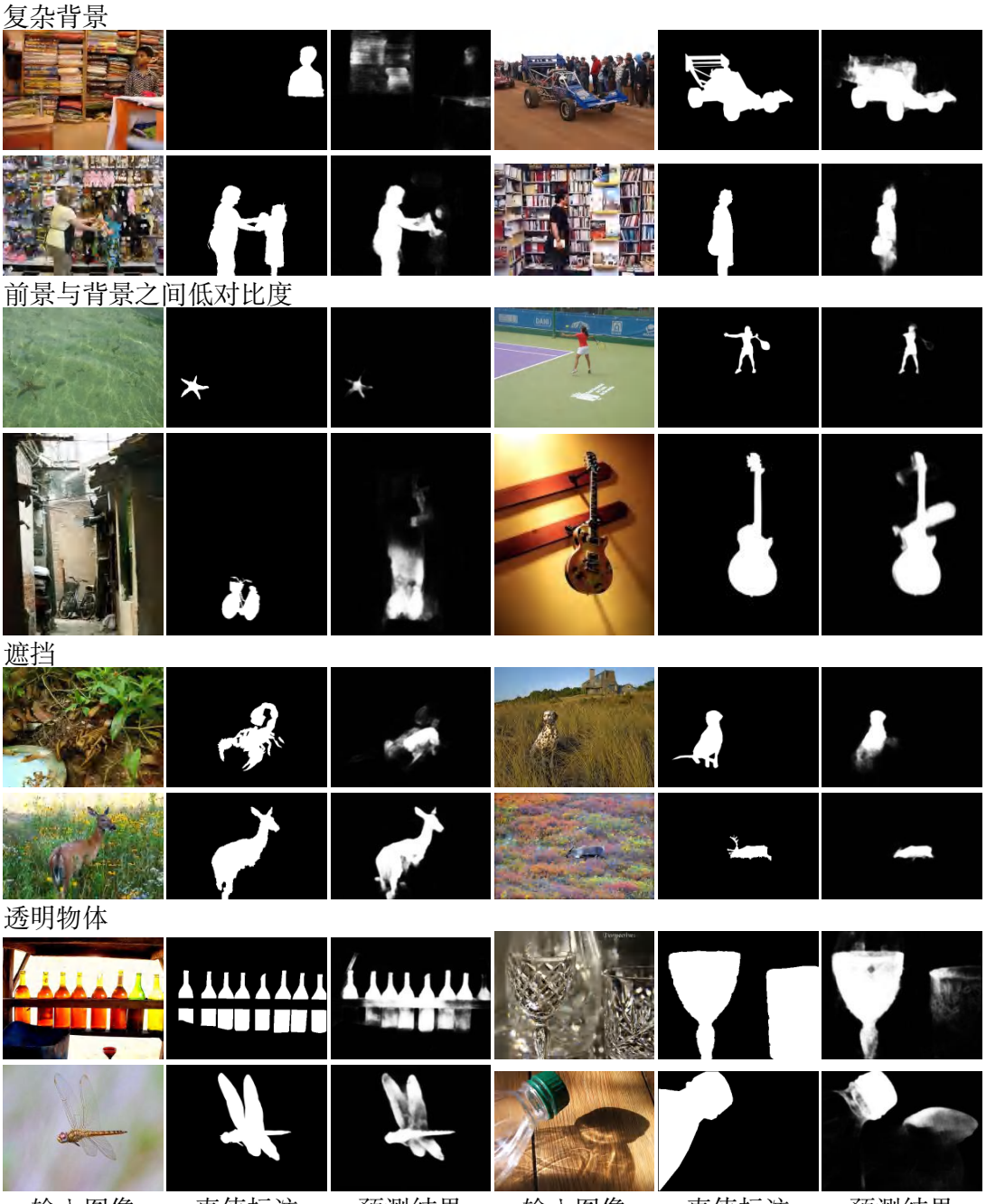


图 3.13 从多个数据集中选取的预测错误案例。这些案例可被分为四种典型情况。

性决定了输入图像是被逐像素处理的，其中可学习的权重决定了预测图中两个位置的相关性。由于相似像素之间是通常是相关联的，所以分割层面的先验可以缓解上面提到的部分预测丢失和模糊的问题。它也可以作为一个后处理步骤来进一步细化所预测的显著性图。设计具有更强大特征提取能力的更先进的模型也是一种解决方案。更多样化和丰富的特征表示通常意味着纠正之前模型的错误预测的可能性更高。

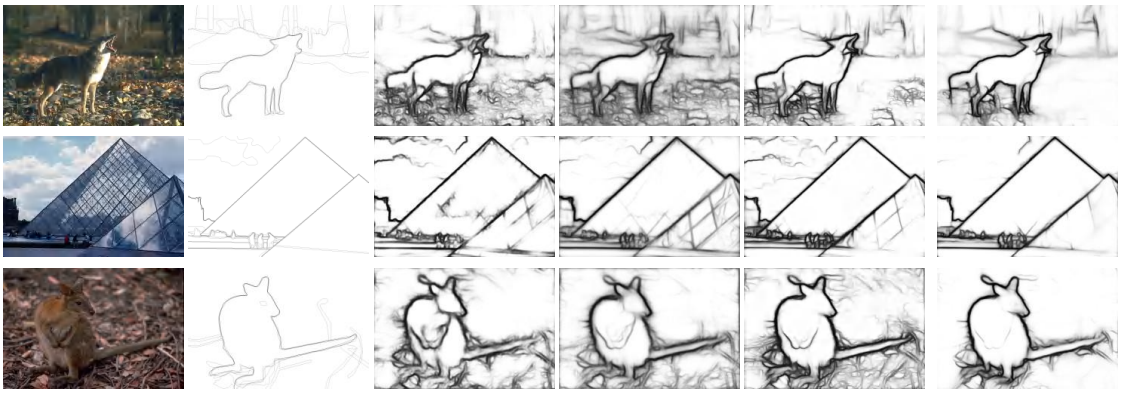
第五节 应用

本节通过将所提出的方法应用到另外三个相关的具有通用属性的浅层视觉任务：边缘检测、RGB-D 显著性目标检测、和伪装物体检测，以探究其泛化能力。

3.5.1 边缘检测

实现细节：遵循边缘检测领域的通常做法[93]，本小节的模型均采用原始大小的输入图像进行训练和测试，并且训练批大小设置为 1，同时也引入了深度监督策略。除了批归一化（BN）层之外，PoolNet-R+ 被不作修改地直接应用到边缘检测任务。具体而言，PoolNet-R+ 中除了骨干网络（ResNet-50）部分之外的批归一化层均被移除，剩下的批归一化层中的参数在训练和测试中均被冻结。骨干网络中的参数使用来自 ImageNet 数据集预训练过的相应模型的权重进行初始化，其余参数为随机初始化。整体训练过程包含 12 个周期，初始学习率设为 5e-5，并在 9 个周期后除以 10。模型使用 Adam[183] 优化器进行优化，其中的权重衰减设置为 5e-4。BSDS 500 数据集[85] 的不同部分被分别用作训练和测试，该数据集包含 200 个训练样本，100 个验证样本和 200 个测试样本，每个样本都包含精确标注的边界。为公平对比并参考通用做法，PASCAL Context 数据集[190] 中样本也被包含进了训练集。同时，算法使用了与[45, 93] 相同的数据增强策略。对于测试而言，最优数据集尺度（Optimal Dataset Scale，ODS）和最优图像尺度（Optimal Image Scale，OIS）被用作为评估标准。在性能评估之前，标准的非极大抑制算法被用来获取细化后的边缘。

与领先方法的对比：表格 3.9中展示了近年来一系列基于 CNN 的领先方法和本方法的定量对比。可以看出，通过简单地将面向显著性目标检测而设计的 PoolNet-R+ 模型应用于边缘检测，所得到的边缘结果优于之前大多数基于 CNN



(a) 输入图像 (b) 真值标注 (c) HED[45] (d) RCF[93] (e) CED[92] (f) PoolNet-R+

图 3.14 同多个现有领先的边缘检测算法的视觉对比。可以看出，与其他三种方法相比，所提出的方法可以生成更清晰的背景并有效捕获较弱的物体边缘。这种现象在第二张图像中尤为明显。图中所有样本图像均来自 BSDS 500 数据集[85]。

表 3.9 本方法同现有领先的边缘检测方法的定量对比。

方法	ODS	OIS
DeepContour[89]	0.756	0.773
HED[45]	0.788	0.808
CEDN[191]	0.788	0.804
RDS[192]	0.792	0.810
COB[193]	0.793	0.820
DCNN+sPb[194]	0.813	0.831
RCF[93]	0.811	0.830
CED[92]	0.815	0.833
PoolNet-R+	0.819	0.834

的模型，甚至可以与最先进的模型相媲美。这意味着 PoolNet 也适用于边缘检测任务。值得一提的是，虽然设计 PoolNet 的目标是提高显著性目标检测的性能，但得到的最终模型也可以产生的很好的边缘预测结果。

本小节也在图 3.14中展示了一些本模型和其他三个流行方法的预测结果的视觉对比，包括 HED[45]，RCF[93] 和 CED[92]。得益于 PoolNet 提取丰富特征的强大能力，即使不对模型进行明显修改，与那些专门面向边缘检测的领先方法相比，本方法在检测物体的真实边界方面仍然表现良好。如图 3.14 中的（f）列所示，因为 GGM 模块引入了更多的全局信息，PoolNet 可以对不是物体真实边界的边缘进行低置信度的预测，而更多地关注于真实的物体边界。作者相信这个特性可以使本方法在实际应用中相比其他方法带来更多帮助。

表 3.10 五个常用 RGB-D 显著性目标检测数据集上的量化结果。可以看出，本方法在所有五个数据集上的 F-measure、MAE 和 S-measure 三个指标上均取得了最好的结果。

方法年份	参数量 (M)	乘加量 (G)	NJU[195]			STERE[196]			DES[197]			NLPR[198]			SIP[199]		
			$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$
DF_{17}[72]	-	-	0.829	0.141	0.763	0.788	0.141	0.757	0.796	0.093	0.752	0.817	0.085	0.802	0.703	0.185	0.653
$CTMF_{18}$[76]	-	-	0.858	0.085	0.849	0.848	0.064	0.875	0.865	0.055	0.863	0.841	0.056	0.860	0.718	0.139	0.715
PCF_{18}[75]	133.40	-	0.888	0.059	0.877	-	-	-	-	-	-	0.888	0.044	0.874	-	-	-
$AFNet_{19}$[200]	-	-	0.805	0.100	0.772	0.848	0.075	0.825	0.775	0.068	0.770	0.816	0.058	0.799	0.756	0.118	0.720
$MMCI_{19}$[78]	-	-	0.869	0.079	0.858	0.877	0.068	0.873	0.838	0.065	0.848	0.841	0.059	0.856	0.839	0.082	0.813
$TANet_{19}$[79]	232.45	-	0.889	0.060	0.878	0.878	0.060	0.871	0.853	0.046	0.858	0.877	0.041	0.886	0.854	0.075	0.835
$CPFP_{19}$[77]	69.50	-	-	-	-	0.891	0.051	0.879	0.883	0.038	0.872	0.892	0.031	0.899	0.873	0.064	0.850
$DMRA_{19}$[80]	59.66	-	0.896	0.051	0.886	0.867	0.066	0.835	0.906	0.035	0.883	0.888	0.031	0.899	0.852	0.085	0.806
$S2MA_{20}$[81]	86.65	108.22	0.899	0.058	0.887	0.895	0.051	0.890	-	-	-	0.912	0.030	0.916	0.893	0.057	0.872
$D3Net_{20}$[82]	45.23	55.17	0.905	0.051	0.893	0.898	0.054	0.889	0.917	0.033	0.898	0.904	0.033	0.905	0.885	0.063	0.864
PoolNet-R+	34.30	14.15	**0.932**	**0.033**	**0.921**	**0.920**	**0.035**	**0.912**	**0.934**	**0.019**	**0.924**	**0.922**	**0.023**	**0.921**	**0.905**	**0.049**	**0.881**

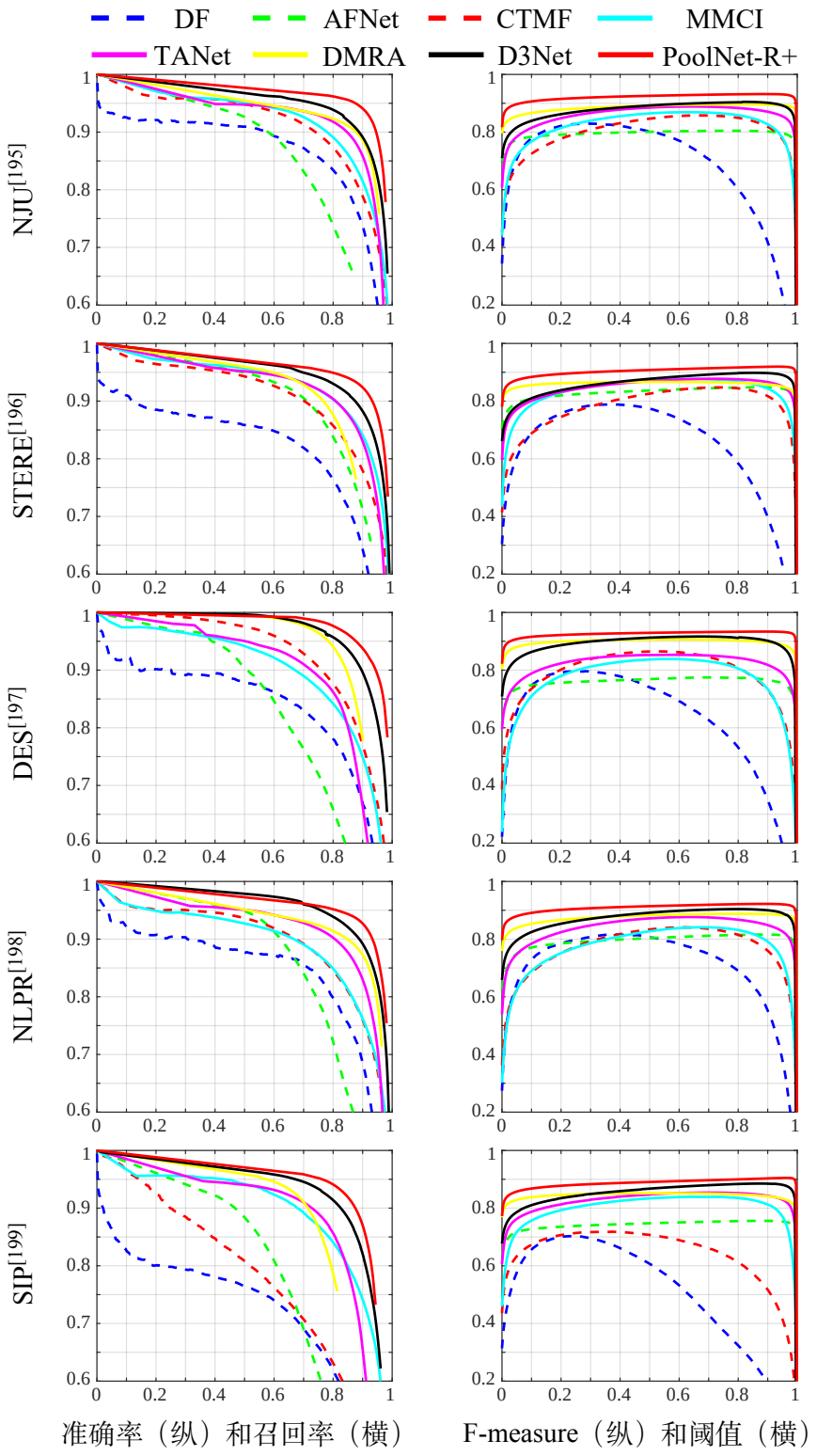


图 3.15 从左至右，每列分别为所提出方法和其他方法在五个流行 RGB-D 显著性目标检测数据集上的：准确率和召回率曲线、F-measure（F_β）值和阈值曲线的对比。

3.5.2 RGB-D 显著性目标检测

实现细节：为满足 RGB-D 显著性目标检测的输入格式：模型同时输入 RGB 图像和对应的深度图，本小节在 PoolNet-R+ 的基础上添加了五个串联的 1×1 卷积层用来提取深度图中的不同层级的深度信息。简便起见，本章依照论文[201, 202] 中做法，直接将上述五个卷积层提取得到的各层级深度信息分别融合进原 PoolNet-R+ 模型中骨干部分的对应的各层级中。本小节使用 Adam[183] 优化器进行优化，初始学习率为 1e-4，批大小为 10。本小节中的模型共训练 200 个周期，学习率每 60 个周期后除以 10。参考本领域通用做法[75, 80, 202]：一个同时包括 NJU[195] 中 1,485 张样本和 NLPR[198] 中 700 张样本的联合数据集被用于训练。NJU 和 NLPR 中除了上述被用作训练样本的其余数据，以及 STERE[196]、DES[197] 和 SIP[199] 共五个数据集被分别用于测试目的。在训练和测试阶段，输入图像的大小都被调整为 352×352 的分辨率。本小节使用与显著性目标检测相同的四个评价指标进行性能评估，包括：准确率-召回率（PR）曲线，特征相似度（F-measure，F_β）、结构相似度（S-measure，S_α）和平均绝对误差（MAE），具体概念和计算方式可以在章节 2.1.4中找到。

与领先方法的对比：表格 3.10中展示了本方法与 10 个现有领先方法的定量对比结果。从表中可以看到本方法在五个数据集的三个测试指标上均超过了其余所有方法。相比现有最好的方法 D3Net[82]，本方法在五个数据集的 F-measure，MAE，和 S-measure 三个指标上平均分别提高了 2.1%，1.5%，和 2.2%。值得一提的是本方法所需要的参数量和乘加量也最少。除了数值结果，本小节也在图 3.15中展示了相应 PR 和 F-measure 曲线对比。可以看到，本方法结果所绘制的曲线（红色）的形状相比其他方法更加凸显。不同于之前大多数方法需要用一个额外的庞大独立网络来提取深度特征，本方法只引入了几个串联的卷积层，即在这种情况下只需要 0.18M 额外参数和 0.12G 额外乘加量。这主要归功于所提出的 PoolNet-R+ 强大的特征提取能力，使得仅仅依靠几个卷积层提取的补充深度特征就足以使模型做出良好的预测。作者认为，设计更强大的深度特征提取分支或更先进的跨 RGB 图和深度图之间的模态特征集成策略可以进一步提升性能。上述现象表明，所提出的方法即使在被迁移应用于从不同域获取输入数据的任务时也能很好地适应和工作。这体现了本方法的泛化能力与鲁棒性。

表 3.11 三个常用伪装对象检测数据集上的量化结果。本方法在几乎所有三个数据集的三个指标上都达到了最优。

方法$_{年份}$	参数量 (M)	乘加量 (G)	CHAMELEON[203]			CAMO[204]			COD10K[100]		
			$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$	$F_\beta\uparrow$	MAE$\downarrow$	$S_m\uparrow$
FPN$_{17}$[121]	36.32	16.84	0.749	0.075	0.794	0.681	0.131	0.684	0.600	0.075	0.697
MaskRCNN$_{17}$[151]	-	-	0.647	0.099	0.643	0.567	0.151	0.574	0.511	0.080	0.613
PSPNet$_{17}$[129]	46.71	37.47	0.747	0.085	0.773	0.656	0.139	0.663	0.575	0.080	0.678
UNet++$_{18}$[205]	36.63	105.70	0.632	0.094	0.695	0.557	0.149	0.599	0.499	0.086	0.623
PiCANet$_{18}$[51]	47.22	54.06	0.777	0.085	0.769	0.596	0.156	0.609	0.587	0.090	0.649
MSRCNN$_{19}$[206]	-	-	0.671	0.091	0.637	0.653	0.133	0.617	0.605	0.073	0.641
BASNet$_{19}$[54]	87.06	97.48	0.633	0.118	0.687	0.584	0.159	0.618	0.504	0.105	0.634
PFANet$_{19}$[48]	16.29	27.82	0.602	0.144	0.679	0.631	0.172	0.659	0.549	0.128	0.636
HTC$_{19}$[207]	-	-	0.502	0.129	0.517	0.432	0.172	0.476	0.505	0.088	0.548
CPD$_{19}$[47]	47.85	-	0.824	0.052	0.853	0.724	0.115	0.726	0.669	0.059	0.747
EGNet$_{19}$[208]	111.66	120.85	**0.830**	0.050	0.848	0.733	0.104	0.732	0.683	0.056	0.737
PoolNet-R+	34.12	14.03	0.828	**0.042**	0.838	**0.761**	**0.095**	**0.750**	**0.718**	**0.049**	**0.754**

3.5.3 伪装对象检测

实现细节：与显著性目标检测任务相似，伪装对象检测任务也使用 RGB 图像作为输入，并输出二元预测。因此本小节直接将 PoolNet-R+ 应用到伪装对象检测任务而不做任何结构上的修改。参考[100] 中的做法，本小节使用随机梯度下降优化器（SGD）进行模型优化，相应的动量设置为 0.9，权重衰减为 5e-5。本小节中的模型共训练 32 个周期，批大小为 30。初始学习率设为 5e-3，同时使用预热和余弦学习率更新策略。本小节使用随机水平翻转和裁剪作为数据增强。参考本领域通用做法，COD10K[100] 中的训练集子集被用于训练。COD10K[100] 中的测试子集、CHAMELEON[203]、和 CAMO[204] 数据集被分别用于测试目的。在训练和测试阶段，输入图像的尺寸都被调整为 352×352 的分辨率。本小节使用四个评价指标进行性能评估，包括：准确率-召回率（PR）曲线，特征相似度（F-measure，F_β）、结构相似度（S-measure，S_α）和平均绝对误差（MAE），它们的具体概念和计算方式可以在章节 2.1.4中找到。

和领先方法的对比：表格 3.11中展示了本方法和 11 个领先方法的定量对比结果。可以看到本方法在三个本领域常用数据集的几乎所有指标上都取得最好的结果。具体而言，尽管本方法在 CHAMELEON[203] 数据集（76 张测试图片）上的 F-measure 和 S-measure 指标的结果不是最好的，但在更大的 COD10K[100] 数据集（2,026 张测试图片）上的性能则比之前最优的方法高很多。这一现象

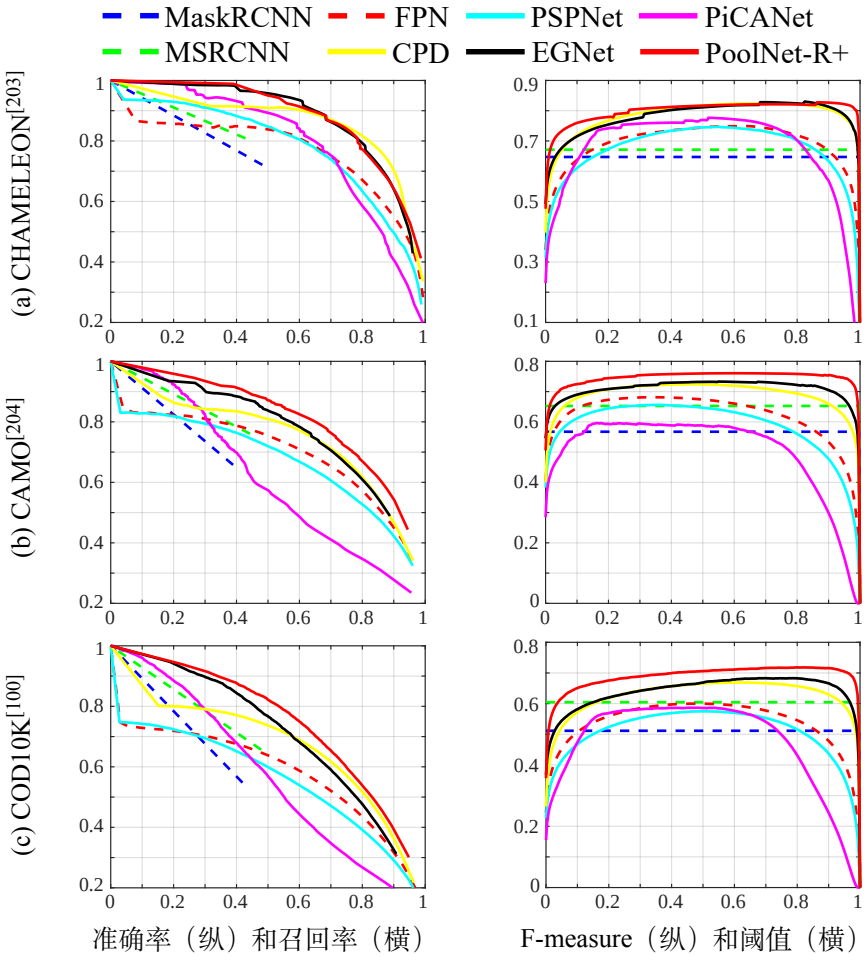


图 3.16 从左至右，每列分别为所提出方法和其他方法在三个流行的伪装对象检测数据集上的：准确率和召回率曲线、F-measure（F_β）值和阈值曲线的对比。

也能更好地体现了本方法的有效性。至于计算资源的消耗，只有 PFANet[48] 在参数量上比本方法更低。然而 PFANet[48] 在性能上远低于本方法（F-measure 和 S-measure 指标平均分别低 17% 和 12%），并且在乘加量要高 98%。与之前表现最好的方法相比，本方法在 F-measure 指标上仍要平均高出 2% 左右，而分别只需要 30% 和 12% 的参数量和乘加量。在 PR 和 F-measure 曲线中也可以观察到类似的现象，本方法（红色）的曲线相比其他方法更加突出，如图 3.16 所示。上述实验结果表明，本方法在被应用于具有非常不同目的的任务时仍表现良好（如显著性到伪装对象检测），验证了本方法具有良好的泛化能力和鲁棒性。

第六节 本章小结

本章通过设计两个简单的基于不同池化操作的模块，探索了高效池化技术在显著性目标检测中的潜力。考虑到精确定位显著性目标的重要性，本章设计了一个全局引导模块（GGM）来扩大自底向上通路的有效感受野，并确保位置信息在自顶向下通路中的引导作用。本章进一步提出了一个改进的特征聚合模块（FAM+）来弥合局部上下文信息和全局指导信息之间的感受野差距。在五个流行的显著性目标检测数据集上的大量实验表明，所提出的方法显著优于现有的最先进的方法。此外，为了满足移动设备上极低的计算开销限制，本章提出了一个名为 PoolNet-M+ 的轻量级版本模型，它能以不到原先十分之一的参数量和乘加量实现优异的预测精度，并且运行速度更快。

本章在多个数据集上从三个方面进行了一系列精心设计的消融实验，以探究所提出的两个模块如何工作以及为何有效。本章首先将 GGM 和所有的 FAM+s 分别视为两个整体部分，来验证它们在模型结构层面上的影响。由于模型中有多个位置可以用来放置所提出的两个模块，因此本章进一步从模块层面拆解 GGM 和每个 FAM+ 中的组件，以从模块层面验证它们的贡献。最后，本章从操作层面比较了两个模块的不同设计选择之间的差异。除了数字和曲线，本章还可视化了不同情形下的中间特征图来直观地说明上述因素的影响。

本章也分析了所提出方法的运行效率，并表明可以在不损害性能的情况下减少一半以上的计算开销。为了验证所提出结构的泛化能力，本章将其应用于三个相关且具有通用图像属性的浅层视觉任务上，包括边缘检测、RGB-D 显著性目标检测和伪装对象检测。实验结果表明，所提出的结构只需稍加修改，就分别在三个任务上的多个数据集上实现了相较于各任务领先方法的实质性提升。作者希望本章的设计理念和实验能够启发显著性目标检测和其他相关视觉任务探索更多有前景的未来研究方向。

第四章　基于高效信息集中交互与融合的显著性目标检测算法

第一节　引言

得益于视觉显著性目标检测不依赖于待检测对象的语义类别的特点，其作为一项计算机视觉浅层领域重要的研究方向，已被广泛应用到如弱监督语义分割[175, 209]，视觉追踪[169]，内容感知图像编辑[171]，和机器人路径规划[174] 等各类相关的下游任务中。传统的显著性目标检测方法很依赖于相关领域的专家依据他们的经验而手工设计的各类特征检测器。但这些手工设计的检测器不能有效利用隐藏在图片以及数据集中的深层语义信息，以致在复杂场景中表现不佳。近年来，基于卷积神经网络的方法得益于其同时且高效地提取由浅到深等多层次、多尺度特征的能力而得以快速发展。

而设计出能够提取更丰富、更有表征能力的多尺度特征的网络结构，往往意味着检测性能的提升，这也是显著性目标检测领域一个持续性的研究热点方向。在各类负责特征提取的网络结构中，一个颇具代表性的类型就是 U 型结构[120, 121]。正如图 4.1所示，一个典型的 U 型结构通常由一个自顶向下的通路，一个自底向上的通路，以及两个通路之间的一些连接所构成。研究者们提出了各种方法来改进 U 型结构，但其中大部分方法关注于如何提升其中自底向上通路对多尺度特征的提取能力，以及（或者）其中自顶向下通路对多尺度特征的聚合能力。而很少有工作关注位于这两条通路之间的连接。一个常见的做法是直接将这两条通路之间的对应层级分别进行连接，或者简单地使用一个卷积层来映射不同的特征通道。因此，一个自然而然的问题便是：是否可以通过设计一个结构来挖掘和发挥这些之前被忽视的连接的价值，从而在两个通路之间进一步提升模型对于多尺度特征的整合能力。本章将以一个不同的视角来关注如何增强所提取出特征的表征能力。具体而言，本章重新设计了 U 型结构的自底向上和自顶向下通路之间的连接，而不是这两条通路本身。为了达到上述目的，一个简单而直接的方法是在 U 型结构的自底向上通路后便对其所提取出的属于不同层级的特征进行融合（例如通道维度拼接或像素维度相加）[210, 211]。多尺度特征融合中一个不可避免的步骤便是空间插值操作，然而这些操作却可能会带

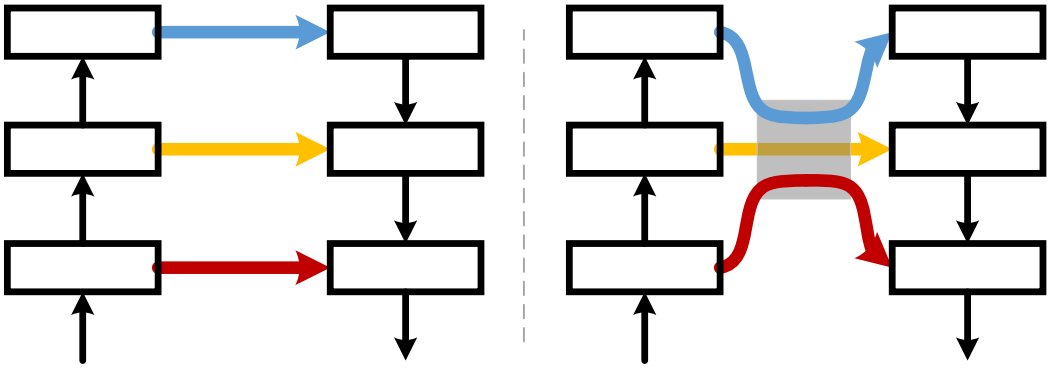


图 4.1 本章提出的集中信息交互策略的示意图。**左**：一个典型 U 型结构，其中自底向上和自顶向下通路中对应的层级被直接相连; **右**：本章提出的集中式策略，其中的圆角灰色矩形表示一个可以并行处理所有层次（尺度）特征的共享模块。

来一些负面影响。为了更加直观的感受这一影响，图 4.2中对空间插值操作前、中和后各过程中的特征图进行了可视化比较。从图中可以注意到，先进行下采样然后再上采样插值操作所得到的特征图同它的初始值相比相差很多，而互换采样操作顺序也会得到相似的结论。上述因为插值操作所导致的影响在下采样倍率变大的时候变得更加严重，因为更多的空间信息在这一过程中丢失了，这一现象值得额外注意。为此，本章提出将多尺度信息编码进不需要空间插值操作的、可学习的滤波器中，而不是特征图本身，以期在达到多尺度信息交互的目的的同时获取细节更丰富的特征。如图 4.1的右侧所示，本章设计了一个基于参数共享的跨尺度信息集中交互策略。该策略能够将从 U 型结构的自底向上通路提取得到的多尺度特征被用于建立自顶向下的通路之前进行并行处理，以达到多尺度信息交互的目的。本章将所提出的基于滤波器的信息集中化交互策略命名为集中信息交互策略（Centralized Information Interaction，CII）。CII 通过鼓励其中心处模块中的共享可学习滤波器去主动适应多尺度的输入，隐式地确保了跨尺度信息之间的交互。同时，CII 可靠地保留了每个输入尺度下确切的空间位置，从根本上避免了空间插值操作对特征可能产生的负面影响。CII 不特指任何具体的模块，而是一种可以与各种多尺度模块相结合的策略。然而，现存的多尺度模块（例如 PPM[129]，ASPP[130]，和 SE[212] 等），基本都是面向单尺度输入特征而设计的。但是，随着 CII 的提出而出现的新常态是：CII 中的共享模块的输入特征天然地包含有多个不同尺度。为了配合 CII 并发挥其潜力，本章进一步提出了一个相对的全局校准（Relative Global Calibration, RGC）模块。不同于之前模块倾向于对尽可能多的多尺度信息进行建模，RGC 被设计成只对必要尺

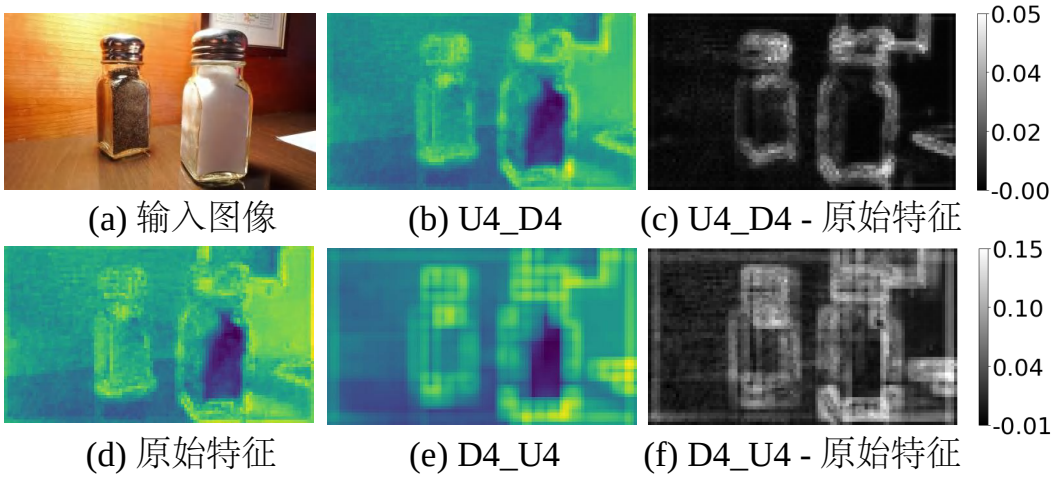


图 4.2 一个说明空间插值影响的例子：输入图像（a）和其对应的在 ResNet-18 网络的 `res1` 层的输出特征图（d）。（b）是（d）依次经过四倍双线性上采样和下采样插值而得到的，而（e）则是先进行四倍下采样再上采样。（c）和（f）是（b,d）和（e,d）之间分别做差而得到的图像。可以看到即使是最简单的空间插值操作也会造成明显的特征图像差异。

度信息进行建模。RGC 由两条并行支路构成：一条用于保持输入特征原有尺度的空间上下文信息，另一条用于捕获该输入尺度下的相对全局向量。这些全局向量由于具有较大的感受野优势，可以被用于指导原始特征空间的特征转换过程。尽管对于每个特定的单尺度输入而言，只有其原始和全局两个尺度的信息被获取。但考虑到 CII 的输入本身就包括多个尺度，RGC 中的可学滤波器也自然而然地编码了丰富的多尺度信息。RGC 进一步利用了 CII 的优势，并且避免了在之前的多尺度模块中容易发生的尺度多样性爆炸问题[129, 130, 180]。本章同时也展示了只需要对 RGC 模块的输入流进行一个细微的修改，模型的整体性能可以在几乎没有额外资源和计算代价的前提下获得进一步提升。

为了评估所提出方法的性能，本章详细对比了本方法在五个流行的显著性目标检测数据集上的结果。本章也同时进行了广泛的对比实验，并展示了众多可视化的示例，以帮助读者更好地理解本方法不同部分的作用和效果。本章提出的模型可以在单块 RTX-2080Ti 显卡上以不到 1.5 小时的时间在包含有 10,553 张图片的训练数据集上完成训练。本模型在测试时能以超过 50FPS 的速度处理 300×400 分辨率大小的输入图片。总而言之，本章主要的贡献可以总结如下：

- 本章重新思考了 U 型结构在显著性目标检测中的意义，通过关注之前被忽视的位于其自底向上和自顶向下通路之间的连接，提出了一个集中信息交

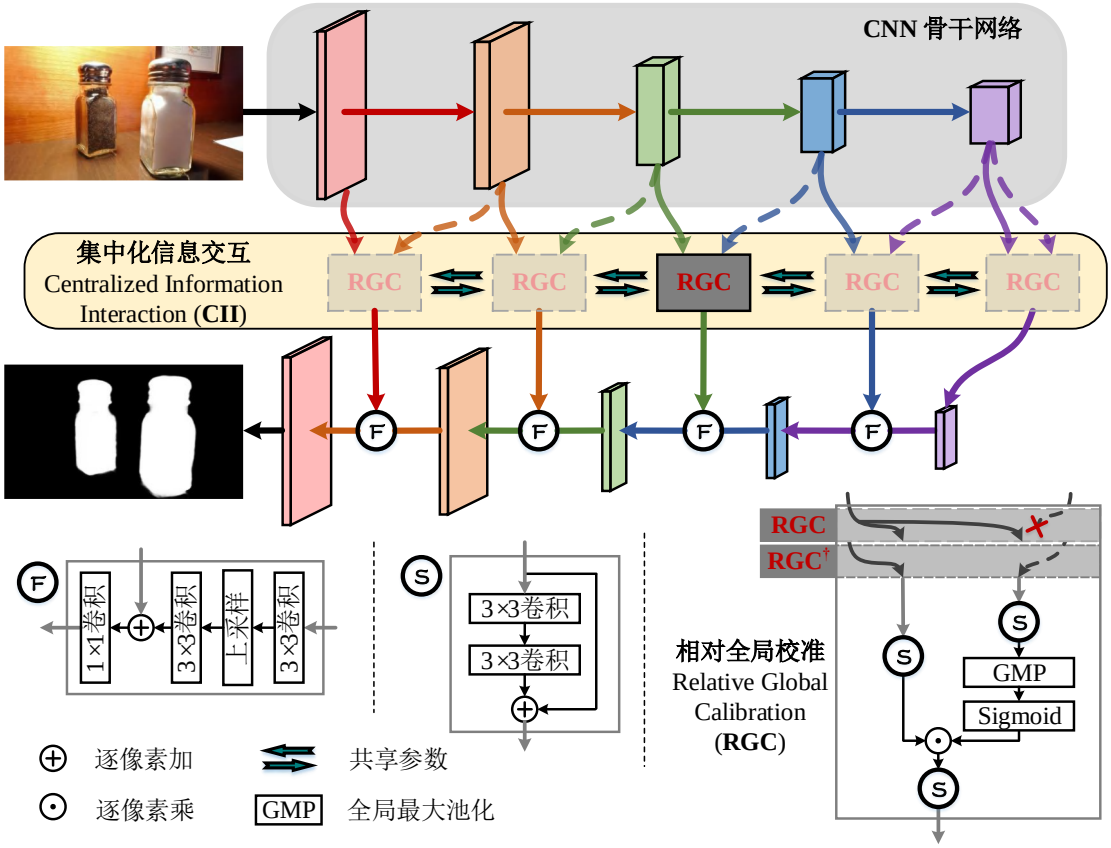


图 4.3 本章方法的总体流程图。虚箭头只在 **RGC**† 中生效，如右下角所示。

互的策略（CII）。CII 将跨尺度信息编码进可学习的滤波器而不是特征中，从根本上避免了特征图因为空间插值所带来的负面影响。

- 本章设计了一个相对的全局校准模块（RGC），该模块利用 CII 天然的多尺度输入特性，对每个输入尺度只对其必要尺度的信息进行建模。RGC 体现了 CII 的优势和潜力，并展现了一个对于多尺度模块设计而言具有价值的发展方向。
- 本章进行了全面的对比实验来解释所提出的策略和模块的设计理念，并探究了它们的有效性。
- 本章提出的方法相较于之前的顶尖方法在五个具有挑战性的基准数据集上达到了更为优越的性能，同时只需要更少的参数和浮点运算量。

第二节 集中交互网络

本小节将详细介绍基于多层级信息集中交互的网络：首先介绍本网络的整体流程，接着依次描述本网络的两个主要部分，即信息交互策略和特征校准模块。

4.2.1 整体流程

本章提出的方法基于被广泛使用的 U 型结构，它由一个用于提取多尺度特征的自底向上通路和一个用于融合这些特征的自顶向下通路构成。如图 4.3所示，从自底向上通路提取出的多尺度特征被逐层级地并行送入信息交互模块（实心灰色矩形部分）。本模型通过共享信息交互模块中的参数来学习得到强大的滤波器，以达到高效的跨尺度信息交互目的。交互后的特征按照从深层到浅层的层级顺序再被逐步用于建立自顶向下通路。本章将上述把多尺度信息编码进共享滤波器的信息交互策略称为集中信息交互（CII）。考虑到 CII 的输入特征现在包括有多个不同尺度的信息，本章进一步提出了一个相对的全局校准模块（RGC）来与其协作。RGC 能够依据每个不同输入尺度的特点，自适应地运用其中的相对全局的信息，从而达到深层全局语义和局部纹理信息间的平衡。接下来的章节将细致介绍上述提及的策略与模块。

4.2.2 集中信息交互策略

U 型结构中一个典型的设计是其自底向上和自顶向下两条通路之间，在具有相同空间尺度的层级间的连接。这一设计提供了一个简单有效的用于融合所提取出的多尺度特征图的方法。本模型用基于 ResNet-18[11] 骨干网络的经典 U 型结构作为例子。$\mathbb{B} = \{B_i\}$ $(1 \leq i \leq M$ 且 $M = 5)$ 表示从 `conv1`，`res1`，`res2`，`res3`，`res4` 层输出的特征图，它们通常被用于建立自底向上通路的特征金字塔。在自顶向下通路中，最高层级的特征图被逐步上采样，并依次与具有对应下采样倍率的特征图相融合。值得注意的是，特征图 B_i 在与自顶向下通路中的 B_{i+1} 级特征进行融合之前，并不能从那些属于更高层级的特征图 B_j $(1 \leq i < j \leq M)$ 中得到任何信息。假设将自底向上和自顶向下通路间的连接视为相互独立的部分，而在这种情况下，不同尺度间的信息流是彼此独立且互不可知的。

一个颇具参考价值的观察是：更低层次的特征包含更多局部纹理与图案，而更深层次的特征则体现了整体目标物体的位置。这也显示出了增强它们之间交互的必要性，从而能以更准确的定位和精确的分割信息辅助其他特征。因此

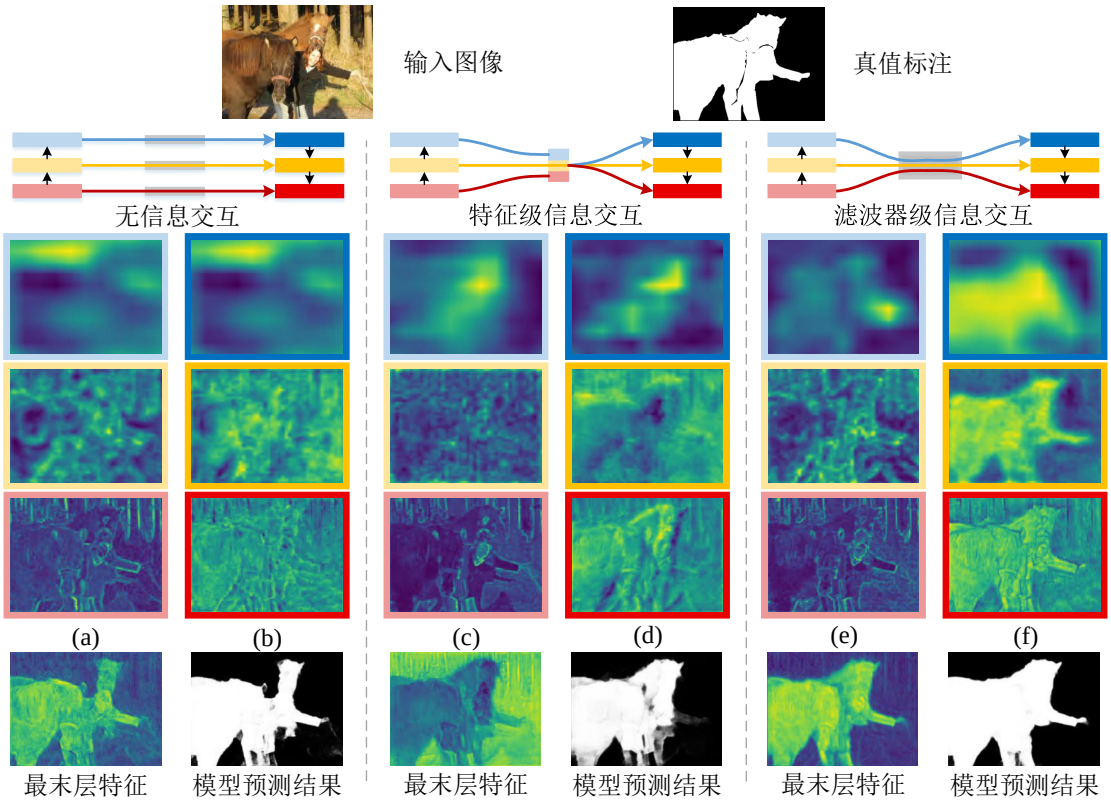


图 4.4 不同信息交互策略下得到的特征图的可视化对比。从左到右依次为：不包含交互，特征级的交互和滤波器级的交互。（a,c,e）列分别是不同空间层级特征在交互前的情况，而（b,d,f）列分别是交互过程后的特征。最后一行的子图分别是模型最后一个卷积层产生的特征和对应的网络的预测结果。可以看到滤波器级的交互策略（CII）可以更好地利用编码在多尺度特征中的互补信息（列 (f) 对比列 (e)）。

本章提出在多尺度特征图被从自底向上通路传递到自顶向下通路之前来增强它们之间的信息交互。而经典的 U 型结构直接将提取出的多尺度特征图传送给自顶向下通路。如图 4.3中的圆角黄色矩形所示，CII 则以不同的方式使用一系列相同的信息交互模块（实心灰色矩形），来与编码在其中的信息进行交互。这些信息交互模块被放置在经典 U 型结构的中心，并且它们的参数是共享的。多尺度信息因而可以通过被编码进共享可学习的滤波器的方式实现彼此间的交互。需要特别强调的是 CII 并不是一个具体的模块而是一种信息交互策略。CII 中被用于信息交互目的的模块的设计可以很灵活，只需要保证相应输入和输出的尺寸保持一致即可，因此也可以简单的用各种成功的模块[129–131, 133] 所代替。

当以 ResNet-18 作为骨干网络时，$\mathbb{B}$ 中各层级特征所对应的通道数分别设置为 $\{64,64,128,256,512\}$。本模型在每层特征 $B_i \in \mathbb{B}$ 后额外添加了一个 1×1 的

卷积层，来将输入通道映射到相同维度的输出通道（即 64）。出于简洁表述的目的，在本章接下来的部分中均省略了每个卷积层后面的批量归一化层（BN[213]）和非线性激活层（ReLU[214]）。之后 CII 中的信息交互模块会将通道映射后的特征图分别进行处理以产生 CII 的输出特征图：$\mathbb{C} = \{C_i\}$ $(1 \leq i \leq M$ 且 $M = 5)$。CII 的整体过程可以总结为：

$$C_i = InI_i(f_i^{1\times1}(B_i)),\ 1 \leq i \leq M, \tag{4.1}$$

其中 C_i 和 B_i 的空间尺寸一致，InI_i 表示相同的、在每个 i 间共享参数的信息交互模块。$\mathbb{C}$ 接下来会被用于建立自顶向下通路。与之前一些方法[210, 211] 通过对多尺度特征进行直接融合（例如拼接或相加）以达到信息交互的目的所不同的是，本章提出的 CII 将交互的信息编码为共享可学习的滤波器。CII 中的信息交互模块可以同时获得从高到低多个层级特征中的优化信号，从而学习到具有更强语义和更准确位置信息的模式特征。CII 的一个优势在于其从根本上避免了上采样操作导致的混叠效应。每个信息交互模块（参数共享）的输入和输出特征图有着相同的空间尺寸，因而不需要进行空间插值操作。同时得益于本模型无需针对每个输入尺度分别设置单独的模块，CII 只引入了少量的额外参数。

为了更加直观感受 CII 的优越性，本小节展示了一些来自不同空间层级的中间阶段特征图在经过 CII 前后（分别对应图 4.4的（e）和（f）列）在视觉感官上的对比。如果不对信息交互模块进行参数共享，便可以得到一个普通的 U 型结构基准模型（（a）列和（b）列）。通过对比（e）和（f）列，可以看到在经过 CII 后（f），低层级特征（红框）倾向于去更多地突出那些结构性相关的区域，而不仅仅是那些位于边缘上的像素。高层级特征（蓝框），与之相反，则不仅仅只是关注激活区域的中心部分，而是在物体边缘部分也变得更加精细且有着高置信度。相比之下，在没有 CII 时，经过信息交互模块之前与之后的特征图在视觉上拥有相似的信息层级（列（b）对比于列（a））。这一视觉现象也佐证了本章提出的 CII 在跨尺度信息互补中的重要作用。

4.2.3 相对的全局校准模块

CII 引入了一个新的策略以用于在经典的 U 型结构上获得更加高效的跨尺度间信息交互。如上节所述，当构建信息交互模块时，一个简单的由两个 3×3 卷积层构成的序列相较于其基线版本（表格 4.2第 3 行对比第 2 行）有着显著的性能提升。众所周知在分割任务中，一个有效的多尺度模块总能对模型的整体

性能带来提升。例如著名的 PPM[129] 包括四条并行的有着不同下采样倍率的池化操作支路来挖掘输入特征图的多尺度信息。该模块最先在语义分割中被提出，后来被成功地应用于许多显著性物体检测方法[41, 215]。基于这一前提以及 PPM 本身也被设计为即插即用的特点，本小节尝试将其作为信息交互模块在 CII 中使用（公式 (4.1) 中的 InI_i ）。然而，除了 PPM 之外，其他很多先前成功的多尺度模块（例如 ASPP[130]，SE[212] 等）也并不能很好地与 CII 相协作（如表格 4.6所示，更多细节将在 章节 4.3.2.3中介绍）。这些现有的多尺度模块主要被设计为用于单尺度输入，从而尽可能多地从多个感受野尺寸来收集信息。而当输入就包括多尺度时（即 $\mathbb{B}$），这些模块会引起感受野尺寸范围的迅速增长（即 1×4 相比 $M\times4$）。然而，正如许多之前的论文[129, 130, 180] 所指出的那样，更大的多尺寸范围并非一定对应更好的效果，过多的尺寸种类甚至可能干扰后续的模型层。

考虑到 CII 的输入（即 $\mathbb{B}$）天然地包括有多个尺寸的感受野，因此有必要对其中冗余的部分进行剔除，而只保留那些必须的尺度种类。如图 4.3的右上部分所示，本章设计了一个相对的全局校准模块 RGC，它包括两条并行的支路，分别用于局部信息的保留和相对全局信息的采集目的。具体而言，在 RGC 的两条支路中，输入 $B_i\in\mathbb{B}$ 首先被两个串联的 3×3 卷积层（分别表示为 $f_{L_2}^{3\times3}$ 和 $f_{R_2}^{3\times3}$）处理。由于输入特征图空间尺寸的不同，这些卷积层中的可学习的参数为特征的调整预留出了适当的空间。在右侧支路的卷积层后面，本模型使用了一个全局最大池化模块来收集相对于输入层级为 i 的特征 B_i 的全局信息 G_i:

$$G_i=\sigma(GMP((f_{R_2}^{3\times3}+1)(B_i))),\ 1\le i\le M, \tag{4.2}$$

其中 σ 表示 sigmoid 函数。在此之后，右侧支路收集得到的全局信息被用于校准左侧支路中保留下来的局部特征。最后再通过另外两个串联的 3×3 卷积层，便可以得到对应的输出 R_i:

$$R_i=(f_{F_2}^{3\times3}+1)(G_i\odot(f_{L_2}^{3\times3}+1)(B_i)),\ 1\le i\le M. \tag{4.3}$$

在公式 (4.2) 和公式 (4.3) 中所有的可学习的参数都跨每个层级 i 所共享。本章将在章节 4.3.2.3中展示，尽管包含的支路更少，RGC 在与 CII 协作时能够获得相比之前各类多尺度模块更好的表现。为了进一步探究 RGC 的潜力，本小节对 RGC 的输入流做了一些小的修改，以在不通过对输入进行空间插值的情况下引入更大范围感受野下的全局信息。通过简单地将右侧支路的输入替换为其紧接

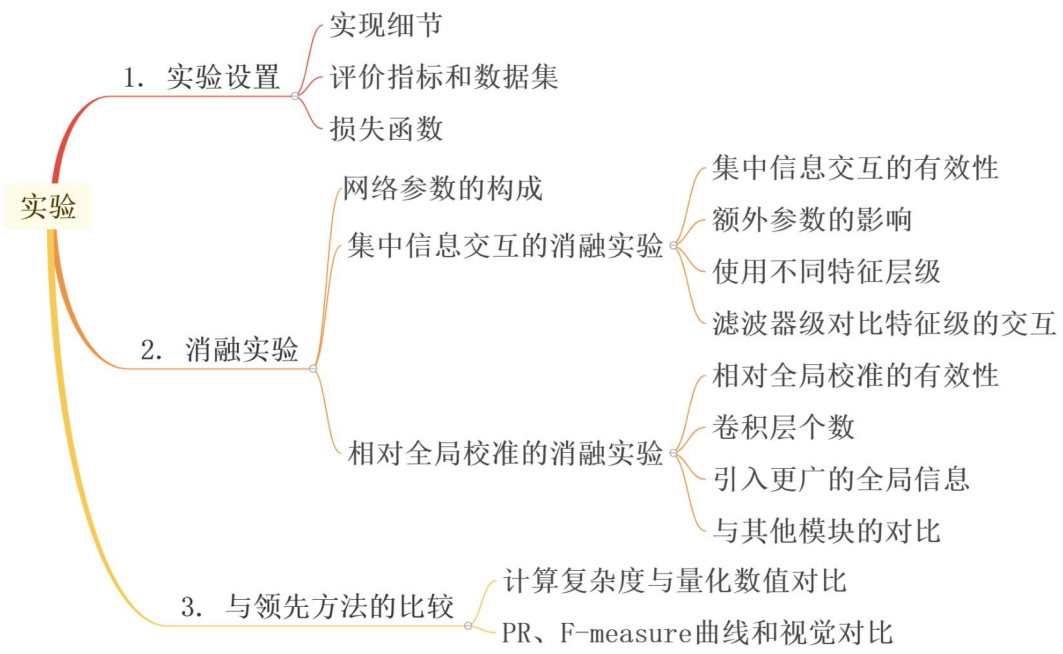


图 4.5 本章实验部分的总体实验方案导图。

着的后继层级的特征图（即 B_i 替换为 B_{i+1}），便可以得到 RGC†，它可以在不引入额外参数，甚至更少计算量的情况下进一步提升性能。更多定量的分析将在实验部分提供。

第三节 实验

本小节的总体实验方案设计如图 4.5所示。本小节首先介绍了实验设置，包括实现细节，使用的数据集、评价指标、以及损失函数等。接着进行了一系列消融实验来说明所提出模型的每个部分对性能的影响。最后展示了所提出的方法在不同设置下的性能，并与现有的领先方法作了对比。

4.3.1 实验设置

实现细节：本模型主要使用了开源的 PyTorch 库[1]来实现，并在一块 RTX-2080Ti 显卡上完成了所有训练和测试过程。本模型中骨干网络的参数（即 ResNet-18 和 ResNet-50[11]）使用了在 ImageNet[9] 进行过预训练的相应模型的参数作为初始化，而其余的可学习参数均采用高斯随机数初始化。对于所有的实

[1]https://pytorch.org

验，本模型均以 30 的批量大小被训练了 32 个周期。本章采用了随机梯度下降（SGD）方法来训练网络，其中动量为 0.9，权重衰减为 5e-5。骨干网络的最大学习率设为 0.005，其余部分设为 0.05。本模型在前 8 个和最后 24 个周期中采用了预热和余弦学习率策略来分别进行控制。在训练和测试时输入图片大小都被调整为 352×352。在训练过程中随机水平翻转与随机裁剪被用于数据增强的目的。除特殊说明外，本章的消融实验默认均基于 ResNet-18[11] 骨干网络。在所有本模型相关的实验中，没有使用任何预处理和后处理的技术。

评价指标和数据集：参考本领域通常的做法，对于所有实验，本章使用 DUTS-TR[63] 数据集进行训练。对于性能评估，本章使用四个广泛使用的评价指标在五个流行数据集上进行性能评估和对比。评价指标包括：准确率-召回率（PR）曲线，特征相似度（F-measure，F_β）、结构相似度（S-measure，S_α）和平均绝对误差（MAE），具体概念和计算方式可以在章节 2.1.4中找到。测评数据集包括：ECSSD[62], PASCAL-S[61], DUT-OMRON[29], HKU-IS[32] 和 DUTS-TE[63]，它们的具体信息可以在2.1.3中找到。

损失函数：本模型使用了二元交叉熵损失（BCE）和交并比损失（IoU）之和作为总损失函数：

$$l = l_{bce} + l_{iou}. \tag{4.4}$$

BCE 损失函数因为鲁棒性较好，在二元分类和分割任务中被广泛应用，它以逐像素的方式来计算图像损失：

$$l_{bce}(x,y) = -\frac{1}{n}\sum_{k=1}^{n}[y_k log(x_k) + (1-y_k) log(1-x_k)], \tag{4.5}$$

其中 x 和 y 分别代指预测出的结果图片和对应的真值标签，而 k 是像素的索引，n 代表 x 中的像素数目。不同于 BCE 损失函数主要关注于像素级别的差异，IoU 损失则考虑了全图的相似性，其定义如下：

$$l_{iou}(x,y) = 1 - \frac{\sum_{k=1}^{n}(y_k * x_k)}{\sum_{k=1}^{n}(y_k + x_k - y_k * x_k)}. \tag{4.6}$$

4.3.2 消融实验

本小节首先进行了几个直观的实验从整体的角度来说明 CII 和 RGC 的有效性。接着设计了更多细致的消融实验来进一步探究和分析了 CII 和 RGC 中的一些设计选择和对应的设置。

表 4.1 所提出模型的参数构成。从表中可以看出，特征提取器（ResNet-50）占据了其中的绝大部分。

	ResNet-50	RGC（图 4.3右下）	F（图 4.3左下）	其他	总计
参数量	23.51M	0.22M	0.41M	0.34M	24.48M
占比	96.04%	0.90%	1.67%	1.39%	100.00%

表 4.2 对 CII 策略的消融分析。M 为自底向上和自顶向下两通路间的连接个数。每列最好的结果以**粗体**突出显示。

卷积核尺寸	卷积层数量	是否共享	DUT-OMRON[29]			DUTS-TE[63]		
			$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$
1×1	$1\times M$	✗	0.801	0.075	0.816	0.848	0.060	0.851
3×3	$2\times M$	✗	0.804	0.075	0.818	0.849	0.059	0.852
3×3	2×1	✓	0.810	0.069	0.822	**0.869**	0.050	**0.870**
3×3	4×1	✓	**0.812**	**0.066**	**0.825**	0.868	**0.049**	**0.870**

4.3.2.1 网络参数的构成

表格 4.1中列出了本章所提出网络的参数构成。从表中可以看到，相较于 ResNet-50 骨干网络（23.51M），本模型仅引入了 0.97M（3.96%）的额外参数。特别地，RGC 模块仅仅占据了 0.22M（0.90%）的参数。作为对比，另外 0.75M（3.06%）的参数主要用于构建自顶向下通路中的基本组成部分和整体网络其余的配件部分（0.34M, 1.39%）。通过对从骨干网络中提取出的多尺度特征的有效利用，并增强它们之间跨尺度信息间的交互，本章提出的方法在引入较少额外参数量的情况下大幅提升了网络的整体性能。这一分布差异明显的参数构成体现了本模型的轻量与有效。

4.3.2.2 集中信息交互的消融实验

集中信息交互的有效性：为了验证 CII 相比于经典 U 型结构的有效性，表格 4.2中对比了在自底向上和自顶向下通路间使用不同类型的连接方式的影响。除了连接方式的不同，其余所有实验设定都保持一致。表格 4.2第一行是基于 U 型结构的基线模型，其中的自底向上和自顶向下两通路间的对应层级分别被一个 1×1 卷积层所连接。作为对比，第三行使用了两个串联的 3×3 卷积层作为信息交互模块（公式 (4.1) 中的 InI_i）来验证本章提出的 CII 方法，即这两个卷积层在各层级间保持参数共享。从对比中可以看到，相较于基线的 U 型结构模型，CII 的使用极大增强了网络的整体性能。

表 4.3 在 CII 中使用不同层级特征的消融分析。✓表示自底向上和自顶向下通路的该对应层级间存在连接。每列的最好结果以**粗体**突出显示。

层级					DUT-OMRON[29]			DUTS-TE[63]		
1	2	3	4	5	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$
✓	✓	✓			0.805	0.071	0.822	0.858	0.055	0.861
	✓	✓	✓		0.808	0.067	0.822	0.860	0.052	0.862
		✓	✓	✓	0.807	0.067	0.824	0.859	0.052	0.864
✓	✓	✓	✓		0.805	0.067	0.823	0.864	0.051	0.867
	✓	✓	✓	✓	0.808	0.067	**0.825**	0.867	0.050	0.869
✓	✓	✓	✓	✓	**0.812**	**0.066**	**0.825**	**0.868**	**0.049**	**0.870**

表 4.4 对不同信息交互策略的消融实验。第一列代表自底向上和自顶向下两条通路间额外使用的卷积层的数量。每列的最好结果以**粗体**突出显示。

卷积层数量	信息交互策略	DUT-OMRON[29]			DUTS-TE[63]		
		$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$
5	无交互	0.801	0.075	0.816	0.848	0.060	0.851
5	特征级	0.808	0.069	0.821	0.854	0.055	0.858
4	滤波器级	**0.812**	**0.066**	**0.825**	**0.868**	**0.049**	**0.870**

额外参数的影响：为了排除额外引入的可学习参数可能带来的影响，在表格 4.2的第二行中。本小节各用两个串联的 3×3 卷积层来替换第一行中模型里每个 1×1 卷积层（M 个卷积序列, 总共 $2\times M$ 个卷积层）。虽然在使用更多的卷积层时，模型的整体性能略有提升（第二行对比第一行），但是该模型相比于 CII 仍有较大差距（第二行对比第三行）。这一现象也表明了更多的可学习参数并不一定意味着更好的性能。通过对比表格 4.2中第四行与第三行也可以得出一个相似的结论。

使用不同特征层级：为验证在 CII 中使用不同层级的特征会如何影响性能，本小节也进行了一系列消融实验。从表格 4.3中的数据可以看出，一个整体趋势是使用更多层级的特征通常会带来更好的结果。而通过对比只使用三个层级的特征，也就是表中的前三行，可以看到不论是过度缺失浅层还是深层特征都会严重影响性能。尤其是当只使用前三层级特征（第一行）时，在两个对比数据集上的 MAE 指标都变差了（越低越好）。同时也可以观察到，不论是缺少最浅层还是最深层的特征，都会使整体性能变差。而当五个层级特征都被使用时模型获得了最好的性能。

滤波器级对比特征级的交互：表格 4.4中展示了基于不同类型信息交互策略

下所得到的结果。从表中可以看出，本章提出的滤波器级的信息交互策略相比于基线设置下特征级交互有着更好的性能（第三行对比第二行和第一行），尤其是在更具挑战性的 DUTS-TE 数据集上。本小节同时也在图 4.4中可视化对比了采用不同类型信息交互策略前后的中间特征图。在进行信息交互前（列（a，c，e）），深层特征（浅蓝框）大多关注于显著性物体的粗略位置信息。对比之下，中层（浅黄）和浅层（粉）特征则通常更强调边缘像素。在没有信息交互时（列（b）对比列（a）），对应层级的特征的视觉效果大体上没有什么明显变化。而在使用特征层级的信息交互后（列（d）对比列（c）），深层特征（蓝）中的显著性物体较背景略微更为明显，而中层（黄）和浅层（红）特征开始更加关注显著性物体的整体性而不仅仅是边缘像素。当使用滤波器级的信息交互策略时，更多明显的视觉提升可以被观察到（列（f）对比列（d，e））。具体而言，深层特征中显著性物体的激活信号变得更明显且更完整，而中层和浅层特征中的激活信号则更明显地展现出更精细的局部特征与更清晰的分割效果。上述提升很大程度上得益于本章提出的不需要对特征图进行空间插值的滤波器级别的交互策略。当没有下采样操作时，高分辨率的细节特征得以被更好地保留。而去掉上采样操作则可以有效避免混叠效应的产生。然而，在特征级别的交互中，所有层级的特征被需要先被插值到同一尺寸才能拼接在一起，其中由于插值操作而引入的不实信息可能会变得具有误导性从而损害模型的性能。这也可以从列（d）与列（f）的对比中可以看出，使用了插值操作的中层和浅层特征中的激活区域的边界更为模糊，而且在物体的中心区域的置信度更低。通过对比最后一个卷积层的可视化特征与其对应的模型预测结果（图 4.4最后一行），也可以得出一个相似的结论。

4.3.2.3 相对全局校准的消融实验

相对全局校准的有效性：为了证明 RGC 的有效性，本小节进行了一系列消融实验来对比不同集中信息交互模块（公式 (4.1) 中的 InI_i）的作用。在接下来的实验里，除了信息交互模块本身，其他所有的配置都与表格 4.2中的第四行实验相对齐。通过对比表格 4.6中的第二行与第一行，可以明显看出 RGC 模块的引入有效地帮助模型实现了更好的整体性能。这同时也验证了 RGC 中的额外分支的必要性，该分支能够充分利用对应于每个不同输入尺度下的相对全局信息，进而对相应的局部特征进行有效校准。

卷积层个数：为了确定 RGC 中各分支中使用多少个卷积层串联是最合适

表 4.5 对 RGC 中各分支中的卷积层个数的消融分析（图 4.3中下部的‘S’）。每列最好结果以**粗体**突出显示。

3×3 卷积层的数目	DUT-OMRON[29]			DUTS-TE[63]		
（串联数 × 分支数）	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$
0×3	0.805	0.063	0.815	0.853	0.049	0.854
1×3	0.810	0.064	0.819	0.868	0.047	0.866
2×3	**0.820**	0.061	**0.826**	0.873	0.045	0.870
3×3	0.810	0.061	0.824	**0.874**	**0.044**	**0.872**
4×3	0.810	**0.060**	0.821	0.872	0.045	0.866

表 4.6 RGC 模块的消融分析。第一行使用了四个串联的 3×3 卷积层作为信息交互模块（与表格 4.2第四行一致）。每列中最好的结果以**粗体**突出显示。

信息交互器类型	DUT-OMRON[29]			DUTS-TE[63]		
	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$
3×3 卷积	0.812	0.066	0.825	0.868	0.049	0.870
RGC	0.820	0.061	0.826	0.873	0.045	0.870
RGC†	**0.824**	**0.058**	**0.828**	**0.878**	**0.042**	**0.874**
PPM[129]	0.810	0.066	0.826	0.868	0.047	0.868
PPM†	0.803	0.070	0.821	0.866	0.047	0.869
ASPP[130]	0.814	0.06	0.820	0.872	0.047	0.864
ASPP†	0.806	0.067	0.824	0.867	0.047	0.869
SE[212]	0.809	0.066	0.823	0.869	0.046	0.870
SE†	0.813	0.063	0.827	0.872	0.045	0.869

的，本小节在表格 4.5中进行了一系列实验。从前三行可以看出，随着更多卷积层被使用，网络的整体性能逐渐得以提升。而当进一步增加卷积层的串联数目时，却可以发现整体性能并没有继续提升，反而是略有下降（见最后三行）。通过简单而初步的对比实验，本章最终选择第三行（也就是 RGC 模块每条支路中各包含两个串联的 3×3 卷积层）作为默认设置，这一设置在性能和准确率之间达到了相对的平衡。本章也在图 4.3的右上角注明了这一设置。

引入更广的全局信息：作为 RGC 的改进版本，RGC† 将 RGC 的右侧（相对全局）分支的输入从本层级改为后继层级的特征（B_i 到 B_{i+1}）。这一对输入流的修改不需要额外的参数和计算负担。对比表格 4.6中的第三行和第二行，可以看到模型的性能得以被进一步提高。本小节想要表达的是：RGC† 只是一个展示设计能够与 CII 更好协作的新模块或策略的潜力的例子。作者希望这些设计理念与实验可以为将来的研究提供更多启发。

与其他模块的对比：为探究之前提出多尺度模块与 CII 的协作效果如何，本小节向 CII 迁移了一些成功且具有代表性的模块（例如 PPM[129]，ASPP[130], 以及 SE[212]），并用它们来直接对 RGC 模块进行替换。然而，如表格 4.6所示，这些之前成功的模块并没有获得很好的表现。PPM 和 SE 模块的表现甚至略微差于只使用 3×3 卷积层序列（第一行）。本小节也尝试了这三个模块的 † 版本，即对于 PPM 和 SE 模块而言，作者将它们中包含有空间插值操作的分支的输入从本层级改为后继层级的特征，分别得到 PPM† 和 SE†)。而对于 ASPP 模块，作者则将它包含有空洞卷积操作的分支的输入进行了修改，得到 ASPP†。在意料之中的是，上述经过修改的模块的结果依旧不太理想（表格 4.6的第五，七，九行）。一个有趣的现象是 PPM† 和 ASPP† 的性能明显下降，而 SE† 对应的性能却有所提升。这些对比数据结果表明一些之前成功的模块并不一定在 CII 中也能适用。这也体现了设计能够与 CII 更好协作的多尺度模块这一方向的可研究价值。

4.3.3 与领先方法的比较

这一小节对比了所提出的方法与 26 个之前的领先方法。包括 DCL[40], RFCN[37], WSS[63], MSR[177], DSS[44], NLDF[46], Amulet[179], SRM[41], C2SNet[185], PAGR[50], DGRL[52], RAS[216], PiCANet[51], AFNet[53], MLMS[57], JDFPR[186], PAGE[187], CapSal[58], ICTB[188], CPD[47], BASNet[54], PoolNet[215], CSNet[189], GateNet[217], ITSD[218], 和 MINet[219]。为了保证对比的公平，其他方法用来作为对比的显著性图均由其作者所开源的原始代码和设置所生成，或者直接由其作者所提供。所有结果的测评方法和代码均是一样的。

计算复杂度与量化数值对比：在表格 4.7中列出了本模型与 26 个现有方法在五个流行数据集上的量化对比结果。本模型基于 ResNet-50 骨干网络的版本在大部分的数据集与测评指标上达到了的最优的性能，同时相较于那些现有的基于相同骨干网络的方法只需要更少的参数量和 FLOPs。当与之前最优的方法 ITSD[218]（基于 ResNet-50）相比较时，本模型在大部分数据集上达到了明显更好的结果，并且需要的额外参数少 67%（0.97M 对比 2.96M）、额外的 FLOPs 少 14%（4.74G 对比 5.53G）。本小节也在表格 4.7中展示了本模型基于 ResNet-18 的结果。值得一提的是，本模型基于 ResNet-18 版本的方法仍然比现有的大部分基于更强大的 ResNet-50 骨干网络的方法表现得更好，并有着明显更少的参数量和 FLOPs。这些对比结果展示了本模型在提升所提取出的多尺度特征中的信息交互效率上的高效和有效性。

表 4.7 本模型与 26 个现有方法在五个流行数据集上的量化对比结果。每列最好的结果以**粗体**突出显示。表中也展示了本模型基于 ResNet-18 的版本。灰色或蓝色背景的行分别代表具有相似计算量的方法。所有方法的 FLOPs 都在 224×224 输入分辨率下测得。

方法年份	参数 (M)	FLOPs (G)	ECSSD[62]			PASCAL-S[61]			DUT-OMRON[29]			HKU-IS[32]			DUTS-TE[63]		
			$F_\beta \uparrow$	$MAE \downarrow$	$S_\alpha \uparrow$	$F_\beta \uparrow$	$MAE \downarrow$	$S_\alpha \uparrow$	$F_\beta \uparrow$	$MAE \downarrow$	$S_\alpha \uparrow$	$F_\beta \uparrow$	$MAE \downarrow$	$S_\alpha \uparrow$	$F_\beta \uparrow$	$MAE \downarrow$	$S_\alpha \uparrow$
DCL_{16}[40]	66.25	-	0.896	0.080	0.869	0.805	0.115	0.800	0.733	0.094	0.762	0.893	0.063	0.871	0.786	0.081	0.803
$RFCN_{16}$[37]	-	-	0.898	0.097	0.856	0.827	0.118	0.808	0.747	0.094	0.774	0.895	0.079	0.860	0.786	0.090	0.793
WSS_{17}[63]	14.70	-	0.855	0.106	0.806	0.771	0.140	0.740	0.694	0.110	0.726	0.862	0.079	0.819	0.740	0.099	0.743
MSR_{17}[177]	-	-	0.903	0.059	0.887	0.839	0.083	0.835	0.790	0.073	0.805	0.907	0.043	0.896	0.824	0.062	0.834
DSS_{17}[44]	62.23	52.20	0.906	0.064	0.880	0.821	0.101	0.804	0.760	0.074	0.789	0.900	0.050	0.881	0.813	0.065	0.826
$NLDF_{17}$[46]	35.48	-	0.903	0.065	0.870	0.822	0.098	0.805	0.753	0.079	0.770	0.902	0.048	0.878	0.816	0.065	0.816
$Amulet_{17}$[179]	33.16	20.70	0.911	0.062	0.876	0.826	0.092	0.816	0.737	0.083	0.784	0.889	0.052	0.866	0.773	0.075	0.800
SRM_{17}[41]	53.14	-	0.916	0.056	0.891	0.838	0.084	0.834	0.769	0.069	0.798	0.906	0.046	0.887	0.826	0.058	0.836
$C2SNet_{18}$[185]	137.05	-	0.910	0.055	0.894	0.842	0.082	0.836	0.757	0.072	0.798	0.896	0.048	0.883	0.807	0.062	0.828
$PAGR_{18}$[50]	-	-	0.924	0.064	0.883	0.847	0.089	0.822	0.771	0.071	0.775	0.919	0.047	0.889	0.854	0.055	0.839
$DGRL_{18}$[52]	-	-	0.921	0.043	0.899	0.844	0.072	0.836	0.774	0.062	0.806	0.910	0.036	0.895	0.828	0.049	0.842
RAS_{18}[216]	20.23	21.24	0.918	0.059	0.888	0.829	0.101	0.799	0.786	0.062	0.814	0.913	0.045	0.887	0.831	0.059	0.839
$PiCANet_{18}$[51]	47.22	54.06	0.932	0.048	0.912	0.864	0.075	0.854	0.820	0.064	0.830	0.920	0.044	0.904	0.863	0.050	0.868
$AFNet_{19}$[53]	25.78	-	0.932	0.045	0.907	0.861	0.070	0.849	0.820	0.057	0.825	0.926	0.036	0.906	0.867	0.045	0.867
$MLMS_{19}$[57]	74.38	58.18	0.924	0.048	0.905	0.853	0.074	0.844	0.793	0.063	0.809	0.922	0.039	0.907	0.854	0.048	0.862
$JDFPR_{19}$[186]	87.61	42.96	0.925	0.052	0.902	0.854	0.082	0.841	0.802	0.057	0.821	-	-	-	0.833	0.058	0.836
$PAGE_{19}$[187]	-	-	0.928	0.046	0.906	0.848	0.076	0.842	0.791	0.062	0.825	0.920	0.036	0.904	0.838	0.051	0.855
$CapSal_{19}$[58]	-	-	-	-	-	0.862	0.073	0.837	-	-	-	0.889	0.058	0.851	0.844	0.060	0.818
$ICTB_{19}$[188]	-	-	0.935	0.045	0.912	0.855	0.071	0.850	0.811	0.060	0.837	0.925	0.037	0.909	0.855	0.043	0.865
CPD_{19}[47]	47.85	7.23	0.936	0.042	0.913	0.859	0.071	0.848	0.796	0.056	0.825	0.925	0.034	0.907	0.865	0.043	0.869
$BASNet_{19}$[54]	87.06	97.65	0.939	0.040	0.911	0.857	0.076	0.838	0.811	0.057	0.836	0.930	0.033	0.908	0.860	0.047	0.866
Ours(ResNet-18)	11.89	6.49	0.937	0.042	0.910	0.868	0.068	0.851	0.824	0.058	0.828	0.933	0.032	0.912	0.878	0.042	0.874
$PoolNet_{19}$[215]	68.26	38.19	0.940	0.042	0.914	0.865	0.075	0.850	0.830	0.055	0.836	0.934	0.032	0.917	0.886	0.040	0.883
$CSNet_{20}$[189]	36.37	11.75	0.940	0.041	0.914	0.866	0.073	0.851	0.821	0.055	0.831	0.930	0.033	0.911	0.881	0.040	0.879
$GateNet_{20}$[217]	128.63	55.23	0.942	0.043	0.914	0.877	0.068	0.858	**0.831**	0.055	0.838	0.935	0.033	0.915	0.889	0.040	0.885
$ITSD_{20}$[218]	26.47	9.67	0.944	0.037	0.919	0.871	0.066	0.859	0.823	0.061	**0.840**	0.933	0.031	0.916	0.883	0.041	0.885
$MINet_{20}$[219]	162.38	42.73	0.945	**0.036**	0.920	0.874	0.064	0.856	0.826	0.056	0.833	0.936	**0.028**	**0.920**	0.888	0.037	0.884
Ours(ResNet-50)	24.48	8.88	**0.947**	**0.036**	**0.921**	**0.882**	**0.062**	**0.865**	**0.831**	**0.054**	0.839	**0.939**	0.029	**0.920**	**0.890**	**0.036**	**0.888**

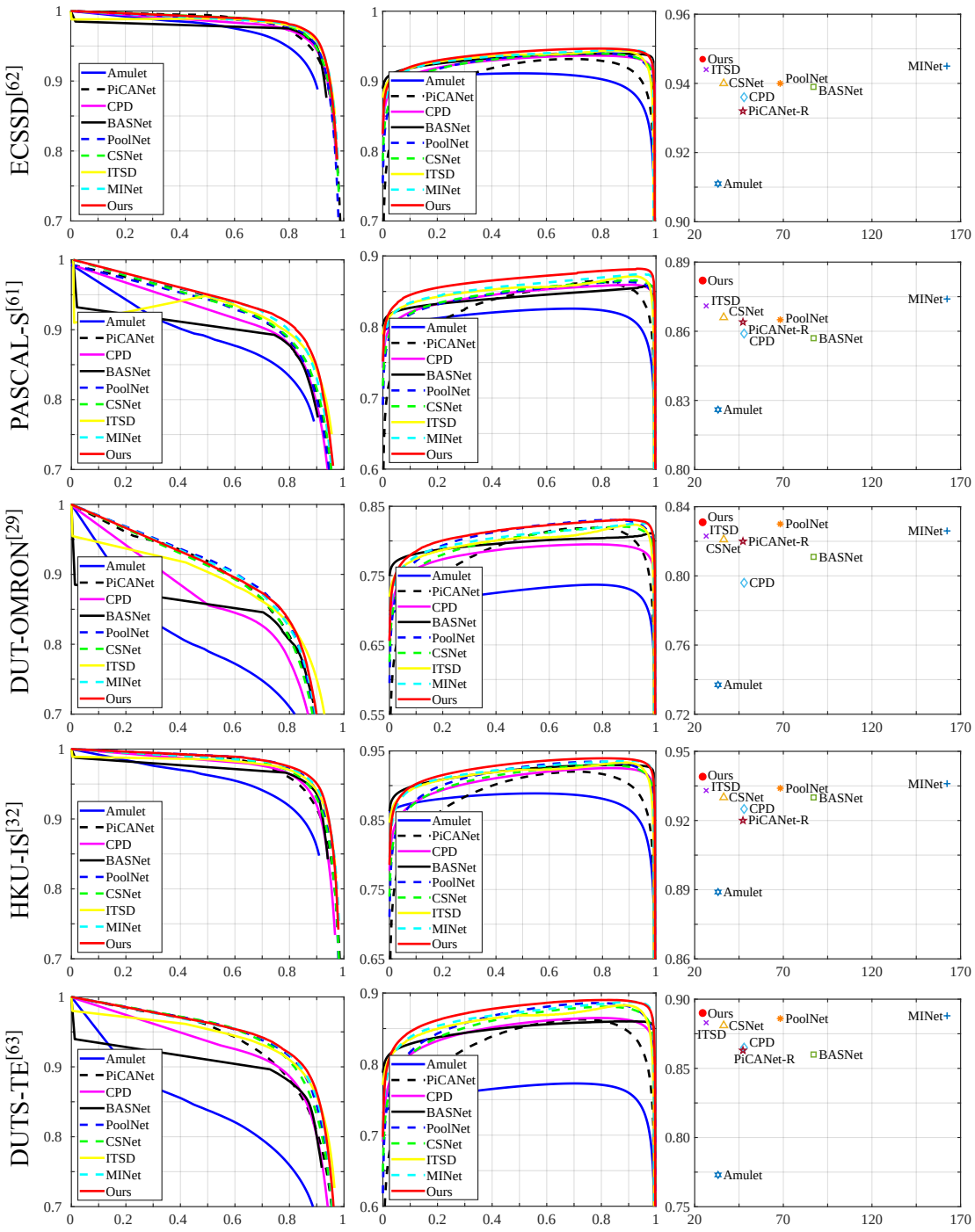


图 4.6 从左至右，每列分别为所提出方法和其他方法在五个流行显著性目标检测数据集上的：准确率和召回率曲线、F_β 值和阈值曲线、以及 F_β 值和参数量的关系对比。

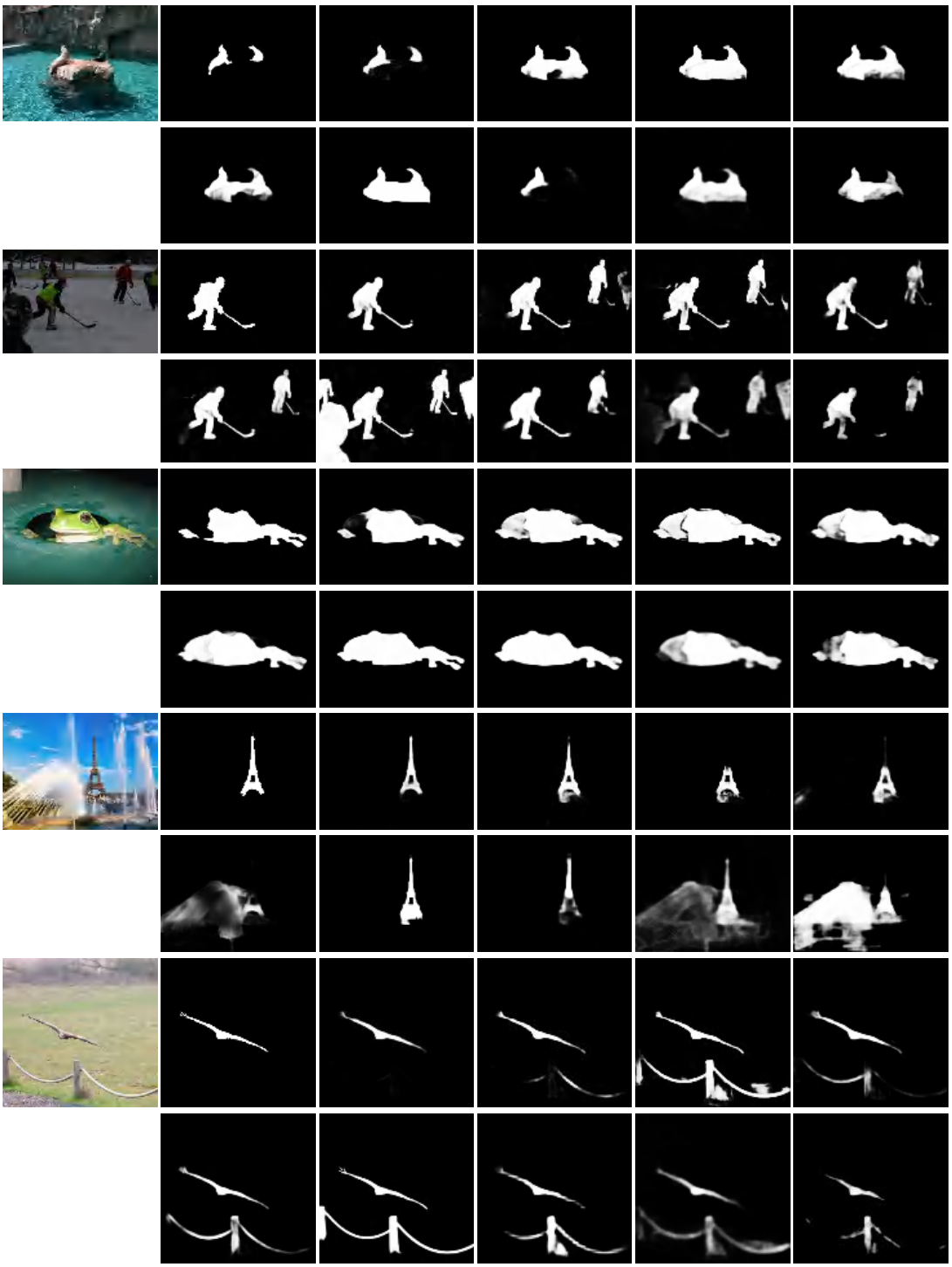


图 4.7 来自不同数据集的显著性图（一）。每组图像对应的标签分别为: 输入图像、标注图像、本方法、MINet[219]、ITSD[218]、CSNet[189]、PoolNet[215]、BASNet[54]、CPD[47]、PiCANet[51] 以及 Amulet[179]。相比于其他方法，本模型不仅能从杂乱的背景中更好地定位出显著性物体，而且生成的显著性图更加完整。

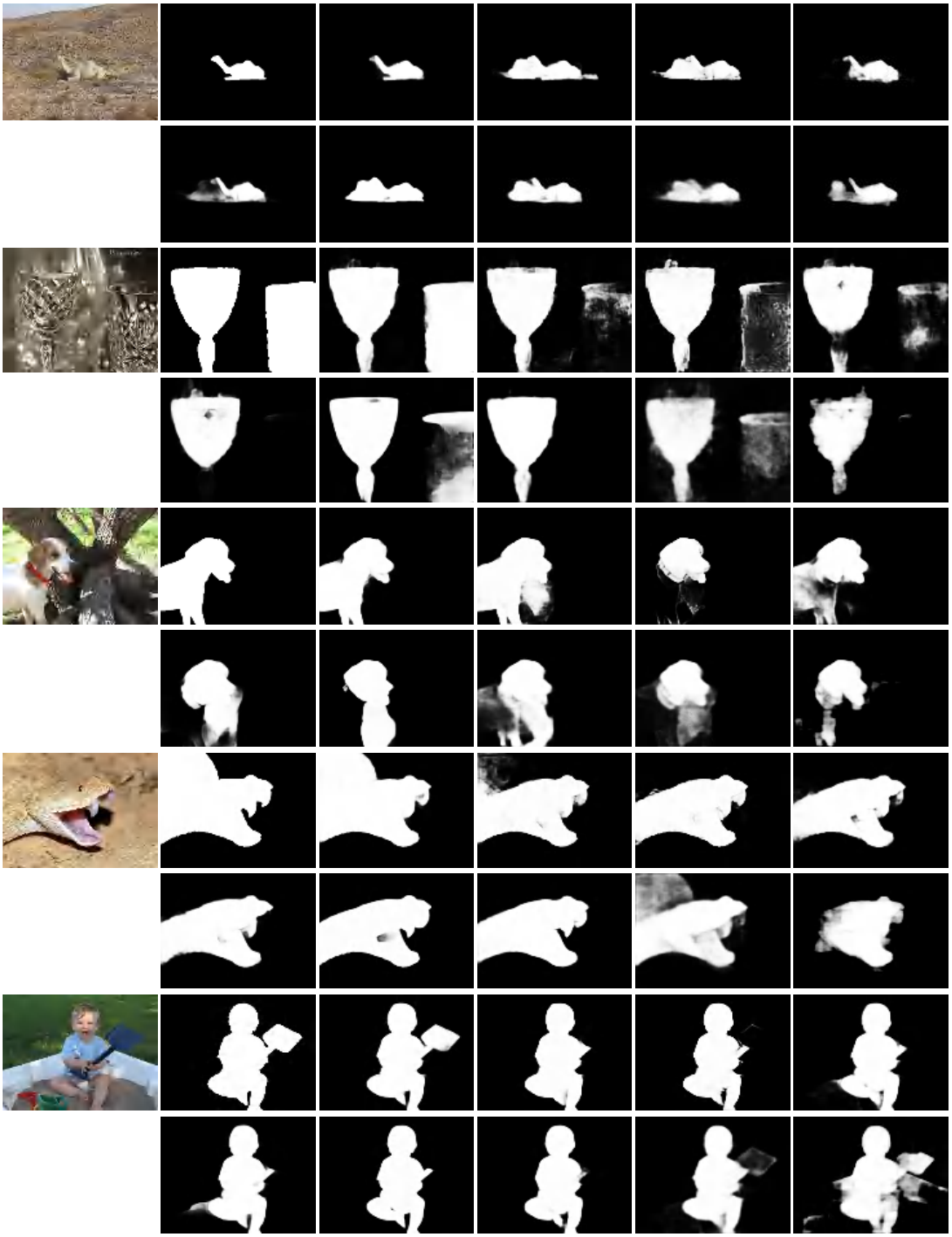


图 4.8 来自不同数据集的显著性图（二）。每组图像对应的标签分别为: 输入图像、标注图像、本方法、MINet[219]、ITSD[218]、CSNet[189]、PoolNet[215]、BASNet[54]、CPD[47]、PiCANet[51] 以及 Amulet[179]。相比于其他方法，本模型不仅能从杂乱的背景中更好地定位出显著性物体，而且生成的显著性图更加完整。

表 4.8 CII 和 RGC 在其他 U 型结构上的泛化效果实验。每列最优结果以**粗体**突出显示。

方法	引入 CII	引入 RGC	DUT-OMRON[29]			DUTS-TE[63]		
			$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$
BASNet[54]	✗	✗	0.811	0.057	0.836	0.860	0.047	0.866
	✓	✗	0.837	**0.053**	0.833	0.890	0.037	0.885
	✓	✓	**0.839**	0.054	**0.840**	**0.895**	**0.036**	**0.891**
PoolNet[215]	✗	✗	0.830	0.055	0.836	0.886	0.040	0.883
	✓	✗	0.831	0.054	0.838	0.894	0.035	0.891
	✓	✓	**0.839**	**0.052**	**0.842**	**0.900**	**0.034**	**0.892**

PR、F-measure 曲线和视觉对比：除了量化的数值比较以外，本小节还在图 4.6中的第一和第二列分别展示了在五个数据集上的 PR 曲线和 F-measure 值曲线的对比。从中可见本模型（红色实线）的 PR 和 F-measure 曲线在大部分数据集上达到了与其他现有方法相当的表现，甚至在部分数据集上表现得更好。尤其是在颇具挑战性的 PASCAL-S 数据集上，本模型在大部分阈值下超过了几乎其他所有方法，值得一提的是本方法只在骨干网络之外引入了 3.96% 的额外参数。本模型在仅需要更少参数的前提下达到了更好的性能（图 4.6第三列中的实心红点）。这也体现了本模型的简洁与鲁棒。

本段在图 4.7和图 4.8中展示了一些具有代表性的例子来可视化评估本模型的效果。可以很容易地看出，本模型不仅可以准确地突出显著性物体，还能在几乎所有情景下将它们完整地分割出来。不同于文献[187, 208, 220, 221] 中的模型，本模型无需对边缘像素区域引入额外的监督信息。这表明本模型在增强多尺度特征间信息交互的有效性。本模型可以生成具备更强语义与更准确位置信息的特征。

第四节 讨论

4.4.1 预测错误案例分析

图 4.9展示了所提出方法一些典型的预测错误样例。通过观察这些错误的例子，可以发现其中的大部分可被归因于以下三种情形：复杂的背景，遮挡，以及前景与背景间的低对比度。如图中前两行所示，这些例子中的显著性物体周围都有着非常杂乱的背景。在中间的两行，一些无关物体部分地遮挡住了显著性物体。而在最后两行，一些显著性物体和其背景有着非常相似的颜色。在一些情形下，本模型只检测到了部分的显著性物体。而在其他情形中，背景中非

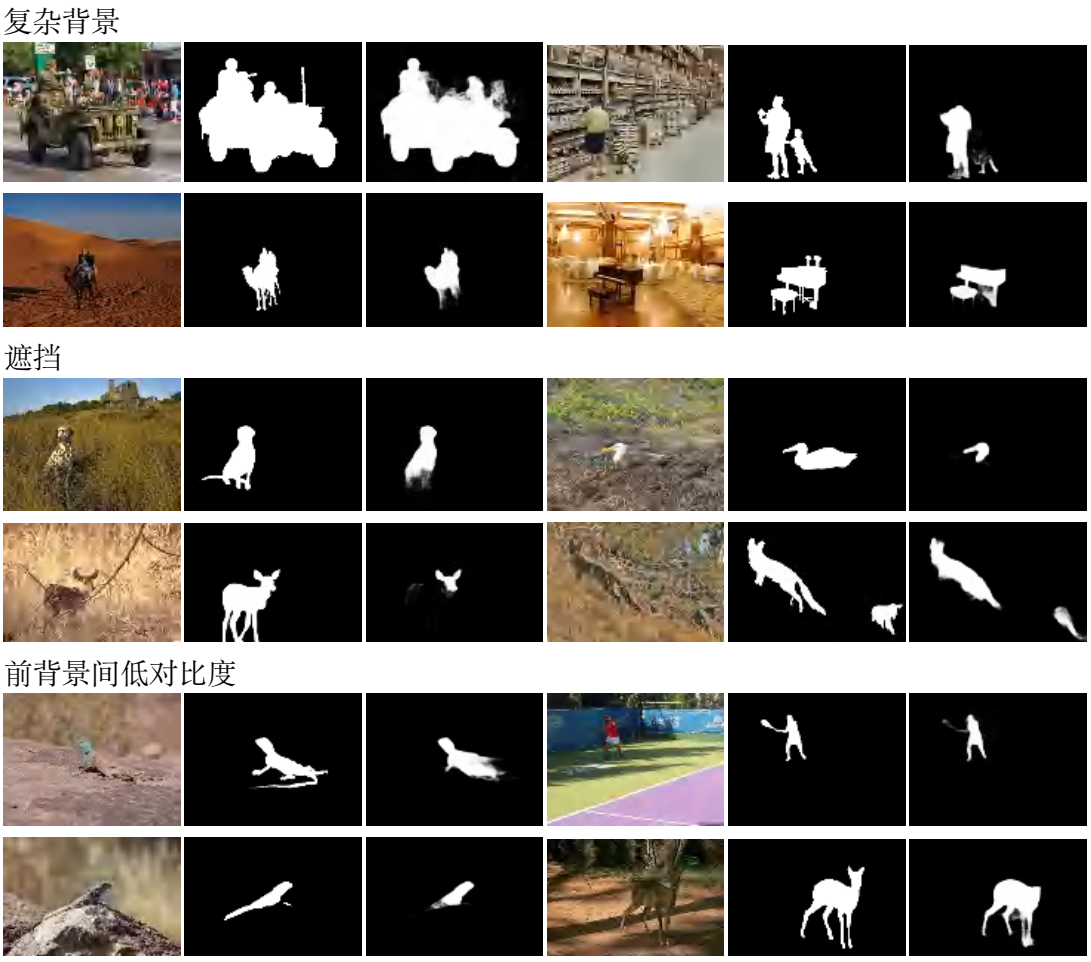


图 4.9 从多个数据集中选取的部分典型的预测错误案例。

显著性的区域被错误地预测为了显著性物体。作者认为即使对于人类而言，要精确地分辨出上述大部分场景中的前景与背景间的边界也是很困难的。

4.4.2 与其他 U 型结构的协作

为了探究 CII 和 RGC 的泛化能力，本小节将它们应用到了两个颇具代表性的基于 U 型结构的现有显著性目标检测方法上：BASNet[54] 和 PoolNet[215]。由于 CII 和 RGC 并不依赖于某种特定的自底向上和自顶向下通路的实现形式，本段仅替换了其他方法位于上述两条通路之间的短连接部分。数值结果展示在表格 4.8中。可以看到，无论将 CII 策略应用于哪个方法均获得了性能的提升。一个有趣的现象是，CII 策略在 BASNet 上相比于 PoolNet 带来了更多提升（在 F_β 值上平均提升 3.05% 对比于 0.45%）。作者认为这可能是因为 PoolNet 根据观察

经验额外地向自顶向下通路的每个层级引入了全局信息，从而在一定程度已经达到了跨尺度信息交互的效果。与之相反，BASNet 只使用了基本的 U 型结构，所以 CII 的引入带来了更多提升。当引入 RGC 模块时，上述两个方法的整体性能可以进一步在几乎所有数据集和指标上获得提升，达到新的领先表现。上述实验表明 CII 和 RGC 能够被用来补充增强那些关注于提升 U 型结构的自顶向下和/或自底向上通路的方法。

第五节 本章小结

本章关注了之前 U 型结构中位于自顶向下和自底向上通路间相互独立的连接，提出对这些连接进行集中化处理以有效鼓励多尺度特征间的信息交互，从而获得更具表征力的特征。为证明所提出的集中信息交互（CII）策略的可行性，本章提出了一个相对全局校准模块（RGC）来与其更好协作。通过将 CII 和 RGC 结合进经典的 U 型结构，新得到的模型在仅仅引入了少量额外的参数和 FLOPs 的情况下。在五个广泛使用的显著性目标检测数据集获得了相较于现有领先方法更优的表现。本章提出的策略和模块与 U 型结构的自顶向下和自底向上通路相独立，因此可以被灵活应用到任何基于 U 型结构的模型上。RGC 的良好表现也表明了进一步研究和设计能够与所提出的 CII 策略更好协作的新多尺度模块具有广阔的前景。

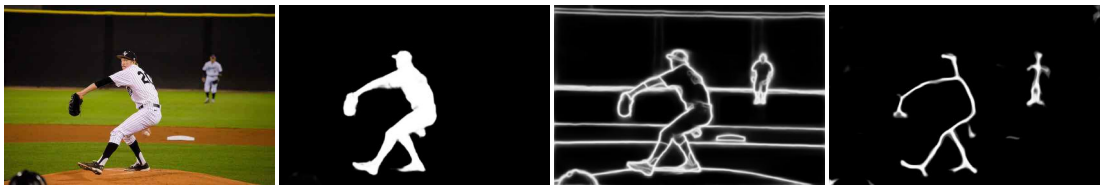
第五章　基于高效特征动态选择与融合的多任务协同学习算法

第一节　引言

随着移动设备的迅速普及，越来越多基于深度学习的计算机视觉应用已经从计算机平台移植到移动平台中。许多浅层的计算机视觉任务得益于它们具有类别无关的通用图像属性，也逐渐成为了移动设备的基本组件之一。例如，在使用智能手机拍照时，许多作为支撑的视觉任务在后台运行，以帮助用户获得更美观的照片并同时提供实时效果预览。单摄像头的智能手机通常利用显著性目标检测任务来模拟本来需要深度信息[44, 222] 才能实现的背景虚化效果。为了帮助用户拍摄出具有更加赏心悦目的内容结构的图片，厂商通常会利用边缘检测任务[93, 223] 来提取结构信息以辅助用户构图。而骨架提取任务[111] 也在辅助拍照上起着重要的作用，它可以通过检测出待拍摄对象的躯体动作来示意和指导用户摆出更加有趣的姿势。然而，由于移动设备的存储和计算资源有限，为每个不同的应用存储预训练过的模型并逐个执行多个不同的任务既不方便且又低效。

一个可行的解决方案是在一个模型中同时执行上述多个任务，但其中主要存在以下两个挑战：一是如何对不同的任务同时进行学习，二是如何解决不同任务在特征域和优化目标之间的分歧。大多数先前的工作[57, 144, 215, 221] 通过人工观察不同任务所具有的特性，进而为每个任务手动设计专门的网络结构来解决第一个挑战。他们通常会假设所有被联合学习的任务之间是互补的，并且其中有一些为辅助任务，（例如，利用额外的边缘信息来帮助显著性目标检测任务在边缘区域获得更精确的分割）。而辅助任务的性能会被牺牲和忽略。但是当面临第二个挑战，即所要同时解决的任务之间差异很大时，如图 5.1所示，直接应用这些现有的方法往往会失败。如表格 5.3 中第三行所示，当与其他两个不同的任务联合训练时，骨骼提取任务的性能会被严重影响。

以前工作中模型的设计标准通常是面向特定任务的，这极大地限制了它们在其他任务上的可适用性[144]。从网络架构的角度来看，尽管本章将要同时解决的三个任务之间是不同的，但是它们都需要多层级的特征，只是偏好和程度不同。显著性目标检测侧重于提取同质区域的能力，因此更依赖于高层级特征[44]。



输入图像　　显著性结果　　边缘结果　　骨架结果

图 5.1　当同时学习显著性、边缘和骨架时，一个可能发生信息冲突的例子。输入图像中远处的人不显著，但有包含骨架结构。而边缘检测任务需要检测所有可能的边缘区域，无论是否显著或者拥有骨架结构。以上所有的预测都是通过本章提出的方法直接预测得到。

边缘检测旨在检测出精确的边界，因此需要更多的低层级特征来锐化由较深层特征生成的粗糙边缘图[45, 193]。而骨架提取[108, 110] 更喜欢低、中和高层级信息的适当组合，以检测不同尺度（或厚或薄）的骨架。因此，一个自然而然的问题是：是否有可能设计一种架构，能够将这三个迥异的、浅层的视觉任务涵盖进一个统一且端到端可训练的网络中，同时不牺牲任何一个任务的性能。

考虑到每项任务的不同特点，本章提出了一个新的、统一的模型来解决上述挑战。具体来说，该模型包含一个共享的主干网络和三个设计相同的任务分支，如图 5.2所示。为了便于每个任务分支在主干网络的不同层级特征中自动选择适当的特征，本模型引入了一种动态特征融合策略，它能够以端到端学习的方式动态选择对各任务有利的特征。这种动态策略可以极大地简化网络架构构建的过程，并促进主干网络适当地调整其参数来适应同时解决多个问题的需求。紧接着，一个任务自适应的注意力模块被引入，该模块以分离-聚集的方式实现不同任务分支之间的信息交换。通过耦合以前通常被设计为互相独立的任务分支，可以有效避免网络的偏向性优化。本章提出的模型简单易用，可以直接在单个显卡上进行完整的训练。当输入 300×400 分辨率的图像并同时执行这三个任务时，本模型可以在不牺牲性能的情况下达到 40FPS 的速度。

为了评估所提出的模型的性能，本章将其与这三个任务各自目前最好的方法分别进行了比较。实验结果表明，在多个广泛使用的测试基准上，本模型都优于现有的、面向单一任务设计的方法。具体来说，本章将所提出方法与当前最好的显著性目标检测工作在六个流行的数据集上进行了测试对比，在 F-measure 指标上平均获得了 1.2% 的性能增益。对于骨架提取任务，所提出方法在 SK-LARGE 数据集[109] 上相对于目前最好的方法在 F-measure 指标上提高了 1.9%。此外，为了让读者更好地理解本章提出的方法，本章对所提出的架构

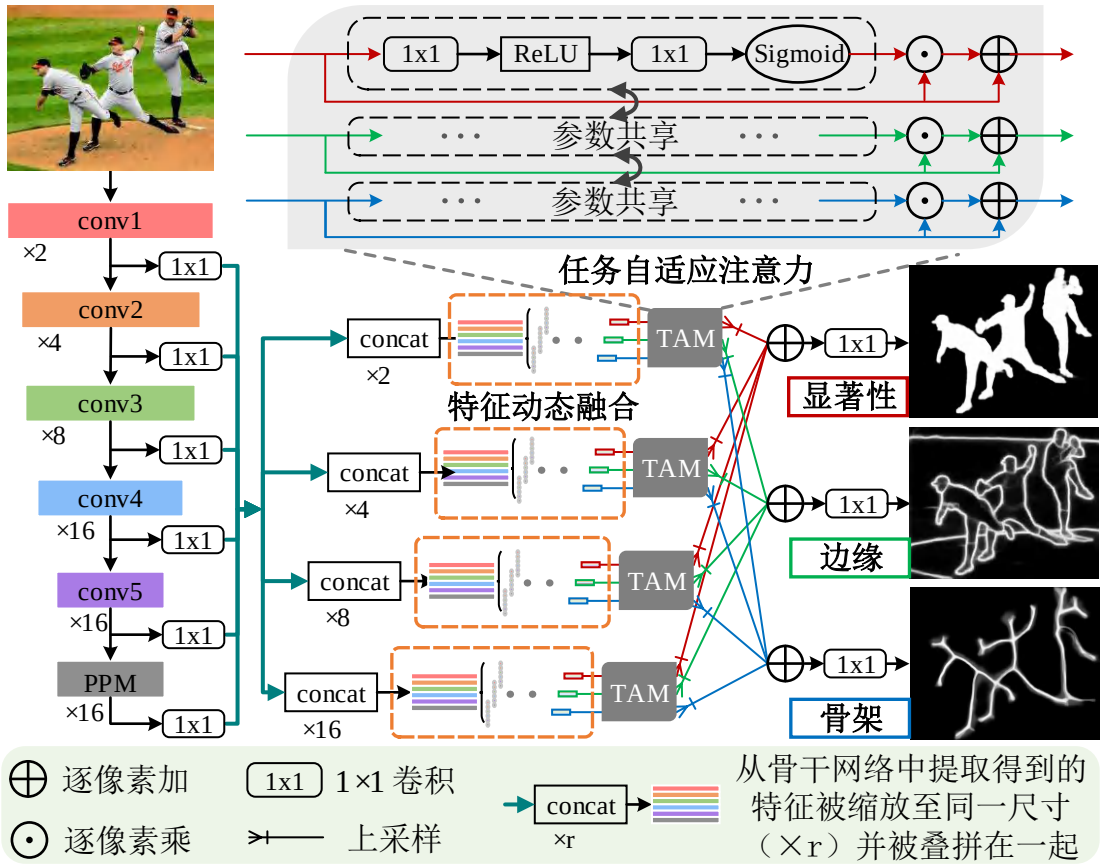


图 5.2 本章所提出的方法的整体流程图（彩色图视觉效果最佳）。

的各部分进行了大量的消融实验。综上所述，本章的贡献可以概括为:

- 本章设计了一种动态特征融合策略，该策略可以根据不同的输入内容和具体任务自适应地探索多尺度特征的组合形式，并能够在一个统一且端到端的框架中以 40FPS 的速度同时解决三个迥异的任务。
- 为了有效平衡不同任务在数据分布和特征偏好之间的差异性，本章提出了一种任务自适应注意力模块。该模块以极小的计算代价搭建起不同任务之间必要的信息交互，能够有效避免不同任务之间可能存在的冲突，使各任务获得更好的整体收敛效果。
- 本章的多任务方法与那些针对各任务而专门设计的、目前最好的方法分别进行了比较，并获得了更好的性能。

第二节 多任务协同网络

本小节提出让网络自身去根据每个任务的偏好和每个输入的内容动态地选择不同阶段的特征，而不是试图手工地设计一个可能适用于所有三个任务的架构，如第一节所述。

5.2.1 总体流程

本章在一个可以端到端训练的统一网络中，在多个独立的数据集上同时完成了三个不同的任务的学习和预测（即，用于显著性目标检测的 DUTS 数据集[63]，用于边缘检测的 BSDS 500[85] 和 VOC Context[190] 数据集，用于骨架提取的 SK-LARGE[108] 或 SYMPASCAL[110] 数据集）。需要强调的是，本章使用上述所有的数据集的方式均与现有的、为每个任务所专门设计的、单一目标的方法对这些数据集的使用方法保持一致，而无需额外处理。

图 5.2显示了所提出框架的总体流程。本模型使用 ResNet-50[11] 网络作为特征提取器。本模型将 `conv_1` 输出的特征图作为 S_1，并将 `conv2_3`，`conv3_4`，`conv4_6`，和 `conv5_3` 的输出分别作为 S_2 至 S_5。参考像素级预测任务中的通常做法，本模型将 `conv5` 中的 3×3 卷积层的空洞步进设置为 2。此外，参考方法[41, 215]，本模型在 ResNet50 网络的顶部添加了一个金字塔池化模块（Pyramid Pooling Module, PPM）[129] 以捕获更多的全局信息，并将其输出记为 S_6。不同于大多数现有的单任务方法中将手工设计的特定的特征融合策略嵌入到网络结构设计中，本章提出通过利用一系列有着不同输出下采样率的动态特征融合模块（Dynamic Feature Integration Modules，DFIMs，图 5.2中橙色虚线圆角矩形）来动态地、适应性地为三个任务选择并融合从骨干网络中提取的特征（即 $\{S_i\}$，其中 $1 \leq i \leq M$）。以本章模型为例，当以 ResNet-50[11] 网络作为特征提取器时 $M = 6$。然后紧接每个 DFIM 之后会连接一个任务自适应注意模块（Task-adaptive Attention Module，TAM），以达到各任务间信息的分配与调整目的，防止网络有偏向地优化。最后，TAMs 中对应于每个任务的输出特征图被上采样并分别求和，然后再分别通过一个 1×1 卷积层用于各任务最终的结果预测。

5.2.2 动态特征融合

如在许多现有的多任务方法[141, 144–146] 中所提到的那样，不同任务所偏好的特征差异很大。而且大多数方法都依赖于在单个数据集内包含多种标注的设

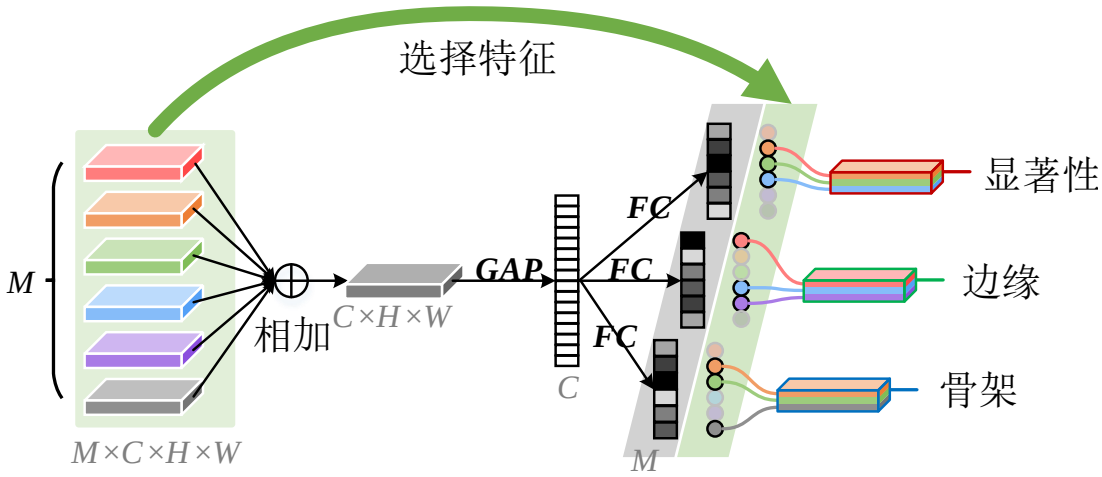


图 5.3 动态特征融合模块的详细结构。该模块以从骨干网络中提取的多尺度特征集合作为输入，首先将其空间大小调整为相同的尺寸，然后每个任务会动态选择不同阶段的特征进行融合。图中，GAP 表示全局平均池化操作，FC 表示全连接层。

定，通常而言这是很难获得的。与现有方法不同的是，本模型直接利用面向不同任务而标注的多个独立数据集的训练数据，而这更加适合不同任务之间所需的特征互相冲突的情况，如图 5.1所示。为了解决这个问题，本章提出了动态特征融合模块，它在训练和测试期间会根据每个任务偏好和不同输入内容动态地调整对应的特征融合策略。相比于现有方法通过人工观察不同任务的特点，进而手动地从骨干网络中融合特定级别特征，DFIM 可以自动且自适应地学习这些特征融合策略。

具体来说，每个 DFIM 均以从骨干网络提取的特征集 $\{S_i\}$ 作为输入，并且每个 DFIM 对应的输出下采样率 $\times r$ 在网络定义期间便已经被确定。如图 5.3所示，对于下采样率为 $\times r$ 的 DFIM，本模型首先将特征集 $\{S_i\}$ 中的所有特征分别通过一个 1×1 的卷积层和双线性插值映射到具有相同的特征维度（$C\times$）和下采样率（$\times r$），并记为 $\{S_i^r\}$。为了使 DFIM 具有能够覆盖所有特征的视野，本模型紧接着对 $\{S_i^r\}$ 进行求和，并在其后连接一个全局平均池化（Global Average Pooling，GAP）层以获得一个紧全的全局特征（$C\times$），就如 SENet[212] 做的那样。对于每个任务 $t \in \{$显著性,边缘,骨架$\}$，本模型分别使用一个单独的全连接（Fully-Connected，FC）层将 $C\times$ 维的特征映射到 $M\times$ 维，然后应用 softmax 算子进一步将 $M\times$ 维的特征转换为概率的形式 $\{p_i^{r,t}\}$ $(1 \le i \le M)$，而该概率可用作为选择特征的指标。由于并非来自骨干网络的每个阶段的特征总是有益的，

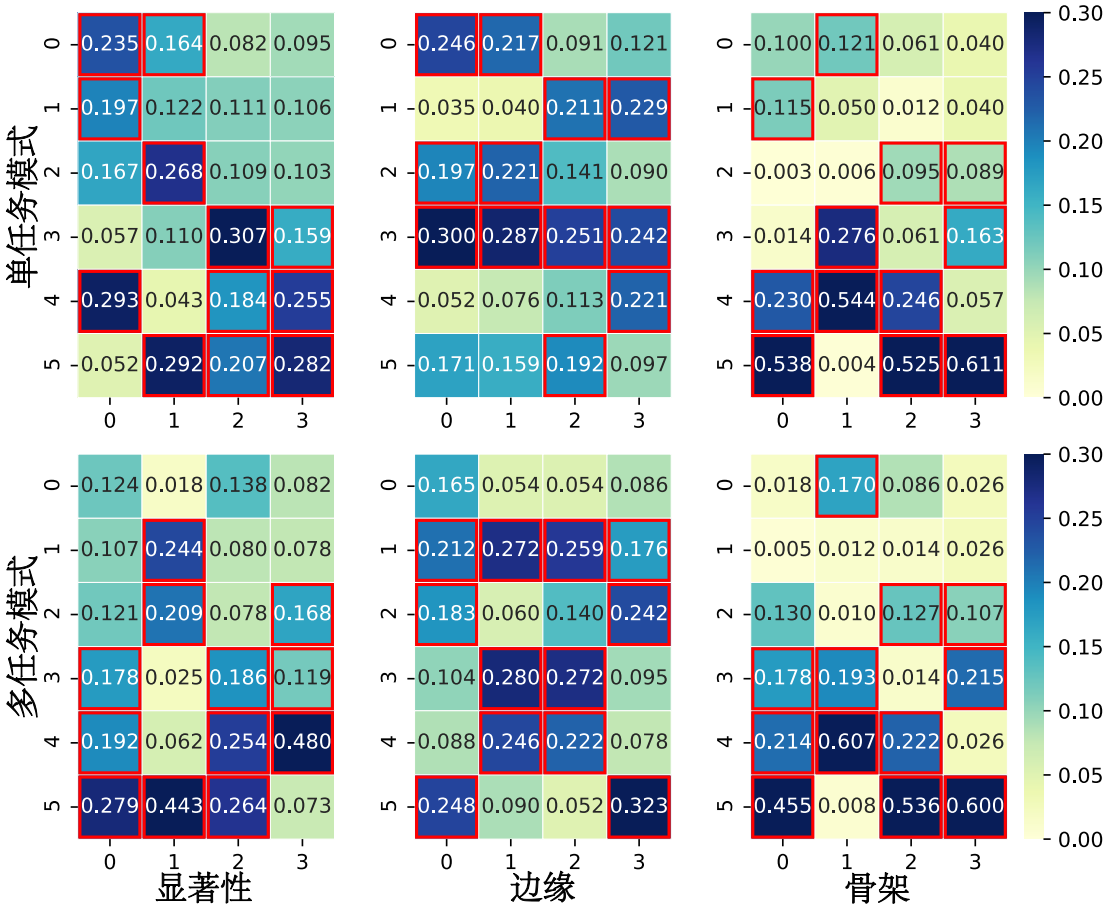


图 5.4 通过学习得到的各层级特征被不同 DFIM 所选择的权重情况。在每个子图中，行表示 DFIMs 的索引序号，列表示特征所属的层级。每个 DFIM 只保留权重高于前 50% 的特征（红色矩形）。

不同于现有的浅层视觉方法[45, 93, 111] 通常对所有的 $\{S_i^r\}$ 保持紧密的连接，本模型提出仅保留其中激活最高的一半的连接，并记为 $\{S_i^{r,t}\}$ $(1 \le i \le M)$:

$$S_i^{r,t} = \begin{cases} p_i^{r,t} * S_i^r, & \text{if } p_i^{r,t} \ge median(\{p_i^{r,t}\}) \\ 0, & \text{else,} \end{cases} \tag{5.1}$$

其中，$median(\cdot)$ 表示取中值操作，且 $1 \le i \le M$。于是对于面向任务 t 的下采样率为 $\times r$ 的 DFIM 的输出可通过如下公式获得：

$$D^{r,t} = \sum_i S_i^{r,t}. \tag{5.2}$$

本章将在章节 5.3.2.2中对这种设计的出发点和效果进行详细地实验和分析。

通过排列一系列覆盖不同下采样率范围的 DFIMs，便可以得到通过动态结合而得到的特征图集合 $\{D^{r,t}\}(r \in \{2,4,8,16\}, t \in \{\text{显著性,边缘,骨架}\})$，如图 5.2所示。由于特征融合策略完全取决于输入内容和任务类型，网络能够以端到端的方式在更宽泛、更灵活的特征组合空间内学习对应于每个输入和任务的融合策略。

5.2.3 任务自适应注意力

当一个模型同时利用来自多个独立数据集的训练数据时，它们之间的域偏移[137, 139] 问题不可忽视。如何有效地整合来自差异巨大的各数据集中的信息对于维持所有任务的整体性能而言是至关重要的。如图 5.4的第一行（单任务）所示，不同任务在对不同特征层级的偏好之间差异很大。如果直接使用由 DFIMs 生成的任务特定特征图 $\{D^{r,t}\}(r \in \{2,4,8,16\})$ 来预测每个任务，则可能出现一些任务分支回传到网络共享部分的梯度明显偏离于其他任务的情况，从而导致网络整体的优化方向偏转到局部最小值并引起欠拟合。

为此，本章提出在来自骨干网络的共享特征被动态融合并为每个任务定制学习之后，让网络具有智能地为不同任务分配信息的全局能力。如图 5.2的右上角所示，来自 DFIM 的输出特征图 $\{D^{r,t}\}(t \in \{\text{显著性,边缘,骨架}\})$ 在 $\times r$ 的下采样率下被进一步输送到 TAM 中。在每个 TAM 模块中，本模型首先将输入特征图 $D^{r,t} \in \mathbb{R}^{C \times H \times W}$ 送入到一个 1×1 卷积层（$f_1^{1\times1}$）中，以减少特征图上采样后的累加操作所导致的混叠效应（公式 (5.2)），随后是一个 ReLU 激活函数以引入非线性。然后本模型利用另一个 1×1 卷积层（$f_2^{1\times1}$）来映射跨通道维度之间的信息。之后本模型使用一个 sigmoid 层（σ）来计算相应的空间注意力图 $A^{r,t} \in \mathbb{R}^{C \times H \times W}$：

$$A^{r,t} = \sigma(f_2^{1\times1}(ReLU(f_1^{1\times1}(D^{r,t})))), \tag{5.3}$$

其中 $f_1^{1\times1}$ 和 $f_2^{1\times1}$ 中的参数在各任务间共享。有了输入特征图和它对应的注意力图，就可通过以下公式获得 TAM 最终的输出特征图:

$$T^{r,t} = D^{r,t} \odot (1 + A^{r,t}), \tag{5.4}$$

其中 $\odot$ 表示逐元素乘法。$D^{r,t} \odot A^{r,t}$ 的作用是输入特征图的残差。

为了达到跨任务间信息交换的目的，本模型在任务间共享 TAM 中可学习的参数。与直接使用每个任务的独立输出相比，对所有任务关系的额外建模使

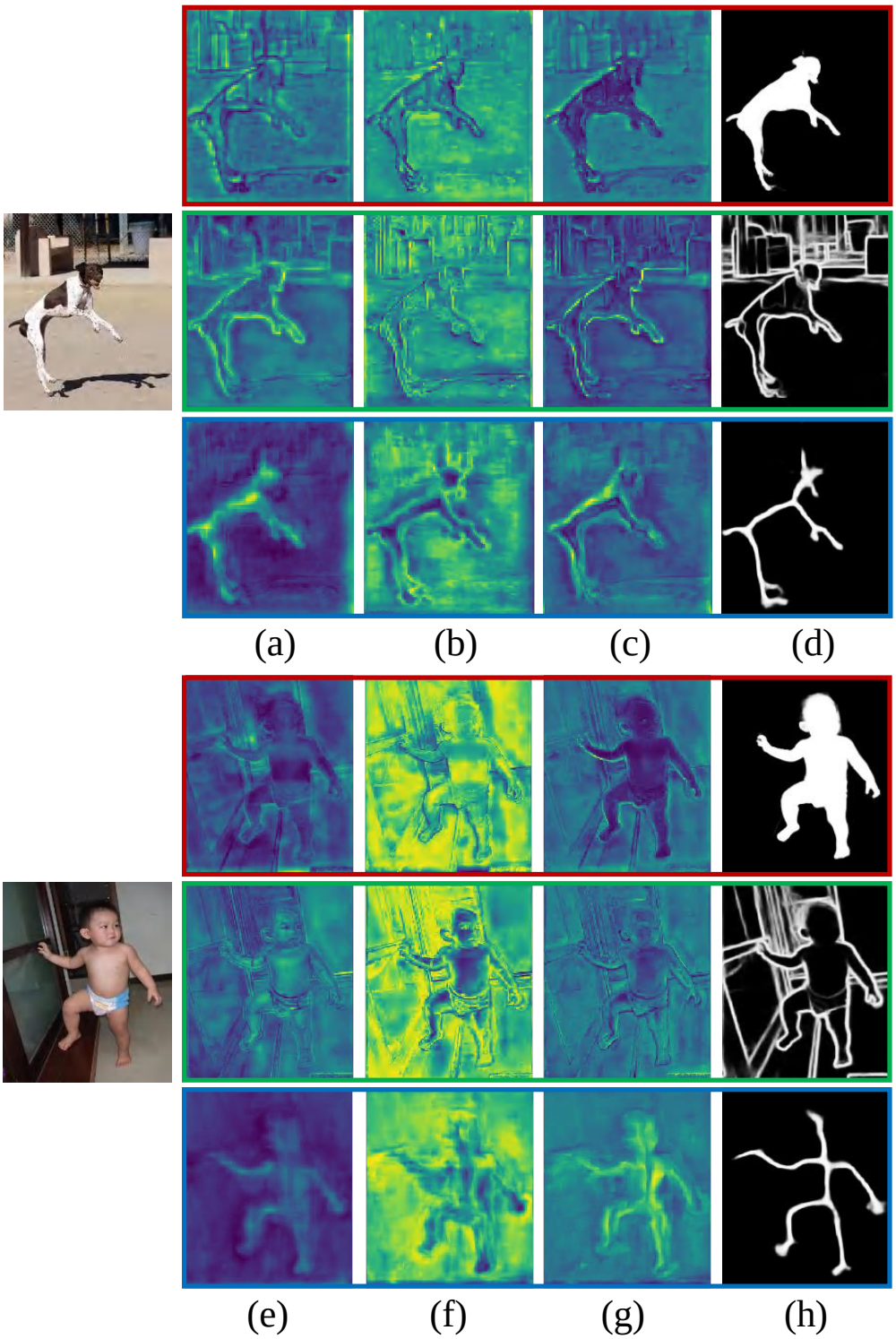


图 5.5 在 TAM 模块前、中和后的特征可视化对比。(a，e) TAM 之前；(b，f) TAM 中；(c，g) TAM 之后；(d，h) 预测结果。可以看出，TAM 可以自适应地为每个任务精细化调整其对应的特征图。各样例中从上到下依次为: 更容易分辨的显著性区域，更锐利的边缘和更突出的骨架。

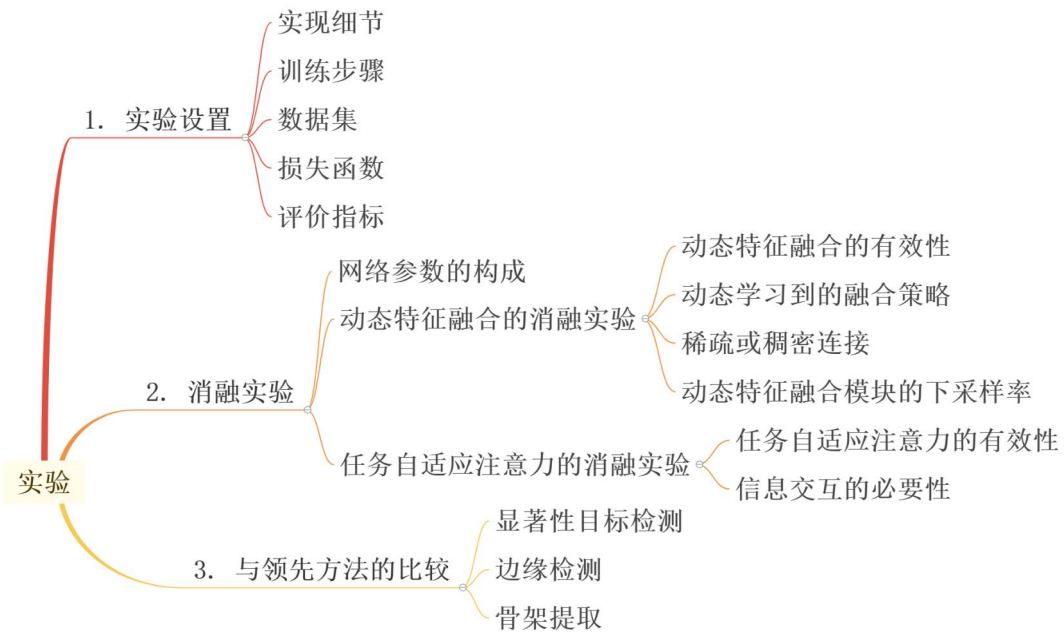


图 5.6 本章实验部分的总体实验方案导图。

DFIM 能够通过同时考虑输入内容和所有任务的特性，来自适应地调整每个任务对共享骨干网络的影响。即使在每个 DFIM 之后，各任务分支中的特征均已被面向不同任务而深度分离和调整，TAM 也能通过物理结构上的设计来强制保证跨任务间的信息交互。这与现有的方法[57, 144, 145] 大不相同，他们通常会保证不同任务的分支相互独立直到最后的结果输出。

为了帮助读者更好地理解上述优势，本小节在图 5.5中可视化了 TAM 模块周围网络部分中的中间特征图。可以看到，第一行中对于显著性目标检测任务而言，在 TAM 之前（a，e），很难从背景中分辨出显著性的前景：狗（小孩）。TAM 中（b，f）学习到的注意力图有效地擦除了背景部分的激活信号。而在 TAM 之后（c，g），狗（小孩）就清楚地凸显出来了。在第二行中，对于边缘检测任务，TAM 之后（c，g）的特征图与 TAM 之前（a，e）的模糊且粗的边缘激活相比，在可能存在边缘的区域中具有明显更清晰和锐利的激活。骨架提取任务中也可以观察到类似的现象。如最后一行所示，经过 TAM 之后狗（小孩）的骨架变得更明显和集中。所有上述的现象和分析都验证了 TAM 在更好地为不同任务分配信息方面的显著效果。

第三节 实验

本小节的总体实验方案设计如图 5.6所示。本小节首先介绍了实验设置，包括实现细节，训练步骤，使用的数据集、损失函数，以及三项任务的评价指标等。接着进行了一系列消融实验来说明所提出模型的每个部分对性能的影响。最后展示了所提出方法在不同设置下的性能，并与现有的领先方法作了对比。

5.3.1 实验设置

实现细节：本章主要基于开源的 PyTorch 库[1]实现了所提出的方法。所有实验都是在一个配有一个 Intel Xeon 12 核 CPU （3.6GHz）、64GB 内存和一个 NVIDIA RTX-2080Ti 显卡的工作站上进行的。本章使用 Adam[183] 优化器来优化网络，初始学习率和权重衰减分别设置为 5e-5 和 5e-4。本模型一共训练了 12 轮，9 轮之后学习率被下降 10 倍。本模型的骨干网络部分（即 ResNet50[11]）的参数是用 ImageNet[9] 预训练过的相应模型进行初始化的，而其他所有参数都是随机初始化。除骨干部分外，本模型在每个卷积层之后使用了组归一化层（GN)[224]。除了骨干部分的批处理归一化层（BN）的参数是在训练和测试期间均被固定之外，本模型中所有其他的参数的优化策略和配置均保持一致。

训练步骤：为了以端到端的方式在三个单独的数据集上联合解决三个不同的任务，对于每次训练迭代过程，本模型分别为三个任务中的每一个任务随机采样一组图像-真值标注对。然后，依次将三组图像-真值标注对中的每一组前向传递到网络中，并计算相应的损失。最后，本模型通过简单地将三个任务计算得到的损失相加，并通过网络一次性完成梯度回传和网络参数更新。除了上述提及部分，所有其他训练步骤与典型的各单一目标方法相同。

数据集：本模型为不同的任务使用单独的数据集，每个数据集均只有一种类型的标注信息。在表格 5.1中列出了详细的训练和测试数据集的相关信息。所有数据集的使用方法都与为每个任务[93, 111, 179] 专门提出的现有单一目标任务方法对相应数据集的使用方式一致，均不包括任何额外的预处理过程。

损失函数：本章中涉及的三个任务所使用的损失函数与大多数以前所对应的单目标任务方法相同。具体而言，对于显著性目标检测本模型使用标准的二元交叉熵损失[44, 179]；而对于边缘检测和骨架提取任务，本模型使用平衡二元交叉熵损失[45, 93, 111] 。本模型使用的损失函数的详细公式如下。给定某图像的

[1] https://pytorch.org

表 5.1 用于训练和测试的数据集介绍。

任务	训练集	样本数量	测试集	样本数量
显著性	DUTS-TR[63]	10,553	ECSSD[62], PASCAL-S[61], DUT-OMRON[29], SOD[60], HKU-IS[32], DUTS-TE[63]	1,000, 850, 5,166, 300, 1,447, 5,019
边缘	BSDS500[85] 和 VOC Context[190]	300 + 10,103	BSDS500[85]	200
骨架	SK-LARGE[109]	746	SK-LARGE[109]	745
	SYM-PASCAL[110]	648	SYM-PASCAL[110]	788

预测结果图 $\hat{Y}$ 及其对应的真值标注图 Y，对于所有的像素 (i,j)，标准二元交叉熵损失可以计算为:

$$\begin{aligned}\mathcal{L}_s(\hat{Y},Y) = -\sum_{i,j}[&Y(i,j)\cdot\log\hat{Y}(i,j)+\\&(1-Y(i,j))\cdot\log(1-\hat{Y}(i,j))],\end{aligned} \tag{5.5}$$

而平衡的二元交叉熵损失计算为:

$$\begin{aligned}\mathcal{L}_b(\hat{Y},Y) = -\sum_{i,j}[&\beta\cdot Y(i,j)\cdot\log\hat{Y}(i,j)+\\&(1-\beta)\cdot(1-Y(i,j))\cdot\log(1-\hat{Y}(i,j))],\end{aligned} \tag{5.6}$$

其中 $\beta = |Y^-|/|Y^+ + Y^-|$，而 Y^+ 和 Y^- 分别指前景和背景像素。

- 显著性目标检测:

$$\mathcal{L}_{sal}(\hat{Y}_{sal},Y_{sal}) = \mathcal{L}_s(\hat{Y}_{sal},Y_{sal}). \tag{5.7}$$

- 边缘检测:

$$\mathcal{L}_{edg}(\hat{Y}_{edg},Y_{edg}) = \mathcal{L}_b(\hat{Y}_{edg},Y_{edg}), \tag{5.8}$$

  其中 $\mathcal{L}_b$ 的 β 中的 Y^+ 指边缘像素，而 Y^- 指非边缘像素。
- 骨架提取:

$$\mathcal{L}_{skl}(\hat{Y}_{skl},Y_{skl}) = \mathcal{L}_b(\hat{Y}_{skl},Y_{skl}), \tag{5.9}$$

  其中 $\mathcal{L}_b$ 的 β 中的 Y^+ 指骨架像素，而 Y^- 指非骨架像素。

总损失计算为三项任务损失的简单总和，即它们的比重相同：

$$\mathcal{L} = \mathcal{L}_{sal}(\hat{Y}_{sal}, Y_{sal}) + \mathcal{L}_{edg}(\hat{Y}_{edg}, Y_{edg}) + \mathcal{L}_{skl}(\hat{Y}_{skl}, Y_{skl}). \tag{5.10}$$

评价指标：

- 显著性目标检测：本章使用四个广泛使用的评价指标对显著性目标检测任务的性能进行评估和对比：准确率-召回率（PR）曲线，特征相似度（F-measure，F_β）、结构相似度（S-measure，S_α）和平均绝对误差（MAE），它们的具体概念和计算方式可以在章节 2.1.4中找到。
- 边缘检测：本模型遵循领域通用做法[45, 93]，在评估之前使用了标准的非极大值抑制（Non-Maximal Suppression，NMS）算法[87] 来获得细化的边缘。为了产生二值的边缘图，有以下两种选择来设置阈值。一种是对数据集中的所有图像使用固定的阈值，从而在整个数据集上达到最佳的整体性能，称之为最优数据集尺度（Optimal Dataset Scale，ODS）；另一种是为每幅图像单独选择一个最佳阈值，称为最优图像尺度（Optimal Image Scale，OIS）。对于 ODS 和 OIS，其相应的 F-measure 分数计算如下:

$$F_m = \frac{2 \times Precision \times Recall}{Precision + Recall}. \tag{5.11}$$

- 骨架提取：本章遵循[105] 中的评估方案，使用准确率-召回率（PR）曲线和最大 F-measure 分数（$MaxF_m$）作为评价指标。参考本领域通用做法，预测的骨架图在评估之前会通过 NMS 算法进行细化。为了获得 PR 曲线，给定一个 NMS 细化后的骨架图，本章首先将其阈值化为一个二值图，然后将其与相应的真值标注图进行匹配。在匹配期间，在预测图中的骨架像素和真值标注中的骨架像素之间允许有着微小的定位误差。通过对预测的骨架图采用不同的阈值，可以得到一系列的准确率和召回率对来绘制 PR 曲线。最大 F-measure 则是在整个数据集的最佳阈值下通过公式 (5.11) 获得。

5.3.2 消融实验

本小节首先分析了所提出的模型的参数组成。然后通过分别在单任务和多任务设置下进行实验来研究和分析了所提出的 DFIM 算法的有效性。最后展示了 TAM 在促进更好的整体收敛效果和性能方面的能力。

表 5.2 网络的参数构成。可以看出，特征提取器（ResNet-50 和 PPM）和共享的部分占据了大部分比例。

总计: 29.57M						
共享参数: 27.01M （91.34%）				任务特定参数: 2.56M （8.66%）		
ResNet-50	PPM	DFIMs	TAMs	显著性	边缘	骨架
23.46M	1.31M	1.42M	0.83M	0.85M	0.85M	0.85M
79.34%	4.43%	4.80%	2.81%	2.87%	2.87%	2.87%

5.3.2.1 网络参数的构成

在表格 5.2中列出了网络参数的组成。可以看出，91.34% 的参数用于在任务间共享，其中特征提取器部分（ResNet-50 & PPM）占 91.71%。而 DFIMs 和 TAMs 的共享部分只引入了 2.25M （8.33%）额外的参数。每个任务分别拥有 0.85M （2.87%）个任务独享的参数。参数的极化分布从侧面证明了所提出方法的有效性和高效性。通过高效利用从骨干网络提取的共享特征并自适应地重组它们，可以节省更多的参数和储存空间。同时，将特征融合策略交由网络本身去学习，也可以有效地减少所需的人工干预。

5.3.2.2 动态特征融合的消融实验

动态特征融合的有效性：如表格 5.3中的第一行所示, 当本模型被直接应用于单个任务时，在显著性目标检测和边缘检测任务上，可以获得能与目前最好的方法相当的效果。在骨架提取任务上则可以观察到更大的性能提升（1.7%）。这表明所提出的 DFIM 能够根据所要解决的目标任务的特性来调整所对应特征选择策略。与现有的方法中通常手工地为不同的任务设计特定网络结构不同，DFIM 所需的人工交互明显更少。

当对三项任务进行协同学习时（表格 5.3中第五行），在显著性目标检测任务的两个数据集上的几乎所有测评指标上都有明显的提升。这一结果与以前的研究结论一致，即边缘信息可以帮助显著性目标检测任务在边缘区域获得更精细的分割结果。同时，边缘检测任务的性能也获得了提高，这表明显著目标的边缘也可以为此任务提供有用的监督信息。而骨架提取任务的性能仅略微下降。

为了对协同训练这三个不同任务的难度进行量化估计，本章通过移除本模型中 DFIM 的动态特征选择过程建立了一个作为对比的基线模型（对应于表格 5.3中的第三行, 标记为 “直连”）。这意味着从骨干网络中提取的特征在求和之

表 5.3 在四个广泛使用的数据集上的显著性目标检测、边缘检测和骨架提取的定量比较结果。“单任务”是指直接应用本模型，但只执行单一任务。每一列中的最佳结果以**粗体**突出显示。

序号	DFIM	TAM	显著性						边缘		骨架
			PASCAL-S[61]			DUTS-TE[63]			BSDS 500[85]		SK-LAR[109]
			$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	ODS ↑	OIS ↑	$MaxF_m\uparrow$
本章提出的方法（单任务）											
1	稀疏	无	0.860	0.075	0.849	0.875	0.042	0.878	0.815	0.831	0.749
2	稀疏	独立	0.859	0.081	0.849	0.880	0.045	0.878	0.812	0.826	0.746
本章提出的方法（多任务）											
3	直连	无	0.877	**0.062**	**0.865**	0.885	**0.038**	0.886	0.811	0.828	0.708
4	稠密	无	0.872	0.064	0.859	0.877	0.039	0.881	0.810	0.825	0.740
5	稀疏	无	0.874	0.064	0.862	0.884	**0.038**	**0.887**	0.818	0.834	0.744
6	稀疏	独立	0.873	0.065	0.861	0.879	0.039	0.883	0.815	0.832	**0.753**
7	稀疏	共享	**0.880**	0.065	**0.865**	**0.888**	**0.038**	**0.887**	**0.819**	**0.836**	0.751
其他现有方法（多任务）											
8	UberNet_{17}[144]		0.823	-	-	-	-	-	0.785	0.805	-
9	MLMS_{19}[57]		0.853	0.074	0.844	0.854	0.048	0.862	0.769	0.780	-

后的所有操作都将被删除，如图 5.3所示。这也等同于将公式 (5.2) 替换为下式：

$$D^{r,t} = \sum_i S_i^r. \tag{5.12}$$

通过将“直连”版本与所提出的“稀疏”特征选择版本（第五行）进行比较，可以观察到在边缘检测和骨架提取任务上明显的性能下降，幅度分别达到了 0.7% 和 3.6%。这些现象表明，简单地融合所有级别的特征对边缘和骨架的检测而言是有害的。当所涵盖的任务具有不同的优化目标并从不同的独立数据集获取训练样本时，很难以手工的方式来设计网络结构。类似的情况在以前的工作[57, 144]中也有出现，即其中部分任务的性能在联合解决多个不同任务时会显著下降，如表格 5.3的最后两行所示。但是对于 DFIM 来说，通过让网络自身动态地和自适应地融合特征，所有三个任务的整体性能表现相当于对每个任务分别训练的效果。

动态学习到的融合策略：为了更好地理解本模型学习到了何种类型的特征融合策略，本章从 DUTS-TE（显著性）、BSDS 500（边缘）和 SK-LARGE（骨架）三个对应于不同任务的测试集中各随机选择了 100 对样本来组成一个包含

表 5.4 本模型在 DFIMs 中采用不同的下采样率组合时的性能分析。每一列中的最好结果以**粗体**突出显示。

下采样倍率	显著性						边缘		骨架
	DUT-OMRON[29]			DUTS-TE[63]			BSDS 500[85]		SK-LARGE[109]
	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	ODS ↑	OIS ↑	$MaxF_m\uparrow$
2,4,8	0.814	0.057	0.839	0.879	**0.038**	0.885	0.815	0.829	0.742
4,8,16	0.809	0.057	0.837	0.880	**0.038**	0.884	0.814	0.829	**0.745**
2,4,8,16	**0.817**	**0.056**	**0.842**	**0.884**	0.038	**0.887**	**0.818**	**0.834**	0.744

300 对样本的综合测试集。通过将这些图像送入模型并对所有的 $\{p_i^{r,t}\}$ 值取平均，便可以得到用来决定特征选择过程的具体数值。图 5.4绘制了每个 DFIM 在每个阶段为不同任务从骨干网络中选择特征的概率分布情况。通过纵向比较各子图可以发现不同任务在偏好的特征层级之间存在明显的差异。这可以用来解释为什么面向某个特定任务所设计的具有良好性能的架构不能在其他任务上很好适应[44, 93, 112]。如果横向比较各子图，当三个任务中的每个任务以单任务方式单独训练时所选择的特征层级也与它们以多任务方式联合训练时选择的特征层级有很大不同。这可能是为何这三个任务中的每个任务都曾被很好地单独研究过，但很少有文献试图去在一个结构中协同解决它们的原因。在多任务协同学习的设定下试图手工地设计架构通常会失败，因为骨干网络提取出的共享特征现在将同时受到所有任务的影响。

稀疏或稠密连接：表格 5.3将本模型中稀疏连接的动态特征选择网络同其稠密连接的版本进行了比较，稠密连接版本的 $\{S_i\}$ $(1 \leq i \leq M)$ 中的所有特征图都被保留，而不是像公式 (5.1) 中那样只保留一半的特征。如表中第四行和第五行所示，稠密连接版本的网络几乎在所有的三个任务上的性能都更差。这表明并不是提取自骨干网络的每个层级的特征总是有正向帮助的[225]。例如，对于边缘检测，更多低层级的特征图对于边缘像素的精确定位是必要的[45, 93]；而对于骨架提取，更多高层级的信息对于确定像素是否属于骨架部分来说则是更重要的[111, 112]。

动态特征融合模块的下采样率：表格 5.4中对动态特征融合模块中采用的下采样倍率的组合进行了消融实验。从表中结果可以看出，更广范围的下采样倍率展示出更好的整体平均性能，特别是在显著性目标检测和边缘检测两个任务上。这也与更丰富的多尺度信息通常是有益的常识相一致。

5.3.2.3 任务自适应注意力的消融实验

任务自适应注意力的有效性： DFIM 的引入使得本模型可以在一个统一的架构下端到端地联合训练三个不同任务。但是，如表格 5.3中的第五行所示，相比于各任务单独训练时（第一行），骨架提取任务的性能有所下降。由于显著性目标检测和边缘检测任务的真值标注中更多地关注边缘部分的像素，这与骨架提取任务的目标之间有着明显的不同，因此骨架提取任务的优化可能会受到影响甚至被误导至完全相反的方向。有了 TAM 的帮助，网络通过自适应地调整每个任务传递给共享骨干部分的梯度，能够从全局角度整合进而分配所有任务的信息。从表格 5.3中第七行与第五行的对比中可以看出，有 TAM 加持的模型取得了更好的整体性能。显著性目标检测和边缘检测的性能稍有提升，而骨架提取的性能则提升了近 0.7%。

信息交互的必要性：为了研究 TAM 带来的提升是否是由于引入了额外的可学习参数所导致的，本小节也进行了消融实验。本小节通过保持 TAM 中不同分支的参数之间相互独立，使得不同的任务分支在从共享的骨干网络中选择特征后互不影响。因 TAM 中的参数不再共享，额外的 1.66M 参数被进一步引入。然而，从表格 5.3的第六行可以看出，即使引入更多参数，参数独立版本的 TAM 的整体性能却明显不如共享版本的 TAM（第七行）。虽然骨架提取任务的表现稍好，但在其他两个任务上的性能却大幅下降。这些现象表明，在每个任务分别进入各自独立的分支之后，通过网络结构设计来强制地保证跨任务间的信息交互有助于所有任务的整体更好地收敛，而仅仅简单地使用注意力机制却无法很好地发挥作用。这也可以从表格 5.3的前两行观察到，当每个任务被单独训练时，添加 TAM 对三个任务中的多数没有帮助，甚至会产生负面影响。

5.3.3 与领先方法的比较

这一节将所提出的方法（为了方便起见记为 DFI）与现有的领先方法分别在显著性目标检测、边缘检测和骨架提取三个任务上进行了比较。由于之前只有很少的文献尝试去同时解决这三个任务，例如 UberNet[144]（CVPR'17）和 MLMS[57]（CVPR'19）只提出同时去解决显著性目标检测和边缘检测两个任务。因此为了更好地比较和说明，本小节主要与这三个任务对应的领先的单一目标方法进行对比。为了公平比较，对于每个任务，其他方法的预测结果图（例如，显著图、边缘图、骨架图）均由其作者发布的原始代码和设置所生成或由他们

表 5.5 本模型与 16 个现有方法在六个广泛使用的显著性目标检测数据集上的量化对比结果。每列的最佳结果以**粗体**突出显示。可以看出，在 F-measure, MAE 和 S-measure 指标上本章提出的方法在几乎所有的数据集上都取得了最好的结果。

	ECSSD[62]			PASCAL-S[61]			DUT-OMRON[29]			HKU-IS[32]			SOD[60]			DUTS-TE[63]		
方法年份	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_\alpha\uparrow$
DCL_{16}[40]	0.896	0.080	0.869	0.805	0.115	0.800	0.733	0.094	0.762	0.893	0.063	0.871	0.831	0.131	0.763	0.786	0.081	0.803
$RFCN_{16}$[37]	0.898	0.097	0.856	0.827	0.118	0.808	0.747	0.094	0.774	0.895	0.079	0.860	0.805	0.161	0.722	0.786	0.090	0.793
MSR_{17}[177]	0.903	0.059	0.887	0.839	0.083	0.835	0.790	0.073	0.805	0.907	0.043	0.896	0.841	0.111	0.782	0.824	0.062	0.834
DSS_{17}[44]	0.906	0.064	0.880	0.821	0.101	0.804	0.760	0.074	0.789	0.900	0.050	0.881	0.834	0.125	0.764	0.813	0.065	0.826
$NLDF_{17}$[46]	0.903	0.065	0.870	0.822	0.098	0.805	0.753	0.079	0.770	0.902	0.048	0.878	0.837	0.123	0.759	0.816	0.065	0.816
$Amulet_{17}$[179]	0.911	0.062	0.876	0.826	0.092	0.816	0.737	0.083	0.784	0.889	0.052	0.866	0.799	0.146	0.729	0.773	0.075	0.800
$PAGR_{18}$[50]	0.924	0.064	0.883	0.847	0.089	0.822	0.771	0.071	0.775	0.919	0.047	0.889	-	-	-	0.854	0.055	0.839
$DGRL_{18}$[52]	0.921	0.043	0.899	0.844	0.072	0.836	0.774	0.062	0.806	0.910	0.036	0.895	0.843	0.103	0.774	0.828	0.049	0.842
$MLMS_{19}$[57]	0.924	0.048	0.905	0.853	0.074	0.844	0.793	0.063	0.809	0.922	0.039	0.907	0.857	0.106	0.790	0.854	0.048	0.862
$JDFPR_{19}$[186]	0.925	0.052	0.902	0.854	0.082	0.841	0.802	0.057	0.821	-	-	-	0.836	0.121	0.767	0.833	0.058	0.836
$PAGE_{19}$[187]	0.928	0.046	0.906	0.848	0.076	0.842	0.791	0.062	0.825	0.920	0.036	0.904	0.837	0.110	0.775	0.838	0.051	0.855
$CapSal_{19}$[58]	-	-	-	0.862	0.073	0.837	-	-	-	0.889	0.058	0.851	-	-	-	0.844	0.060	0.818
CPD_{19}[47]	0.936	0.042	0.913	0.859	0.071	0.848	0.796	0.056	0.825	0.925	0.034	0.907	0.857	0.110	0.771	0.865	0.043	0.869
$PiCA_{18}$[51]	0.932	0.048	0.912	0.864	0.075	0.854	0.820	0.064	0.830	0.920	0.044	0.904	0.861	0.103	0.792	0.863	0.050	0.868
$AFNet_{19}$[53]	0.932	0.045	0.907	0.861	0.070	0.849	0.820	0.057	0.825	0.926	0.036	0.906	-	-	-	0.867	0.045	0.867
$BASNet_{19}$[54]	0.939	0.040	0.911	0.857	0.076	0.838	0.811	0.057	0.836	0.930	0.033	0.908	0.849	0.112	0.772	0.860	0.047	0.866
DFI （本章）	**0.945**	**0.038**	**0.921**	**0.880**	**0.065**	**0.865**	**0.829**	**0.055**	**0.839**	**0.934**	**0.031**	**0.919**	**0.878**	**0.100**	**0.802**	**0.888**	**0.038**	**0.887**

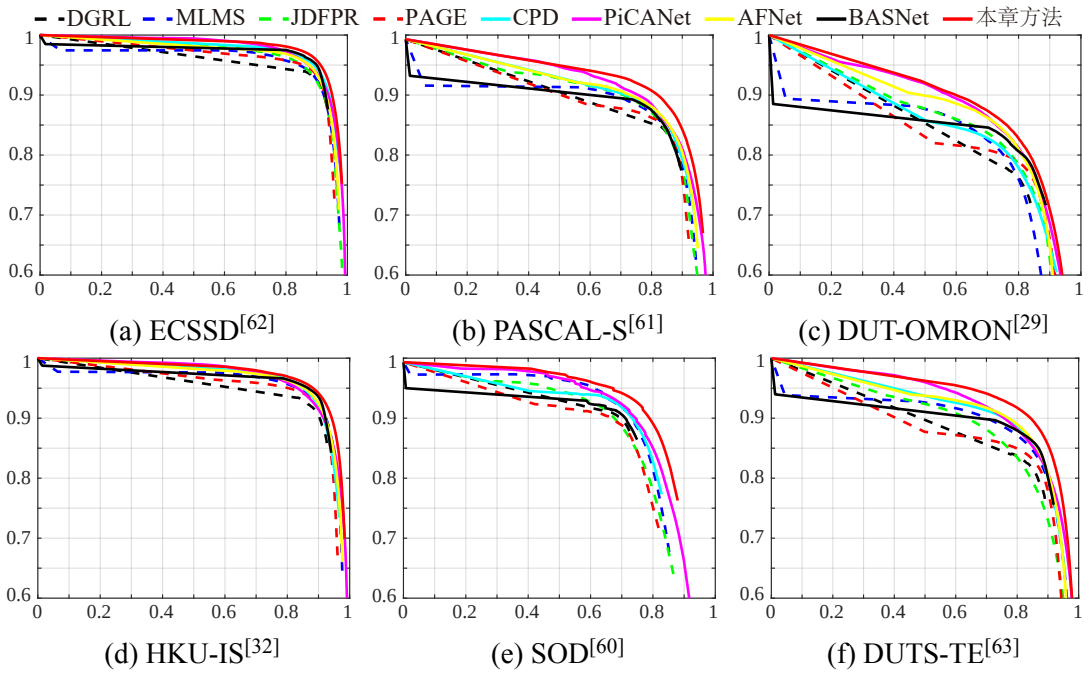


图 5.7 六个常用显著性目标检测出数据集上的准确率（纵轴）-召回率（横轴）曲线对比。

直接提供。除了评估边缘和骨架图之前的 NMS 过程[45, 93, 111] 之外，所有结果都是直接从单模型测试中获得，而不依赖于任何其他预处理或后处理过程。对于每个任务，所有的预测结果图均使用了相同的测评代码进行评估。

5.3.3.1 显著性目标检测

本小节将 DFI 和现有的 16 种领先的显著性目标检测方法进行了详尽的比较，包括 DCL[40], RFCN[37], MSR[177], DSS[44], NLDF[46], Amulet[179], PAGR[50], DGRL[52], MLMS[57], JDFPR[186], PAGE[187], CapSal[58], CPD[47], PiCANet[51], AFNet[53], 和 BASNet[54]。

本段比较了 DFI 和前面提到的方法在 F-measure，MAE 和 S-measure 三个指标上的优劣（见表格 5.5）。可以看出，DFI 在六个数据集上的表现均优于所有其他方法。与每个数据集上的第二好的方法相比，DFI 在 F-measure 和 S-measure 指标上的平均提升分别为 1.2% 和 1.0%。特别是在具有挑战性的 DUTS-TE 数据集上，可以看到 F-measure 和 S-measure 指标上有 2.1% 和 1.8% 的提升。在 MAE 指标上也可以观察到类似的结果。此外，与联合学习显著性目标检测和边缘检测的 MLMS[57] 方法相比，DFI 在这两项任务上都有着巨大的提升，如表格 5.3的

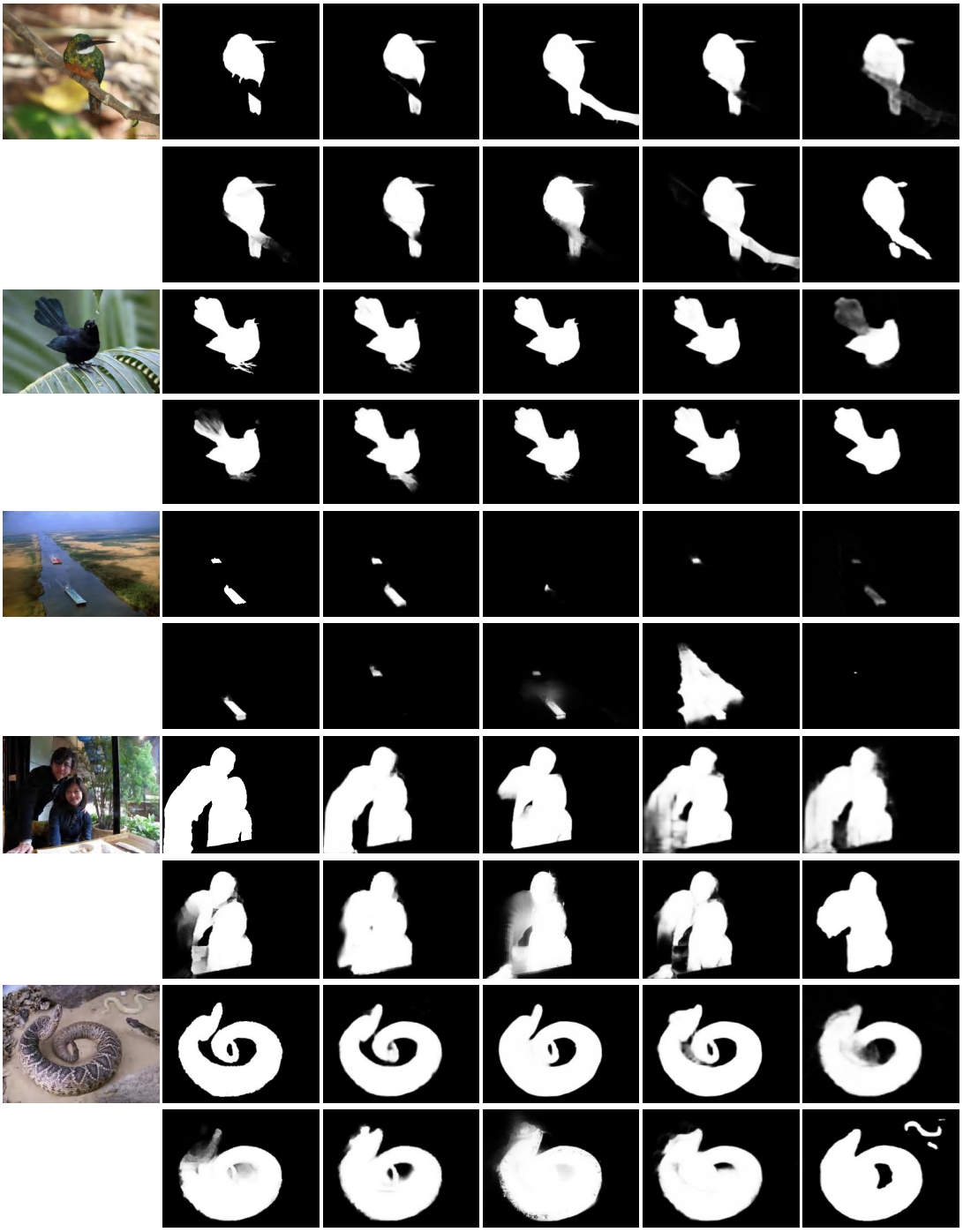


图 5.8 不同显著性目标检测方法的结果的视觉比较（一）。每组图像对应的标签分别为: 输入图像、标注图像、本方法、BASNet[54]、CPD[47]、PiCANet[51]、AFNet[53]、PAGE[187]、JDFPR[186]、MLMS[57] 以及 DGRL[52]。

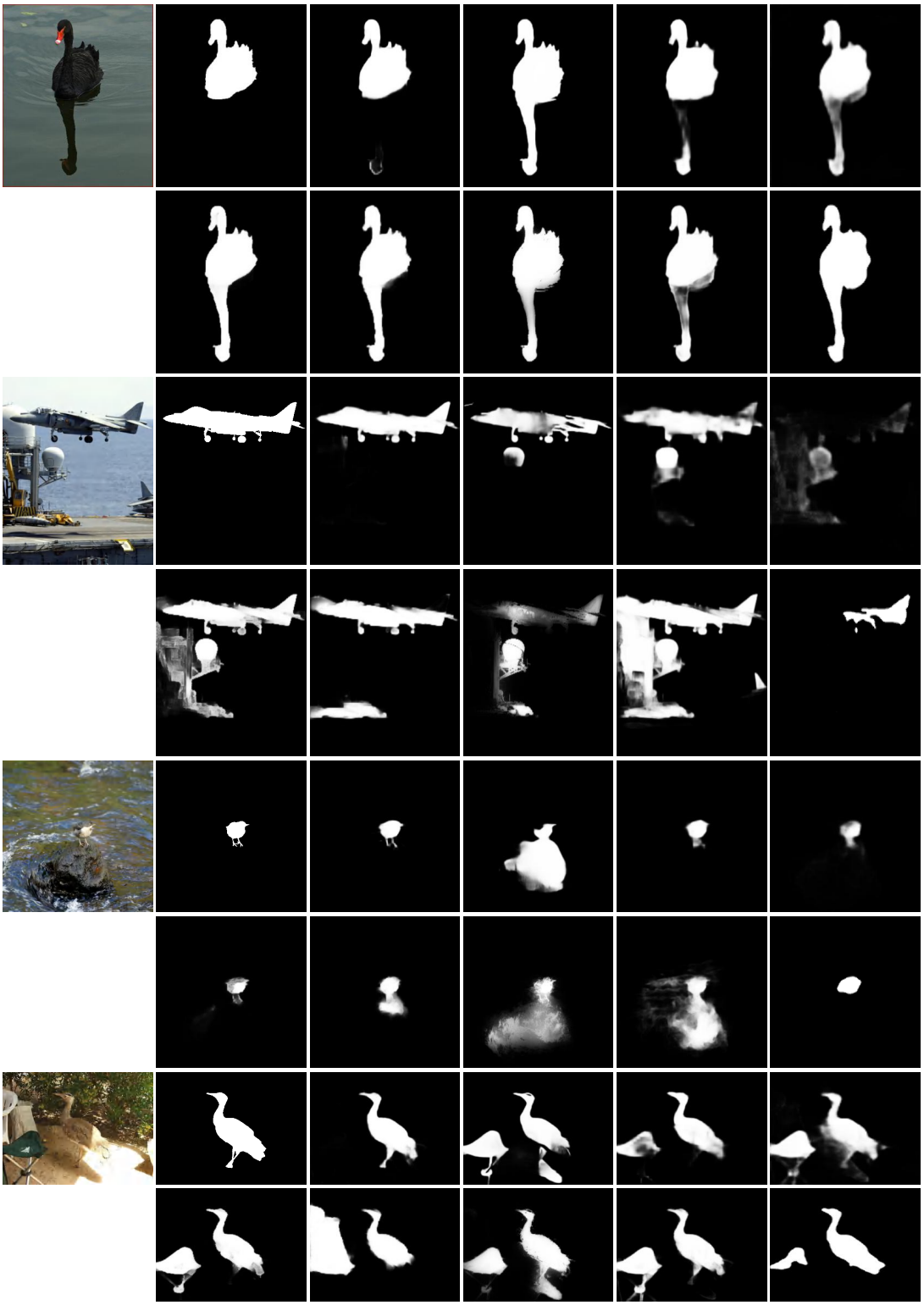
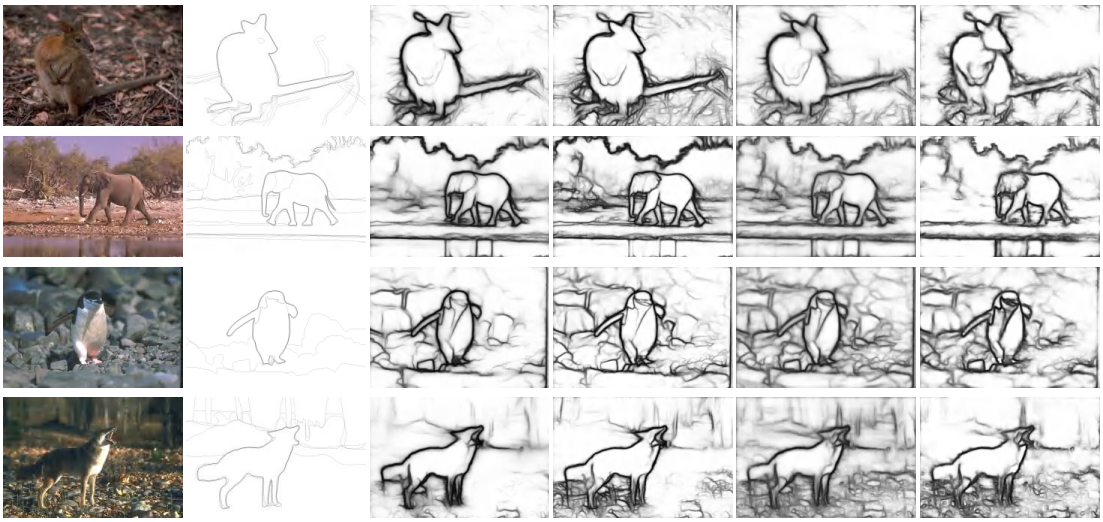


图 5.9 不同显著性目标检测方法的结果的视觉比较（二）。每组图像对应的标签分别为: 输入图像、标注图像、本方法、BASNet[54]、CPD[47]、PiCANet[51]、AFNet[53]、PAGE[187]、JDFPR[186]、MLMS[57] 以及 DGRL[52]。



(a) 输入图像 (b) 标注图像 (c) 本方法 (d) CED[226] (e) RCF[93] (f) HED[45]

图 5.10 与最近几个领先的边缘检测方法的视觉结果比较。可以看出，与其他方法相比，DFI 不仅能够生成更清晰的背景，而且在物体边缘部分上表现精细。

第七行和第九行所示。即使没有 TAM，DFI 的表现仍然远远超过 MLMS[57]（表格 5.3的第三行对比于第九行）。这一结果证明了所提出的 DFIM 和 TAM 的有效性。

除了数值结果，本小节还在图 5.7中展示了所提出的方法在这六个数据集上的 PR 曲线对比结果。可以看出，DFI 的 PR 曲线（红色实线）在大部分数据集上与其他方法有着相当的表现，而在一些数据集上甚至更好。特别是在 PASCAL-S 和 DUTS-TE 数据集上，DFI 比以前的所有方法都要突出。当召回率接近 1 时，本模型的准确率远远高于其他方法，这表明本模型的显著图中的假阳率更低。

图 5.8和图 5.9中提供了 DFI 与现有的几种领先的方法的视觉比较。在最上面的样本中，显著性目标被部分遮挡，但 DFI 能够完整分割出整个目标，而不会混入不相关的区域。如第二个样本所示，DFI 还能够以更精确的边界和细节分割出显著性目标。当面对显著物体微小且不规则或者前景和背景之间对比度较低的样本时，DFI 依旧能够很好的检测并分割出显著性对象。例如，图 5.9底部的两个样本。这些结果表明，DFI 能够更好地区分边缘像素并分割出整个目标，这可能是与边缘检测和骨架提取任务协同训练所得到的优势。

表 5.6 DFI 与现有边缘检测方法的量化对比结果。每列的最佳结果以**粗体**突出显示。

方法$_{年份}$	BSDS 500[85]	
	ODS ↑	OIS ↑
gPb-owt-ucm$_{11}$[85]	0.726	0.757
SE-Var$_{15}$[87]	0.746	0.767
MCG$_{17}$[227]	0.747	0.779
DeepEdge$_{15}$[228]	0.753	0.772
DeepContour$_{15}$[89]	0.756	0.773
HED$_{15}$[45]	0.788	0.808
CEDN$_{16}$[191]	0.788	0.804
RDS$_{16}$[192]	0.792	0.810
COB$_{17}$[193]	0.793	0.820
RCF$_{17}$[93]	0.811	0.830
DCNN+sPb$_{15}$[194]	0.813	0.831
CED$_{17}$[226]	0.815	0.833
LPCB$_{18}$[229]	0.815	0.834
DFI （本章）	**0.819**	**0.836**

5.3.3.2 边缘检测

本小节将 DFI 与现有的 13 种领先的边缘检测方法的结果进行了比较，包括 gPb-owt-ucm[85], SE-Var[87], MCG[227], DeepEdge[228], DeepContour[89], HED[45], CEDN[191], RDS[192], COB[193], RCF[93], DCNN+sPb[194], CED[226] 和 LPCB[229]，其中大多数是基于 CNN 的方法。

表格 5.6中展示了量化的结果比较。DFI 得到了 0.819 的 ODS 和 0.836 的 OIS 分数，这甚至比之前那些为边缘检测而单独精心设计的方法还要好。得益于了 DFIM 和 TAM，来自其他任务的信息不仅不会影响反而还有助于提升边缘检测的性能，如表格 5.3的“边缘”列的第一行和第七行所示。

本模型和部分领先方法在 BSDS 500 数据集[85] 上的准确率-召回率曲线对比可以在图 5.11中找到。可以观察到，由本模型产生的 PR 曲线在某些情况下已经优于人工标注的表现，并且与之前的方法表现相当，尤其是在准确率方面。

图 5.10中展示了 DFI 与一些领先的代表性方法[45, 192, 226] 之间的一些视觉比较。可以观察到，DFI 在检测边界方面比其他方法表现得更好。如图 5.10的最后一行，显然狼的真实边缘部分被很好地突显了。得益于所提出的动态融合机制，样例中没有边缘的区域被描绘得更干净清晰，这一现象表明与[45, 192] 相比 DFI 学习的特征更加强大。总而言之，除了在 ODS 和 OIS 指标上有所提升之外，本

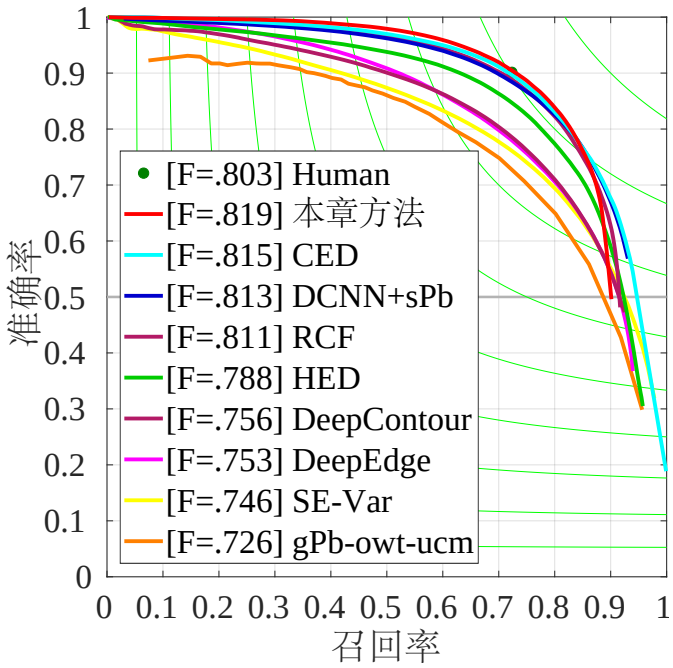


图 5.11 BSDS 500 数据集[85] 上的准确率-召回率曲线对比。

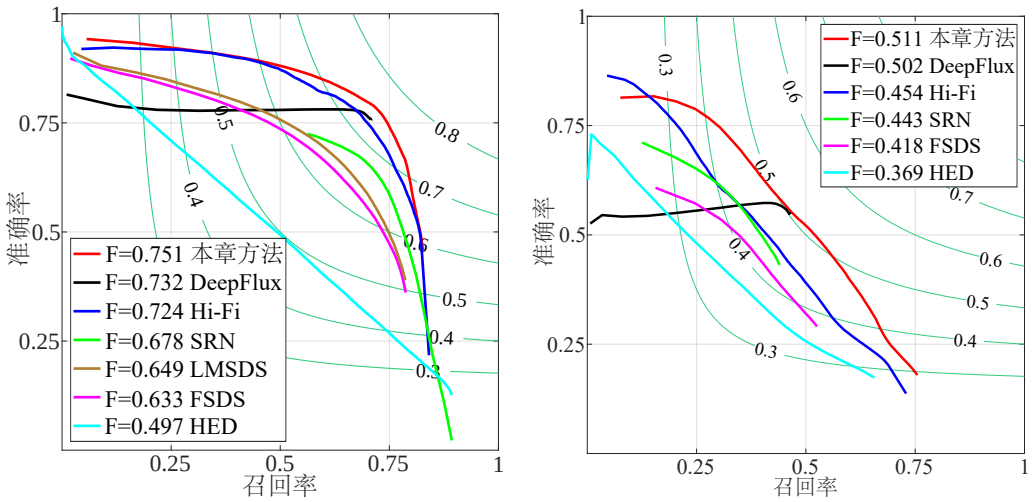
表 5.7 DFI 与现有骨架提取方法的量化对比结果。每列的最佳结果以**粗体**突出显示。

	SK-LARGE[109]	SYM-PASCAL[110]
方法$_{年份}$	$MaxF_m$ ↑	$MaxF_m$ ↑
MIL_{12}[105]	0.353	0.174
HED_{15}[45]	0.497	0.369
RCF_{17}[93]	0.626	0.392
$FSDS_{16}$[108]	0.633	0.418
$LMSDS_{17}$[109]	0.649	-
SRN_{17}[110]	0.678	0.443
LSN_{18}[230]	0.668	0.425
Hi-Fi_{18}[111]	0.724	0.454
$DeepFlux_{19}$[112]	0.732	0.502
DFI （本章）	**0.751**	**0.511**

模型的结果在视觉质量上有着更明显的提升。

5.3.3.3 骨架提取

本小节比较了 DFI 和最近九个基于 CNN 的领先骨架提取方法，包括 MIL[105], HED[45], RCF[93], FSDS[108], LMSDS[109], SRN[110], LSN[230], Hi-Fi[111],



(a) SK-LARGE[109] (b) SYM-PASCAL[110]

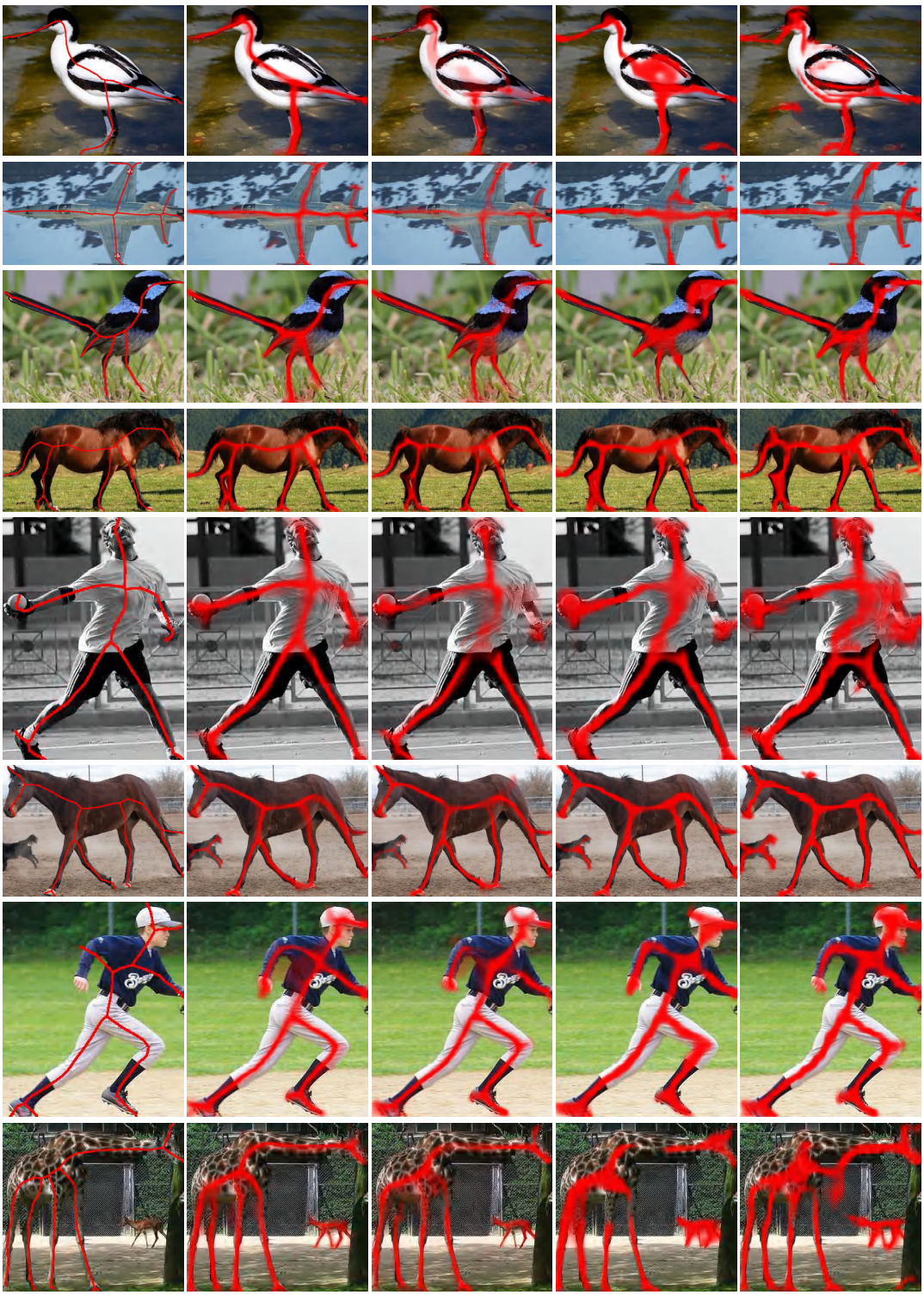
图 5.12 在 SK-LARGE 数据集[109] 和 SYM-PASCAL 数据集[110] 上部分骨架提取方法的准确率-召回率曲线对比。

和 DeepFlux[112]。对比在该领域两个流行且具有挑战性的数据集上进行，包括 SK-LARGE[109] 和 SYM-PASCAL[110]。为了公平比较，本小节参照该领域现有方法的做法，分别使用这两个数据集来训练并得到了两个对应的模型以作为对比。

在表格 5.7中展示了 DFI 与现有方法的定量比较结果。可以看出，DFI 在 SK-LARGE 数据集[108] 上大幅度 (1.9%) 领先于其余方法。并且在 SYM-PASCAL 数据集[110] 上也有 0.9% 的提升。

在图 5.12中还展示了本模型和部分骨架提取方法的准确率-召回率曲线的对比结果。从曲线之间的差距可以看出，本模型在这两个数据集上的表现明显优于其他现有方法。

在图 5.13中显示了一些预测结果的视觉比较。得益于所提出的特征融合策略是根据任务和输入而动态调整的，DFI 能够更精确地定位骨架的准确位置。本模型的预测图比其他工作的结果要更加精细，并且更突出。这也佐证了所提出方法的有效性。定量和可视化结果共同表明，即便是在多任务协同学习的设定中，DFI 也能够更好地结合不同层级的特征以促进骨架提取的效果。



(a) 标注图像 (b) 本章方法 (c) Hi-Fi[111] (d) SRN[110] (e) FSDS[108]

图 5.13 与三种领先且具有代表性的骨架提取方法的视觉结果对比。很容易发现，本模型的结果比其他三种方法要细得多，且有着更高的置信度。此外，本模型的预测结果中的骨架是连续的，这对于实际的应用来说是至关重要的。

表 5.8 DFI 和现有领先方法的平均运行速度（FPS）比较。

	DFI（多任务）	**DFI（单任务）**	BASNet[54]	AFNet[53]	PiCANet[51]
输入分辨率	400 × 300	400 × 300	256 × 256	224 × 224	224 × 224
速度（FPS）	40	57	25	26	7
	PAGE[187]	CPD[47]	DGRL[52]	Amulet[179]	DSS[44]
输入分辨率	224 × 224	352 × 352	384 × 384	256 × 256	400 × 300
速度（FPS）	25	61	8	16	12
	RCF[93]	CED[226]	LPCB[229]	Hi-Fi[111]	DeepFlux[112]
输入分辨率	480 × 320	480 × 320	480 × 320	300 × 200	300 × 200
速度（FPS）	36	35	35	32	55

第四节 讨论

5.4.1 运行时间的比较

表格 5.8将 DFI 的运行速度与本章中评估的包括所有三个任务在内的部分其他开源方法进行了比较。表中同时注明了不同方法的平均速度（FPS）以及相应的输入分辨率（在相同的环境中测试）。DFI 可以在单任务模式下以 57FPS 的速度运行，这与其他方法相当，但是能获得更好的检测结果。此外，DFI 即使在多任务模式下也能以 40FPS 的速度运行，即在一次前向传递中同时完成三个不同任务的预测。

5.4.2 关于训练时间的分析

由于现有方法中没有能够同时完成这三项任务的。为了突出比较所提出的 DFIM 和 TAM 带来的影响，本段将所提出的方法与其基线版本进行了比较（表格 5.3中第七行对比第三行）。所提出的方法需要大约 30 个小时来训练，而基线方法需要 25 个小时。由此可见，DFIM 和 TAM 引入的额外结构和参数增加了约 20% 的训练时间，但也获得了在所有任务中更好、更平衡的整体性能。

5.4.3 ImageNet 预训练的影响

在上面的章节的所有实验中，为了公平比较，作者参考上述三个任务的之前方法中的做法，均使用了 ImageNet 预训练的模型参数作为骨干提取网络的初始化参数。本小节研究了 ImageNet 预训练参数对所提出方法的整体性能的影响。当完全从头开始训练时，网络中的所有参数都是随机初始化的。而其他所有的训练设置除了特别说明外均保持了一致。通过比较表格 5.9的第一行和第三行，

表 5.9 对于 ImageNet 预训练影响的消融分析。

训练周期	ImageNet 预训练	显著性			边缘		骨架
		DUTS-TE[63]			BSDS 500[85]		SK-LARGE[109]
		$F_\beta \uparrow$	$MAE \downarrow$	$S_\alpha \uparrow$	ODS ↑	OIS ↑	$MaxF_m \uparrow$
1×	不包括	0.819	0.064	0.831	0.786	0.809	0.663
4×	不包括	0.841	0.053	0.848	0.799	0.820	0.738
1×	包括	**0.888**	**0.038**	**0.887**	**0.819**	**0.836**	**0.751**

可以看到在进行 1× 周期（∼ 12 轮）的训练时，不使用 ImageNet 预训练的模型的整体性能要比使用 ImageNet 预训练版差很多。即使在训练了 4× 周期（∼ 48 轮，36 轮后学习率除以 10），模型的整体表现还是有着明显差距。作者在训练过程观察到中，当使用 ImageNet 预训练时，整体的损失在早期便快速降低并收敛，而随机初始化的版本则需要明显更多的迭代次数来收敛。ImageNet 数据集有 ∼1.28M（1,281,167）张图像，比这三个任务所用的所有图像的数量（21,702，如表格 5.1中所述）还多大约 59 倍。当从头开始训练一个网络时，期望仅仅用约 22K 的图像来很好地优化一个网络是不够的。考虑到这三个任务都以自然图像作为输入，作者认为 ImageNet 预训练有助于在训练开始阶段为模型提供更强大和有效的初始特征提取功能。当从头开始训练时，模型必须从零开始学习如何有效地提取特征，因而需要更多的迭代次数来收敛。通过增加训练周期，随机初始化的模型最终也可能收敛，但因为特征提取能力的缺乏所导致的性能差距并不容易缩小。

第五节 本章小结

本章提出从多任务协同学习的角度来提升显著性目标检测任务的性能。具体而言，本章在一个网络中端到端地同时解决了三个不同的浅层像素级预测任务，包括显著性目标检测、边缘检测和骨架提取。本章提出了一个动态特征融合模块（DFIM）来动态地学习每个任务的特征融合策略，以及一个任务自适应注意力模块（TAM）来到达跨任务信息交互与调整的目的，以获得更好的整体收敛性。在多个广泛使用的数据集上的实验表明，所提出 DFI 方法与各任务对应的单一目标的领先方法的性能相当，有的时候甚至有着更好的性能表现。DFI 的速度也很快，它能以 40FPS 的速度同时执行这三个不同的像素级的预测任务。

第六章 总结与展望

显著性目标检测已然成为计算机视觉和图像处理领域一个重要的研究方向，其目的在于将给定图像中视觉最为显著的物体或者区域准确地检测并完整地分割出来。得益于显著性目标检测的类别无关特性，其作为一种通用的图像属性已被实际应用于众多现实场景中，并同时作为一类通用的预处理方法被广泛应用于各种各样的下游计算机视觉任务中，以辅助它们高效快速地捕获输入图像中最为关键的信息。卷积神经网络由于其自动提取多空间尺度信息的能力，以及对于输入图像有着平移不变的特性，已成为当前基于深度学习技术的显著性目标检测算法的主流。然而，来源于现实场景的图像通常有着结构复杂、目标繁多、以及成像受限等缺点，并且现实应用场景中常常有着硬件以及算法层面各种各样的限制或者需求。因此，设计更加简洁、先进和高效的多尺度特征选择和融合方式，以更少的参数和计算量得到更适合显著性目标检测任务的、更具表征力的特征，对于提升显著性目标检测的性能和效率有着重要意义。

第一节 本文工作总结

为了在提升显著性目标检测模型性能的同时保证其效率，考虑到现有显著性目标检测方法在多尺度特征融合方面的不足和缺陷，本文立足于池化操作和注意力机制这两个神经网络的基础组件，从三个不同方面提出了更为先进和高效的多尺度特征融合方式。下面将按照章节顺序对本文主要工作进行总结。

本文首先介绍了显著性目标检测任务的发展背景和现实意义、当前存在的部分研究难点以及本文主要的研究目标和相应的贡献。

第二章结合本文的研究内容对显著性目标检测领域的模型和算法、常用的数据集和评价指标，相关的二元预测任务所属领域的模型和算法，以及一些相关的基础操作、结构和机制领域的模型和算法分别进行了回顾。

第三章提出了一种基于高效特征池化和融合的显著性目标检测算法(PoolNet)。PoolNet 利用一个基于适应性平均池化的全局信息指导模块来弥补显著性目标检测模型中深度神经网络的实际感受野与其理论值的差异，以及 U 型结构的自顶向下通路中高层语义信息会逐渐被浅层细节所稀释的问题。同时

PoolNet 还提出了一个基于多下采样率的平均池化操作的特征融合模块来桥接来自全局信息引导模块中的全局信息与各浅层局部特征之间的感受野差距，以得到更加无缝的跨尺度信息融合。对模型的中间特征和最终预测结果的可视化对比表明，所提出的两个模块可以有效提升模型对于显著性目标的全局定位准确度和细节分割精细度。为了定量地验证 PoolNet 的有效性，本章在五个显著性目标检测领域中广泛使用的数据集上对 PoolNet 进行了评估。量化实验结果表明，PoolNet 相较于现有的最优方法在多个不同的测评指标上均有较大的提升。例如在最具挑战性的 DUTS-TE 数据集的 F-measure 指标上，PoolNet 有着约 1.5% 的提升，同时运行速度快 1.6 倍。除此之外，本章提出的轻量化 PoolNet-M 在只需要 8.2% 参数量和 10.2% 乘加量的基础上，达到了和现有最优方法相当的效果。本章还进一步将 PoolNet 迁移应用到了边缘检测、RGB-D 显著性目标检测、以及伪装对象检测三个有着不同通用图像属性的任务上，量化实验结果表明 PoolNet 在占用明显更少的参数量和计算量的情况下，在这三个任务上均达到了更加优越的性能。上述实验结果充分验证了所提出的特征池化和融合算法的高效性、强泛化性和鲁棒性。

第四章提出了一种基于高效信息集中交互与融合的显著性目标检测算法（CII）。考虑到 U 型结构在当前显著性目标检测算法中的广泛使用，CII 提出将之前在 U 型结构的自底向上通路和自顶向下通路的相应层级之间被保持相互独立的连接，进行多尺度信息集中化交互，以在 U 型结构内部高效地鼓励更加动态和自由的跨尺度特征信息融合，从而得以在极少量的参数和计算代价下得到具有更丰富表征力的融合特征。该信息交互策略不需要任何特征采样操作，有效地避免了特征下采样过程中的信息损失。同时本章还提出了一个相对全局信息矫正模块，该模块利用多尺度特征在集中信息交互时相邻层级的特征之间天然存在的感受野差距，通过较高层的信息来指导较浅层信息的学习，有效丰富了特征融合的形式和表征力。为了验证 CII 的有效性，本章在五个广泛使用的显著性目标检测数据集上进行了量化对比。实验结果表明，CII 在相对于基线模型几乎不引入额外的参数和计算复杂度的情况下，能够在最具挑战性的 DUTS-TE 数据集的 F-measure 指标上获得约 3.5% 的提升。相较于现有领先的方法，CII 在只需要其 15.1% 参数量和 20.8% 乘加量的前提下，能够获得更加优异的表现。CII 的设计目的在于提升 U 型结构的多尺度信息融合能力，因此可以被灵活应用到任何基于 U 型结构的显著性目标检测模型上。本章进一步将 CII 迁移应用到了

多个现有的具有代表性的基于 U 型结构的显著性目标检测模型上，定量实验结果表明 CII 均能明显提升这些模型的性能。上述实验结果有效验证了所提出的特征集中交互与融合算法的高效性和易扩展性。

第五章提出了一种基于高效特征动态选择与融合的多任务协同学习算法(DFI)。考虑到在显著性目标检测任务中，无论是显著性对象的完整性还是边界部分的准确性都对模型性能有着至关重要的作用，本章提出以多任务协同学习的方式利用骨架提取和边缘检测两个任务来分别提升显著性目标检测任务在主体完整性和边界准确性上的性能。DFI 在一个模型中同时完成了三个不同的任务，它利用一个多尺度特征动态选择机制来让模型根据具体的输入内容和任务特点自适应地选择各任务适用的多尺度特征以进行融合。为了平衡不同任务在训练数据分布和任务偏好等方面的冲突，DFI 提出了一个多任务适用的自适应注意力模块，该模块通过将可学习的滤波器参数进行跨任务共享的方式构建出不同任务之间必要的信息交流，有效促进了模型的整体收敛效果，避免了模型陷入部分任务的局部最优解中。不同于现有多任务方法通常会牺牲辅助任务的性能，DFI 在上述三个任务各自通用的多个数据集上的量化对比结果表明，其相比于各任务现有领先的单一目标的方法均获得了更优的效果。具体而言，DFI 相对于显著性目标检测任务现有最优方法，在 DUTS-TE 数据集的 F-measure 指标上有着 3.3% 的提升；相对于边缘检测任务现有最优方法，在 BSDS 500 数据集的 ODS 指标上有着 0.5% 的提升；相对于骨架提取任务现有最优方法，在 SK-LARGE 数据集的 F-measure 指标上有着 2.6% 的提升。除此之外，DFI 相对于必要的特征提取骨干网络只需要 8.7% 的各任务特定参数，并且整体而言，原来需要三个不同模型通过三次前向传播而分别得到的三个任务的预测结果，DFI 只需要一个模型一次前向传播便可以得到，极大地减少了模型参数量、计算和时间开销，这有效证明了所提出的多任务协同学习算法的高效性。

第二节 未来工作展望

作为一项不依赖于目标对象语义类别的检测技术，显著性目标检测能够快速而精准地定位并分割出复杂真实场景中使人感兴趣的对象，并被广泛地应用于众多现实场景和视觉任务中。虽然目前基于卷积神经网络的显著性目标检测方法的性能相对于传统算法已经有了长足的进步，并在众多真实场景中获得应用，但仍有许多问题亟待研究和解决。本文主要对基于卷积神经网络的显著性

目标检测算法中的多尺度特征高效融合问题展开研究，并从神经网络的两种基本组成部件：池化操作和注意力机制出发，根据具体的细分问题提出了自己的观察和解决方案。但由于作者的水平和能力有限，所提出的研究和解决方案还存在一定的不足，下面将针对本文各章中待研究的方向和后续的工作进行简要的介绍。

第三章提出了基于高效池化操作的显著性目标检测算法。本方法提出的模块均是基于池化操作，而其他类型的高效操作，例如特征平移（feature shifting），在本模型中的表现是一个值得研究的方向。本方法利用简单直连的方式来显式地将全局信息引导到自顶向下各层级中，是否有更有效的模块来替代直连方式也是一个值得研究的方向。另外本方法在所有需要融合全局指导信息和局部信息的地方使用的特征融合模块的设置（下采样倍率的组合）均是一样的，在不同的融合位置使用不同的设置是否能带来更好的效果也是一个值得研究的地方。

第四章提出了基于信息集中交互的显著性目标检测算法。本方法通过改进 U 型结构中的短连接来构建更丰富的特征表达。首先，本方法将所有来自自底向上通路的多尺度特征进行集中交互，是否所有的特征均是有正向收益的，以及如何设计更加自适应的特征选择机制均值得深入研究。其次，如何利用本方法中集中信息交互的优势，构建出更加有效的特征交互模块也是一个值得研究的方向。再者还可以进一步探究本方法在其他基于 U 型网络的模型以及任务中的表现。

第五章提出了基于多任务协同学习的显著性目标检测算法。本方法通过利用来自其他协同学习任务中的监督信息来有效提升显著性目标检测任务的性能。本方法提出的任务自适应注意力模块只使用了几个串联的卷积层，如何设计更加鲁棒和有效的模块是一个值得研究的方向。另外本方法中选择的辅助任务是否有更优的替代、或者是否可以通过添加更多的辅助任务以得到更优的效果也值得继续研究。除此之外，当增加新的协同学习任务时，本方法需要在所有任务上进行重新训练，比较耗费资源和时间，能否通过例如增量学习等方式来改进所提出的方法也是一个值得继续研究的方向。

整体而言，本文的第三、四和五章所提出的特征融合方式相对独立且互不相同，它们是否可以互补地结合为一个统一的系统值得继续研究。此外，本文目前仅在完全监督的任务设定下对所提出的各类特征融合方式的有效性和泛化性进行了研究和验证，它们在例如半监督、弱监督甚至无监督等任务设定下的

表现也值得深入研究。再者，本文目前仅在二元任务上对所提出的各类特征融合方式进行了应用实验，它们在其它更加复杂的任务，例如语义分割和物体检测上的表现同样值得继续研究。

参考文献

[1] GOLDSTEIN E B. Sensation and perception. 8th. [J]. Belmont: Wadsworth, Cengage Learning, 2009, 496 (3): 231–233.

[2] BUNDESEN C. A theory of visual attention. [J]. Psychological Review, 1990, 97 (4): 523.

[3] GRILL-SPECTOR K, MALACH R. The human visual cortex. [J]. Annual Review of Neuroscience, 2004, 27: 649–677.

[4] ITTI L, KOCH C, NIEBUR E. A model of saliency-based visual attention for rapid scene analysis. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, 20 (11): 1254–1259.

[5] 王亚楠. 基于深度神经网络的显著目标检测算法研究. [D]. 中国科学院大学 (中国科学院西安光学精密机械研究所), 2020.

[6] BORJI A, ITTI L. State-of-the-art in visual attention modeling. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35 (1): 185–207.

[7] CHENG M.-M, GAO S.-H, BORJI A, et al. A highly efficient model to study the semantics of salient object detection. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021. DOI: 10.1109/TPAMI.2021.3107956.

[8] LECUN Y, BOSER B, DENKER J S, et al. Backpropagation applied to handwritten zip code recognition. [J]. Neural Computation, 1989, 1 (4): 541–551.

[9] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks. [C] // Advances in Neural Information Processing Systems. Vol. 25: 2012.

[10] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition. [C] // International Conference on Learning Representations: 2015.

[11] HE K, ZHANG X, REN S, et al. Deep Residual Learning for Image Recognition. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2016: 770–778.

[12] SZEGEDY C, LIU W, JIA Y, et al. Going deeper with convolutions. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2015: 1–9.

[13] HUANG G, LIU Z, VAN DER MAATEN L, et al. Densely connected convolutional networks. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 4700–4708.

[14] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need. [C] // Advances in Neural Information Processing Systems. Vol. 30: 2017.

[15] LIU Z, LIN Y, CAO Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows. [C] // IEEE International Conference on Computer Vision: 2021: 10012–10022.

[16] 张平平. 基于全卷积模型的显著性目标检测算法研究. [D]. 大连理工大学, 2020.

[17] 蒋峰岭. 基于结构约束的视觉显著物体检测及其应用. [D]. 中国科学技术大学, 2021.

[18] HOU X, ZHANG L. Saliency detection: A spectral residual approach. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2007: 1–8.

[19] GUO C, MA Q, ZHANG L. Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2008: 1–8.

[20] GUO C, ZHANG L. A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression. [J]. IEEE Transactions on Image Processing, 2009, 19 (1): 185–198.

[21] ACHANTA R, HEMAMI S, ESTRADA F, et al. Frequency-tuned salient region detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2009: 1597–1604.

[22] GOFERMAN S, ZELNIK-MANOR L, TAL A. Context-aware saliency detection. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 34 (10): 1915–1926.

[23] WEI Y, WEN F, ZHU W, et al. Geodesic saliency using background priors. [C] // European Conference on Computer Vision. Springer: 2012: 29–42.

[24] CHENG M.-M, MITRA N J, HUANG X, et al. Global contrast based salient region detection. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 37 (3): 569–582.

[25] HAN B, ZHU H, DING Y. Bottom-up saliency based on weighted sparse coding residual. [C] // ACM International Conference on Multimedia: 2011: 1117–1120.

[26] LI X, LU H, ZHANG L, et al. Saliency detection via dense and sparse reconstruction. [C] // IEEE International Conference on Computer Vision: 2013: 2976–2983.

[27] ZHANG J, SCLAROFF S. Saliency detection: A boolean map approach. [C] // IEEE International Conference on Computer Vision: 2013: 153–160.

[28] JIANG B, ZHANG L, LU H, et al. Saliency detection via absorbing markov chain. [C] // IEEE International Conference on Computer Vision: 2013: 1665–1672.

[29] YANG C, ZHANG L, LU H, et al. Saliency detection via graph-based manifold ranking. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2013: 3166–3173.

[30] HE S, LAU R W, LIU W, et al. Supercnn: A superpixelwise convolutional neural network for salient object detection. [J]. International Journal of Computer Vision, 2015, 115 (3): 330–344.

[31] WANG L, LU H, RUAN X, et al. Deep networks for saliency detection via local estimation and global search. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2015: 3183–3192.

[32] LI G, YU Y. Visual saliency based on multiscale deep features. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2015: 5455–5463.

[33] ZHAO R, OUYANG W, LI H, et al. Saliency detection by multi-context deep learning. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2015: 1265–1274.

[34] LONG J, SHELHAMER E, DARRELL T. Fully convolutional networks for semantic segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2015: 3431–3440.

[35] WANG W, LAI Q, FU H, et al. Salient Object Detection in the Deep Learning Era: An In-depth Survey. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021: 1–1. DOI: 10.1109/TPAMI.2021.3051099.

[36] ZHANG P, WANG D, LU H, et al. Learning Uncertain Convolutional Features for Accurate Saliency Detection. [C] // IEEE International Conference on Computer Vision: 2017: 212–221.

[37] WANG L, WANG L, LU H, et al. Saliency Detection with Recurrent Fully Convolutional Networks. [C] // European Conference on Computer Vision. Springer: 2016: 825–841.

[38] HU P, SHUAI B, LIU J, et al. Deep level sets for salient object detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 2300–2309.

[39] KUEN J, WANG Z, WANG G. Recurrent attentional networks for saliency detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2016: 3668–3677.

[40] LI G, YU Y. Deep Contrast Learning for Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2016: 478–487.

[41] WANG T, BORJI A, ZHANG L, et al. A stagewise refinement model for detecting salient objects in images. [C] // IEEE International Conference on Computer Vision: 2017: 4019–4028.

[42] CHEN X, ZHENG A, LI J, et al. Look, perceive and segment: Finding the salient objects in images via two-stream fixation-semantic cnns. [C] // IEEE International Conference on Computer Vision: 2017: 1050–1058.

[43] ZENG Y, ZHANG P, ZHANG J, et al. Towards High-Resolution Salient Object Detection. [C] // IEEE International Conference on Computer Vision: 2019: 7234–7243.

[44] HOU Q, CHENG M.-M, HU X, et al. Deeply supervised salient object detection with short connections. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41 (4): 815–828.

[45] XIE S, TU Z. Holistically-nested edge detection. [C] // IEEE International Conference on Computer Vision: 2015: 1395–1403.

[46] LUO Z, MISHRA A K, ACHKAR A, et al. Non-local Deep Features for Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 6609–6617.

[47] WU Z, SU L, HUANG Q. Cascaded Partial Decoder for Fast and Accurate Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 3907–3916.

[48] ZHAO T, WU X. Pyramid Feature Attention Network for Saliency Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 3085–3094.

[49] CHEN S, TAN X, WANG B, et al. Reverse attention for salient object detection. [C] // European Conference on Computer Vision. Springer: 2018: 234–250.

[50] ZHANG X, WANG T, QI J, et al. Progressive Attention Guided Recurrent Network for Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 714–722.

[51] LIU N, HAN J, YANG M.-H. PiCANet: Learning Pixel-wise Contextual Attention for Saliency Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 3089–3098.

[52] WANG T, ZHANG L, WANG S, et al. Detect Globally, Refine Locally: A Novel Approach to Saliency Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 3127–3135.

[53] FENG M, LU H, DING E. Attentive Feedback Network for Boundary-Aware Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 1623–1632.

[54] QIN X, ZHANG Z, HUANG C, et al. BASNet: Boundary-Aware Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 7479–7489.

[55] ZHANG L, DAI J, LU H, et al. A Bi-Directional Message Passing Model for Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 1741–1750.

[56] KRUTHIVENTI S S, GUDISA V, DHOLAKIYA J H, et al. Saliency unified: A deep architecture for simultaneous eye fixation prediction and salient object segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2016: 5781–5790.

[57] WU R, FENG M, GUAN W, et al. A Mutual Learning Method for Salient Object Detection With Intertwined Multi-Supervision. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 8150–8159.

[58] ZHANG L, ZHANG J, LIN Z, et al. CapSal: Leveraging Captioning to Boost Semantics for Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 6024–6033.

[59] ZENG Y, ZHUGE Y, LU H, et al. Joint learning of saliency detection and weakly supervised semantic segmentation. [C] // IEEE International Conference on Computer Vision: 2019: 7223–7233.

[60] MOVAHEDI V, ELDER J H. Design and perceptual validation of performance measures for salient object segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2010: 49–56.

[61] LI Y, HOU X, KOCH C, et al. The secrets of salient object segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2014: 280–287.

[62] YAN Q, XU L, SHI J, et al. Hierarchical saliency detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2013: 1155–1162.

[63] WANG L, LU H, WANG Y, et al. Learning to detect salient objects with image-level supervision. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 136–145.

[64] MARTIN D, FOWLKES C, TAL D, et al. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. [C] // IEEE International Conference on Computer Vision. Vol. 2: 2001: 416–423.

[65] EVERINGHAM M, VAN GOOL L, WILLIAMS C K, et al. The pascal visual object classes (voc) challenge. [J]. International Journal of Computer Vision, 2010, 88 (2): 303–338.

[66] DENG J, DONG W, SOCHER R, et al. Imagenet: A large-scale hierarchical image database. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2009: 248–255.

[67] XIAO J, HAYS J, EHINGER K A, et al. Sun database: Large-scale scene recognition from abbey to zoo. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2010: 3485–3492.

[68] BORJI A, CHENG M.-M, JIANG H, et al. Salient object detection: A benchmark. [J]. IEEE Transactions on Image Processing, 2015, 24 (12): 5706–5722.

[69] FAN D.-P, CHENG M.-M, LIU Y, et al. Structure-measure: A New Way to Evaluate Foreground Maps. [C] // IEEE International Conference on Computer Vision: 2017: 4548–4557.

[70] CHENG Y, FU H, WEI X, et al. Depth enhanced saliency detection method. [C] // International Conference on Internet Multimedia Computing and Service: 2014: 23–27.

[71] ZHU C, LI G, WANG W, et al. An innovative salient object detection using center-dark channel prior. [C] // IEEE International Conference on Computer Vision Workshops: 2017: 1509–1515.

[72] QU L, HE S, ZHANG J, et al. RGBD salient object detection via deep fusion. [J]. IEEE Transactions on Image Processing, 2017, 26 (5): 2274–2285.

[73] SHIGEMATSU R, FENG D, YOU S, et al. Learning RGB-D salient object detection using background enclosure, depth contrast, and top-down features. [C] // IEEE International Conference on Computer Vision: 2017: 2749–2757.

[74] LIU Z, SHI S, DUAN Q, et al. Salient object detection for RGB-D image by single stream recurrent convolution neural network. [J]. Neurocomputing, 2019, 363: 46–57.

[75] CHEN H, LI Y. Progressively complementarity-aware fusion network for RGB-D salient object detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 3051–3060.

[76] HAN J, CHEN H, LIU N, et al. CNNs-based RGB-D saliency detection via cross-view transfer and multiview fusion. [J]. IEEE Transactions on Cybernetics, 2017, 48 (11): 3171–3183.

[77] ZHAO J.-X, CAO Y, FAN D.-P, et al. Contrast prior and fluid pyramid integration for RGBD salient object detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 3927–3936.

[78] CHEN H, LI Y, SU D. Multi-modal fusion network with multi-scale multi-path and cross-modal interactions for RGB-D salient object detection. [J]. Pattern Recognition, 2019, 86: 376–385.

[79] CHEN H, LI Y. Three-stream attention-aware network for RGB-D salient object detection. [J]. IEEE Transactions on Image Processing, 2019, 28 (6): 2825–2835.

[80] PIAO Y, JI W, LI J, et al. Depth-induced multi-scale recurrent attention network for saliency detection. [C] // IEEE International Conference on Computer Vision: 2019: 7254–7263.

[81] LIU N, ZHANG N, HAN J. Learning selective self-mutual attention for RGB-D saliency detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2020: 13756–13765.

[82] FAN D.-P, LIN Z, ZHANG Z, et al. Rethinking RGB-D salient object detection: Models, data sets, and large-scale benchmarks. [J]. IEEE Transactions on Neural Networks and Learning Systems, 2020.

[83] CANNY J. A computational approach to edge detection. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986 (6): 679–698.

[84] MARTIN D R, FOWLKES C C, MALIK J. Learning to detect natural image boundaries using local brightness, color, and texture cues. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26 (5): 530–549.

[85] ARBELAEZ P, MAIRE M, FOWLKES C, et al. Contour detection and hierarchical image segmentation. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33 (5): 898–916.

[86] DOLLAR P, TU Z, BELONGIE S. Supervised learning of edges and object boundaries. [C] // IEEE Conference on Computer Vision and Pattern Recognition. Vol. 2: 2006: 1964–1971.

[87] DOLLÁR P, ZITNICK C L. Fast edge detection using structured forests. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37 (8): 1558–1570.

[88] GANIN Y, LEMPITSKY V. N^ 4-fields: Neural network nearest neighbor fields for image transforms. [C] // Asian Conference on Computer Vision. Springer: 2014: 536–551.

[89] SHEN W, WANG X, WANG Y, et al. Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2015: 3982–3991.

[90] XU D, OUYANG W, ALAMEDA-PINEDA X, et al. Learning deep structured multi-scale features using attention-gated crfs for contour prediction. [C] // Advances in Neural Information Processing Systems: 2017: 3961–3970.

[91] YU Z, FENG C, LIU M.-Y, et al. Casenet: Deep category-aware semantic edge detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 5964–5973.

[92] WANG Y, ZHAO X, LI Y, et al. Deep crisp boundaries: From boundaries to higher-level tasks. [J]. IEEE Transactions on Image Processing, 2018, 28 (3): 1285–1298.

[93] LIU Y, CHENG M.-M, HU X, et al. Richer Convolutional Features for Edge Detection. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41 (8): 1939–1946. DOI: 10.1109/TPAMI.2018.2878849.

[94] BEHRENS R R. The theories of Abbott H. Thayer: Father of camouflage. [J]. Leonardo, 1988, 21 (3): 291–296.

[95] BEHRENS R R. Seeing through Camouflage: Abbott Thayer, Background-Picturing and the Use of Cutout Silhouettes. [J]. Leonardo, 2018, 51 (1): 40–46.

[96] CUTHILL I. Camouflage. [J]. Journal of Zoology, 2019, 308 (2): 75–92.

[97] STEVENS M, MERILAITA S. Animal camouflage: current issues and new perspectives. [J]. Philosophical Transactions of the Royal Society B: Biological Sciences, 2009, 364 (1516): 423–427.

[98] CUTHILL I C, STEVENS M, SHEPPARD J, et al. Disruptive coloration and background pattern matching. [J]. Nature, 2005, 434 (7029): 72–74.

[99] PIKE T W. Quantifying camouflage and conspicuousness using visual salience. [J]. Methods in Ecology and Evolution, 2018, 9 (8): 1883–1895.

[100] FAN D.-P, JI G.-P, SUN G, et al. Camouflaged object detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2020: 2777–2787.

[101] LI A, ZHANG J, LV Y, et al. Uncertainty-aware Joint Salient Object and Camouflaged Object Detection. [J]. ArXiv preprint arXiv:2104.02628, 2021.

[102] YU Z, BAJAJ C. A segmentation-free approach for skeletonization of gray-scale images via anisotropic vector diffusion. [C] // IEEE Conference on Computer Vision and Pattern Recognition. Vol. 1. IEEE: 2004: I–I.

[103] JANG J.-H, HONG K.-S. A pseudo-distance map for the segmentation-free skeletonization of gray-scale images. [C] // IEEE International Conference on Computer Vision. Vol. 2. IEEE: 2001: 18–23.

[104] MAJER P. On the influence of scale selection on feature detection for the case of linelike structures. [J]. International Journal of Computer Vision, 2004, 60 (3): 191–202.

[105] TSOGKAS S, KOKKINOS I. Learning-based symmetry detection in natural images. [C] // European Conference on Computer Vision. Springer: 2012: 41–54.

[106] SIRONI A, LEPETIT V, FUA P. Multiscale centerline detection by learning a scale-space distance transform. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2014: 2697–2704.

[107] LEVINSHTEIN A, SMINCHISESCU C, DICKINSON S. Multiscale symmetric part detection and grouping. [J]. International Journal of Computer Vision, 2013, 104 (2): 117–134.

[108] SHEN W, ZHAO K, JIANG Y, et al. Object skeleton extraction in natural images by fusing scale-associated deep side outputs. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2016: 222–230.

[109] SHEN W, ZHAO K, JIANG Y, et al. DeepSkeleton: Learning Multi-task Scale-associated Deep Side Outputs for Object Skeleton Extraction in Natural Images. [J]. IEEE Transactions on Image Processing, 2017, 26 (11): 5298–5311.

[110] KE W, CHEN J, JIAO J, et al. SRN: Side-output residual network for object symmetry detection in the wild. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 1068–1076.

[111] ZHAO K, SHEN W, GAO S, et al. Hi-Fi: Hierarchical Feature Integration for Skeleton Detection. [C] // International Joint Conference on Artificial Intelligence: 2018: 1191–1197. DOI: 10.24963/ijcai.2018/166.

[112] WANG Y, XU Y, TSOGKAS S, et al. DeepFlux for Skeletons in the Wild. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 5287–5296.

[113] LECUN Y, BOSER B, DENKER J, et al. Handwritten digit recognition with a back-propagation network. [C] // Advances in Neural Information Processing Systems. Vol. 2: 1989.

[114] LECUN Y, BOTTOU L, BENGIO Y, et al. Gradient-based learning applied to document recognition. [J]. Proceedings of the IEEE, 1998, 86 (11): 2278–2324.

[115] RANZATO M, BOUREAU Y.-L, LECUN Y, et al. Sparse feature learning for deep belief networks. [C] // Advances in Neural Information Processing Systems. Vol. 20: Vancouver, 2007: 1185–1192.

[116] LEE C.-Y, GALLAGHER P, TU Z. Generalizing pooling functions in cnns: Mixed, gated, and tree. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40 (4): 863–875.

[117] TOUTOUNCHI F, IZQUIERDO E. Advanced super-resolution using lossless pooling convolutional networks. [C] // IEEE Winter Conference on Applications of Computer Vision: 2019: 1562–1568.

[118] GAO Z, WANG L, WU G. Lip: Local importance-based pooling. [C] // IEEE International Conference on Computer Vision: 2019: 3355–3364.

[119] HOU Q, ZHANG L, CHENG M.-M, et al. Strip Pooling: Rethinking Spatial Pooling for Scene Parsing. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2020: 4003–4012.

[120] RONNEBERGER O, FISCHER P, BROX T. U-net: Convolutional networks for biomedical image segmentation. [C] // International Conference on Medical Image Computing and Computer Assisted Intervention: 2015: 234–241.

[121] LIN T.-Y, DOLLÁR P, GIRSHICK R B, et al. Feature Pyramid Networks for Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 2117–2125.

[122] LIU S, QI L, QIN H, et al. Path aggregation network for instance segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 8759–8768.

[123] LIU S, HUANG D, WANG Y. Learning spatial fusion for single-shot object detection. [J]. ArXiv preprint arXiv:1911.09516, 2019.

[124] TAN M, PANG R, LE Q V. Efficientdet: Scalable and efficient object detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2020: 10781–10790.

[125] QIAO S, CHEN L.-C, YUILLE A. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2021: 10213–10224.

[126] GHIASI G, LIN T.-Y, LE Q V. Nas-fpn: Learning scalable feature pyramid architecture for object detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 7036–7045.

[127] XU H, YAO L, ZHANG W, et al. Auto-fpn: Automatic network architecture adaptation for object detection beyond classification. [C] // IEEE International Conference on Computer Vision: 2019: 6649–6658.

[128] ZOPH B, LE Q V. Neural architecture search with reinforcement learning. [C] // International Conference on Learning Representations: 2017.

[129] ZHAO H, SHI J, QI X, et al. Pyramid scene parsing network. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 2881–2890.

[130] CHEN L.-C, PAPANDREOU G, KOKKINOS I, et al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40 (4): 834–848.

[131] YANG M, YU K, ZHANG C, et al. Denseaspp for semantic segmentation in street scenes. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 3684–3692.

[132] LIU C, CHEN L.-C, SCHROFF F, et al. Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 82–92.

[133] WOO S, PARK J, LEE J.-Y, et al. Cbam: Convolutional block attention module. [C] // European Conference on Computer Vision. Springer: 2018: 3–19.

[134] YUAN Y, WANG J. Ocnet: Object context network for scene parsing. [J]. ArXiv preprint arXiv:1809.00916, 2018.

[135] FU J, LIU J, TIAN H, et al. Dual attention network for scene segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 3146–3154.

[136] HUANG Z H ; X W ; Y W ; L H ; H S ; W L ; T S. CCNet: Criss-Cross Attention for Semantic Segmentation. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020: 1–1. DOI: 10.1109/TPAMI.2020.3007032..

[137] CARUANA R. Multitask learning. [J]. Machine Learning, 1997, 28 (1): 41–75.

[138] EVGENIOU T, PONTIL M. Regularized multi–task learning. [C] // ACM SIGKDD International Conference on Knowledge Discovery and Data Mining: 2004: 109–117.

[139] PAN S J, YANG Q. A survey on transfer learning. [J]. IEEE Transactions on Knowledge and Data Engineering, 2009, 22 (10): 1345–1359.

[140] THRUN S, PRATT L. Learning to learn. [M].: Springer Science & Business Media, 2012.

[141] MISRA I, SHRIVASTAVA A, GUPTA A, et al. Cross-stitch networks for multi-task learning. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2016: 3994–4003.

[142] DOERSCH C, ZISSERMAN A. Multi-task self-supervised visual learning. [C] // IEEE International Conference on Computer Vision: 2017: 2051–2060.

[143] RUSU A A, RABINOWITZ N C, DESJARDINS G, et al. Progressive neural networks. [J]. ArXiv preprint arXiv:1606.04671, 2016.

[144] KOKKINOS I. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 6129–6138.

[145] LIU S, JOHNS E, DAVISON A J. End-to-end multi-task learning with attention. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 1871–1880.

[146] KENDALL A, GAL Y, CIPOLLA R. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 7482–7491.

[147] CHEN Z, BADRINARAYANAN V, LEE C.-Y, et al. GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks. [C] // International Conference on Machine Learning: 2018: 793–802.

[148] REBUFFI S.-A, BILEN H, VEDALDI A. Learning multiple visual domains with residual adapters. [C] // Advances in Neural Information Processing Systems: 2017: 506–516.

[149] SERMANET P, EIGEN D, ZHANG X, et al. Overfeat: Integrated recognition, localization and detection using convolutional networks. [C] // International Conference on Learning Representations: 2014.

[150] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks. [C] // Advances in Neural Information Processing Systems: 2015: 91–99.

[151] HE K, GKIOXARI G, DOLLÁR P, et al. Mask r-cnn. [C] // IEEE International Conference on Computer Vision: 2017: 2961–2969.

[152] GKIOXARI G, HARIHARAN B, GIRSHICK R, et al. R-cnns for pose estimation and action detection. [J]. ArXiv preprint arXiv:1406.5212, 2014.

[153] KENDALL A, GRIMES M, CIPOLLA R. Posenet: A convolutional network for real-time 6-dof camera relocalization. [C] // IEEE International Conference on Computer Vision: 2015: 2938–2946.

[154] DU K, LIN X, SUN Y, et al. CrossInfoNet: Multi-Task Information Sharing Based Hand Pose Estimation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 9896–9905.

[155] EIGEN D, FERGUS R. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. [C] // IEEE International Conference on Computer Vision: 2015: 2650–2658.

[156] TEICHMANN M, WEBER M, ZOELLNER M, et al. Multinet: Real-time joint semantic reasoning for autonomous driving. [C] // IEEE Intelligent Vehicles Symposium: 2018: 1013–1020.

[157] GAO Y, MA J, ZHAO M, et al. NDDR-CNN: Layerwise Feature Fusing in Multi-Task CNNs by Neural Discriminative Dimensionality Reduction. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 3205–3214.

[158] ZHANG C, ZHANG Z. Improving multiview face detection with multi-task deep convolutional neural networks. [C] // IEEE Winter Conference on Applications of Computer Vision: 2014: 1036–1041.

[159] ZHANG Z, LUO P, LOY C C, et al. Facial landmark detection by deep multi-task learning. [C] // European Conference on Computer Vision. Springer: 2014: 94–108.

[160] QI G.-J. Hierarchically gated deep networks for semantic segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2016: 2267–2275.

[161] TAKIKAWA T, ACUNA D, JAMPANI V, et al. Gated-scnn: Gated shape cnns for semantic segmentation. [C] // IEEE International Conference on Computer Vision: 2019: 5229–5238.

[162] DING H, JIANG X, SHUAI B, et al. Context contrasted feature and gated multi-scale aggregation for scene segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 2393–2402.

[163] CHENG Y, CAI R, LI Z, et al. Locality-sensitive deconvolution networks with gated fusion for rgb-d indoor semantic segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 3029–3037.

[164] ZHU C, HE Y, SAVVIDES M. Feature selective anchor-free module for single-shot object detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 840–849.

[165] LI S, YANG L, HUANG J, et al. Dynamic anchor feature selection for single-shot object detection. [C] // IEEE International Conference on Computer Vision: 2019: 6609–6618.

[166] CHEN Z, LI Y, BENGIO S, et al. You look twice: GaterNet for dynamic filter selection in CNNs. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 9172–9180.

[167] LI X, WANG W, HU X, et al. Selective kernel networks. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 510–519.

[168] HUA W, ZHOU Y, DE SA C M, et al. Channel gating neural networks. [C] // Advances in Neural Information Processing Systems: 2019: 1884–1894.

[169] HONG S, YOU T, KWAK S, et al. Online tracking by learning discriminative saliency map with convolutional neural network. [C] // International Conference on Machine Learning: 2015: 597–606.

[170] WANG W, SHEN J, LING H. A deep network solution for attention and aesthetics aware photo cropping. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 41 (7): 1531–1544.

[171] CHENG M.-M, ZHANG F.-L, MITRA N J, et al. RepFinder: finding approximately repeated scene elements for image editing. [J]. ACM Transactions on Graphics, 2010, 29 (4): 83.

[172] GAO Y, WANG M, TAO D, et al. 3-D object retrieval and recognition with hypergraph analysis. [J]. IEEE Transactions on Image Processing, 2012, 21 (9): 4290–4303.

[173] WANG W, SHEN J, YANG R, et al. Saliency-aware video object segmentation. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40 (1): 20–33.

[174] CRAYE C, FILLIAT D, GOUDOU J.-F. Environment exploration for object-based visual saliency learning. [C] // IEEE International Conference on Robotics and Automation: 2016: 2303–2309.

[175] WEI Y, LIANG X, CHEN Y, et al. STC: A simple to complex framework for weakly-supervised semantic segmentation. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 39 (11): 2314–2320.

[176] SUN G, WANG W, DAI J, et al. Mining cross-image semantics for weakly supervised semantic segmentation. [C] // European Conference on Computer Vision. Springer: 2020: 347–365.

[177] LI G, XIE Y, LIN L, et al. Instance-Level Salient Object Segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 2386–2395.

[178] LIU N, HAN J. DHSNet: Deep Hierarchical Saliency Network for Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2016: 678–686.

[179] ZHANG P, WANG D, LU H, et al. Amulet: Aggregating Multi-level Convolutional Features for Salient Object Detection. [C] // IEEE International Conference on Computer Vision: 2017: 202–211.

[180] LIU J.-J, HOU Q, CHENG M.-M, et al. Improving Convolutional Networks With Self-Calibrated Convolutions. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2020: 10096–10105.

[181] ZHOU B, KHOSLA A, LAPEDRIZA A, et al. Object detectors emerge in deep scene cnns. [C] // International Conference on Learning Representations: 2015.

[182] SANDLER M, HOWARD A, ZHU M, et al. Mobilenetv2: Inverted residuals and linear bottlenecks. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 4510–4520.

[183] KINGMA D P, BA J. Adam: A method for stochastic optimization. [C] // International Conference on Learning Representations: 2015.

[184] ZHANG D, HAN J, ZHANG Y. Supervision by fusion: Towards unsupervised learning of deep salient object detector. [C] // IEEE International Conference on Computer Vision: 2017: 4048–4056.

[185] LI X, YANG F, CHENG H, et al. Contour knowledge transfer for salient object detection. [C] // European Conference on Computer Vision. Springer: 2018: 355–370.

[186] XU Y, XU D, HONG X, et al. Structured Modeling of Joint Deep Feature and Prediction Refinement for Salient Object Detection. [C] // IEEE International Conference on Computer Vision: 2019.

[187] WANG W, ZHAO S, SHEN J, et al. Salient Object Detection With Pyramid Attention and Salient Edges. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 1448–1457.

[188] WANG W, SHEN J, CHENG M.-M, et al. An Iterative and Cooperative Top-Down and Bottom-Up Inference Network for Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 5968–5977.

[189] CHENG M.-M, GAO S.-H, BORJI A, et al. A Highly Efficient Model to Study the Semantics of Salient Object Detection. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021. DOI: 10.1109/TPAMI.2021.3107956.

[190] MOTTAGHI R, CHEN X, LIU X, et al. The role of context for object detection and semantic segmentation in the wild. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2014: 891–898.

[191] YANG J, PRICE B, COHEN S, et al. Object contour detection with a fully convolutional encoder-decoder network. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2016: 193–202.

[192] LIU Y, LEW M S. Learning relaxed deep supervision for better edge detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2016: 231–240.

[193] MANINIS K.-K, PONT-TUSET J, ARBELAEZ P, et al. Convolutional Oriented Boundaries: From Image Segmentation to High-Level Tasks. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40 (4): 819–833.

[194] KOKKINOS I. Pushing the boundaries of boundary detection using deep learning. [C] // International Conference on Learning Representations: 2016.

[195] JU R, GE L, GENG W, et al. Depth saliency based on anisotropic center-surround difference. [C] // IEEE Conference on Image Processing: 2014: 1115–1119.

[196] NIU Y, GENG Y, LI X, et al. Leveraging stereopsis for saliency analysis. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2012: 454–461.

[197] CHENG Y, FU H, WEI X, et al. Depth enhanced saliency detection method. [C] // International Conference on Internet Multimedia Computing and Service: 2014: 23–27.

[198] PENG H, LI B, XIONG W, et al. RGBD salient object detection: a benchmark and algorithms. [C] // European Conference on Computer Vision. Springer: 2014: 92–109.

[199] FAN D.-P, LIN Z, ZHANG Z, et al. Rethinking RGB-D salient object detection: Models, data sets, and large-scale benchmarks. [J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 32 (5): 2075–2089.

[200] WANG N, GONG X. Adaptive fusion for RGB-D salient object detection. [J]. IEEE Access, 2019, 7: 55277–55284.

[201] FU K, FAN D.-P, JI G.-P, et al. Jl-dcf: Joint learning and densely-cooperative fusion framework for rgb-d salient object detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2020: 3052–3062.

[202] FAN D.-P, ZHAI Y, BORJI A, et al. BBS-Net: RGB-D salient object detection with a bifurcated backbone strategy network. [C] // European Conference on Computer Vision. Springer: 2020: 275–292.

[203] SKUROWSKI P, ABDULAMEER H, BŁASZCZYK J, et al. Animal camouflage analysis: Chameleon database. [J]. Unpublished Manuscript, 2018.

[204] LE T.-N, NGUYEN T V, NIE Z, et al. Anabranch network for camouflaged object segmentation. [J]. Computer Vision and Image Understanding, 2019, 184: 45–56.

[205] ZHOU Z, SIDDIQUEE M M R, TAJBAKHSH N, et al. Unet++: A nested u-net architecture for medical image segmentation. [G] // Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Springer, 2018: 3–11.

[206] HUANG Z, HUANG L, GONG Y, et al. Mask scoring r-cnn. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 6409–6418.

[207] CHEN K, PANG J, WANG J, et al. Hybrid task cascade for instance segmentation. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 4974–4983.

[208] ZHAO J.-X, LIU J.-J, FAN D.-P, et al. EGNet: Edge Guidance Network for Salient Object Detection. [C] // IEEE International Conference on Computer Vision: 2019: 8779–8788.

[209] HOU Q, JIANG P.-T, WEI Y, et al. Self-Erasing Network for Integral Object Attention. [C] // Advances in Neural Information Processing Systems. Vol. 31: 2018.

[210] PANG J, CHEN K, SHI J, et al. Libra r-cnn: Towards balanced learning for object detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 821–830.

[211] LI Z, LANG C, LIEW J H, et al. Cross-layer feature pyramid network for salient object detection. [J]. IEEE Transactions on Image Processing, 2021, 30: 4587–4598.

[212] HU J, SHEN L, SUN G. Squeeze-and-excitation networks. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2018: 7132–7141.

[213] IOFFE S, SZEGEDY C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. [C] // International Conference on Machine Learning: 2015.

[214] NAIR V, HINTON G E. Rectified linear units improve restricted boltzmann machines. [C] // International Conference on Machine Learning: 2010.

[215] LIU J.-J, HOU Q, CHENG M.-M, et al. A Simple Pooling-Based Design for Real-Time Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2019: 3917–3926.

[216] CHEN S, TAN X, WANG B, et al. Reverse Attention-Based Residual Network for Salient Object Detection. [J]. IEEE Transactions on Image Processing, 2020, 29: 3763–3776.

[217] ZHAO X, PANG Y, ZHANG L, et al. Suppress and Balance: A Simple Gated Network for Salient Object Detection. [C] // European Conference on Computer Vision. Springer: 2020: 35–51.

[218] ZHOU H, XIE X, LAI J.-H, et al. Interactive Two-Stream Decoder for Accurate and Fast Saliency Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2020: 9141–9150.

[219] PANG Y, ZHAO X, ZHANG L, et al. Multi-Scale Interactive Network for Salient Object Detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2020: 9413–9422.

[220] LIU J.-J, HOU Q, CHENG M.-M. Dynamic Feature Integration for Simultaneous Detection of Salient Object, Edge and Skeleton. [J]. IEEE Transactions on Image Processing, 2020, 29: 8652–8667. DOI: 10.1109/TIP.2020.3017352.

[221] WU Z, SU L, HUANG Q. Stacked Cross Refinement Network for Edge-Aware Salient Object Detection. [C] // IEEE International Conference on Computer Vision: 2019: 7264–7273.

[222] CHENG M.-M, HOU Q.-B, ZHANG S.-H, et al. Intelligent Visual Media Processing: When Graphics Meets Vision. [J]. Journal of Computer Science and Technology, 2017, 32 (1): 110–121. DOI: 10.1007/s11390-017-1681-7. ISSN: 1860-4749.

[223] CHENG M.-M, LIU X.-C, WANG J, et al. Structure-Preserving Neural Style Transfer. [J]. IEEE Transactions on Image Processing, 2020, 29: 909–920. DOI: 10.1109/TIP.2019.2936746.

[224] WU Y, HE K. Group normalization. [C] // European Conference on Computer Vision. Springer: 2018: 3–19.

[225] HINTON G E, SRIVASTAVA N, KRIZHEVSKY A, et al. Improving neural networks by preventing co-adaptation of feature detectors. [J]. ArXiv preprint arXiv:1207.0580, 2012.

[226] WANG Y, ZHAO X, HUANG K. Deep crisp boundaries. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2017: 3892–3900.

[227] PONT-TUSET J, ARBELAEZ P, BARRON J T, et al. Multiscale combinatorial grouping for image segmentation and object proposal generation. [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39 (1): 128–140.

[228] BERTASIUS G, SHI J, TORRESANI L. Deepedge: A multi-scale bifurcated deep network for top-down contour detection. [C] // IEEE Conference on Computer Vision and Pattern Recognition: 2015: 4380–4389.

[229] DENG R, SHEN C, LIU S, et al. Learning to predict crisp boundaries. [C] // European Conference on Computer Vision. Springer: 2018: 562–578.

[230] LIU C, KE W, QIN F, et al. Linear Span Network for Object Skeleton Detection. [C] // European Conference on Computer Vision. Springer: 2018: 133–148.

致谢

五年光阴转瞬即逝。本人由衷地感谢导师程明明教授在本人五年硕博期间在学习和科研各方面的耐心帮助和悉心指导。作为导师，程明明教授对许多事情有着独到的观察和见解。在程明明教授卓有远见的规划和建议下，本人在很多重要的选择上得以少走弯路。

本人感谢五年多来实验室各位小伙伴的帮助和陪伴，特别是侯淇彬学长在本人科研初期在实验和写作上的手把手指导，以及吹牛吐槽五人组之间无话不说而带来的安心与快乐。本人会一直记得在媒体计算实验室这段积极而自由的时光，可以按照自己的想法和意愿去心无旁骛地研究。

此外，本人感谢论文的各位合作者的努力与付出，以及在字节跳动 AI Lab 和腾讯优图实习期间各位领导和同事的指导和包容，让本人受益良多。

最后，本人感谢我的父母多年来的养育之恩，是你们的支持与宽容塑造了今天的我。本人感谢各位家人、老师及朋友在本人求学途中的陪伴和帮助，是你们的监督与鼓励让我有了突破自我的能力和信心。

特别地，作为新冠爆发初期在武汉的一员，本人由衷感谢全国人民的忘我相助，是你们逆流而上的勇敢和无畏给我们带来了春天。希望疫情早点过去，大家都能健健康康、快快乐乐、无忧无虑地去拥抱这个美丽的世界。

个人简历

刘姜江，男，四川达州人，出生于 1995 年 2 月 18 日。2013 年 9 月在南开大学电子信息与光学工程学院电子信息科学与技术专业就读，2017 年 6 月毕业并获得理学学士学位。2017 年 9 月在南开大学计算机学院计算机科学与技术专业攻读硕士学位，并于 2019 年以硕博连读方式继续攻读该专业博士学位至今。

研究生期间已发表论文:

1. **Jiang-Jiang Liu***, Zhi-Ang Liu*, Pai Peng, and Ming-Ming Cheng. Rethinking the U-Shape Structure for Salient Object Detection[J]. IEEE Transactions on Image Processing (TIP), 2021, 30: 9030-9042. (SCI 源刊, 中科院一区, CCF A 类期刊, 影响因子 10.856.)
2. **Jiang-Jiang Liu**, Qibin Hou, and Ming-Ming Cheng. Dynamic Feature Integration for Simultaneous Detection of Salient Object, Edge and Skeleton[J]. IEEE Transactions on Image Processing (TIP), 2020, 29: 8652-8667. (SCI 源刊, 中科院一区, CCF A 类期刊, 影响因子 10.856.)
3. **Jiang-Jiang Liu***, Qibin Hou*, Ming-Ming Cheng, Changhu Wang, and Jiashi Feng. Improving Convolutional Networks with Self-Calibrated Convolutions[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020: 10096-10105. (EI 源刊, CCF A 类会议.)
4. **Jiang-Jiang Liu***, Qibin Hou*, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A Simple Pooling-Based Design for Real-Time Salient Object Detection[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019: 3917-3926. (EI 源刊, CCF A 类会议.)
5. Jiaxing Zhao, **Jiang-Jiang Liu**, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. EGNet: Edge Guidance Network for Salient Object Detection[C]. IEEE/CVF International Conference on Computer Vision (ICCV), 2019: 8779-8788. (EI 源刊, CCF A 类会议.)
6. Deng-Ping Fan, Ming-Ming Cheng, **Jiang-Jiang Liu**, Shang-Hua Gao, Qinbin

Hou, and Ali Borji. Salient Objects in Clutter: Bringing Salient Object Detection to the Foreground[C]. European Conference on Computer Vision (ECCV), 2018: 186-202. (EI 源刊, CCF B 类会议.)

研究生期间其它成果:

1. 刘姜江; 程明明; 侯淇彬; 范登平; 谭永强. 一种基于深度网络的多类型任务通用的检测方法 [P]. 中国专利: ZL201810173285.7, 2018-08-21.
2. 刘姜江; 程明明; 侯淇彬. 一种基于特征动态选择的多任务联合检测方法 [P]. 中国专利: CN111598107A, 2020-08-28.
3. 刘姜江; 程明明; 彭剑威; 于金波. 检测方法及装置 [P]. 中国专利: CN111833363A, 2020-10-27.
4. 程明明; 刘姜江; 刘志昂. 基于集中式信息交互的显著性目标检测方法及系统 [P]. 中国专利: CN112507933A, 2021-03-16.

研究生期间参与课题:

1. 图像场景理解. 国自科优青项目. 项目号: 61922046.
2. 演化认知深度学习算法规划. 教育部指导高校科技创新规划项目.
3. 知识引导的自适应感知与结构理解. “新一代人工智能”重大项目. 项目号: 2018AAA0100400.
4. 场景语义智能识别与理解技术. 天津市新一代人工智能科技重大专项. 项目号: 18ZXZNGX00110.
5. 认知规律启发的弱监督图像场景理解. 天津市杰出青年科学基金. 项目号: 17JCJQJC43700.
6. 3D 多视点全景视频的室内场景重构理论及算法的国际合作研究. 国自科国际合作重点. 项目号: 61620106008.
7. 移动设备上的图像交互式分析与编辑. 国自科面上项目. 项目号: 61572264.