

分类号 _____

密级 _____

UDC^{注 1} _____



南京理工大学
NANJING UNIVERSITY OF SCIENCE & TECHNOLOGY

博士 学 位 论 文

显著物体检测计算方法及其应用

楼 竞

指导教师姓名 _____ 任明武 教授

学 位 类 别 _____ 工学博士

学 科 名 称 _____ 计算机应用技术

研 究 方 向 _____ 显著物体检测

论文提交日期 _____ 2018.10

注 1: 注明《国际十进分类法 UDC》的类号

显著物体检测计算方法及其应用

南京理工大学

博士 学位 论文

显著物体检测计算方法及其应用

作 者：楼 竞

指导教师：任明武 教授

南京理工大学

2018 年 10 月

Ph.D. Dissertation

Salient Object Detection: Computational Methods and Applications

By
Jing Lou

Supervised by Prof. Mingwu Ren

Nanjing University of Science & Technology

October, 2018

声 明

本学位论文是我在导师的指导下取得的研究成果，尽我所知，在本学位论文中，除了加以标注和致谢的部分外，不包含其他人已经发表或公布过的研究成果，也不包含我为获得任何教育机构的学位或学历而使用过的材料。与我一同工作的同事对本学位论文做出的贡献均已在论文中作了明确的说明。

研究生签名：_____

年 月 日

学位论文使用授权说明

南京理工大学有权保存本学位论文的电子和纸质文档，可以借阅或上网公布本学位论文的部分或全部内容，可以向有关部门或机构送交并授权其保存、借阅或上网公布本学位论文的部分或全部内容。对于保密论文，按保密的有关规定和程序处理。

研究生签名：_____

年 月 日

摘要

人类视觉系统从外部世界接收到海量的视觉输入信息，选择性注意机制使人类具有从中抽取出重要信息并实时诠释复杂场景的能力。在过去几十年里，心理学家和生理学家们一直试图研究和揭示选择性注意的内部机理，并提出了视觉显著性这一概念，已被广泛地应用于诸如目标识别、视频摘要、图像压缩、图像编辑、视觉跟踪等众多计算机视觉任务中。作为视觉显著性计算的研究分支，显著物体检测旨在从场景中检测出显著的物体并得到这些物体的精确分割，一个好的显著物体检测方法应同时满足高检测性能和低计算复杂度的要求。深入研究显著物体检测技术，可以逐步跨越底层特征与图像内容之间的鸿沟，使高层的场景理解成为可能。本文围绕显著物体检测技术与应用展开了相关研究，提出了两种显著物体检测方法和一种协同显著物体检测方法，并将显著性技术应用于单帧彩色图像小目标检测任务中。本文主要工作和创新点包括：

(1) 提出了一种基于区域主色的显著物体检测方法。该方法首先对原始图像在 RGB 空间进行最小方差量化，在 LAB 空间度量颜色差异，计算量化颜色的全局显著性；然后对最小方差量化图像进行超像素分割，将区域内频数最高的颜色作为区域主色计算局部显著性；最后计算区域间距离以及区域与图像中心间距离，并基于这两种空间关系对局部显著性加权。实验结果表明，该方法可在更少的颜色数下获得更好的检测性能，并降低了算法复杂度。

(2) 提出了一种基于颜色名空间的显著物体检测方法。该方法首先将原始图像从 RGB 空间转换到颜色名空间，在对颜色名通道序列化阈值分割的基础上，采用环绕性线索对图形-背景分离计算显著图；然后将原始图像转换为颜色名图像，计算全局颜色名统计特征和对比特征，并与环绕性线索结合计算加权显著图；最后，对显著图与加权显著图采用高亮截断的线性融合和后处理，得到显著物体区域。实验结果表明，该方法可弥补仅基于环绕性线索检测显著物体的不足，并在突出显著物体区域的同时抑制非显著的背景区域。

(3) 提出了一种基于分层结构与颜色名的协同显著物体检测方法。该方法首先对原始图像构造三个图像层，并融合颜色名和背景线索计算单层显著图；然后基于显著一致性和颜色名稀缺性对单层显著图融合并改进，计算单图显著图；最后对单图显著图采用自适应的阈值分割得到显著区域，并基于颜色名相似性去除非协同显著区域获得协同显著图。除可用于检测协同显著物体外，该方法也是一种针对单张图像的显著物体检测方法，实验结果表明提出的融合机制和颜色名线索的有效性。

(4) 针对显著性的具体应用，提出了一种融合区域稳定性与显著性的单帧彩色图像小目标检测方法，标注发布了一个用于单帧彩色图像小目标检测的基准数据库，并提供了相应的性能评估代码。该方法首先构造五种区域描述子和四个稳定性度量准则，并基

于区域稳定性度量，在提取候选小目标的同时得到目标区域精确的特征描述；然后采用局部颜色差异计算像素级显著性，在突出小目标的同时抑制背景区域；最后，融合算法基于候选小目标的平均显著值剔除虚警目标，得到小目标检测结果。实验结果表明，该方法可有效弥补仅基于稳定性或显著性检测小目标的不足，以及融合算法的性能优势。

关键词：选择性注意，视觉显著性计算，显著物体检测，区域主色，颜色名空间，协同显著物体检测，小目标检测，区域稳定性，区域显著性

Abstract

Human visual system receives a mass of information from the outside world. As one of many remarkable abilities of human beings, selective attention allows us to extract important information from abundant visual inputs and interpret complex scenes in real time. Over the past several decades, many psychologists and physiologists have been trying to research how selective attention helps us to deal with a huge amount of visual information efficiently. The visual saliency was proposed and has been widely used in many computer vision tasks, such as object recognition, video summarization, image compression, image editing, and visual tracking, etc. As a branch of visual saliency computation, salient object detection is about detecting salient objects and segmenting the accurate regions of them in visual scenes. A good salient object detection method should meet high detection performance with low computational complexity. In-depth study of salient object detection can help to bridge the gap between low-level features and image contents, and make high-level scene understanding possible. This dissertation proposes two salient object detection methods and a co-salient object detection method, then applies saliency technique to small target detection in a color image. The main contributions of this dissertation are:

(1) A regional principal color based salient object detection method is proposed. The proposed method first quantizes the input image using minimum variance quantization in RGB space, and computes global saliency by measuring color difference in LAB space. After segmenting the quantized image into regions using a superpixel segmentation method, the proposed method chooses the most frequently occurring color of each region as the regional principal color to further compute local saliency. Finally, by exploiting two kinds of spatial relationships, the proposed method can suppress non-salient regions meanwhile highlight central ones. The proposed method can reduce the computational complexity of saliency computation and achieve higher performance with fewer quantitative colors.

(2) A color name space based salient object detection method is proposed. The proposed method first converts the input image from RGB space to color name space, and obtains a saliency map by sequentially segmenting each color name channel and using a surroundedness cue based figure-ground segregation. Then the proposed method converts the input image to the color name image, from which two global color name features are derived and coupled with the surroundedness cue to generate a weighted saliency

map. Finally, the proposed method combines the aforementioned two saliency maps using a truncation operation and a post-processing procedure. The proposed method can suppress background regions effectively and simultaneously highlight salient objects uniformly.

(3) A hierarchical co-salient object detection method is proposed. The proposed method first constructs three image layers for each input image, and generates three single-layer saliency maps by measuring the saliency consistency between the saliency results of color names and background detection. Then these single-layer saliency maps are refined by invoking a global rarity cue of color names. The saliency consistency based combination and the color rarity based refinement are also used in the stage of multi-layer fusion. Finally, the proposed method exploits a color repeatedness cue to discard non-co-salient regions and generate co-saliency maps. In addition to being able to detect co-salient objects from a pair of input images, the proposed method can also detect salient objects in a single image. Experimental results show that the intra- and inter-saliency can benefit from the usage of color names.

(4) A small target detection method combining regional stability and saliency is proposed. By exploiting five regional descriptors and four stability metrics, the proposed method uses a stability measure based regional stability algorithm to generate a set of small target candidates with their precise feature descriptions. Then a local color contrast based regional saliency algorithm is proposed, which can highlight small salient regions and simultaneously suppress background regions. Finally, the proposed method integrates the obtained stability and saliency maps, and uses the mean saliency of each small target candidate to remove false alarms. Experimental results show the validity of the proposed fusion mechanism. What's more, a new benchmark dataset for small target detection is labelled and published, which includes human annotated ground truth and experimental evaluation code.

Keywords: selective attention, visual saliency computation, salient object detection, regional principal color, color name space, co-salient object detection, small target detection, regional stability, regional saliency

目 录

摘要	I
Abstract	III
目录	V
插图目录	IX
表格目录	XIII
1 绪论	1
1.1 论文背景与研究意义	1
1.2 显著物体检测课题的研究现状	3
1.2.1 视觉显著性计算的主要任务	3
1.2.2 显著物体检测的研究现状	6
1.2.3 显著物体检测数据集的建立	11
1.2.4 显著物体检测的性能度量	13
1.3 论文研究内容概述与创新点	20
1.4 论文内容安排	21
2 基于区域主色的显著物体检测	23
2.1 引言	23
2.2 研究动机与出发点	23
2.3 方法框架	24
2.4 基于区域主色的显著物体检测方法	27
2.4.1 基于最小方差量化的全局显著性计算	27
2.4.2 基于区域主色的局部显著性计算	31
2.4.3 基于空间距离的局部显著性加权	33
2.5 实验与讨论	36
2.5.1 数据集简介	36
2.5.2 参数分析	37
2.5.3 量化方式和中心加权的影响	39
2.5.4 与其它显著性方法的性能比较	43
2.5.5 实验结果讨论	47

2.6 本章小结	49
3 基于颜色名空间的显著物体检测	51
3.1 引言	51
3.2 研究动机与出发点	52
3.3 方法框架	55
3.4 基于颜色名空间的显著物体检测方法	55
3.4.1 基于颜色名通道和环绕性线索的注意图生成	55
3.4.2 基于线性融合和后处理的显著性计算	61
3.4.3 基于全局颜色名线索的显著性加权	65
3.4.4 基于高亮截断和后处理的显著性融合	70
3.5 实验与讨论	72
3.5.1 数据集简介	72
3.5.2 参数分析	75
3.5.3 与其它显著性方法的性能比较	78
3.5.4 实验结果讨论	83
3.6 本章小结	85
4 基于分层结构与颜色名的协同显著物体检测	87
4.1 引言	87
4.2 研究动机与出发点	88
4.3 方法框架	89
4.4 基于分层结构与颜色名的协同显著物体检测方法	91
4.4.1 基于显著图一致性的单层融合	91
4.4.2 基于稀缺性与一致性的单层融合图改进	95
4.4.3 基于显著图一致性的多层次融合	98
4.4.4 基于稀缺性与一致性的多层次融合图改进	100
4.4.5 基于颜色名一致性的协同显著物体检测	101
4.5 实验与讨论	104
4.5.1 数据集简介	104
4.5.2 参数分析	105
4.5.3 融合算法的性能评估	109
4.5.4 与其它显著性/协同显著性方法的性能比较	110
4.5.5 实验结果讨论	118
4.6 本章小结	120

5 融合区域稳定性与显著性的单帧彩色图像小目标检测	121
5.1 引言	121
5.2 研究动机与出发点	122
5.3 方法框架	123
5.4 融合区域稳定性与显著性的小目标检测方法	125
5.4.1 基于区域稳定性的候选小目标提取	125
5.4.2 仅基于区域稳定性检测小目标的缺点	133
5.4.3 基于区域显著性的虚警目标去除	134
5.4.4 仅基于区域显著性检测小目标的缺点	136
5.5 实验与讨论	138
5.5.1 数据库简介	138
5.5.2 参数分析	139
5.5.3 与其它方法的性能比较	145
5.5.4 实验结果讨论	151
5.6 本章小结	152
6 结束语	153
6.1 论文工作总结	153
6.2 未来工作展望	154
致谢	157
参考文献	161
附录	175

插图目录

1.1	基于显著物体检测的车牌定位	2
1.2	显著物体检测	3
1.3	协同显著物体检测	4
1.4	人眼视点预测	5
1.5	候选目标生成	6
1.6	Image Set A 标注	12
1.7	Image Set B 标注	12
1.8	显著图	13
1.9	ROC 曲线图与 AUC	15
1.10	$P-R$ 曲线图与 F_β 曲线图	17
1.11	AvgF 指标	18
1.12	$P-R-F_\beta$ 柱状图	20
1.13	本文研究框架	22
2.1	RPC 方法框架	25
2.2	不同量化方式的 RGB 颜色空间分布	26
2.3	均匀量化和最小方差量化	28
2.4	颜色直方图	30
2.5	全局显著图	31
2.6	基于图割方法得到的超像素	32
2.7	区域主色	33
2.8	局部显著图	33
2.9	基于区域间距离的显著性加权	35
2.10	基于区域与图像中心间距离的显著性加权	36
2.11	ASD 数据集特性统计	37
2.12	RPC 方法参数分析	38
2.13	量化方式及颜色数的影响	41
2.14	中心加权的影响	42
2.15	与 24 个显著性方法的性能比较	45
2.16	与 8 个显著性方法的视觉比较	47
2.17	检测失败示例	48

3.1 基于环绕性线索的检测示例	53
3.2 CNS 方法框架	54
3.3 颜色名空间的 11 个颜色名通道	58
3.4 对黑色通道 C_1 序列化分割得到的布尔图	59
3.5 注意图	60
3.6 主注意图和平均注意图	62
3.7 后处理算法 I	63
3.8 颜色名图像和颜色名直方图	66
3.9 基于统计与分布的颜色名索引矩阵和权重矩阵	67
3.10 加权主注意图和加权平均注意图	69
3.11 将后处理算法 I 应用于加权平均注意图 \bar{A}_w	69
3.12 基于高亮截断的线性融合	70
3.13 后处理算法 II	71
3.14 ECSSD 数据集	73
3.15 ECSSD 数据集特性统计	73
3.16 ImgSal 数据集	74
3.17 ImgSal 数据集特性统计	74
3.18 CNS 方法参数分析	76
3.19 与 21 个显著性方法的性能比较	81
3.20 与 8 个显著性方法的视觉比较	83
3.21 检测失败示例	84
4.1 HCN 方法框架	90
4.2 单层融合算法	93
4.3 图像边框的影响	94
4.4 单层融合图改进	98
4.5 多层融合算法	100
4.6 多层融合图改进	101
4.7 基于颜色名的协同显著性	103
4.8 Image Pair 数据集特性统计	105
4.9 HCN 方法参数分析	106
4.10 与 23 个显著性方法的性能比较	112
4.11 与 8 个显著性方法的视觉比较	114
4.12 与 6 个协同显著性方法的性能比较	116
4.13 与 6 个协同显著性方法的视觉比较	117
4.14 检测失败示例	119

5.1 RSS 方法框架	124
5.2 小目标区域稳定性示例	126
5.3 序列化采样分割得到的部分布尔图	129
5.4 目标与表象间的隶属关系	130
5.5 RS _T 算法检测结果	132
5.6 RS _T 算法缺点	133
5.7 小目标的局部差异性	135
5.8 与 12 个显著性方法的视觉比较	136
5.9 RS _A 算法缺点	137
5.10 数据集部分图像与人工标注	138
5.11 MSER 方法参数分析	140
5.12 RSS 方法参数分析	143
5.13 与 12 个显著性方法的性能比较	147
5.14 检测失败示例	151

表格目录

1.1	Image Set A 标注一致性	12
2.1	量化结果与原始图像的比较	29
2.2	RPC 方法参数分析	39
2.3	RPC 方法最优参数	39
2.4	量化方式及颜色数的影响	40
2.5	中心加权的影响	43
2.6	与 24 个显著性方法的性能比较	46
3.1	英语中 11 个基本的颜色名	56
3.2	基于统计的颜色名概率	66
3.3	CNS 方法参数分析	77
3.4	CNS 方法最优参数	78
3.5	与 21 个显著性方法的性能比较	82
4.1	HCN 方法参数分析	107
4.2	HCN 方法最优参数	109
4.3	单层融合算法的性能比较	109
4.4	多层融合算法的性能比较	110
4.5	与 23 个显著性方法的性能比较	113
4.6	与 6 个协同显著性方法的性能比较	118
5.1	五种区域描述子的符号表示与定义	127
5.2	数据集简介	139
5.3	MSER 方法参数分析	141
5.4	MSER 方法最优参数	141
5.5	RSS 方法最优参数	142
5.6	RSS 方法参数分析	144
5.7	稳定性方法的性能比较	145
5.8	显著性方法的 F_β 相关指标比较	148
5.9	显著性方法的精确率、召回率和 F_β 指标的平均值比较	148
5.10	融合方法的性能比较	150

1 绪论

1.1 论文背景与研究意义

早在公元前 350 年，古希腊哲学家亚里士多德就在《On Sense and the Sensible》中提出^①：“人类在同一个感官行为中不可能同时感知到两个物体”，这其中就暗含了选择性注意 (Selective Attention) 的概念。M. Corbetta 认为^[1]：“注意”是一种在与行为相关的刺激、响应、记忆或思考中进行选择的心理能力，但由于脑容量导致的计算资源限制，为了确保行为受相关信息控制必须进行“选择”。当我们在观察一个场景时，似乎总有一种错觉，就是人脑强大的处理能力使我们应该能够掌握场景中的每处细节，但事实真的如此吗？想象这样一个画面，在一片绿色的草地上盛开着一朵红色的小花，在观察这样的场景时，人类视觉系统的注意力似乎总能被小花吸引过去，而忽视冗余的草地区域。事实上，虽然人类具有出色的对场景的整体把握和快速理解能力，但在一个很短的时刻内，人眼只能聚焦于视野中很小的一块区域，并从中观察到具体的细节信息。随着时间的推移，注视点将依据场景内容的重要程度在场景中不断转移，并从当前位置跳转到下一个重要的位置，直至掌握了场景中的大部分重要内容^[2]。

上述的“选择性注意”机制直接带来了两个问题：何种信号刺激能引起人眼的视觉注意，以及人类是如何从这些信号刺激中察觉到重要内容并在场景中进行视点转移？这较早地引起了认知心理学家^[3,4] 和神经生物学家^[5,6] 的关注，并提出了显著性 (Saliency) 和显著图 (Saliency Map) 的概念。特征集成理论^[4] 认为，视觉场景在可分离的维度上被初始地编码为一些基本特征，如颜色、朝向、空间频率、亮度、运动方向等，这些特征以一种早期的、自动的、并行的方式被人类所感知并进入视野，在引起视觉注意后，人类开始将这些特征重组并顺序地对引起刺激的物体进行处理。选择性注意转移理论^[6] 则认为，这些早期特征表示会被集成到显著图中，显著图编码了视觉场景中相对显著的位置，人类以一种“赢者通吃” (Winner-Take-All) 的机制处理显著图并从中挑选出最为显著的位置，之后受到相邻性和相似性偏好的影响，视觉注意会自动地转移到下一个最为显著的位置。受这些理论的启发，L. Itti 等人^[7] 在 1998 年提出了最为经典的视觉显著性计算模型。

视觉显著性计算 (Visual Saliency Computation) 是以可编程的方式检测和理解视觉场景中对人而言相对重要的区域。通过计算模型的表达，帮助理解人类是如何能够在海量的视觉输入信号中，过滤掉那些冗余的信息，并与生俱来地拥有快速掌握重要信息的能力。在计算机视觉领域，显著性研究可以从物体或区域层面找到场景的有效表示，并帮助解决诸如场景理解等复杂的视觉问题^[8]。在早期的视觉显著性计算模型中，以

^① J. I. Beare 译：<http://classics.mit.edu/Aristotle/sense.2.2.html>。

L. Itti 等人的工作^[7]为代表，主要研究人眼视点的预测问题。2007 年，刘铁等人^[9]认为人类在观察图像时会很自然地将注意力聚焦于图中的显著物体，而不是具有高对比结构的小的局部区域，据此提出了显著物体的概念，并将显著物体检测 (Salient Object Detection) 定义为从背景中分离出显著物体的二值标注问题。自此，显著物体检测作为视觉显著性计算的一个分支，在过去的十年里得到了飞速发展，并成为显著性领域最为热门的研究课题。

作为通常用于预处理或后处理阶段的一项辅助技术，显著物体检测的真正价值是由其应用所决定的。当显著物体检测技术用于预处理阶段时，可对场景内容进行有效筛选，剔除掉那些与任务无关的背景区域的干扰，定位到任务相关的位置或直接分割出对应的物体区域，以避免浪费宝贵的计算资源，实现更为高效的处理操作。例如，南开大学的程明教授团队利用显著物体检测，在手机拍摄过程中自动找到场景中的主要物体区域，实现“前清后虚”的大光圈效果，已成功应用于华为 Mate 10 手机^①。图 1.1 则示例了将第 3 章的部分技术应用于车牌粗定位的结果，从图 1.1(e) 可以看到，如果将基于显著物体检测预处理技术得到的候选车牌区域进一步用于车牌检测或车牌识别，可以大大减少在整幅图像中定位车牌区域的时间消耗。本文第 5 章将显著性检测技术应用于单帧彩色图像小目标检测的后处理阶段，通过判断候选小目标是否具有局部显著性来剔除虚警目标，使检测到的小目标同时满足局部稳定性和局部显著性的要求。

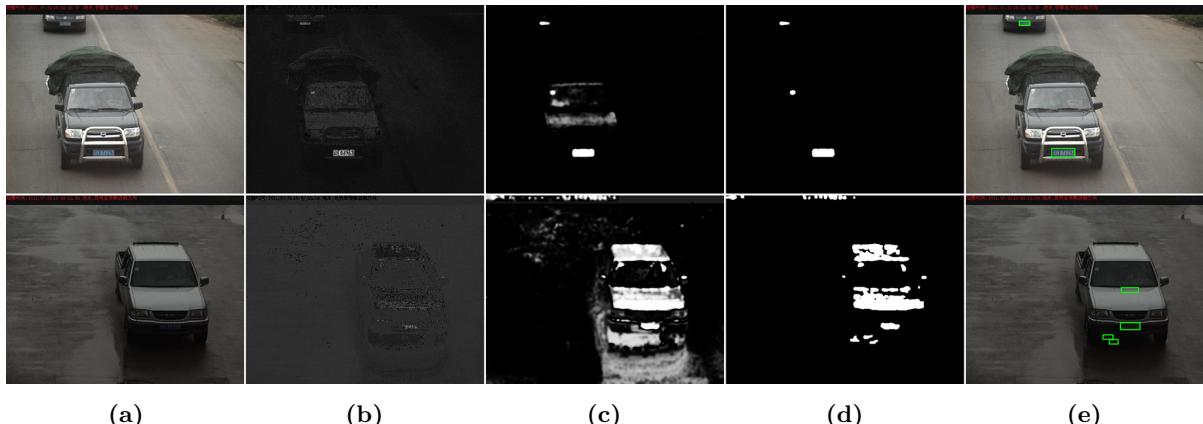


图 1.1 基于显著物体检测的车牌定位。(a) 输入图像；(b) 颜色名通道图；(c) 基于图(b)的亮度全局稀缺性并采用亮度变换得到的显著图；(d) 对图(c)采用自适应阈值分割和形态学处理得到的显著区域；(e) 对图(d)采用环绕性和车牌宽高比约束得到的候选车牌区域

目前，显著物体检测技术已被广泛应用于目标检测^[10,11]、目标识别^[12,13]、物体发现^[14]、目标与图像分割^[15,16]、图像编辑^[17]、图像检索^[18]、图像质量评价^[19]、人机交互^[20]、场景分类^[21]、图像与视频压缩^[22,23]、视频摘要^[24,25]、视觉跟踪^[26,27]等诸多领域。深入研究显著物体检测技术，可以逐步跨越底层特征与图像内容之间的鸿沟，使高层的场景理解成为可能。本文围绕显著物体检测展开的相关研究与应用，只是诸多成果

^① 南开新闻网：<http://news.nankai.edu.cn/nkyw/system/2017/12/24/000362595.shtml>。

中的冰山一角，随着计算机视觉、计算机图形学、机器人等领域的蓬勃发展，该课题未来必定还有更大的发展空间和应用前景。

1.2 显著物体检测课题的研究现状

本节主要从研究任务、现有工作、实验数据集以及性能度量指标与评估方法四个方面，介绍显著物体检测课题的研究现状。

1.2.1 视觉显著性计算的主要任务

依据研究目标和输出结果在视觉呈现上的不同，针对静态场景的显著性检测的主要任务包括显著物体检测、人眼视点预测、候选目标生成，以及近年来从显著物体检测衍生出的一个新的分支——协同显著物体检测。

(1) 显著物体检测 (Salient Object Detection)

计算机视觉中，显著物体检测的主要任务是检测出场景中最为显著的物体，并得到显著物体区域的精确分割。因此，显著物体检测本质上是一个图形–背景 (Figure–Ground) 分割问题，其最终目的是将显著的前景图形从背景中分割出来^[28]。与传统的图像分割问题不同，图像分割需要将整个场景划分为一组感知上相似的子块区域，这其中不仅包括对前景图形的分割，也包括对图像背景的分割，而显著物体检测仅需要分割出显著的前景图形。由于图像分割属于计算机视觉的另一研究范畴，因此除少部分工作^[29–31] 明确地将检测与分割这两个任务加以区分外，绝大部分的显著物体检测方法都遵循文献 [9, 32] 的处理方式，仅关注场景中的显著区域检测问题。

图 1.2 给出了显著物体检测的典型示例，第 1 行是来自于 MSRA 数据集^[9] 的原始图像，第 2 行是文献 [32] 提供的人工标注结果。可以看到，每幅原始图像中都有明确的单个显著物体存在，而对应的人工标注是一幅二值图像，其中显著物体区域的像素值为 1，背景区域的像素值为 0。既然显著物体检测最终都要回归到对显著物体的分割上来，那么显著物体检测方法理想的输出结果也应接近于人工标注图。

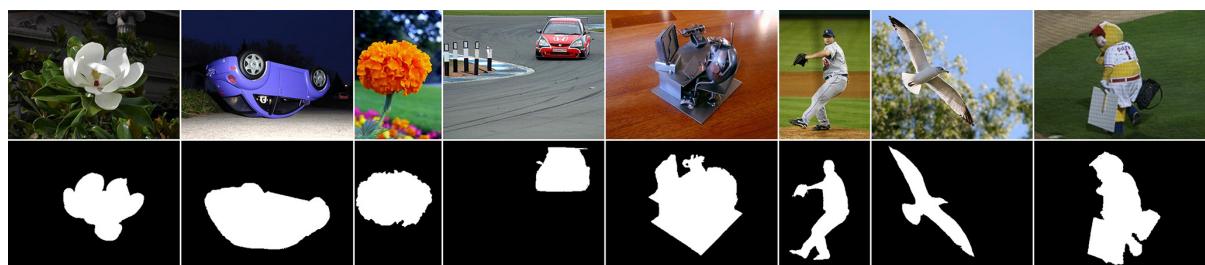


图 1.2 显著物体检测

(2) 协同显著物体检测 (Co-salient Object Detection)

近年来，随着显著物体检测领域的迅速发展，逐渐形成了一个新的研究分支——协同显著物体检测。与仅针对单张图像的显著物体检测不同，协同显著物体检测旨在从一

组包含有两张或多张的图像中检测出共同显著的前景区域。对一组图像而言，可以大致地将其中包含的内容划分为四个类别^[33]，即：共同的前景、共同的背景、非共同的前景、非共同的背景，其中共同的前景，可能是具有相似外观的同一类别物体、或仅仅是不同视角下的同一物体。协同显著物体检测的任务就是从这组图像中寻找出共同的前景，即共同的显著区域。

图 1.3 是对两幅图像进行协同显著物体检测的示例，其中第 1 行是来自于 Image Pair 数据集^[34] 的原始图像，第 2 行是对应的人工标注图，不同图像对之间使用绿色加粗的竖线隔开。从第二对图像可以明确看到，协同显著物体检测除了要得到单张图像中的显著物体外，还应是不同图像间具有相似外观的同一类别物体。与显著物体检测相同，协同显著物体检测也是一个图形 – 背景分割问题，只是要求分割得到的是不同图像间的共同前景。因此，协同显著物体检测与图像协同分割^[35,36] 之间也存在着紧密联系，区别在于除了前景，图像协同分割还需要将背景划分为一组感知上相似的子块区域。此外，从图 1.3 的人工标注图也可以看出，与显著物体检测相同，协同显著物体检测的理想输出也应是二值的显著图。



图 1.3 协同显著物体检测

(3) 人眼视点预测 (Eye Fixation Prediction)

早期的显著性检测研究工作^[7,37]，受人类视觉注意机制^[6,38,39] 的启发，主要聚焦于人眼视点的预测问题。由于人在观察一幅图像时通常会随意地扫视整个场景，并首先短暂停留在最能引起视觉注意的区域，而后快速转移到下一个引起视觉注意的区域，因此人眼视点预测的主要任务就是预测出准确的视点位置并给出视点之间的转移路径。虽然输出结果同样需要凸显出能够引起视觉注意的区域，但与显著物体检测不同，人眼视点预测并不需要得到精确的物体区域或进行显著物体分割。在人机交互和图形学领域，人眼视点预测有着广泛的应用^[40–42]，也有研究人员通过分析视点指示区域，来辅助实现目标跟踪^[43] 或动作识别^[44]。

人眼视点预测的输出结果是一组视点集，因此对不同方法的性能评估需要借助于眼动仪实现。通过与眼动仪真实采集到的视点数据相比较，判断不同方法是否能够准确预测出真实的人眼视点位置。图 1.4 是来自于 ImgSal 数据集^[45,46] 的人眼视点预测示例，其中第 1 行是原始图像，第 2 行是使用 Tobii T60 眼动仪^① 采集到的 21 名志愿者

^① <https://www.tobiipro.com/product-listing/tobii-t60-and-t120/>。

的视点图 (Fixation Map)。视点图记录了志愿者观察图像时的真实视点位置，视点的像素值为 1、背景的像素值为 0。由于同一视点位置可能叠加有多名志愿者的视点，在绘制视点图时对未重叠的视点半径大小设置为 2 个像素，其它视点的半径大小设置为叠加次数的 3 倍，即图中半径越大、颜色越深的点，表明有越多志愿者的视点落在该位置。第 3 行是对视点图使用标准差为 10、大小为 40 的二维高斯低通滤波器平滑得到的密度图 (Density Map)，这也是人眼视点预测方法理想的输出结果，第 4 行是将密度图叠加到原图上生成的热图。从第 3 行可以看到，人眼视点预测的输出结果并不需要精确地得到显著物体区域。



图 1.4 人眼视点预测

(4) 候选目标生成 (Object Proposal Generation)

传统的目标检测方法采用滑动窗口方式^[47,48]，通过设计有效的分类器来检测每个窗口中是否有目标存在，考虑到目标宽高比变化，一幅图像的子窗数量通常会达到 $10^6 - 10^7$ 个。由于更好的检测性能往往意味着更为复杂的检测算法，过多的子窗数量，使得整个检测过程的算法复杂度控制面临着巨大的挑战。一种有效解决这一问题的处理方式^[49-51] 是不进行窗口滑动，而是假设目标之间存在某些有别于背景的视觉特征，通过构造或训练候选目标生成方法，得到一组可能包含任意类别目标的子窗口，并在这些子窗中检测是否有特定目标的存在。好的候选目标生成方法，可以在较高的检测召回率下，将子窗数量减少到几百至几千个，这可以大大降低计算时间的消耗。

生成候选目标区域的过程也被称为似物性采样 (Objectness Proposal)，与显著物体检测或人眼视点预测不同的是，似物性采样的主要任务并不揭示图像中的显著区域，而是要给出完整覆盖图像中每个目标的一组外接矩形^[52]。图 1.5 是候选目标生成示例，其中第 1 行是来自于 PASCAL VOC 2007 数据集^[53] 的原始图像。第 2 行是文献 [49] 的

检测结果，结果中给出了 1000 个可能包含有目标的采样窗口，这里仅取似物性得分排名前 10 的窗口以红色矩形框标注，青色矩形框里包含的是人工标注的真值目标。如果将 1000 个采样窗口依次与每个人工标注窗口计算面积的交除并，并以黄色矩形框标注比值最大的采样窗口，不难看出采样结果较好地覆盖了图像中的目标区域。如果将这 1000 个采样窗口内部的像素值设为 1，全部叠加后再执行归一化操作，得到的热图如第 3 行所示。可以看到，虽然候选目标生成方法并不检测显著性，但产生的采样窗口在一定程度上指明了图像中的显著区域，两者之间高度的关联性，使得可以利用显著性线索来帮助似物性方法获得更好的采样性能^[49,54]。

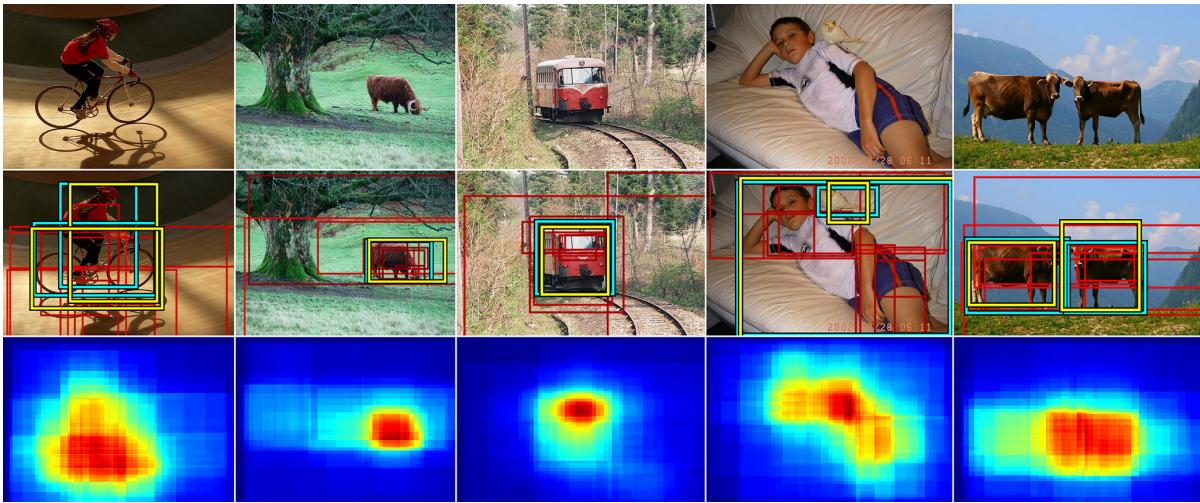


图 1.5 候选目标生成

1.2.2 显著物体检测的研究现状

由于本文主要研究静态场景下的显著物体检测和协同显著物体检测的计算方法与应用，下面分别从这两个方向介绍相关的研究工作。

1.2.2.1 显著物体检测

纵观显著物体检测的发展历程，迄今为止大致经历了三个阶段^[8]：

- 第一阶段。里程碑式的工作是 1998 年由 L. Itti 等人^[7]提出的视觉显著性计算方法，一般称为 IT 方法。这一工作实际上并非针对显著物体检测问题，而是围绕人眼视点预测问题开展了研究，除了生成显著图外，还讨论了视点之间的转移机制。IT 方法之所以被认为是里程碑式的工作，是因为之前针对视觉显著性主要是从认知心理学^[3,4] 和神经生物学^[5,6] 的角度研究认知上的视觉注意问题。而 IT 方法将该问题表达为一个可编程的计算模型，通过线性滤波器提取底层的颜色、亮度和方向特征后，采用竞争机制得到图像局部最显著的位置，并对所有特征图进行线性融合生成显著图，最后利用显著图解释和模拟视觉认知上的搜索策略。

- 第二阶段。此阶段里程碑的工作是 2007 年刘铁等人的研究^[9]，以及 2009 年由

R. Achanta 等人提出的 FT 方法^[32]。在 IT 方法之后，视觉显著性方法主要都围绕人眼视点预测问题，显著区域检测方法^[37,55]也是在视点的基础上寻找图像中能够引起视觉注意的大致区域。这些方法都没有回归到显著物体检测的本质，即显著物体检测是一个图形–背景分割问题。刘铁等人^[9]的贡献在于：一是将视觉显著性从人眼视点预测引导到显著物体检测上来，提出了显著物体检测是一个将显著物体从图像背景中分离出来的二值标注问题；二是公开了一个含有 20,840 张图像的数据集，并对每张图像中的显著物体使用外接矩形予以标注，这就为这一课题的发展提供了实验基础。FT 方法^[32]同样把显著物体检测看作是一个二值分割问题，其贡献在于将矩形标注改为更为精确的基于物体轮廓的标注，并提出了基于固定阈值和自适应阈值的显著图分割方法。在此之后，大量的显著性方法开始把研究的目标转到对显著物体的检测上^[29,56–60]，同时对实验数据集的建立^[34,45,46,60,61]以及对方法性能的评估也转为采用更为精确的显著物体区域二值分割图。

- 第三阶段。在此阶段之前，显著物体检测方法的性能优劣主要依赖于手工提取的图像特征以及组合方式。随着卷积神经网络 (Convolutional Neural Network，简记为 CNN)^[62] 的复兴，基于 CNN 的显著物体检测方法登上历史舞台。在早期的研究工作中，何盛烽等人^[63]认为基于手工提取特征的计算模型只能抓住底层的对比信息，因此提出了一种基于超像素对的卷积神经网络，通过将显著物体检测问题形式化为一个二值标注问题来获取分层的对比特征。由于基于超像素的方法获取到的是局部上下文信息，而 CNN 的一大优势是可以建模图像的全局上下文，因此赵瑞等人^[64]就考虑同时对全局和局部上下文进行优化，提出了多上下文深度模型。侯淇彬等人^[65]则在 HED 网络结构^[66]的基础上，提出了端到端的显著物体检测网络，通过引入短连接，显式地让高层语义信息帮助低层更好地凸显整个显著物体并获得精确的物体轮廓。张静等人^[67]利用全卷积神经网络，融合语义线索和高阶区域统计进行显著物体检测。罗志明等人^[68]则融合局部和全局特征，引入边界损失项训练端到端网络。此外，张平平等^[69]人提出了通用聚合多级卷积特征框架，该框架可以集成多级特征图到多个分辨率中，自适应地学习特征图的融合，并利用组合特征预测显著图。事实上，由于显著物体检测是一个将图像分为显著的前景图形和非显著的图像背景两个部分的二分类问题，因而用于分类的学习模型可以很容易地扩展到本课题中。随着深度卷积神经网络的迅猛发展，基于该技术的显著物体检测方法将逐渐成为研究的主流。

认知心理学^[4]和神经生物学^[6]通常将视觉注意机制分为两个阶段：1) 快速的、自底向上的、数据驱动的；2) 慢速的、自顶向下的、任务驱动的。自顶向下的方法与所面向的具体任务有关，并不属于一般意义上的显著性检测问题。自底向上的显著性检测方法主要关注由图像底层特征刺激带来的视觉注意，因其不限定特定的检测目标，而受到更为广泛的关注。如果把输入图像看作是由一组子图像所构成，那么依据所利用的子图像的不同，可以将显著物体检测方法分为两大类^[8]：基于图像块 (Block) 的检测方法以及基于区域 (Region) 的检测方法。

(1) 基于图像块的检测方法

图像块是指矩形的图像区域，如果把像素也看作是 1×1 的图像块的话，那么基于像素的检测方法也归为此类。文献 [9] 首先对输入图像构造六层的高斯金字塔，并将任意一层中每个像素的显著值定义为该像素与其 9×9 的中心子块之间的差异，通过多尺度的差异对比，来凸显显著物体的轮廓像素并抑制物体内部的显著值。FT 方法^[32] 也利用差异对比来获得显著性，将每个像素的显著值定义为该像素的特征与图像的平均特征之间的差异，使得全局稀缺特征获得更高的显著值。为了抑制高频信号，FT 方法对输入图像使用一个 5×5 的二维高斯核进行平滑，并在 LAB 颜色空间中度量特征差异，图像的平均特征采用高斯平滑结果的 LAB 均值向量来表示。程明明等人提出的 HC 方法^[29] 同样利用 LAB 颜色的全局稀缺性定义像素的显著值，为了降低像素间颜色对比的计算复杂度，HC 方法通过在 RGB 空间中对颜色均匀量化并构造颜色直方图进行计算加速。

由于显著的像素通常在空间上聚集在一起，因此许多方法利用图像块之间的对比来获取显著性。例如，文献 [70] 提出了一种上下文感知显著性检测方法，在利用局部对比机制的同时也考虑了对全局重复模式的抑制。该方法首先通过度量图像块之间的 LAB 颜色距离以及空间位置距离来获得图像块之间的差异，并通过前 64 个相似图像块计算像素显著值，然后通过多尺度的线性平均来抑制背景的显著值，以进一步加大与显著区域的对比度，最后通过上下文关系提升显著物体附近的显著值。文献 [71] 在检测显著性的同时考虑颜色和模式的独特性，在模式独特性度量时同样采用图像块进行操作，由于显著的图像块在高维空间上的分布相比非显著的图像块要更为分散，因此利用每个图像块到平均图像块的距离来定义其独特性，该距离是通过计算图像块到图像主成分的投影距离之和获得。然而，由于基于图像块的检测方法通常会在高对比度的边缘处产生较高的显著值，同时较大的图像块可能导致物体轮廓的模糊，因此更多的方法选择基于区域来检测图像中的显著物体。

(2) 基于区域的检测方法

区域，是指对输入图像分割得到的感知上同质的超像素。与图像块相比，由于不需要考虑块与块之间的重叠，使得较少的超像素可以带来更高的计算效率。同时相较于像素，区域级的视觉特征更容易引起人的视觉注意，因而此类方法往往具有更好的检测性能。在计算显著性时，通常会利用超像素之间的全局或局部对比机制得到每个超像素的显著值，进而获得显著图。例如，RC 方法^[29] 通过累计每个超像素与图像中其它所有超像素之间的颜色差异来计算显著性，同时考虑超像素的面积大小以及超像素之间的距离对显著性的影响，并采用颜色直方图降低大量比较所带来的计算复杂度。但文献 [61] 认为，基于全局对比的方法可能将高显著性错误地引导到小尺寸的高对比区域，因此提出了基于分层结构的显著性检测模型，在下两层中通过颜色、大小和距离进行局部度量，只在最顶层中通过参数控制产生接近于全局对比的结果，最后通过最小化一个层间不同区域一致性相关的能量函数推断出显著性。

B. Tatler^[72] 和 T. Judd^[73] 等人的研究结果表明，相较于图像的外围区域，人们会更趋向于关注图像的中心，这意味着显著物体更有可能位于图像的中心区域，或者说图像的边框附近更有可能是背景区域。因此，一些方法利用背景先验计算显著性。例如，文献 [74] 将 LAB/RGB 颜色以及空间位置作为超像素的特征向量，同时利用稠密和稀疏的重构误差来计算基于特征向量表示的每个超像素的显著值，重构误差则利用由图像边界超像素构造的边界模板计算得到。然而，由于基于背景先验的方法通常将图像边框作为背景区域，当显著物体是边界对象时会面临方法的失效。因此基于边界连通性的方法^[58] 将区域的贴边程度度量为区域落在图像边界上的像素个数与自身面积的比值，同时考虑到直接分割区域会比较困难，改为利用超像素之间的测地距离计算连通性，并利用对比模型计算出背景概率，最后通过优化一个同时考虑背景概率、前景概率和相邻平滑性的二次目标函数计算得到每个超像素的显著值。

除了显式地利用背景先验外，基于环绕性线索的检测方法^[75] 认为边界区域更有可能是背景，因此在布尔图中剔除掉与图像边框相连接的区域，利用注意图的线性融合计算得到显著图。虽然这种处理方式更多考虑了显著物体作为前景图形应具有环绕闭合的外部轮廓，但在某种程度上，其去除边界对象的操作也隐式地利用到了背景先验。综上，由于超像素是颜色、亮度等特征高度同质的像素在局部空间上的集合，并且生成超像素的分割方法可以很好地保持物体轮廓，与基于图像块的检测方法相比，基于区域的检测方法在提高计算效率、改善整体检测性能等方面有着明显的优势。

1.2.2.2 协同显著物体检测

显著物体检测方法更多关注的如何挖掘输入图像的内在特征，来有效表征图像中的显著区域并定义其显著性。但也有学者认为，显著性线索不仅与输入图像自身有关，也可以来自于其它的相似图像。作为显著物体检测衍生出的一个研究分支，协同显著物体检测的目的是在不同图像间寻找共性的显著性线索以获得共同的显著前景区域，并受到越来越多研究人员的关注^[19,34,76–79]。协同显著性的概念，最早是 2010 年由 D. Jacobs 等人^[80] 在研究针对不同图像的视点预测问题时所提出，随后与显著物体检测发展历程大致相同，在经历过手工提取特征的阶段后，逐渐进入到机器学习和深度学习阶段。由于协同显著图中也同时反映了单张图像中的显著区域或部分显著区域，协同显著性方法通常会借鉴显著性方法的思想，使两者具有许多相似之处。依据检测机制的不同，现有方法主要分为三类^[33]：

(1) 自底向上的检测方法：自底向上的协同显著物体检测方法通常对图像块或超像素手工提取特征，在利用一种或多种特征从图像中抽取出显著性信息后，基于协同显著性线索将这些信息加以融合，最终得到协同显著物体检测结果，即协同显著图。上述步骤的关键在于如何发现和有效利用协同显著性线索，一种常见的手段是利用显著性特征的重复性^[81]。例如，李宏亮等人^[34] 提出了一种在单张图像中寻找显著区域，在图像间度量相似性，然后将两者加权融合的协同显著物体检测模型，其中单图显著图是通过

加权融合 IT^[7]、FT^[32]、SR^[37] 三个方法的显著图获得，协同显著图则是利用协同多层图度量超像素之间的颜色和纹理特征的相似性获得。文献 [79] 提出了一种基于分层分割的检测方法，该方法在超度量轮廓图的基础上，利用阈值控制轮廓图中保留的区域数分别得到精分割和粗分割的结果，并在精分割的基础上利用颜色直方图度量区域相似性得到图内显著性，在粗分割的基础上通过度量区域与图像边界的连接度得到目标先验图，最后与全局相似性结合得到协同显著图。自底向上的检测方法的优点在于其简单高效、同时也更为直观，但此类方法依赖于手工提取的视觉特征的可靠性以及对协同显著性的显式定义，在应对新的检测场景时缺乏一定的泛化能力，同时这种显式定义能否表征人类对于协同显著区域的视觉注意机制也有待商榷。

(2) 基于融合的检测方法：此类方法的基本思想是利用已有的显著性或协同显著性方法，通过协同显著性线索计算自适应的融合权重，最后对这些方法的检测结果图加权融合获得协同显著图。例如，文献 [78] 提出了一种基于秩约束的自适应权重求解机制，利用现有方法得到粗糙的显著物体分割，再用颜色直方图表示每个显著区域的特征，得到一个大小由图像数、不同方法对单张图像产生的子图数、特征向量维度所共同确定的特征矩阵，由于协同显著区域的相似性和一致性导致的低秩，通过低秩矩阵恢复得到误差矩阵，最后计算出自适应的融合权重。基于融合的检测方法的优点在于可以在现有显著性/协同显著性方法的基础上，利用它们较好的性能优势进一步提升检测结果，同时这种机制也很容易推广到对其它方法的融合。然而，由于此类方法在很大程度上依赖于现有方法的检测结果，当参与融合的方法提供的是错误的或不一致的显著区域时，基于融合的方法同样会产生错误的融合结果。

(3) 基于学习的检测方法：与手工提取特征的方法不同，基于学习的检测方法试图从不同图像中自动挖掘出有效的协同显著模式。例如，文献 [18] 基于简单的草图，通过为目标区域和背景构建的外观模型来帮助改进显著性检测和图像分割，实现组显著性 (Group Saliency) 检测、兴趣目标分割和图像检索等应用。文献 [82] 则提出了一个自步多示例学习 (Self-paced Multiple-instance Learning) 模型，该模型将待检测图像组中的图像作为正包，从其它图像组检索出的相似图像作为反包，每幅图像中的超像素作为待分类的示例。在预测出每个示例的协同显著值后，通过相邻超像素之间的空间关系计算协同显著图，超像素的特征表示是通过 ImageNet 上的 CNN 预训练网络获得。除了用于提取区域特征外，深度学习也被直接用于解决协同显著物体检测问题，其中开创性的工作是张鼎文等人^[83] 提出的同时利用深度和广度信息的深度学习模型。在深度方面，该模型利用带迁移层的 CNN 获得协同显著物体的深度信息表示，在广度方面，则利用跨组信息对背景建模以抑制组内共同的背景区域，最后通过形式化为贝叶斯问题并利用前景区域的一致性度量来获得协同显著图。虽然有着明显的性能优势，但深度学习方法主要的缺点在于训练过程中的时间消耗，特别是当模型长时间得不到收敛时，需要在权重初始化和学习率等方面做出诸多调整。

1.2.3 显著物体检测数据集的建立

对不同显著物体检测方法的性能评估，离不开实验数据集的参与。如图 1.2 所示，数据集中除了应包含用于检测的原始图像外，也应包含二值的人工标注图 (Ground Truth)。本节将对目前被广泛使用的 ASD 数据集^[9,32] 的建立过程做一简介，在后续的章节中，将仅介绍所用数据集的统计特性，不再赘述相关的建立过程。

ASD 数据集^① 也被称为 MSRA1000 或 MSRA1K 数据集，其源自于 MSRA 数据集^{[9] ②}。MSRA 数据集由刘铁等人建立，为了研究单张彩色图像中的显著物体检测问题，他们从互联网上收集了 130,099 张高质量图像，从中人工筛选出 60,000 余张仅含有一个显著物体或前景目标的图像作为候选集。随后从候选集中进一步剔除掉那些包含大尺寸显著物体的图像，最终保留了 20,840 张图像构成 MSRA 的第一个子集，称为 Image Set A，并进行了两阶段的人工标注。

在第一个阶段，通过 3 名志愿者对 Image Set A 进行标注。对于每幅图像，每个志愿者将其认为的显著物体区域框选一个外接矩形。如果将矩形内每个坐标 (x, y) 处的像素值 $a_{x,y}$ 设为 1，矩形外的其它像素值设为 0，就得到了该名志愿者的二值标注图。接着对每幅图像的 3 个二值标注图进行平均，就得到了该幅图像的显著性概率图，概率图中坐标 (x, y) 处的概率值 $g_{x,y}$ 为

$$g_{x,y} = \frac{1}{M} \sum_{m=1}^M a_{x,y}^m, \quad (1.1)$$

其中， M 表示志愿者人数 (这里 $M = 3$)， m 表示志愿者的编号。为了度量 3 名志愿者的标注一致性，利用一个阈值 t 来统计达成一致性标注意见的像素比例 C_t ，计算如下

$$C_t = \frac{\sum_x \sum_y g_{x,y}}{\sum_x \sum_y \sum_{(x,y) \in \{(x,y) | g_{x,y} > t\}} g_{x,y}}, \quad (1.2)$$

其中，阈值 t 表示志愿者比例。例如，假设 $C_{0.9} = 0.975$ 即 $t = 0.9$ ，就表示有超过 90% 的志愿者在 97.5% 的标注像素上达成了一致的标注意见，由于第一阶段仅有 3 名志愿者，这意味着所有的志愿者产生了近乎一致的标注结果。图 1.6 示例了对 Image Set A 的标注过程，第 1 行是原始图像，第 2 行是 3 名志愿者的标注结果，第 3 行是显著性概率图。可以看到，3 名志愿者对于图 1.6(a) – 1.6(c) 中的显著物体区域产生了较大分歧，而在后两张图上达成了非常统一的意见。表 1.1 也表明，对于像图 1.6(d) 和 1.6(e) 这样达成高度一致性标注意见的图像， $C_{0.9}$ 将接近于 1。

在第二个阶段，从 Image Set A 中随机选取 5,000 张 $C_{0.9} > 0.8$ 的图像，构成 MSRA 的第二个子集，称为 Image Set B，并通过 9 名志愿者进行标注。图 1.7(a) 和 1.7(c) 给出了对图 1.6(d) 和 1.6(e) 进行第二阶段标注的结果，由于在这一阶段志愿者之间产生的分

^① https://ivrl.epfl.ch/supplementary_material/RK_CVPR09/。

^② http://research.microsoft.com/en-us/um/people/jiansun/SalientObject/salient_object.htm。

歧较少，因此最终对区域级的显著物体检测性能度量就直接利用如图 1.7(b) 和 1.7(d) 所示的概率图实现。

但是，这种基于外接矩形的标注结果显然远不够精确。由于显著物体检测的最终目的是为了分割出图像中的显著物体，因此当矩形内部含有大面积的背景区域或者同时包含多个物体时，将不能为检测方法性能提供可靠的评估。基于此，文献 [84] 将基于外接矩形的标注方式改为基于目标轮廓的标注方式。受此标注方式的启发，R. Achanta 等人^[32] 从 Image Set B 中选取了 1,000 张图像，标注出精确的显著物体区域二值分割图（见图 1.2 第 2 行），构成了 ASD 数据集。

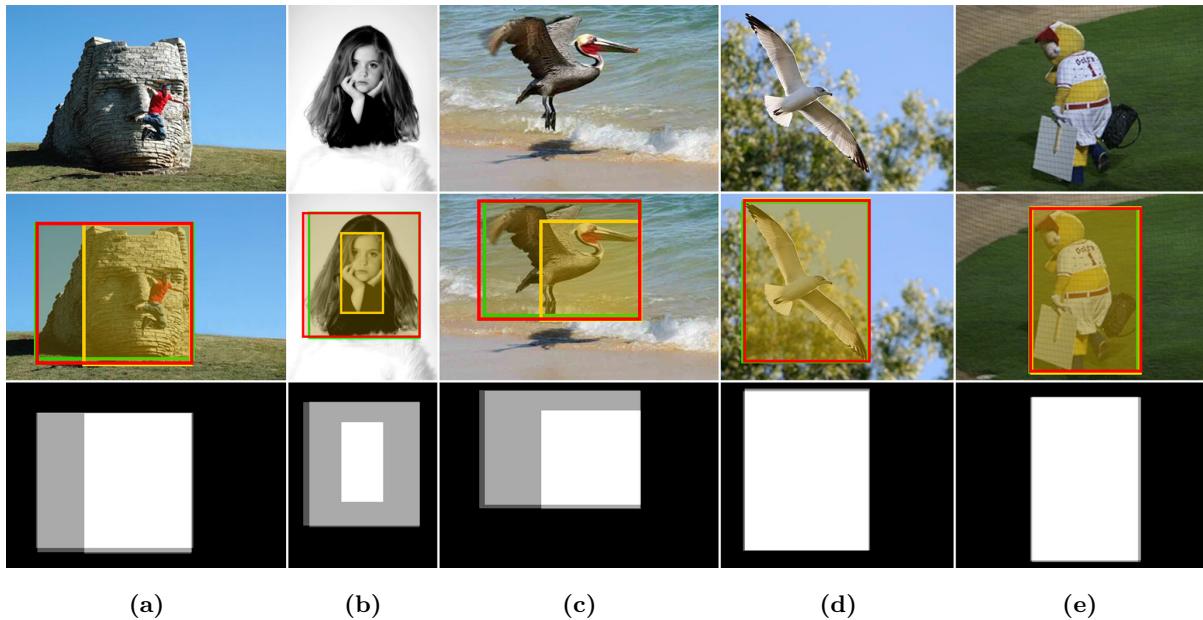


图 1.6 Image Set A 标注

表 1.1 Image Set A 标注一致性

C_t	图 1.6(a)	图 1.6(b)	图 1.6(c)	图 1.6(d)	图 1.6(e)
$C_{0.9}$.7468	.3170	.6009	.9784	.9649

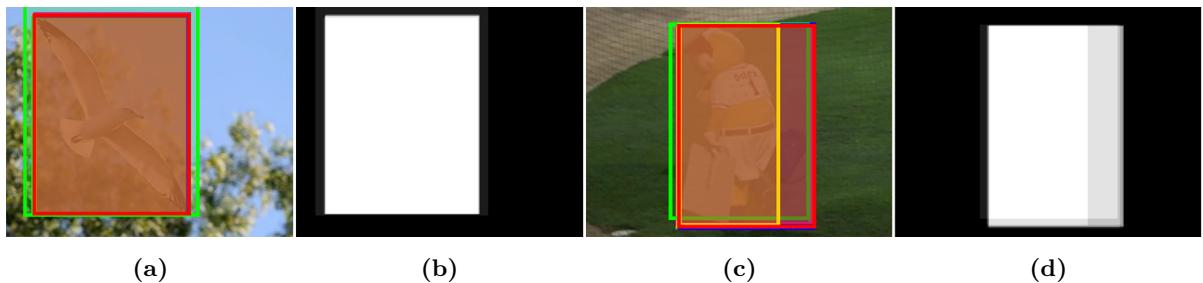


图 1.7 Image Set B 标注

1.2.4 显著物体检测的性能度量

1.2.4.1 显著图

显著物体检测方法的输出结果通常是与输入图像具有相同大小的“显著图”(Saliency Map)，图中每一个像素值表示该点属于显著物体的概率，并以取值范围为 $[0, 1]$ 区间的概率值表示，即显著图也是概率图。为了便于存取和显示，一般将概率图归一化到更为常见的8比特图像，以整数取值范围 $[0, 255]$ 表示为一幅灰度图(也可称为亮度图或强度图)。图1.8给出了采用第2章提出的RPC方法进行显著物体检测的结果，第1行是来自于ASD数据集^[9,32]的8幅图像，检测得到的显著图如第3行所示。可以看到，显著图中像素越亮表示该像素的显著值越大，即该像素越有可能属于图中的显著物体。一个好的显著物体检测方法应在凸显显著物体的同时，抑制非显著的图像背景，并尽可能得到与人工标注图(第2行)相同的结果。本文的第4章提出了一种协同显著物体检测方法，协同显著物体检测方法的输出结果被称为“协同显著图”(Co-saliency Map)，虽然协同显著图凸显的是多幅图像的共同前景物体，但本质上仍然属于显著图。



图 1.8 显著图

1.2.4.2 常用的性能度量指标

对不同显著物体检测方法进行性能评估时，需要有定量评价检测能力的衡量标准，即性能度量指标。通常在横向对比不同方法的检测能力时，会使用到以下几种性能指标^[28,85,86]，这些指标同样也适用于对协同显著物体检测方法的性能评估。

(1) 精确率 – 召回率 (Precision – Recall, 简记为 $P - R$)

一种直观比较不同方法性能的方式是通过绘制“精确率 – 召回率曲线图”，这里首先给出精确率和召回率的定义：

- 精确率 (Precision, 简记为 P)

“精确率”也被称为准确率或查准率，该指标反映了检测出的显著物体中有多少比例是属于人工标注的显著物体区域。对于一幅显著图 S 而言，假设 M 是对 S 二值分割的结果， G 是人工标注图，由于 M 和 G 都是二值图像，精确率可通过如下方式计算

$$P = \frac{|M \cap G|}{|M|}, \quad (1.3)$$

其中，符号 $|\cdot|$ 表示计算前景区域的面积，即统计像素值为 1 的像素个数。

- 召回率 (Recall, 简记为 R)

“召回率”也被称为查全率，该指标反映了人工标注的显著物体区域中有多少比例被检测了出来。同样，对于一幅显著图 S 而言，召回率可通过二值分割结果 M 和人工标注图 G ，采用如下方式计算

$$R = \frac{|M \cap G|}{|G|}, \quad (1.4)$$

其中， $|\cdot|$ 同样表示统计像素值为 1 的像素个数。

依据式(1.3)和(1.4)的定义，可以看到对显著图 S 分割得到 M 的过程是性能评估的关键。由于 S 是一幅灰度图，因此通常采用两种分割方式：1) 使用固定阈值 $T_f \in [0, 255]$ 做 256 次分割；2) 使用自适应阈值 T_a 做 1 次分割。下一节将详细介绍这两种分割过程。

(2) F_β 度量 (F_β -measure 或 F_β -score，简记为 F_β)

精确率 P 和召回率 R 是一对相互矛盾的性能度量指标。通常，当精确率高的时候，召回率会偏低；而召回率高的时候，精确率又会偏低。为了综合考察精确率和召回率，一般使用的是基于精确率和召回率调和平均的“ $F1$ 度量”

$$\frac{1}{F1} = \frac{1}{2} \cdot \left(\frac{1}{P} + \frac{1}{R} \right), \quad (1.5)$$

由上式得

$$F1 = \frac{2 \times P \times R}{P + R}. \quad (1.6)$$

在评估显著物体检测方法的性能时，更为常见的是使用 $F1$ 度量的一般形式，即基于精确率和召回率加权调和平均的“ F_β 度量”

$$\frac{1}{F_\beta} = \frac{1}{1 + \beta^2} \cdot \left(\frac{1}{P} + \frac{\beta^2}{R} \right), \quad (1.7)$$

由上式得

$$F_\beta = \frac{(1 + \beta^2) \times P \times R}{\beta^2 \times P + R}, \quad (1.8)$$

其中 $\beta > 0$ 为加权调和系数，用于平衡精确率 P 和召回率 R 两者的相对重要性^[87]。当 $\beta = 1$ 时， F_β 的计算与 $F1$ 相同；当 $\beta < 1$ 时，精确率的大小对 F_β 指标有更大的影响，意味着在性能评估时更为看重精确率；而当 $\beta > 1$ 时，则召回率的大小对 F_β 指标有更大的影响，意味着相对精确率而言，在性能评估时更为看重召回率。

至于加权调和系数 β 的具体大小，不同文献中有不同的建议。例如，文献 [9, 88] 建议将 β^2 设置为 0.5，而 R. Achanta 等人^[32] 则建议将 β^2 设置为 0.3，即 $\beta = \sqrt{0.3}$ 。不

难看出，虽然对 β 的建议取值有所不同，但这些文献都将其设置为 < 1 ，以给精确率更大的权重倾斜。这表明在显著物体检测任务中，召回率并不如精确率那么重要。正如刘铁等人^[89]所提出的，相对于精确率，召回率可以很容易获得一个较高的值。比如，假设某个显著物体检测方法不做任何算法处理，而是直接将整幅图像都判为显著物体区域，那么由式(1.4)计算得到的召回率就是 100%。但显然，此时的精确率将处于一个较低的水平，从而整体上拉低了 F_β 指标，使得该方法的实际检测性能并不占优。

在本文后续的各章实验中，除了第 5 章对区域稳定性方法以及融合方法的性能评估采用 F_1 指标外，其它对显著性/协同显著性方法的性能评估将统一采用 F_β 指标。至于加权调和系数 β ，则与目前绝大部分工作一样，遵循文献^[32]的建议将 β^2 设置为 0.3。

(3) 受试者工作特征 (Receiver Operating Characteristic, 简记为 ROC)

ROC 曲线^[90]的 x 轴表示“假正例率”(False Positive Rate, 简记为 FPR)， y 轴表示“真正例率”(True Positive Rate, 简记为 TPR)。对于显著物体检测，基于式(1.3)中的符号 M 和 G ，FPR 和 TPR 的定义分别为

$$\text{FPR} = \frac{|M \cap \bar{G}|}{|\bar{G}|}, \quad (1.9)$$

$$\text{TPR} = \frac{|M \cap G|}{|G|}, \quad (1.10)$$

其中， \bar{G} 表示人工标注图 G 的补图像，即对 G 取反的结果。

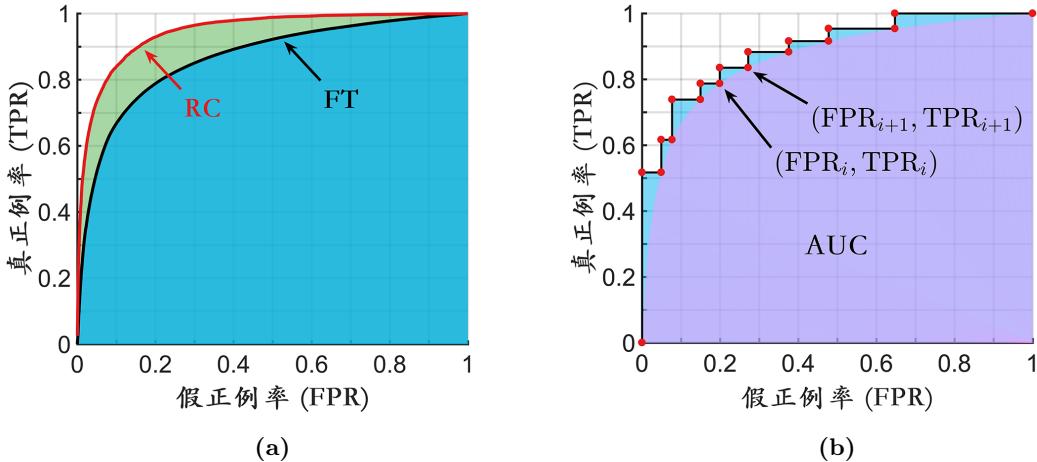


图 1.9 ROC 曲线图与 AUC。(a) ROC 曲线图与 AUC；(b) 基于 10 个采样点绘制的 ROC 曲线图与 AUC

ROC 曲线的绘制过程如下：首先对显著图 S 采用固定阈值 $T_f \in [0, 255]$ 做 256 次分割，得到 256 个二值结果 M ，然后将通过式(1.9)和(1.10)计算得到的 256 个坐标点 $\{(\text{FPR}_0, \text{TPR}_0), (\text{FPR}_1, \text{TPR}_1), \dots, (\text{FPR}_{255}, \text{TPR}_{255})\}$ 连接而成。图 1.9(a)示例了两种显著物体检测方法对 ASD 数据集^[9,32]检测得到的 ROC 曲线，其中红色曲线为 RC 方

法^[29]、黑色曲线为 FT 方法^[32]。可以看到，由于 RC 方法的 ROC 曲线将 FT 方法的曲线完全包住，可以断言 RC 方法的性能优于 FT 方法。但是，有时不同显著物体检测方法的 ROC 曲线会发生交叉，这就很难直接断言孰优孰劣。如果一定要进行比较，可以通过 ROC 曲线下的面积进行判断。

(4) ROC 曲线下的面积 (Area Under ROC Curve, 简记为 AUC)

对 ROC 曲线下的面积，即 AUC 的计算过程如下：以图 1.9(b) 为例，图中紫色标示了 FT 方法^[32] 的 ROC 曲线下覆盖的区域，假设从 ROC 曲线上抽取出 10 个坐标点，那这些坐标点与 x 轴共构成了 9 个矩形，通过对这 9 个矩形的面积求和就可近似得到 AUC。由于 ROC 曲线是由 256 个离散的坐标点连接而成，AUC 可近似为

$$AUC = \sum_{i=0}^{254} (FPR_{i+1} - FPR_i) \cdot TPR_i, \quad (1.11)$$

或

$$AUC = \sum_{i=0}^{254} (FPR_{i+1} - FPR_i) \cdot TPR_{i+1}. \quad (1.12)$$

上述两式的计算结果，式(1.11)会导致 AUC 偏小，而式(1.12)则会使 AUC 相对偏大。因此，通常会对 AUC 采用更为精确一些的计算方式，即取上述两式的平均值

$$AUC = \frac{1}{2} \sum_{i=0}^{254} (FPR_{i+1} - FPR_i)(TPR_i + TPR_{i+1}). \quad (1.13)$$

从图 1.9(a) 可以看到，由于 RC 方法 ROC 曲线下的面积（即 AUC）要大于 FT 方法，表明 RC 方法更优。

(5) 平均绝对误差 (Mean Absolute Error, 简记为 MAE)

上面几种性能度量指标的计算都需要先对显著图 S 执行分割操作，一种不需要分割操作的度量方式是直接计算显著图 S 与人工标注图 G 的平均绝对误差^[56]。由于 S 与 G 的大小相同，将 S 归一化到 $[0, 1]$ 区间后，MAE 的计算方式为

$$MAE = \frac{1}{W \times H} \sum_{x=1}^W \sum_{y=1}^H |S(x, y) - G(x, y)|, \quad (1.14)$$

其中，符号 $|\cdot|$ 表示绝对值运算， W 和 H 分别表示显著图 S 的宽度和高度。

1.2.4.3 本文使用的性能度量指标

使用不同度量指标进行性能评估时，同一方法的性能优劣通常不尽相同。在本文后续对显著物体检测/协同显著物体检测方法的参数分析和性能评估中，将使用最为广泛采用的“精确率–召回率”和“ F_β 度量”两种度量指标，计算这些指标时需要对显著图 S 执行两种阈值分割操作：

(1) 使用固定阈值 T_f 分割

对一幅像素值范围 $[0, 255]$ 的显著图 S 使用 $T_f \in [0, 255]$ 的固定阈值进行分割，可以得到 256 个二值分割结果 M ，进而利用人工标注图 G 和式(1.3)与(1.4)，可以计算出 256 组“精确率–召回率指标”。而对于包含有一组图像的数据集而言，其在阈值 i 下的精确率–召回率指标 (R_i, P_i) ，是该阈值下所有图像的精确率–召回率指标的平均值。在得到 256 组精确率–召回率 $\{(R_0, P_0), (R_1, P_1), \dots, (R_{255}, P_{255})\}$ 后，以召回率 R_i 为 x 坐标、精确率 P_i 为 y 坐标，将这些坐标点连接，可以绘制出“精确率–召回率曲线”，简称“ $P-R$ 曲线”，由 $P-R$ 曲线构成的图被称为“ $P-R$ 曲线图”或“ $P-R$ 图”。

图 1.10(a)示例了两种显著物体检测方法对 ASD 数据集^[9,32] 检测得到的 $P-R$ 曲线图，其中红色曲线为 RC 方法^[29]、蓝色曲线为 FT 方法^[32]，分析如下：

首先，两条曲线都汇聚于同一点，该点坐标为 $R = 1, P = 0.1984$ 。实际上，对于某个数据集而言，所有显著物体检测方法在该数据集上得到的 $P-R$ 曲线都将汇聚于同一点，该点坐标的 P 值表示显著物体面积在图像中的平均占比。也就是说，在 ASD 数据集中，人工标注的显著物体区域面积平均占到每幅图像面积的 19.84%。

其次，由于 RC 方法的 $P-R$ 曲线将 FT 方法的曲线完全包围，可以断言 RC 方法的性能优于 FT 方法。但有时不同方法的 $P-R$ 曲线会发生交叉，此外当检测方法在凸显显著物体的同时，最大限度地抑制了背景区域，且检测出的显著物体区域具有平滑一致的显著值时， $P-R$ 曲线会被压缩在一段很窄的区间内，这时就很难直接断言参与比较的方法孰优孰劣，而应通过 F_β 曲线图来进行判断。

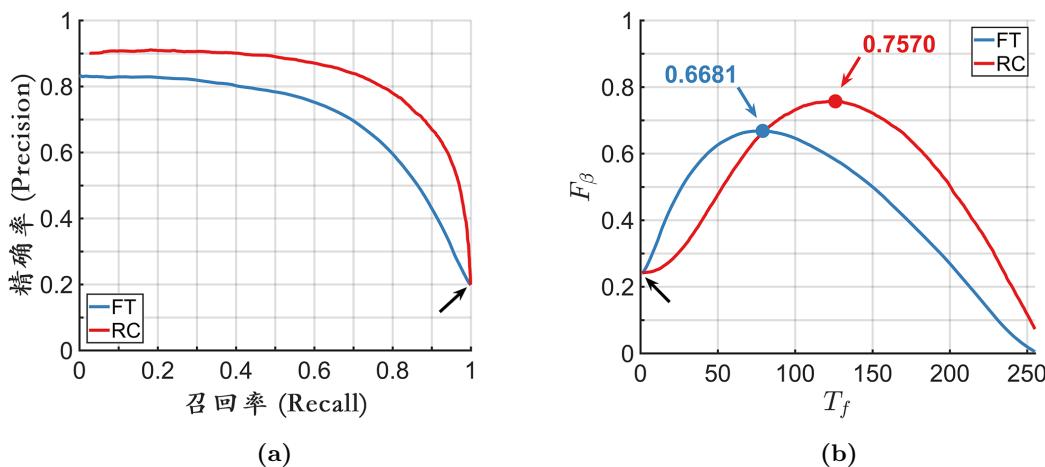


图 1.10 $P-R$ 曲线图与 F_β 曲线图。(a) $P-R$ 曲线图： x 轴表示召回率 R ， y 轴表示精确率 P ；(b) F_β 曲线图： x 轴表示固定分割阈值 T_f ， y 轴表示 F_β

对于某个显著物体检测方法，其在数据集上的 F_β 曲线绘制过程如下：对固定阈值区间 $[0, 255]$ 内的每个阈值 i ，由数据集中每幅图像的精确率和召回率，通过式(1.8)计算出该幅图像的 F_β 指标，进而整个数据集在阈值 i 的 F_β ，通过计算所有图像的 F_β 平均值得到。再利用 256 组“阈值– F_β 指标” $\{(0, F_{\beta,0}), (1, F_{\beta,1}), \dots, (255, F_{\beta,255})\}$ ，以阈

值 i 为 x 坐标、 $F_{\beta,i}$ 为 y 坐标，将坐标点连接就绘制出“ F_{β} 曲线”，由 F_{β} 曲线构成的图被称为“ F_{β} 曲线图”。

在 F_{β} 曲线图中，如果一个方法的 F_{β} 曲线能够把另一个方法的曲线完全包住，则可以断言前者的性能优于后者。但同样，当两种方法的 F_{β} 曲线发生交叉时，很难直接断言两者孰优孰劣。以 ASD 数据集的 F_{β} 曲线图 1.10(b) 为例，图中 RC 方法（红色）与 FT 方法（蓝色）的曲线发生了交叉，在这种情况下，为了定量地评估方法性能，本文采用由 F_{β} 曲线产生的两种指标进行判断：

- F_{β} 曲线的均值 (Average F_{β} , 简记为 AvgF)

由于 F_{β} 曲线是一条横跨阈值区间 $[0, 255]$ 的连续曲线，如果一个方法的 F_{β} 曲线能够把另一个方法的曲线完全包住，那通过计算 F_{β} 曲线下的面积同样可以断言前者的性能优于后者。另一种更为简单的替代评估手段就是直接计算 F_{β} 曲线的均值，本文称为“AvgF 指标”，计算如下

$$\text{AvgF} = \frac{1}{256} \sum_{i=0}^{255} F_{\beta,i}, \quad (1.15)$$

其中 i 表示 $[0, 255]$ 区间内的固定分割阈值。

图 1.11(a) 示例了计算 AvgF 指标的意义^①，可以看到，红色曲线完全包住了蓝色曲线，可以断言前者代表的方法优于后者。在具体指标值上，红色曲线的 $\text{AvgF} = 0.8915$ ，蓝色曲线的 $\text{AvgF} = 0.6674$ ，前者也具有比后者更大的 AvgF 值。类似的，图 1.10(b) 中 RC 方法的 $\text{AvgF} = 0.5192$ ，FT 方法的 $\text{AvgF} = 0.4342$ ，也可以认为 RC 方法的性能优于 FT 方法。但是，仅依赖于 AvgF 指标，是否一定可以断言 AvgF 较大的方法优于 AvgF 较小的方法？

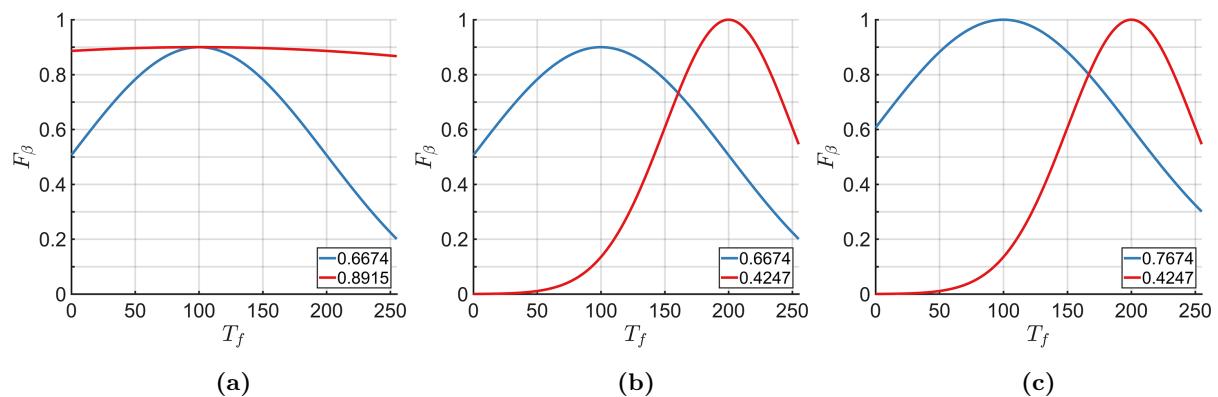


图 1.11 AvgF 指标

^①如图 1.10(b) 中黑色箭头所示，在同一个数据集上，当固定分割阈值 $T_f = 0$ 时，不同方法的 F_{β} 曲线会汇聚到同一点。但在第 4 章可以看到，如果一个检测方法得到的显著图接近于二值图像的话，确实会存在除 $T_f = 0$ 这一点外， F_{β} 曲线呈现出一条近似于水平线的现象。因此这里暂时不考虑 $T_f = 0$ 时的情况，具体到图 1.11 中， F_{β} 曲线并没有汇聚到同一点上。

- F_β 曲线的峰值 (Maximal F_β , 简记为 MaxF)

再次观察图 1.11(a), 可以发现两条 F_β 曲线具有相同的峰值 $F_\beta = 0.9$, 现在考虑不同峰值的情况。图 1.11(b) 给出了这样的示例, 图中红色曲线的 AvgF = 0.4247, 蓝色曲线的 AvgF = 0.6674, 如果仅依赖 AvgF 指标进行评估, 应认为蓝色方法的性能优于红色方法, 但事实并非如此。图 1.11(b) 中, 蓝色曲线峰值 $F_\beta = 0.9$ 、峰值点 x 坐标为 100, 红色曲线峰值 $F_\beta = 1$ 、峰值点 x 坐标为 200, 后者比前者具有更大的 F_β 峰值。由于 F_β 曲线通常是单峰的, 因此曲线峰值代表了其最优性能, 同时峰值点对应的 x 坐标为获得最优性能时的分割阈值, 即最优分割阈值。也就是说, 在 [0, 255] 的阈值区间内, 蓝色方法生成的显著图在阈值 100 上达到整体上的最优性能, 而红色方法的这一阈值为 200。如果从获得最优性能的角度来看, 应认为红色方法优于蓝色方法。

因此, 本文在进行方法性能比较时, 也使用了被普遍采用的 F_β 曲线峰值作为度量指标^[88], 简称“MaxF 指标”, 以评估方法的最优检测性能。对于图 1.10(b), 由于 RC 方法的 MaxF = 0.7570、FT 方法的 MaxF = 0.6681, 可以认为 RC 方法的性能优于 FT 方法。除了用于性能比较外, MaxF 指标在本文中也被用于调参过程, 以确定提出方法的最优参数取值。

本文之所以除 MaxF 指标外, 还引入了 AvgF 指标, 主要出于如下考虑:

首先, 虽然很少出现这种情况, 但当两种方法的 MaxF 指标值相同时, 显然 AvgF 指标值越大的方法性能越好。图 1.11(c) 给出了这样的示例, 两个方法具有相同的 MaxF 指标值 1, 但由于蓝色方法的 AvgF 指标值 0.7674, 要大于红色方法的 AvgF 指标值 0.4247, 可认为蓝色方法的性能要优于红色方法。

其次, 如果一种方法得到的显著图与人工标注结果完全一致, 那么对应的 F_β 曲线应是一条 y 坐标为 1 的水平线, 即 AvgF = 1。当两种方法的 MaxF 指标值较为接近时, 如果其中一种方法的 AvgF 指标值更大, 就意味着该方法的 F_β 曲线在更长的一段阈值区间内更为平坦; 同时更接近于 1 的 AvgF 值也意味着该方法得到的显著图更接近于人工标注结果。事实上, 以 MaxF 指标进行性能度量的前提是已提供了人工标注结果, 如果应用在不带人工标注的新数据集上, 调参数据集上得到的 MaxF 对应的最优分割阈值并不一定是新数据集的最优阈值。但如果一种方法的 F_β 曲线可以在很长一段阈值区间内保持相对平坦, 就意味着在该段区间内使用任何阈值都可以获得相似的分割结果, 这能够为后续的显著物体分割带来更大的便利。以表 4.6 为例, 虽然第 4 章提出的 HCN 方法的 MaxF 指标略低于 IPTDIM 方法^[19](相差 0.06%), HCN 方法的 AvgF 指标却要高出 IPTDIM 方法 24.59%。在两种方法的 MaxF 值较为接近的情况下, 本文认为具有明显 AvgF 指标优势的 HCN 方法具有更好的检测性能。

(2) 使用自适应阈值 T_a 分割

基于固定阈值 T_f 的繁琐之处是需要对显著图执行 256 次分割, 而另一种仅需要执行 1 次分割操作的性能度量方式是利用 R. Achanta 等人提出的自适应阈值 T_a ^[32], 即以显著图均值的两倍作为分割阈值

$$T_a = \frac{2}{W \times H} \sum_{x=1}^W \sum_{y=1}^H S(x, y), \quad (1.16)$$

其中 W 和 H 分别表示显著图 S 的宽度和高度。利用 T_a 对显著图 S 分割出二值图 M , 再通过式(1.3)、式(1.4)和式(1.8)计算出该图的精确率 P 、召回率 R 以及 F_β 指标, 整个数据集的 P 、 R 和 F_β 是通过分别计算所有图像的三项指标的均值得到。

- 自适应阈值的 F_β (Adaptive F_β , 简记为 AdpF)

基于自适应阈值 T_a 的性能度量方式是通过绘制“精确率–召回率– F_β 度量”柱状图, 简称“ $P-R-F_\beta$ 柱状图”, 通常以 F_β 值的大小做升序排列。图 1.12 示例了 FT 方法^[32] 和 RC 方法^[29] 在 ASD 数据集^[9,32] 上得到的 $P-R-F_\beta$ 柱状图, 其中蓝柱表示精确率 P , 红柱表示召回率 R , 绿柱表示 F_β 度量。对不同方法的定量评估是基于自适应阈值 T_a 得到的 F_β 指标, 本文称为“AdpF 指标”。具有最大 AdpF 指标值, 即在 $P-R-F_\beta$ 柱状图中排在最后的方法, 被认为是在该数据集上拥有最优检测性能的方法。图 1.12 中, 尽管 RC 方法的召回率不如 FT 方法, 但由于 RC 方法具有更大的 AdpF 值, 可认为 RC 方法的性能优于 FT 方法。

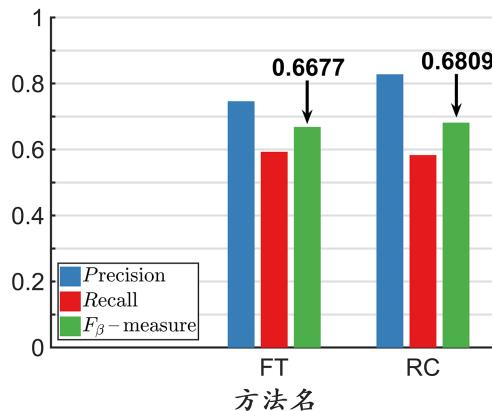


图 1.12 $P-R-F_\beta$ 柱状图

1.3 论文研究内容概述与创新点

本文针对静态图像场景, 基于自底向上的、底层数据驱动的图像处理技术, 围绕显著物体检测的计算方法及其应用开展了相关的研究工作, 提出了两种显著物体检测方法和一种协同显著物体检测方法, 并将显著物体检测技术应用到实际场景, 提出了一种单帧彩色图像小目标检测方法。本文的主要工作和创新点如下:

(1) 采用全局颜色差异以及区域空间关系, 提出了一种基于区域主色的显著物体检测方法。该方法首先对原始图像进行最小方差量化, 可在计算全局显著性时降低颜色差异度量的算法复杂度; 考虑到区域更易引起视觉注意且区域内部具有高度的颜色一致

性，对量化图像进行超像素分割后，对每个局部区域仅保留其区域主色，将区域间的颜色差异度量转换为区域主色间的差异度量，计算得到局部显著性；最后计算区域间距离以及区域与图像中心间距离，并利用这两种空间关系对局部显著性进行加权，在抑制非显著区域的同时提升中心区域的显著值。实验结果表明，区域主色可有效降低局部显著性计算时的颜色差异比较次数，最小方差量化和中心加权模式可有效提升检测性能。

(2) 采用环绕性线索和全局颜色名线索，提出了一种基于颜色名空间的显著物体检测方法。该方法首先将原始图像转换到颜色名空间，对各颜色名通道采用基于环绕性线索的图形-背景分离得到注意图，并对注意图进行线性融合和后处理操作生成显著图；然后将原始图像转换为颜色名图像，计算出全局的颜色名统计特征和对比特征，并将这两种全局特征与注意图相融合生成加权显著图；最后对显著图与加权显著图采用高亮截断的融合操作以及后处理操作，得到显著物体检测结果。实验结果表明，该方法可有效抑制图像背景，并在突出显著物体的同时使其获得均匀且平滑的显著值，得到的显著物体检测结果更接近于二值图像，便于后续对显著物体的分割。

(3) 采用颜色名线索、环绕性线索以及边界连通性线索，提出了一种基于分层结构与颜色名的协同显著物体检测方法。该方法首先对原始图像构造三个图像层，单层融合算法在每个图像层上利用显著一致性计算单层融合图，并基于颜色名的全局稀缺性改进得到单层显著图；然后采用基于显著一致性和颜色名稀缺性的多层融合算法对三个单层显著图进行融合，得到单图显著图；最后，对单图显著图进行自适应的阈值分割得到单图显著区域，通过颜色名相似性去除非协同显著区域生成协同显著图。实验结果表明融合机制和颜色名线索对协同显著物体检测的有效性。该方法除可用于检测一对图像的协同显著物体外，也是一种可用于单张图像的显著物体检测方法。

(4) 采用五种区域描述子、四个稳定性度量准则以及颜色的局部差异性，提出了一种融合区域稳定性与显著性的单帧彩色图像小目标检测方法。该方法首先将小目标检测看作是前景图形分割问题，通过序列化的阈值分割获得二值区域，并基于稳定性度量得到每个候选小目标的一组表象，在生成候选小目标的同时，获得目标区域精确的特征描述；然后将小目标检测看作是局部显著性检测问题，采用颜色的局部差异度量计算像素级的显著性，在突出小的目标区域的同时抑制均匀的背景模式；最后对稳定性算法生成的稳定图以及显著性算法生成的显著图进行融合，利用候选小目标的平均显著值剔除虚警目标。实验结果表明区域稳定性与区域显著性算法的性能优势以及融合机制的有效性。此外，标注并发布了一个用于彩色图像小目标检测的基准数据库，并提供了相应的性能评估代码。

1.4 论文内容安排

本文共分为六章，除本章以外，其余各章内容安排如图 1.13 所示，具体如下：第 2 章针对单张彩色图像，介绍了一种基于区域主色的显著物体检测方法；第 3 章同样针对单张彩色图像，介绍了一种基于颜色名空间的显著物体检测方法；第 4 章针对一对彩色图

像,介绍了一种基于分层结构与颜色名的协同显著物体检测方法;第5章是显著物体检测的一个具体应用,介绍了一种融合区域稳定性与显著性的单帧彩色图像小目标检测方法;第6章将总结全文并对后续的研究工作提出展望。

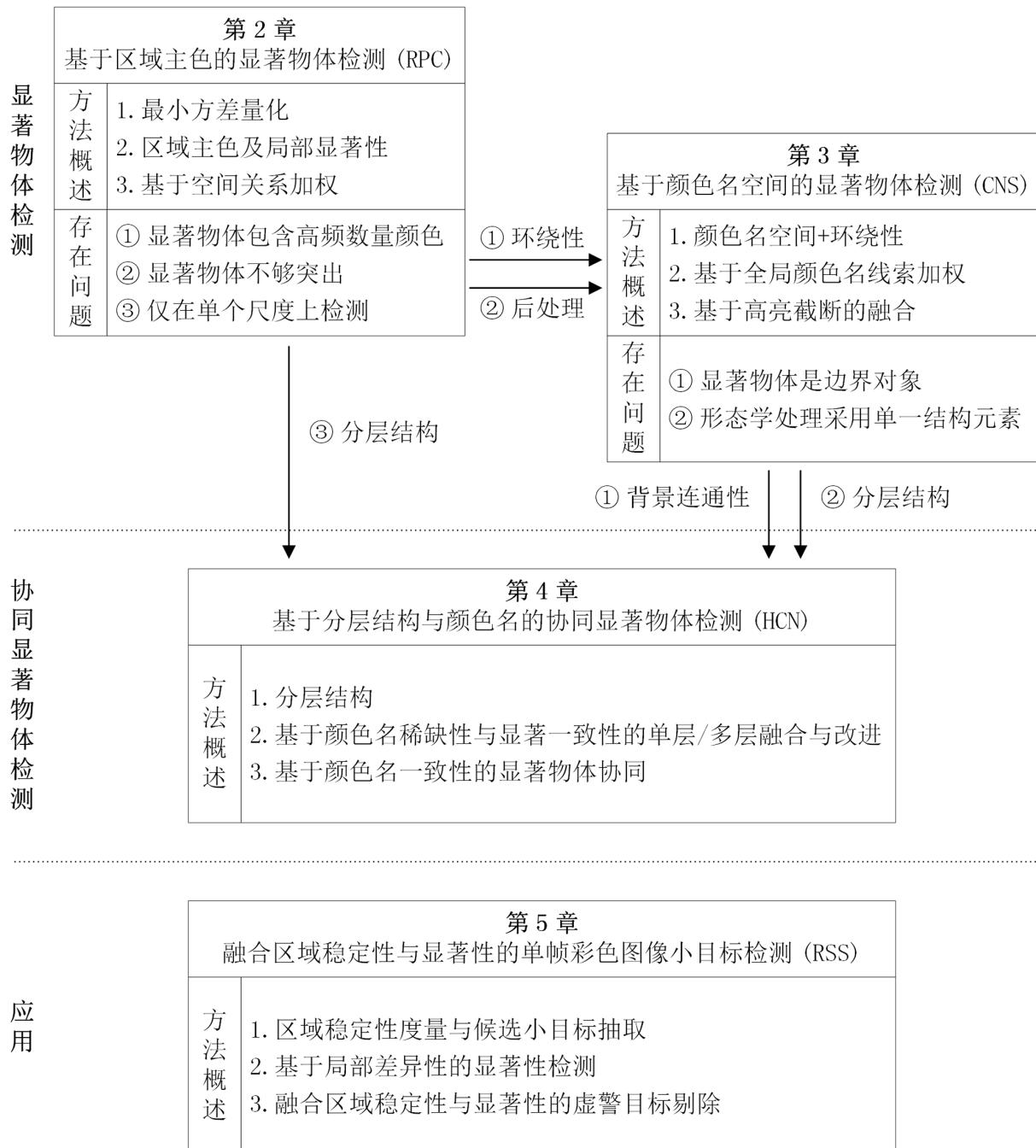


图 1.13 本文研究框架

2 基于区域主色的显著物体检测

2.1 引言

针对单张彩色图像显著物体检测，本章聚焦于快速的、自底向上的、底层颜色特征驱动的计算方法。此类方法不需要依赖与图像内容有关的先验知识或认知假设，而是从人类视觉注意机制出发，采用图像底层特征在局部或全局的对比关系来实现显著物体检测^[70]。通常，从局部角度考虑的方法基于局部区域与其环绕邻域的对比来计算显著性^[7,91]，但往往忽视了特征在整个场景中的发生频率或出现次数，因而会在局部差异明显的交界处产生较高的显著值；而从全局角度考虑的方法则利用特征在整个图像场景中的稀缺性（Rarity）来检测显著物体^[32]，但像素级的计算忽视了显著物体作为整体区域时内部像素的聚集性。因此，合理结合这两类方法，在全局显著性的基础上融合局部空间关系，可有效弥补两者的不足，进而得到更为理想的输出结果。

本章提出了一种基于区域主色的显著物体检测方法，该方法是一种全局结合局部的计算方法：一方面，采用整幅图像的颜色差异来计算最小方差量化后各颜色的全局显著性；另一方面，采用区域主色降低局部显著性计算时的算法复杂度，并采用两种空间关系进行加权。本章工作的创新点为：在最小方差量化的基础上，采用区域主色计算局部显著性，并基于两种空间距离对局部显著性加权，实验结果表明提出的方法可在更少的颜色数下获得更好的检测性能，并降低了算法复杂度。

为方便起见，本章提出的基于区域主色（Regional Principal Color）的显著物体检测方法，简记为“RPC”方法^①。本章内容安排如下：第2.2节介绍研究动机与出发点；第2.3节介绍RPC方法框架；第2.4节详细介绍RPC方法，包括基于最小方差量化的全局显著性算法、基于区域主色的局部显著性算法、以及基于两种空间距离的局部显著性加权；第2.5节是数据集介绍、参数分析、量化方式和中心偏置对性能影响的评估，以及实验结果和结果分析；最后是本章小结和未来工作展望。

2.2 研究动机与出发点

文献[29]从高效性、智能性、集约性、准确性四个方面出发，提出了基于直方图对比（Histogram Contrast）的HC方法以及基于区域对比（Region Contrast）的RC方法。HC方法首先采用均匀量化将原始图像的RGB每个通道量化为12级，统计量化图像的颜色直方图并抛弃5%的低频数量颜色，并采用颜色直方图实现颜色比较的快速计算。随后在LAB颜色空间度量颜色差异^[92]，利用颜色显著性平滑抵消相似颜色量化为

^① RPC方法的MATLAB源代码、显著图、性能指标结果以及幻灯片等资源，已发布在论文主页：<http://www.loujing.com/rpc-saliency/>。

不同值的影响，得到全局显著性检测结果。考虑到区域相比像素更易引起人的视觉注意，RC 方法在利用图割算法^[93] 将图像划分为超像素后，利用区域间的空间关系结合区域间的颜色比较，得到区域级的显著图。

然而，由于 HC 方法中采用的均匀量化方式对 RGB 颜色空间进行等间隔的划分，忽视了大量图像中存在的因不同颜色间像素数不平衡带来的非均匀分布问题，因而在一定程度上导致对颜色空间划分的失效^[94]。此外，由于超像素分割得到的每个区域依然有多种量化颜色，虽然 RC 方法在进行区域比较时为每个区域单独构造了颜色直方图进行加速，但仍然带来了一定的计算工作量，执行效率有进一步提升的空间。

针对上述问题，本章提出的 RPC 方法在量化阶段采用最小方差量化，可在降低量化颜色数的同时真实体现图像颜色的非均匀分布，并可在更少的颜色数下获得更好的检测性能。同时，对每个超像素采用区域主色表示，将区域间的颜色比较转换为区域主色间的比较，降低了局部显著性计算时的算法复杂度。最后，采用两种空间关系对局部显著性加权，有效抑制了局部显著图中的非显著区域并提升了中心区域的显著值。

2.3 方法框架

RPC 方法框架如图 2.1 所示，包括基于最小方差量化的全局显著性计算、基于区域主色的局部显著性计算、以及基于两种空间关系的局部显著性加权，具体如下：

- 模块 I—最小方差量化

为了减少参与比较的颜色数量并降低计算复杂度，本模块将原始图像最小方差量化为 32 种颜色，得到的量化结果与原始图像具有相似的视觉效果，也在更少的颜色数下体现出原始图像中非均匀的颜色分布。

- 模块 II—全局显著性

在对原始图像完成最小方差量化后，利用 HC 方法^[29]，本模块基于量化颜色在整个图像中的稀缺性以及颜色之间的差异性，计算得到量化颜色的全局显著性。同时，为了抵消在量化阶段将相似颜色量化为不同颜色值的影响，所有量化颜色被送入显著性平滑模块，进一步计算得到“全局显著图”。

- 模块 III—局部显著性

相较于像素，区域级的颜色信息更容易引起视觉注意，本模块首先采用图割算法将最小方差量化得到的结果划分为超像素。考虑到区域间颜色比较以及局部显著性的计算效率问题，采用每个区域的主色来表征区域的整体颜色，并利用区域主色的显著值得到“局部显著图”。

- 模块 IV—局部显著性加权

考虑到区域空间关系对局部显著性的影响，计算区域间距离以及区域与图像中心间的距离，并利用这两种空间关系对局部显著性加权。经过加权得到的显著图，可以抑制原局部显著图中的非显著区域，并提升中心区域的显著值。

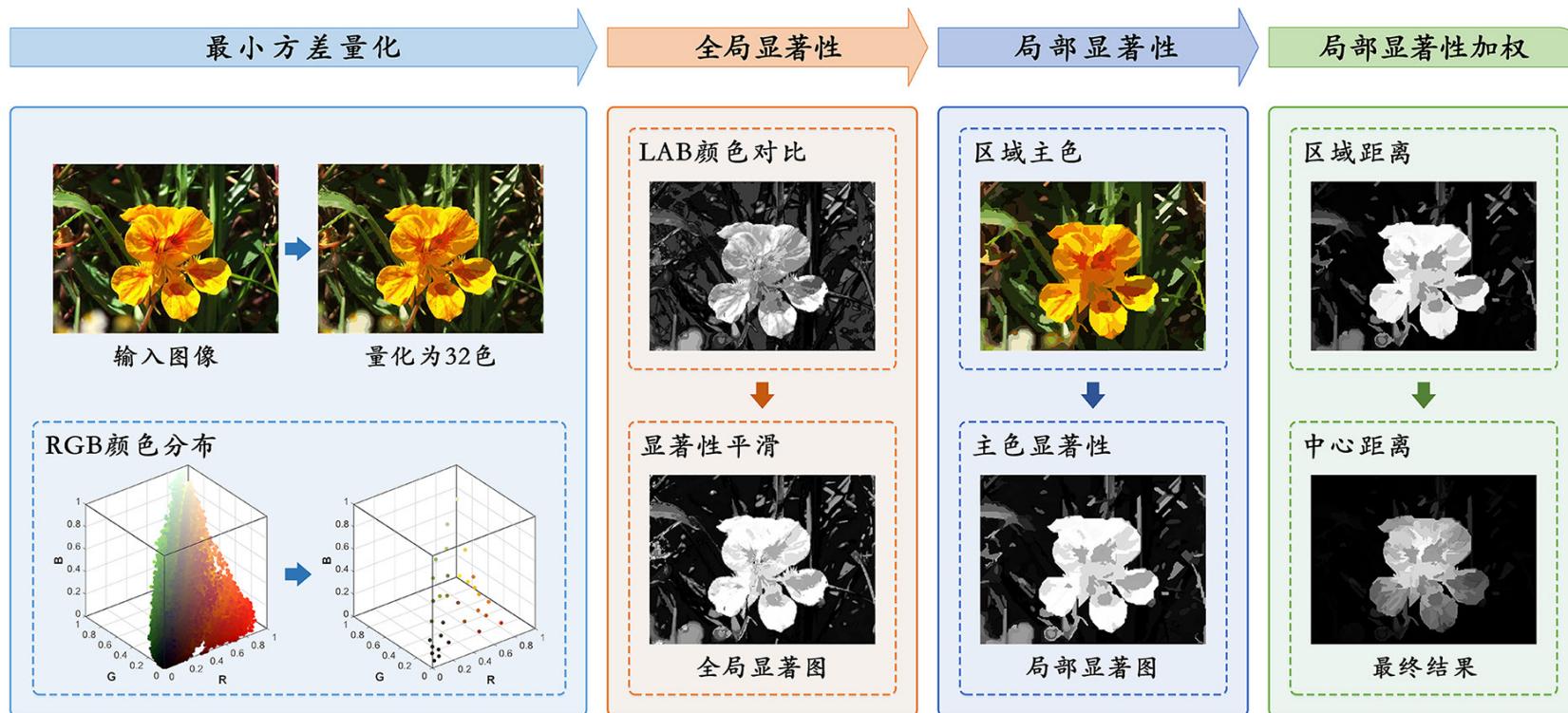


图 2.1 RPC 方法框架

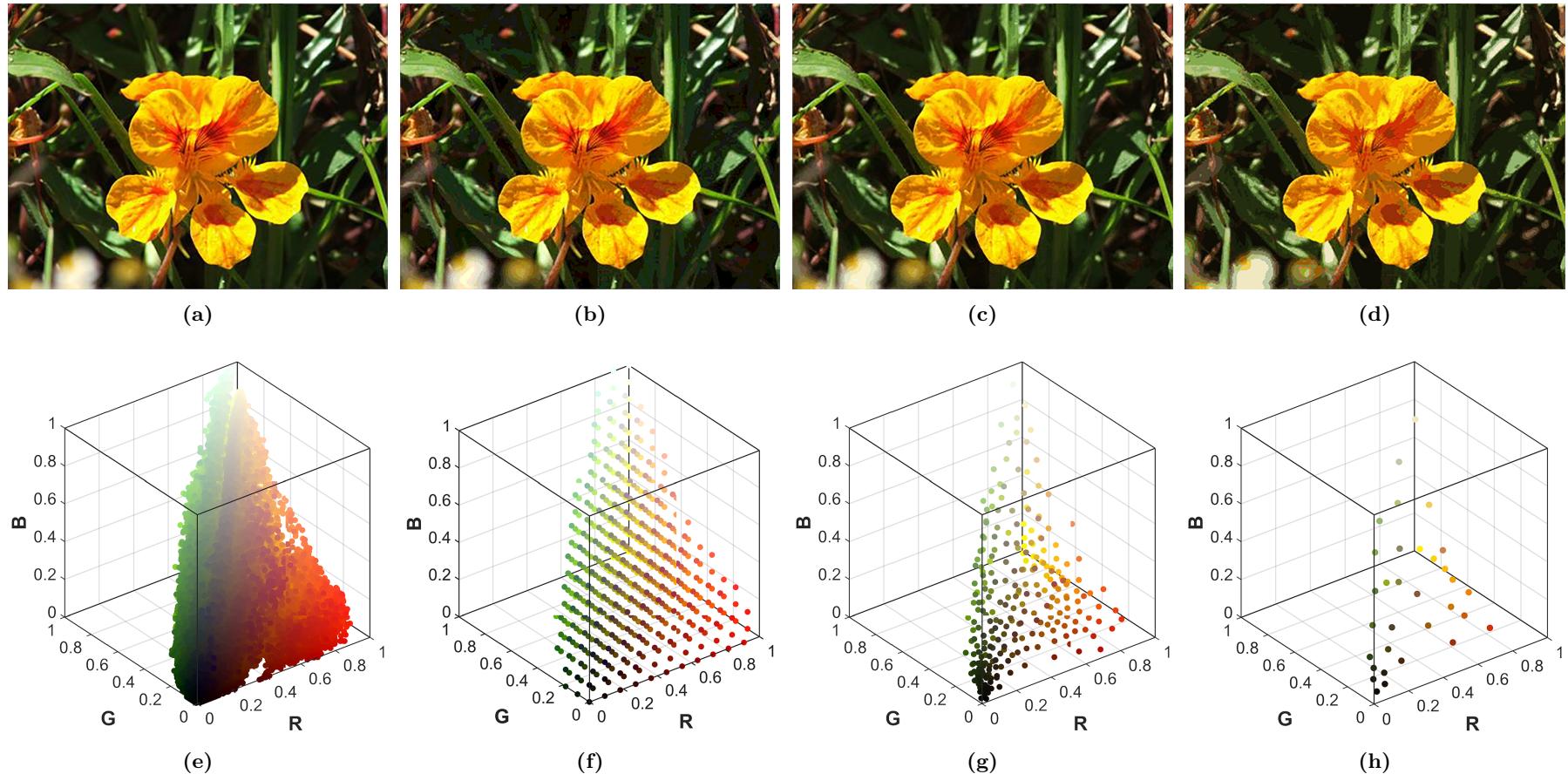


图 2.2 不同量化方式的 RGB 颜色空间分布。(a) 原始图像; (b) 将图(a)中每个颜色通道均匀量化为 12 级; (c) 将图(a)最小方差量化为 256 色; (d) 将图(a)最小方差量化为 32 色; (e)–(h) 分别为对应图(a)–(d) 的 RGB 颜色空间分布, 其中 x 轴、 y 轴、 z 轴分别表示红(R)、绿(G)、蓝(B)三个颜色分量, 各分量都已归一化到 $[0, 1]$ 区间

2.4 基于区域主色的显著物体检测方法

2.4.1 基于最小方差量化的全局显著性计算

基于颜色对比的显著性方法利用颜色差异计算显著性^[92]，这个过程需要不断进行两两颜色比较，对于一幅 24 比特的 RGB 图像，假设包含 N 种颜色，共需要比较 $N(N - 1)/2$ 次。一种极端的情况是图像覆盖了整个 RGB 颜色空间，由于各颜色通道的取值范围是 $[0, 255]$ ，即图像颜色数 $N = 256^3 = 16,777,216$ ，那比较次数将达到可怕的 140 万亿次。显然，如此大的计算量将导致实现的程序几乎没有运行的可能。

但是，对一幅输入图像的显著性计算并不真的需要如此多的颜色比较，这是因为：

(1) 一幅图像中所包含的颜色数，仅占整个颜色空间中相当小的比例。例如，图 2.2(a) 为来自于 ASD 数据集^[9,32] 的图像，图中一共包含有 60,904 种颜色，这个颜色数要远远小于 256^3 ，大约只占全彩色空间总颜色数的 0.36%。

(2) 由于图像包含的颜色丰富多彩，那些“极度稀缺”(极度稀少)的颜色将不被认为是显著的颜色，并通常被人类视觉系统忽略。以图 2.2(a) 为例，图中共含有 60,904 种颜色，但其中的 44,973 种颜色只占据了 1 个像素，也就是说，有 73.84% 的颜色在这幅图像中有且仅出现了一次。虽然这些像素的数量占到了图像的 37.48%，但即便将这些颜色从原图像中除去，就视觉上对人而言也并没有多大区别。

(3) 更重要的是，在全彩色空间中，人眼实际上很难分辨出两个相近颜色之间的差异。这就意味着对于与人类视觉注意紧密关联的显著性检测任务而言，完全可以合并掉其中一些颜色，以从整体上减少图像中所包含的颜色数量。

2.4.1.1 颜色空间的最小方差量化

为了降低颜色比较时的计算复杂度，一种有效的方式是对原图像进行颜色量化，即通过合并相近颜色来达到直接减少图像中所含颜色数量的目的。HC 方法^[29] 将 RGB 空间的每个颜色通道均匀地量化为 12 个等级，这使得量化后的图像中至多包含 $12^3 = 1728$ 种颜色。如果把整个 RGB 空间看作是一个颜色立方体(见图 2.3(a))^①，均匀量化实际上是把该立方体等间隔地划分为一组具有相同大小的小立方体，然后将每个小立方体内包含的颜色全部映射为中心颜色。关于均匀量化的一个简单示意如图 2.3(b) 所示，该图是将每个颜色通道均匀量化为 4 级后的 B-G(蓝色-绿色) 平面，其中“x”标记的位置表明落在该平面上的颜色。可以看到均匀量化后整个平面被等间隔地划分为 16 个相同大小的方格，而由于 5 个“x”标记的颜色全部位于同一个方格之内，因此这 5 个颜色将被统一映射为方格中心颜色。对于其它 15 个不含有任何颜色的方格而言，将直接丢弃而不会有任何颜色被映射为它们的中心颜色。

图 2.2(b) 是对图 2.2(a) 采用 12 级均匀量化后的结果，量化后图像中的颜色数从 60,904 减少为 518 种，与原图相比除了在边缘细节上有所损失外，肉眼很难发现两者

^① 参考 <https://www.mathworks.com/help/images/reduce-the-number-of-colors-in-an-image.html> 中的示例图像重新绘制。

之间的差别。虽然均匀量化机制看起来简单直接，也获得了视觉上相近的量化结果，但其本身还是有一定的缺陷。究其原因，在于自然场景中普遍存在着颜色非均匀分布的情况，即一幅图像中的颜色往往大量聚集于颜色空间中的某些位置，但均匀量化只是将颜色空间做了等间隔的划分，而忽视了这种颜色数量上的不平衡性。以图 2.2(a) 为例，其所有像素颜色在 RGB 空间中的分布如图 2.2(e) 所示。虽然原始图像中占据较大面积的花朵处于显著的中心地位，但背景中存在着包括黑色和深绿色在内的大量深颜色像素，这使得在 RGB 空间中这些深色像素密集地位于原点坐标 (0, 0, 0) 处，而花朵中包含的黄色、红色和咖啡色分布得则较为稀疏。但是，从均匀量化为 12 级后的颜色分布图 2.2(f) 中可以看到，包括原点附近位置在内，整个 RGB 空间并没有表现出颜色分布应有的疏密关系，所有颜色都以等间隔的方式呈现出规则的排列。因此，这种对颜色非均匀分布的忽视，使得均匀量化无法真实地体现图像原始的颜色分布情况，从而在一定程度上导致对颜色空间划分的失效^[94]。

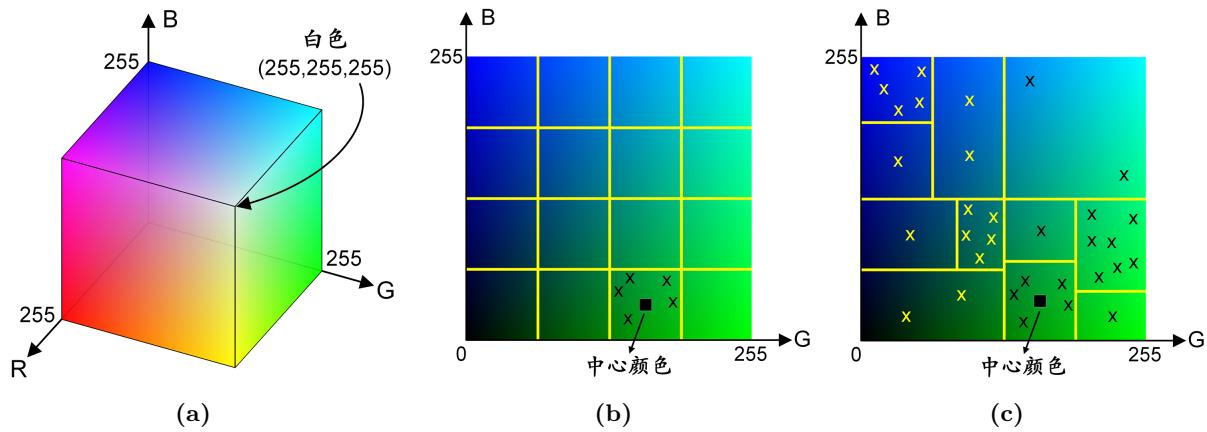


图 2.3 均匀量化和最小方差量化 (各颜色通道取值范围 [0, 255])。(a) RGB 彩色立方体; (b) 均匀量化; (c) 最小方差量化

为解决上述问题，RPC 方法使用最小方差量化^[95,96] 取代均匀量化。最小方差量化最初被用于处理连续色调的黑白图像，后被用于彩色图像处理。假设原始图像中坐标 (x, y) 处的颜色为 $c_{x,y}$ ，量化后的颜色为 $q_{x,y}$ ，量化前后的误差用 $d(c_{x,y}, q_{x,y})$ 表示，则整幅图像的量化误差为 $D = \sum_{x,y} d(c_{x,y}, q_{x,y})$ ，而量化过程即为最小化 D 的过程。由于该问题没有一个最优的解决方案^[97]，因此通常使用一些近似算法^[98-101] 来处理。如 P. Heckbert^[98] 对原始图像首先构造一个初始的颜色表，将所有颜色以大致相同的数量分配给初始颜色表中的各个颜色，然后在 RGB 空间采用颜色距离平方作为误差度量，利用原始颜色与颜色表中颜色的误差，不断迭代最小化 D ，得到最终的量化结果。

RPC 方法中，最小方差量化是通过直接调用 MATLAB 内置的不带颜色抖动的 `rgb2ind` 函数实现，如图 2.3(c) 所示。与均匀量化不同，最小方差量化不再将 RGB 空间等间隔地划分为相同大小的小立方体，而是依据颜色分布的疏密自适应地划分颜色空间。假设原始图像中包含较多的绿色、较少的蓝色，则最小方差量化将为颜色立方

体中分布较为密集的区域(即高频数量颜色),分配更多的量化颜色数(见绿色区域);而对于分布较为稀疏的区域(即低频数量颜色),分配较少的量化颜色数(见蓝色区域);当整个颜色立方体划分结束后,同样将每个小立方体内包含的所有颜色映射为中心颜色。

图 2.2(c) 和 2.2(d) 是将图 2.2(a) 分别最小方差量化到 256 色和 32 色的结果。为了展示量化结果与原始图像之间的实际差异,表 2.1 给出了 R、G、B 三个颜色通道在量化前后的 MAE 值(定义见第 1.2.4.2 节),计算时归一化到 [0, 1] 区间。当量化到相同的颜色数 12^3 即 1728 种颜色时,最小方差量化的 MAE 明显小于均匀量化,表明前者的量化结果更接近于原始图像。当同样采用最小方差量化时,随着量化颜色数的减少,伪轮廓和马赛克效应逐渐加剧,量化结果与原始图像之间的差异逐渐增大,表明颜色数确实会对量化结果产生影响。但从图 2.2 中可以看到这种差异并不明显,即使量化到 32 色,还是能够真实地重现原始图像中主要的颜色分布。

表 2.1 量化结果与原始图像的比较

量化方式	量化颜色数	红色 (R)	绿色 (G)	蓝色 (B)	平均
均匀量化	$12^3 = 1728$.0210	.0234	.0212	.0219
	$12^3 = 1728$.0086	.0086	.0083	.0085
最小方差量化	256	.0151	.0146	.0146	.0148
	32	.0320	.0283	.0284	.0295

这里需要重点关注的是两种量化方式产生的结果在 RGB 空间中分布的不同。与均匀量化得到的分布图 2.2(f) 相比,最小方差量化至 256 色时,更多的颜色被分配给靠近原点处(见图 2.2(g)),这与之前讨论的原始图像内包含的颜色特征相一致;即便是量化到 32 色(见图 2.2(h)),仍然会将更多的颜色分配给这一区域。这表明,最小方差量化会针对颜色数量的不平衡,依据图像数据动态划分颜色空间。在第 2.5.3.1 节中,将继续讨论两种量化方式在不同颜色数下的性能比较,以表明最小方差量化的优势。最小方差量化颜色数对 RPC 方法性能的影响,将在第 2.5.2 节中予以讨论。

2.4.1.2 全局显著性度量与平滑

在完成对原始图像的最小方差量化后,就可以比较量化颜色之间的差异来计算各颜色的全局显著性。在 RPC 方法中,这一过程是利用 HC 方法^[29] 实现。假设最小方差量化后保留的颜色数为 N ,则量化颜色 c_i 的全局显著值为

$$S(c_i) = \sum_{j=1}^N f_j D(c_i, c_j), \quad (2.1)$$

其中, f_j 为颜色 c_j 的频数,即该颜色在量化图像中出现的次数; $D(c_i, c_j)$ 为颜色 c_i 与 c_j 的颜色差异,通过在 LAB 颜色空间计算颜色之间的欧氏距离^[92] 获得,即

$$D(c_i, c_j) = \|(L_i, A_i, B_i) - (L_j, A_j, B_j)\|_2, \quad (2.2)$$

其中, 符号 $\|\cdot\|_2$ 表示 ℓ_2 范数, (L_i, A_i, B_i) 和 (L_j, A_j, B_j) 分别表示颜色 c_i 和 c_j 在 LAB 空间三个通道 L 、 A 和 B 上的分量值。

式(2.1)利用了颜色在整幅图像中的全局稀缺性, 即相对于图像中的高频数量颜色, 那些低频数量颜色要更为显著。对于颜色 c_i 而言, 其它数量越多、且与 c_i 差异越大的颜色, 将对 c_i 显著值的贡献越大。为了降低计算复杂度, HC 方法对量化后的图像构造颜色直方图, 仅保留其中 $\alpha = 95\%$ 的高频数量颜色, 而对于剩余 5% 的低频数量颜色, 则使用与其最接近的高频数量颜色取代。最后, 为了抵消量化阶段将相似颜色量化为不同颜色值的影响, HC 方法利用与颜色 c_i ($i \in [1, N]$) 最相似的 m 个颜色的显著值进行显著性平滑, 得到颜色 c_i 平滑后的显著值

$$\tilde{S}(c_i) = \sum_{j=1}^m \frac{T - D(c_i, c_j)}{(m-1)T} S(c_j), \quad (2.3)$$

其中, m 被设置为量化颜色数的 $1/4$, 即 $m = N/4$; $T = \sum_{j=1}^m D(c_i, c_j)$ 表示颜色 c_i 与其它 m 个相似颜色的距离之和。 $\frac{T-D(c_i, c_j)}{(m-1)T}$ 是 $S(c_j)$ 的权重。对于颜色 c_i , 由于 $(m-1)T$ 是一个定值, 因此 c_j 与 c_i 的颜色越相似, 则 $S(c_j)$ 被赋予的权重越大。

图 2.4(a) 和 2.4(b) 分别是最小方差量化前后的图 2.2(a) 和图 2.2(d) 的颜色直方图。在量化之前的原始图像中, 频数最高的颜色的 RGB 值为 $(24, 34, 10)$, 像素数为 149; 量化之后, 频数最高的颜色值为 $(15, 14, 5)$, 占 15,776 个像素。从图 2.4(b) 的颜色直方图可以看到, 最小方差量化后, 黑色和深绿色依然属于高频数量颜色, 花朵区域所包含的颜色处于整个直方图的中间位置, 既不会因像素数过少被忽略, 同时又与高频数量颜色之间有着明显的颜色差异, 这确保了花朵区域能在整个场景中凸显出来。

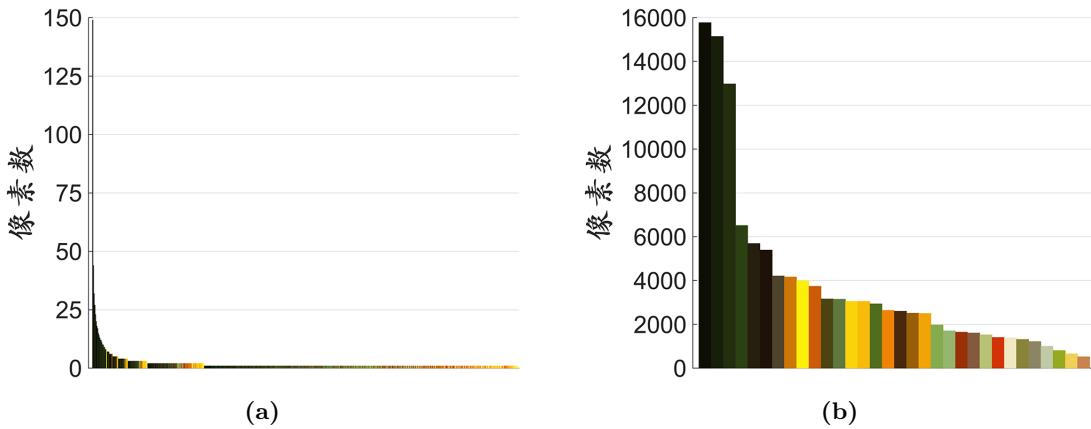


图 2.4 颜色直方图。(a) 图 2.2(a) 的颜色直方图; (b) 图 2.2(d) 的颜色直方图

由于式(2.1)和(2.3)计算的是与颜色有关的显著值, 将这些显著值依据颜色对应地回填到图 2.2(d)中, 即可得到全局显著图。图 2.5 给出了显著性平滑前后的全局显著图对比, 其中图 2.5(a) 是在显著性平滑之前, 基于 32 种最小方差量化颜色比较得到的全局显著图; 图 2.5(b) 是对前者显著性平滑后的结果。可以看到, 经过显著性平滑处理后的

直观效果是显著物体区域的显著值被进一步提升，而原显著值较低的区域被有所抑制，这就更加凸显了图像中的花朵区域。需要说明的是，出于论文展示的需要，图 2.5(a) 经过了归一化处理，使得像素值的范围在 $[0, 255]$ 区间，而在实际的代码中，归一化处理只在显著性平滑之后才被执行。

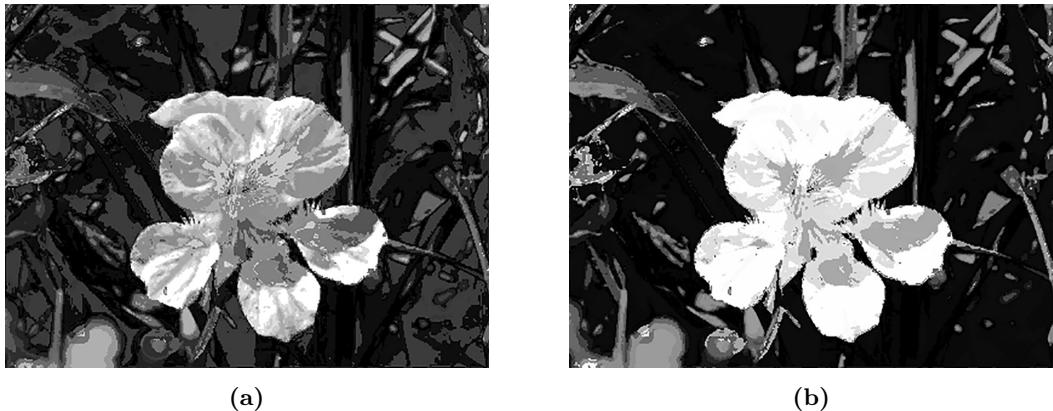


图 2.5 全局显著图。(a) 最小方差量化为 32 色后，基于颜色差异对比得到的全局显著图；(b) 对图(a)进行显著性平滑后的结果

需要特别指出的是，本章工作已发表成果^[102] 中，对保留的高频数量颜色比例参数 α 仅讨论了 0.9 和 0.95 时的情况，并依据性能评估确定 α 的最优参数为 0.95。 α 的参数取值将在第 2.5.2 节中被重新讨论，预设的参数范围是 $[0.9 : 0.01 : 1.0]$ ，且由于参数分析中使用了不同的性能指标，导致 α 的最优取值有所不同。参数分析表明，当 $\alpha = 1$ 时性能最优，这意味着 RPC 方法在最小方差量化之后不需要抛弃 $1 - \alpha$ 的低频数量颜色，而应将 32 种量化颜色全部送入后续的显著性平滑模块。原因在于经过最小方差量化后，图像中仅包括少量颜色，如若再抛弃颜色，反而会因颜色数过少导致检测性能的下降。因此颜色直方图 2.4(b) 仅为示例，RPC 方法实现并不需要颜色直方图的参与。此外，在显著性平滑阶段，相似颜色数量 m 的取值被设置为与量化颜色数 N 有关，即 $m = \lceil \delta \times N \rceil$ 。在已发表成果^[102] 中，仅评估了参数 $\delta \in \{1/16, 1/4\}$ 时的检测性能， δ 也将在第 2.5.2 节中被重新讨论，预设的参数范围是 $1/2^k, k \in \{0, 1, \dots, 6\}$ 。

2.4.2 基于区域主色的局部显著性计算

为了进一步减少参与区域间比较的颜色数量并加速计算过程，本章提出的 RPC 方法采用区域主色来取代区域内包含的所有颜色，这里的区域主色，是指每个区域内出现频数最高的颜色。采用区域主色表征区域颜色的出发点如下：1) 就颜色特征而言，相较于像素，区域更容易引起视觉注意，特别是那些与周围颜色差异明显的区域^[103]；2) 超像素分割会把局部范围内具有相似颜色的像素划分到同一区域中，区域内个别像素的颜色差异会被区域的整体颜色相似性所淹没；3) 由于区域内颜色高度的相似性，这些颜色在经过全局显著性计算以及显著性平滑后也会获得接近的显著值，这使得区域内呈现出较为一致的显著性。因此，使用区域主色表征区域整体颜色所带来的直接好处，

是可以利用区域主色的显著性来取代区域的显著性，从而大大降低引入空间关系后区域之间颜色比较的计算复杂度。

RPC 方法直接调用文献 [93] 提供的 C++ 程序完成量化图像的超像素分割，并以超像素作为局部区域。考虑到每个区域内部应尽可能具有高度一致的颜色，同时将区域内所有颜色替换为区域主色后，区域之间的颜色比较仅需要执行一次，因此适当增加了超像素个数，以在较小的粒度上进行划分。实验中，设置尺度参数以及最小区域大小为 50 个像素，高斯滤波器的宽度被设置为 0.5。图 2.6 是对最小方差量化为 32 色的图像 2.2(d) 进行超像素分割的结果，图中共包含 518 个超像素。可以看到，每个超像素内部的颜色值保持高度一致，同时各区域包括物体区域的边界得到了较好的保持。

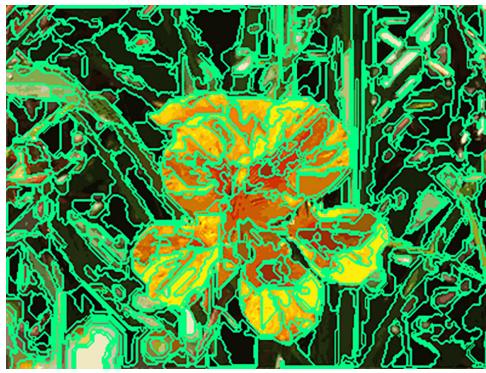


图 2.6 基于图割方法得到的超像素

在完成超像素分割后，就可以利用区域主色的显著值得到区域的显著值，即使用每个区域的区域主色的全局显著值取代该区域内其它所有像素的显著值。为方便期间，这里对相关的符号重新进行定义。假设分割过程得到 k 个区域，第 i 个区域被记为 r_i ，其中 $i \in [1, k]$ 。假设 r_i 中颜色 c 出现的频次最高，则 c 就为区域 r_i 的主色，主色经过显著性平滑处理后的显著值 $\tilde{S}(c)$ 采用式 (2.3) 计算得到。而局部显著性就是将区域 r_i 中所有像素的显著值统一替换为主色 c 的显著值 $\tilde{S}(c)$ ，由于 r_i 中所有像素共享相同的显著值，将区域 r_i 的显著值记为 $S(r_i)$ 。最后，将所有区域的显著值归一化到 $[0, 1]$ 区间，即得到基于区域主色的局部显著图。

图 2.7(a) 是将每个区域使用区域主色填充的结果。客观来看，采用区域主色确实带来了一定程度上的失真，导致填充结果类似于在量化颜色基础上产生的伪彩色图像。然而，除了一些细节上的丢失外，主色填充图中仍然保留了原始图像中显著物体以及背景的主要颜色分布。为了展示填充前后的差异，图 2.7(b) 和 2.7(c) 分别给出了区域主色填充图与原始图像 2.2(a) 以及最小方差量化为 32 色图像 2.2(d) 的差异图。可以看到，由于最小方差量化图接近于原始图像，两个差异图之间也趋于相同。它们与区域主色填充图的差异主要在区域边界上，物体内部的颜色差异实际上并不明显。

图 2.8(a) 是基于区域主色显著值得到的局部显著图，该图与全局显著图 2.5(b) 的差异热图如图 2.8(b) 所示。可以看到，相较于全局显著图，局部显著图中的物体区域和背景区域中的孔洞被填充，部分噪声点和噪声区域得以抑制，使得显著性结果在整体上

更为平滑。就显著物体本身而言，两个显著图在物体区域的边界上差异较为明显，这主要是由于经过全局显著性计算后边缘位置已被平滑，而在较小粒度上的超像素分割以及区域主色填充之后，颜色差异被进一步累积。但局部显著图存在的一个问题是对背景显著性的抑制不足，其中部分背景区域与显著物体内部具有相近甚至更高的显著值，这直接导致了检测精度的下降。为此，下一节将采用两种空间关系对局部显著性加权，以在突出显著物体的同时抑制背景区域。

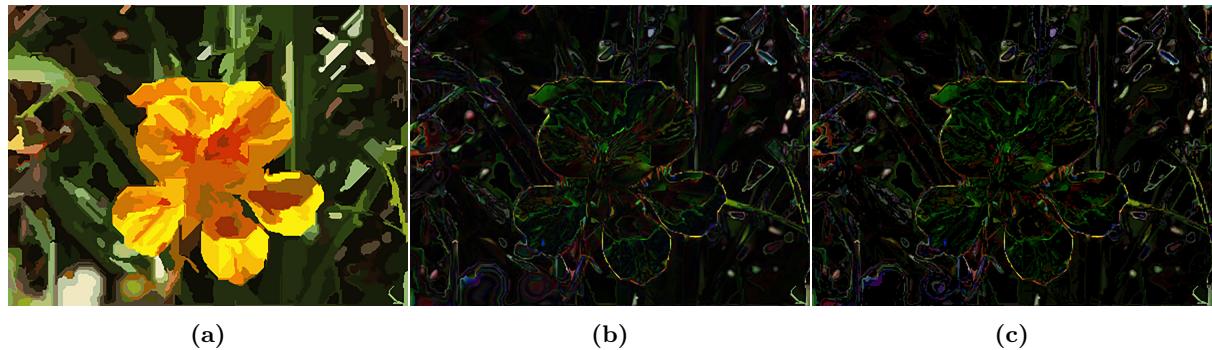


图 2.7 区域主色。(a) 使用区域主色填充的结果；(b) 图(a)与原始图像 2.2(a) 的差异；(c) 图(a)与最小方差量化为 32 色图像 2.2(d) 的差异

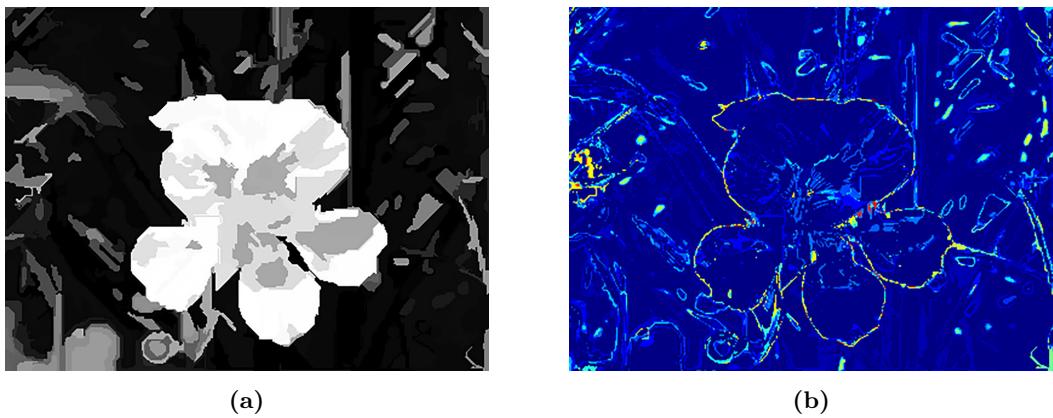


图 2.8 局部显著图。(a) 基于区域主色的局部显著图；(b) 图(a)与全局显著图 2.5(b) 的差异

2.4.3 基于空间距离的局部显著性加权

除了区域之间的颜色比较外，空间关系也在显著性度量时发挥着重要作用。从视觉注意机制的角度来看，人类会动态地扫描场景，并不断基于中心–环绕原则^[9,104] 在场景不同位置间转移视点。通常，视点会首先定位在一个引起视觉注意的显著区域，然后快速转移到下一个在空间上与前者相邻的显著区域处^[7]。B. Tatler^[72] 和 T. Judd^[73] 等人的研究结果表明，相较于外围区域，人们更倾向于首先关注场景的中心，即视点会更高频地围绕在中心区域。虽然没有直接的证据表明这种中心注视是否由场景自身的中心偏置 (Center Bias) 所致^[72]，但图像中心常常在初始化上下文建模时被用来抽取全局

的场景特征^[105]，而中心偏置概念，即显著物体通常位于一幅图像的中心区域，也往往以二维高斯分布的形式被用在视觉显著性检测任务中^[57,61,106,107]。

本节对局部显著性进行加权的两种空间关系，分别是区域间的距离以及区域与图像中心间的距离：对于前者，度量的是区域间空间关系对局部显著性的影响；而后者，影响的是在整幅图像不同全局空间位置上的局部显著性。

2.4.3.1 基于区域间距离的局部显著性加权

RPC 方法中采用的区域间距离，是指两个不同区域质心间的欧氏距离。为了简化质心的计算过程，这里简单地将每个区域内部所有像素的 x 坐标和 y 坐标的均值作为质心的 (x, y) 坐标，同时将计算得到的所有质心距离归一化到 $[0, 1]$ 区间。假设符号 $D(r_i, r_j)$ 表示区域 r_i 和 r_j 质心间的欧氏距离，对于 r_i ，基于区域间距离加权的局部显著性 $\tilde{S}(r_i)$ 计算为

$$\tilde{S}(r_i) = f_i S(r_i) + \sum_{j=1}^k \frac{f_j \phi(r_i, r_j)}{\exp\{D(r_i, r_j)^2\}}, \quad (2.4)$$

其中 k 是总的区域个数， f_i 和 f_j 分别是区域 r_i 和 r_j 中包含的像素数，且

$$\phi(r_i, r_j) = \begin{cases} S(r_i) - S(r_j), & \text{if } S(r_i) > S(r_j); \\ 0, & \text{otherwise.} \end{cases} \quad (2.5)$$

式(2.4)可以看作是式(2.1)的扩展，是在颜色比较的基础上引入了区域间的空间关系。这主要来自于两个朴素的动机：1) 对于基于区域主色得到的局部显著图中的显著性区域而言，自身的显著性应予以保留；2) 对于局部显著图中的非显著性区域，显著值不应被其它显著的区域所间接影响。因此，式(2.4)得到的显著值由两部分构成：

- $f_i S(r_i)$ 决定了加权后的显著值首先由其自身的显著值所决定，对于那些原来显著值高、所占面积大的区域，在加权结果中依然保持较高的显著值。

- $\sum_{j=1}^k \frac{f_j \phi(r_i, r_j)}{\exp\{D(r_i, r_j)^2\}}$ 中， r_i 加权后的显著值除了与其它区域的面积以及空间距离有关，还取决于因子 $\phi(r_i, r_j)$ 。那些距离 r_i 近、且面积大的区域，会对 r_i 的显著值产生更多的影响，但这种影响主要来自于比 r_i 显著值低的区域。可以想象，假设 $S(r_i)$ 较大（即 r_i 原为显著性区域），那么显著值低于 r_i 的区域会使 $\tilde{S}(r_i)$ 进一步提升；相反，如果 $S(r_i)$ 较小（即 r_i 原为非显著性区域），那么那些显著值高于 r_i 的区域不会对 $\tilde{S}(r_i)$ 产生实质影响。采用这种方式，可以避免那些紧邻显著性区域的背景区域的显著值被显著性区域一起带动提升，从而保持对背景区域的抑制。

图 2.9(a) 是经过空间距离加权并归一化得到的局部显著图，它与未加权的局部显著图 2.8(a) 的差异如图 2.9(b) 所示。可以看到，经过空间距离加权后，原局部显著图中非显著的背景被进一步抑制，导致加权前后背景部分的显著值差异较大。虽然归一化处理会导致显著物体的显著值有所下降，但下降的程度并不明显。

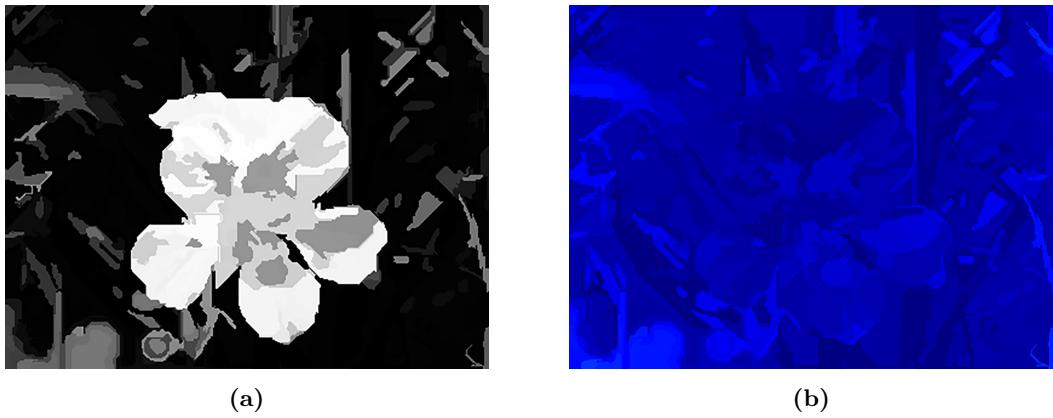


图 2.9 基于区域间距离的显著性加权。(a) 基于区域间距离加权的显著图; (b) 图(a)与局部显著图 2.8(a) 的差异

最后,来看一下采用区域主色后式(2.4)的算法复杂度。 $\tilde{S}(r_i)$ 的计算主要取决于 r_i 与其它 $k - 1$ 个区域之间的比较,由于对每个区域采用区域主色, r_i 与其它任意一个区域 r_j 仅需要比较一次,故 $\tilde{S}(r_i)$ 的算法复杂度为 $\mathcal{O}(k)$ 。整幅图像共包含 k 个区域,因此整体的算法复杂度为 $\mathcal{O}(k^2)$,也就是说,算法复杂度仅与超像素分割后产生的区域个数有关。下面考虑不采用区域主色的情况,此时区域 r_i 的加权显著值 $\tilde{S}(r_i)$ 不仅取决于区域个数 k ,还与每个区域内部所包含的颜色数有关。以理论上限、即每个区域内部都包含有 N 个量化颜色计算,每一对区域之间颜色显著性比较的算法复杂度为 $\mathcal{O}(N^2)$,从而 $\tilde{S}(r_i)$ 的算法复杂度为 $\mathcal{O}(kN^2)$,整体算法复杂度为 $\mathcal{O}(k^2N^2)$ 。因此,在不采用区域主色的情况下,算法复杂度不仅与区域个数有关,还取决于量化所产生的颜色数量。当然,加速上述步骤的一种有效方式是直接将比较操作改为查表操作,即通过简单的查询来替换重复的比较过程。但由于区域个数 k 与量化颜色数 N 大致在一个数量级上,即便采用查表操作,还是会得到相同的算法复杂度。由此可见,RPC 方法采用的区域主色,可以有效降低基于空间距离进行局部显著性加权时的算法复杂度。

2.4.3.2 基于区域与图像中心距离的局部显著性加权

基于区域与图像中心距离的局部显著性加权,度量的是每个区域质心与图像中心的欧氏距离。假设符号 I_c 表示图像中心的 (x, y) 坐标,区域 r_i 与图像中心的欧氏距离用 $D(r_i, I_c)$ 表示。将所有距离归一化到 $[0, 1]$ 区间后,加权显著性的计算公式为

$$\hat{S}(r_i) = \exp \left\{ -D(r_i, I_c)^2 / \sigma \right\} \tilde{S}(r_i), \quad (2.6)$$

其中 σ 是一个待定参数。

式(2.6)中的 $\exp \left\{ -D(r_i, I_c)^2 / \sigma \right\}$ 作为权重,控制着各区域受抑制的程度:一方面,当 σ 固定时,区域 r_i 距离图像中心越远,则对应的 $D(r_i, I_c)$ 越大,导致 r_i 受抑制的程度越大;另一方面,参数 σ 的值越小,距离图像中心越远的区域受抑制的程度越大。上述两种情况本质上都采用了中心加权的思想,越远离图像中心的区域的显著值受

抑制的程度越大，表明 RPC 方法越希望显著性区域聚集在图像的中心。

将经过中心加权后的各区域的显著值归一化，就可以得到如图 2.10(a) 所示的 RPC 方法的结果显著图，与基于区域距离加权的局部显著图 2.9(a) 的差异如图 2.10(b) 所示。可以看到，尽管背景区域的显著值已被最大限度地抑制为 0，但花朵区域特别是远离图像中心的花瓣部分，也受到了一定程度的抑制。虽然花朵区域依然占据了整幅图像中绝对的显著地位，但相对于人工标注结果 2.10(c) 还是有一定的差距。实际上，对于像图 2.10(a) 这样的显著图，通过简单的亮度变换就可以在突出显著物体区域的同时使其获得均匀且平滑的显著值。第 3 章将基于统计直方图进行自适应的亮度变换，以使得显著图更接近于二值图像。

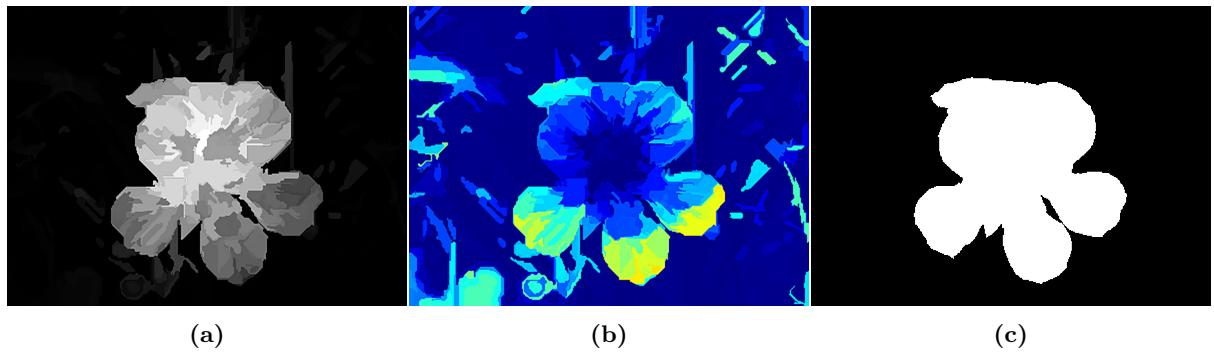


图 2.10 基于区域与图像中心间距离的显著性加权。(a) 基于区域与图像中心间距离加权的显著图；(b) 图 (a) 与基于区域间距离加权的显著图 2.9(a) 的差异；(c) 人工标注结果^[32]；

2.5 实验与讨论

为了实际评估提出的 RPC 方法的有效性，本章在 ASD 数据集上与 24 个显著性方法进行性能比较，包括：AC^[108]、AIM^[109]、CA^[70]、COV^[110]、FES^[111]、FT^[32]、GB^[91]、HC^[29]、HFT^[46]、IT^[7]、LC^[92]、MNP^[17]、MSS^[112]、MZ^[55]、OBJ^[49]、RC^[29]、SEG^[113]、SeR^[114]、SIM^[115]、SR^[37]、SS^[116]、SUN^[117]、SVO^[118]、SWD^[119]。这些方法名的英文缩写，与文献 [28] 保持一致。

2.5.1 数据集简介

有关 ASD 数据集^[9,32]的采集和人工标注过程，已在第 1.2.3 节中做过介绍。ASD 数据集中共有 1,000 张图像，其中包含 1,206 个人工标注的显著物体区域。在 1,000 张图像中，有 896 张图像仅包含一个显著物体区域，剩余的 104 张图像中则含有两个及以上的显著物体区域。图 2.11(a) 是所有人工标注显著物体的平均标注图，可以看到该数据集带有中心偏置现象。然而，平均标注图中心点的显著值约为 0.7137 (区间 [0, 1])，表明约有 28.63% 的显著物体区域内不包含图像的中心 (否则中心点的显著值应为 1)。在 1,206 个标注区域中，有 16 个区域与图像边界相连接，它们的平均标注图如图 2.11(b) 所示。此外，整个数据集的图像平均分辨率为 373×326 像素 (宽度 \times 高

度)。如果对所有人工标注的显著物体区域的面积大小进行归一化处理, 概率密度分布如图 2.11(c) 所示。平均而言, 在 ASD 数据集中, 显著物体区域的面积大小约占整个图像面积大小的 19.85% (绿色实线所示)。概率密度曲线峰值处所对应的 x 轴坐标约为 0.1952 (红色虚线所示), 这意味着在所有人工标注的显著物体区域中, 占图像面积大小 19.52% 的区域数量最多。

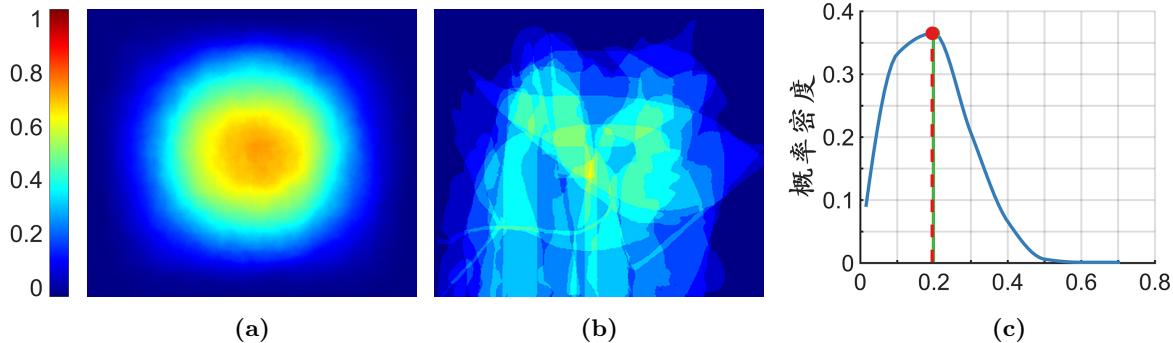


图 2.11 ASD 数据集特性统计。(a) 数据集的平均标注图 (归一化到 $[0, 1]$); (b) 与图像边界连接的显著区域的平均标注图 (归一化到 $[0, 1]$); (c) 显著区域大小的概率密度曲线: x 轴表示显著区域占图像面积大小的比例

2.5.2 参数分析

RPC 方法共需要 3 个参数, 包括最小方差量化颜色数 N 、颜色显著性平滑时相似颜色比例 δ 、以及局部显著性中心加权的权重 σ 。最优参数的选取过程是通过分析参数曲线完成, 即预先设定每个参数的取值范围, 并计算每个参数取值下的 MaxF 指标值 (定义见第 1.2.4.3 节), 最后以 MaxF 曲线峰值所在位置的 x 坐标作为参数的最优取值。具体到对某个参数调参时, 其它参数的取值保持固定不变, 即仅对单个参数执行调参操作。3 个参数预设的取值范围分别是: $N = 2^k, k \in [2 : 12]$ 、 $\delta = 1/2^k, k \in [0 : 6]$ 、 $\sigma \in [0.1 : 0.1 : 1.0]$ 。此外, 本节也分析了如果在最小方差量化之后, 仅保留部分高频数量颜色对 RPC 方法性能的影响, 这里采用参数 α 表示高频数量颜色占量化颜色数 N 的比例, 该参数的预设取值范围为 $[0.9 : 0.01 : 1.0]$ 。

图 2.12(a) – 2.12(d) 是对应的参数曲线, 曲线峰值点使用红色圆点标记。参数曲线中, x 轴表示预设的参数取值范围, y 轴表示不同参数取值下的 MaxF 指标。所有参数曲线的具体指标值见表 2.2, 其中 MaxF 的最大值使用红色标记。相关分析如下:

(1) 随着最小方差量化保留的颜色数 N 的不断增大, 方法性能不断提升, 直至处于平缓变化阶段。在参数曲线中, 有两个局部峰值点存在, 分别是当颜色数为 32 色和 1024 色时, MaxF 指标值分别为 0.8002 和 0.8008。理论上, 设置最小方差量化颜色数为 1024 色, 方法性能可以达到最优。但保留的颜色数越多就意味着计算时间的消耗, 考虑到方法整体的执行效率, RPC 方法使用次优参数, 即在量化阶段将输入图像最小方差量化为 32 色。

(2) 对于两个具有相似变化趋势的参数 δ 和 σ , 参数曲线中指标性能在逐渐上升到峰值点后, 随着参数值的进一步增大而逐渐下降。对于参数 δ , 峰值点处的参数值为 $1/4$, 表明 25% 的相近颜色参与显著性平滑计算时性能最优, 这与 HC 方法^[29] 作者的建议一致。对于参数 σ , 由于其决定了各区域距离图像中心远近时受抑制的程度, 最优取值 0.2 表明 RPC 方法更趋向于将显著物体位置置于图像中心, 这符合大部分显著性数据集 (包括 ASD 在内) 建立时被主观引入的中心偏置事实。

(3) 最后是关于高频数量颜色数占量化颜色数的比例参数 α 。参数曲线显示, 检测性能随着参数取值的不断增大而逐渐提升, 直至参数值为 1.0 (即 100%) 时达到性能最优。这说明, 所有量化得到的颜色都需要参与到显著性平滑计算中, 但在平滑之前并不需要替换掉低频数量颜色, 这种现象的产生与 RPC 方法的量化方式有关。由于最小方差量化更能体现原始图像中非均匀的颜色分布, 因此可在更少的颜色数下获得更好的检测效果。如果在显著性平滑前替换掉低频数量颜色, 反而会因参与对比的颜色数过少而导致不具有差异性, 从而拉低了显著性方法的检测性能。因此, 本节虽然对 α 进行了参数分析, 但在 RPC 方法的实现中并不需要引入该参数, 也就是说, 最小方差量化得到的颜色被 100% 地用于显著性平滑阶段。

基于上述分析, RPC 方法最终确定的 3 个参数的最优取值如表 2.3 所示。

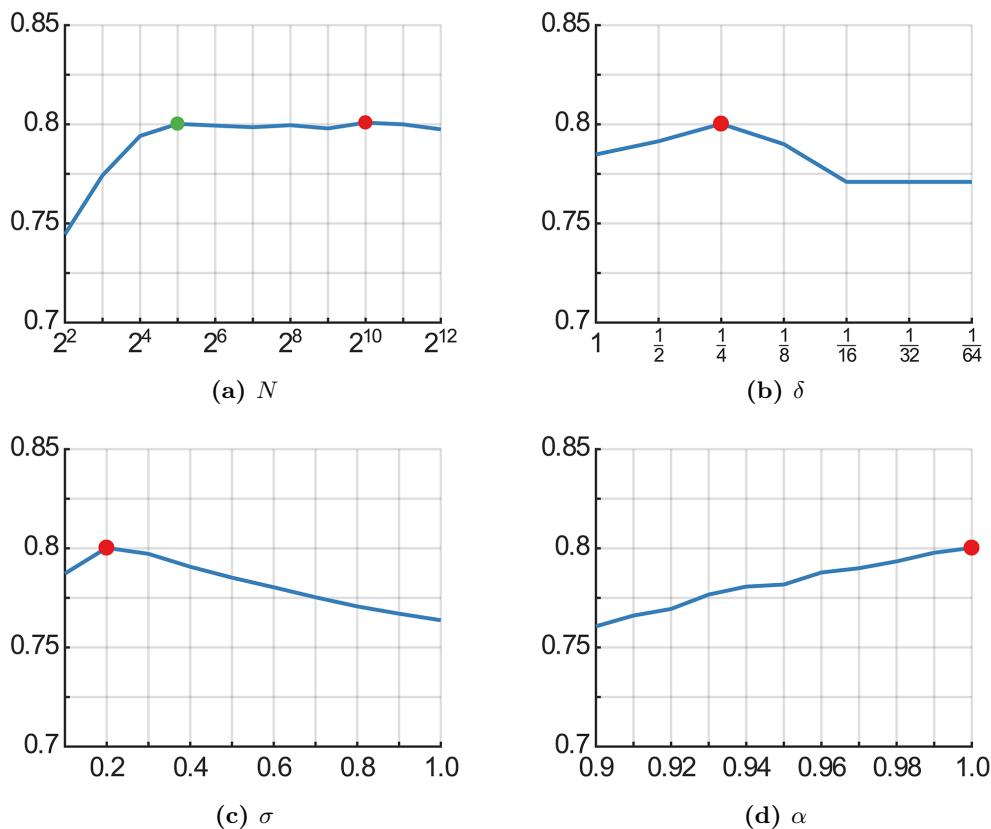


图 2.12 RPC 方法参数分析: x 轴表示参数取值范围, y 轴表示 MaxF 的值

表 2.2 RPC 方法参数分析

(a) N		(b) δ		(c) σ		(d) α	
N	MaxF	δ	MaxF	σ	MaxF	α	MaxF
$2^2 = 4$.7446	$1/2^0 = 1$.7848	0.1	.7873	0.90	.7606
$2^3 = 8$.7742	$1/2^1 = 1/2$.7915	<u>0.2</u>	<u>.8002</u>	0.91	.7661
$2^4 = 16$.7941	<u>$1/2^2 = 1/4$</u>	<u>.8002</u>	0.3	.7972	0.92	.7694
<u>$2^5 = 32$</u>	<u>.8002</u>	$1/2^3 = 1/8$.7900	0.4	.7907	0.93	.7767
$2^6 = 64$.7994	$1/2^4 = 1/16$.7710	0.5	.7852	0.94	.7807
$2^7 = 128$.7985	$1/2^5 = 1/32$.7710	0.6	.7803	0.95	.7817
$2^8 = 256$.7996	$1/2^6 = 1/64$.7710	0.7	.7753	0.96	.7878
$2^9 = 512$.7979			0.8	.7707	0.97	.7900
$2^{10} = 1024$	<u>.8008</u>			0.9	.7670	0.98	.7934
$2^{11} = 2048$.8000			1.0	.7637	0.99	.7978
$2^{12} = 4096$.7975					<u>1.0</u>	<u>.8002</u>

表 2.3 RPC 方法最优参数

参数名	最优参数值
N	32
δ	1/4
σ	0.2

2.5.3 量化方式和中心加权的影响

在第 2.4.1.1 和 2.4.3.2 节中，分别从理论层面探讨了量化方式和中心加权对 RPC 方法性能的影响，本节给出具体的定量评估。

2.5.3.1 量化方式的影响

在上一节参数分析中，已经评估了最小方差量化为不同颜色数后对 RPC 方法性能的影响。为了横向对比在全局颜色显著性计算过程中引入均匀量化和最小方差量化导致的性能差异，需要对 RPC 方法进行适当改造。具体实现是将最小方差量化模块替换为均匀量化模块，而之后的所有模块保持不变，这样就可以确保除量化方式外，两种显著性检测方法具有完全相同的处理流程。同时，为了让均匀量化获得与最小方差量化相同或相近的量化颜色数，这里将 RGB 空间中每个颜色通道均匀量化为 $k \in [3 : 16]$ 级，使得均匀量化后颜色数的理论范围在 27 (即 3^3) 色至 4096 (即 16^3) 色之间^①。

^①如第 2.4.1.1 节中所述，受不同输入图像所含颜色的影响，量化时的颜色空间划分通常有不包含任何颜色的小立方体存在，因此实际得到的颜色数一般要小于设定的量化颜色数。

表 2.4 量化方式及颜色数的影响

(a) 均匀量化					(b) 最小方差量化				
颜色数	AvgF	MaxF	AdpF	平均	颜色数	AvgF	MaxF	AdpF	平均
$3^3 = 27$.3986	.6113	.6250	.5450	$2^2 = 4$.5942	.7446	.7461	.6950
$4^3 = 64$.3989	.6344	.6441	.5592	$2^3 = 8$.5647	.7742	.7686	.7025
$5^3 = 125$.4137	.6707	.6740	.5861	$2^4 = 16$.5796	.7941	.7851	.7196
$6^3 = 216$.4315	.7022	.7056	.6131	$2^5 = 32$.5762	.8002	.7880	.7215
$7^3 = 343$.4502	.7236	.7247	.6328	$2^6 = 64$.5779	.7994	.7883	.7219
$8^3 = 512$.4610	.7357	.7369	.6445	$2^7 = 128$.5783	.7985	.7877	.7215
$9^3 = 729$.4719	.7477	.7494	.6563	$2^8 = 256$.5777	.7996	.7888	.7220
$10^3 = 1000$.4823	.7603	.7606	.6678	$2^9 = 512$.5732	.7979	.7889	.7200
$11^3 = 1331$.4926	.7673	.7668	.6755	$2^{10} = 1024$.5713	.8008	.7937	.7219
$12^3 = 1728$.4983	.7713	.7713	.6803	$2^{11} = 2048$.5635	.8000	.7939	.7191
$13^3 = 2197$.5084	.7763	.7758	.6868	$2^{12} = 4096$.5559	.7975	.7934	.7156
$14^3 = 2744$.5117	.7786	.7772	.6892					
$15^3 = 3375$.5148	.7815	.7805	.6923					
$16^3 = 4096$.5209	.7846	.7833	.6963					

表 2.4 给出了两种量化方式及在不同颜色数下的 F_β 相关性能指标:

• 首先要关注的是颜色数问题, 可以看到, 虽然由于量化方式的不同, 两者得到的颜色数不尽相同, 但在部分颜色数上却有一定的重叠。例如, 均匀量化中的 4^3 、 8^3 和 16^3 , 分别与最小方差量化中 2^6 、 2^9 和 2^{12} 所得到的颜色数相同, 而在其它量化级上, 两种量化方式也具有相近的颜色数。通过构造量化颜色数序列, 使得两种量化方式的性能比较具有一定的参考价值。

• 其次是各量化方式在不同颜色数下的变化趋势, 相应的性能变化曲线如图 2.13 所示, 图中包括固定阈值分割得到的 MaxF 指标, 以及自适应阈值分割得到的 AdpF 指标。可以看到, 对于均匀量化, 随着颜色数的不断增多, AvgF、MaxF 和 AdpF 三项指标性能快速提升, 当均匀量化颜色数为 4096 色时达到峰值, 此时 MaxF 指标值为 0.7846。理论上, 均匀量化得到的颜色数越多, 对原始图像颜色分布的还原就越准确, 从而可以获得更好的检测性能。反观最小方差量化, 其三项指标的变化并不呈现固定的上升趋势, 而是在缓慢上升到一定程度后, 呈现出小幅波动。对于最小方差量化的 MaxF 指标, 当量化为 1024 色时达到性能峰值 0.8008; 而对于 RPC 方法最终选定的量化颜色数 $N = 32$, 获得了次优结果 0.8002。

• 第三, 两种量化方式在颜色数区间的增幅也存在较大差异。均匀量化从 27 色到 4096 色的变化过程中, MaxF 指标上升了 17.33%; 而最小方差量化从 4 色变化为 4096 色时, MaxF 指标仅上升了 5.29%。这表明虽然随着颜色数的不断增多, 最小方差量化

导致的方法性能同样因其对图像原始颜色分布的刻画更为准确而不断提升，但由于该量化方式能在较少的颜色数时就很好地体现出这种颜色分布，因而其对性能提升的幅度不如均匀量化来得明显。

最后，是关键的两种量化方式之间的横向比较：

(1) 对于 AvgF 指标，最小方差量化在所有颜色数上都要高于均匀量化最高的 4096 色。最小方差量化在最小颜色数 4 时，要高出均匀量化 4096 色 7.33%；即便是指标最低的 4096 色，最小方差量化也要高出均匀量化 4096 色 3.5%。

(2) 从图 2.13(a) 的曲线中可以看到，对于 MaxF 指标，在相同颜色数下最小方差量化要始终优于均匀量化，当最小方差量化仅为 16 色时，已经超过了均匀量化 4096 色的结果。此外，指标最高的 1024 色以及 RPC 方法最终选定的参数值 32 色，要分别比均匀量化 4096 色高出 1.62% 和 1.56%。

(3) 同样，在 AdpF 指标上，图 2.13(b) 也展示了最小方差量化明显的性能优势。在最小的颜色数 4 上，指标已超过均匀量化的 512 色；而当量化为 32 色时，性能较均匀量化的最高指标即 4096 色高出 0.47%。

通过上述的实验和结果分析，不难看出最小方差量化的优势所在。实际上，由于 RPC 方法本质上检测的是全局的稀缺颜色，采用非均匀的颜色映射可以使 RGB 空间中原来高频数量颜色依然聚集得相对紧密，而低频数量颜色则保持了相对的稀缺性。同时，这种疏密分布可以通过更少的颜色数来刻画，这也正是 RPC 方法中使用最小方差量化取代 HC 方法^[29] 中均匀量化的原因所在。

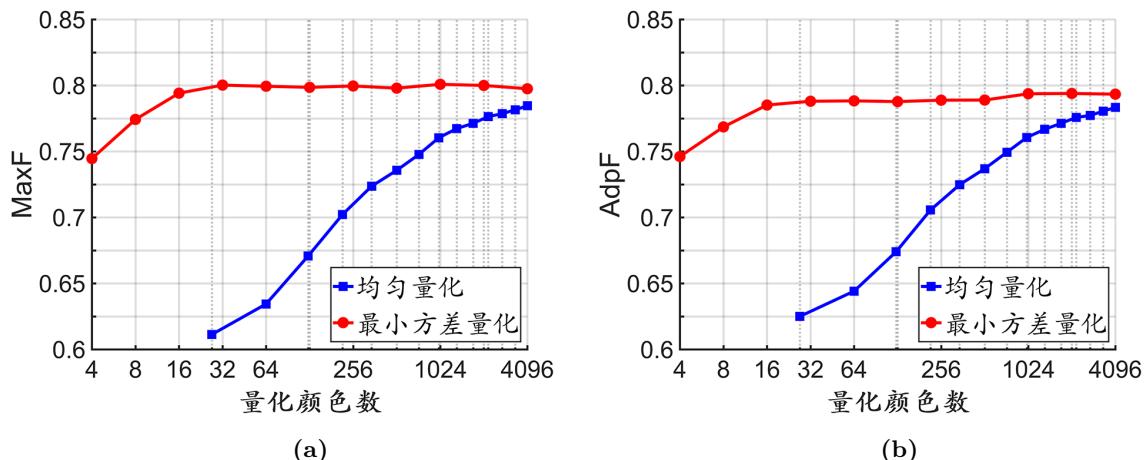


图 2.13 量化方式及颜色数的影响。(a) x 轴为量化颜色数，y 轴为 MaxF 指标值；(b) x 轴为量化颜色数，y 轴为 AdpF 指标值

2.5.3.2 中心加权的影响

在第 2.4.3.2 节中，RPC 方法利用区域与图像中心的空间关系，引入中心权重对基于区域主色得到的局部显著性进行加权。这直接导致了中心偏置现象，即检测出的显著

物体更聚集于图像的中心。在文献 [46] 中, 李健等人考虑到中心偏置因素的影响, 修正了后处理步骤, 通过设计最优中心偏置因子使 ROC 指标最大化。为了评估引入中心加权对 RPC 方法性能的影响, 这里简化这一过程, 通过直接去掉中心加权模块使新产生的方法不被中心偏置因素所干扰。

图 2.14 给出了去除中心加权前后的结果, 该结果通过采用第 2.5.2 节给出的最优参数计算得到。从 $P-R$ 曲线图中可以看到, RPC 方法的曲线将去除中心加权后的曲线完全包围其中, 表明前者性能优于后者。 F_β 曲线图中, RPC 方法的曲线峰值要高于去除中心加权后的峰值, 即前者的 MaxF 指标更优。在自适应阈值分割得到的 $P-R-F_\beta$ 柱状图中, RPC 方法具有更高的精确率和 F_β 值, 仅在召回率上低于去除中心加权后的结果。表 2.5 给出了具体的指标值, 可以看到, 由于 F_β 曲线更为平缓, 去除中心加权后的 AvgF 指标要高出 RPC 方法 5.93%, 但在 MaxF 和 AdpF 两项指标上, RPC 方法要分别高出 6.78% 和 7.18%。结果表明, 在 RPC 方法中引入中心加权, 有助于对检测性能的提升。

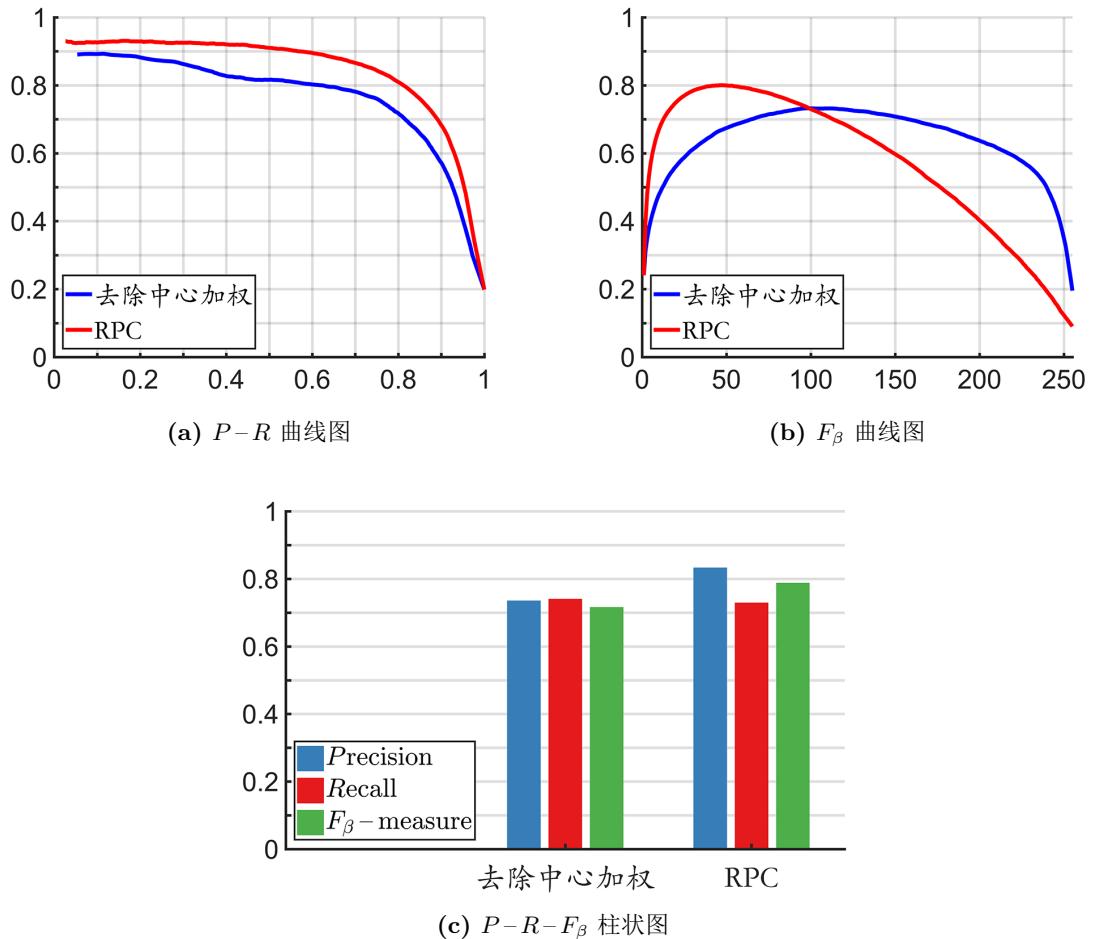


图 2.14 中心加权的影响。(a) $P-R$ 曲线图: x 轴表示召回率 R , y 轴表示精确率 P ; (b) F_β 曲线图: x 轴表示固定分割阈值 T_f , y 轴表示 F_β ; (c) 使用自适应分割阈值 T_a ^[32] 得到的 $P-R-F_\beta$ 柱状图: 以 F_β 值升序排列

表 2.5 中心加权的影响

方法	AvgF	MaxF	AdpF	平均
去除中心加权	.6355	.7324	.7162	.6947
RPC	.5762	.8002	.7880	.7215

实际上，引入中心加权的主要问题在于对非图像中心区域显著值的抑制，特别是当显著物体确实远离图像中心时，物体内部的显著值会受此影响而降低，但这种影响很大程度上还是取决于显著性检测方法自身的优劣。可以想见，只要显著性方法对显著物体区域检测得足够精确，那么即便因远离图像中心而使得显著值有所降低，最终的归一化操作也会将物体区域的显著值拉升到一个较高的水平。更重要的是，既然人类在观察自然场景时更倾向于定位在中心位置^[72,73]，同时显著性数据集在构建时往往也被主观地引入了偏置因素^[120]（见第 2.5.1、3.5.1 和 4.5.1 节），那么在引入中心加权后，无论是从视觉注意机制的角度还是从实验结果来看，都表明其可以有效帮助提升 RPC 方法的检测性能。

2.5.4 与其它显著性方法的性能比较

将 RPC 方法与其它 24 个显著性方法进行性能比较时，利用了已在第 1.2.4.3 节中介绍过的性能度量指标。在计算各方法在 ASD 数据集^[9,32] 上的精确率 P 、召回率 R 以及 F_β 度量之前，统一将所有显著图归一化到 $[0, 255]$ 区间。实验中各方法的结果显著图来源如下：

- 对于 AC^[108]、CA^[70]、FT^[32]、GB^[91]、HC^[29]、IT^[7]、LC^[92]、MZ^[55]、RC^[29] 和 SR^[37] 这 10 个方法，直接使用文献^[29] 提供的显著图①；
- 对于 HFT^[46] 方法，通过运行原作者发布的 MATLAB 代码生成显著图②；
- 对于参与比较的其它方法，则从 MSRA10K 数据集^[8,28,121,122] 中检索出所需的 1,000 张显著图③。

图 2.15 给出了基于固定分割阈值 T_f 得到的 $P-R$ 曲线图和 F_β 曲线图，以及基于自适应分割阈值 T_a ^[32] 得到的 $P-R-F_\beta$ 柱状图，性能分析如下：

(1) 对于 $P-R$ 曲线：RPC 方法覆盖了参与比较的大部分方法，在召回率 R 区间的前半段也获得了最好的性能。但在召回率 R 区间的后半段（即 $R \in (0.8, 1]$ 之间），精确率 P 要低于 RC 和 SVO 方法。

(2) 对于 F_β 曲线：RPC 方法获得了最大峰值，这意味着如果所有方法都基于峰值

① 下载网址：<http://cg.cs.tsinghua.edu.cn/people/~cmm/Saliency/Index.htm>。

② 代码网址：<http://www.escience.cn/people/jianli/publications.html>。

③ MSRA10K^[8,28,121,122] 是由程明等人发布的 MSRA 数据集^[9] 人工标注结果，共包括 10,000 张像素级标注图，其中含 ASD 数据集的 1,000 张图像，即 ASD 是 MSRA10K 的子集。除人工标注数据外，MSRA10K 还提供了部分显著性方法的结果显著图。本节实验所需显著图，是通过直接在 MSRA10K 中检索 ASD 子集得到。MSRA10K 数据集下载网址：<http://mmcheng.net/msra10k/>。

点阈值进行分割的话，RPC 方法可以得到最好的分割结果。但在阈值 100 以后，RPC 方法性能下降较为明显，相对而言，HC 方法曲线的中段更为平坦。

(3) 在 $P-R-F_\beta$ 柱状图中，RPC 方法在 P 、 R 和 F_β 三个度量上都获得了最高指标。在指标 P 上，RPC 略高于排名第二和第三的 RC 和 SEG 方法，分别高出 0.53% 和 6.26%；但在指标 R 上，则分别高出这两个方法 14.67% 和 36.45%，这意味着 RPC 方法的检测结果更能完整覆盖人工标注的显著物体区域。在指标 R 上，紧随 RPC 的是 HC 和 MSS 方法，RPC 方法分别高出这两个方法 1.56% 和 4.44%，同时在指标 P 上也超过了这两个方法 10.22% 和 5.05%。而对于综合性能的 F_β 指标，RPC 要比第二和第三名的 MSS 和 HC 方法分别高出 5.11% 和 3.6%。

对性能指标定量的评估结果见表 2.6，表中包括 AvgF、MaxF、AdpF 三项指标及均值，各项指标的前三名分别使用红色、绿色、蓝色标记：

(1) 对于关键的 MaxF 指标：RPC 方法排名第一，紧随其后的是 SVO 和 RC 方法，RPC 方法分别高出这两个方法 2.42% 和 4.32%。

(2) 对于 AdpF 指标：RPC 方法依然排名第一，紧随其后的是 MSS 和 HC 方法，RPC 方法分别高出这两个方法 5.11% 和 8.71%。

(3) 对于 AvgF 指标：RPC 方法仅排在第二，低于第一名的 HC 方法 3.51%，但要比第三名的 RC 方法高出 5.7%。AvgF 指标高，表明 HC 方法的 F_β 曲线整体上要比 RPC 平坦。在第 1.2.4.3 节中已讨论过，更为平坦的 F_β 曲线实际上更有利于图形–背景分离。虽然 RPC 对比 HC 方法在 AvgF 指标上有一定劣势，但在 MaxF 和 AdpF 两项指标上要分别高出 7.47% 和 8.71%。

(4) 三个指标平均而言，PRC 方法排名第一，比第二和第三名的 HC 和 RC 方法分别高出 4.22% 和 6.92%。

最后是不同显著性方法检测结果在视觉上的比较，这里从 24 个横向对比的显著性方法中选取出了 8 个方法。图 2.16 给出了 RPC 方法与这 8 个方法对 10 幅输入图像计算得到的显著图，其中图 2.16(c) 是 RPC 方法的检测结果。从显著图结果来看，RPC、HC (图 2.16(d))、RC (图 2.16(e)) 这三个方法对显著物体区域的检测更为精确。特别是 HC 方法，除了在第六个输入样本上错误地将显著区域定位到图像左边的白色云朵外，该方法能够在凸显物体区域的同时，使物体内部具有更为一致的显著值。然而，HC 和 RC 方法对于背景区域的抑制程度不够，导致生成的显著图与 RPC 相比要更为杂乱，相同的情况也出现在 SVO 方法上 (图 2.16(k))。另一个对显著物体区域检测较为准确的是 FT 方法 (图 2.16(h))，可以看到该方法产生的显著图能较好地保持显著物体区域的轮廓。但是，除了在部分示例图像上同样存在背景抑制不足的问题，FT 方法的一个缺点在于没有能够整体上凸显物体区域。相较而言，RPC 方法在突出整个显著物体区域并使其具有相对一致的显著值方面，有更大的优势。

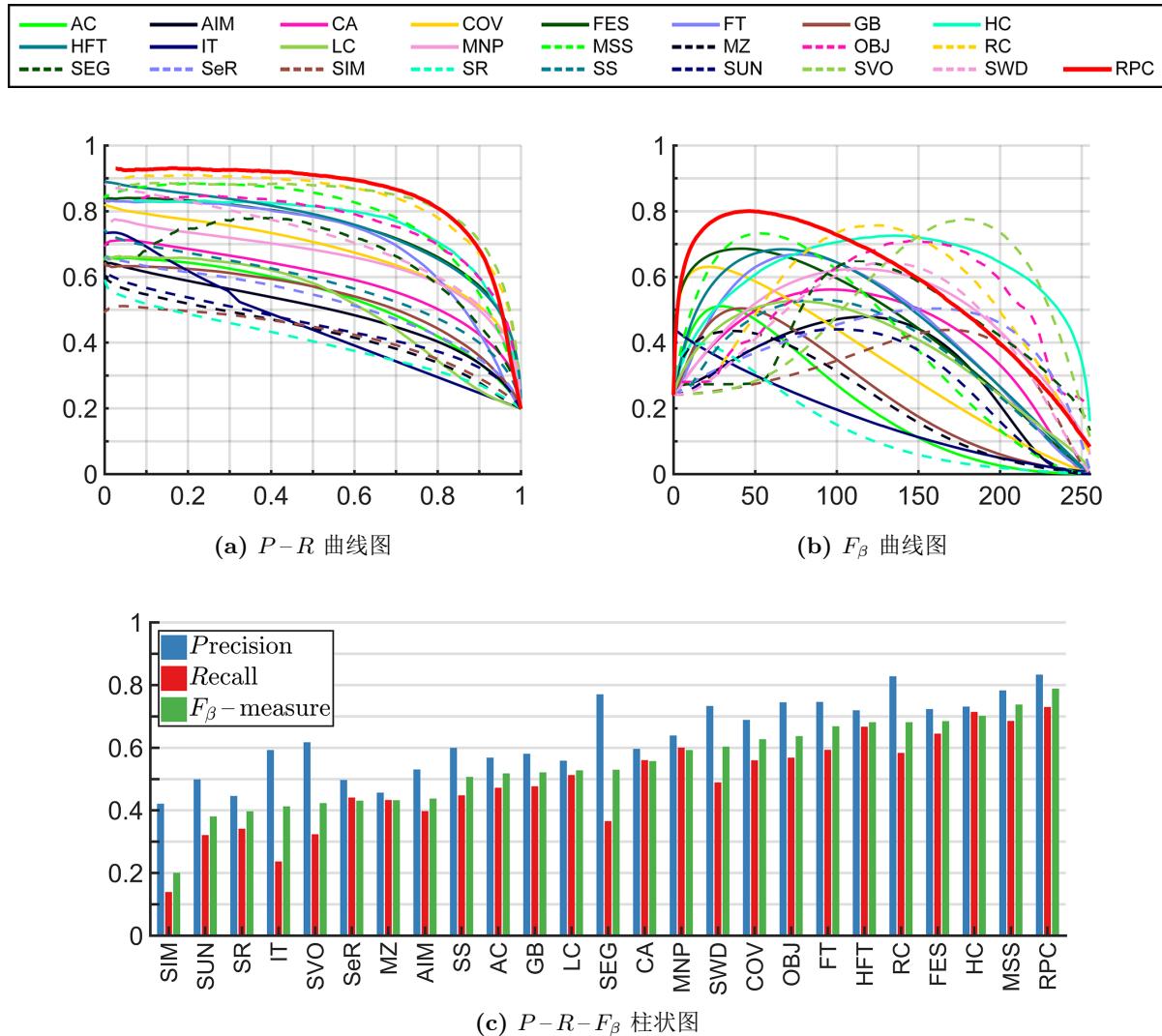


图 2.15 与 24 个显著性方法的性能比较。(a) $P-R$ 曲线图: x 轴表示召回率 R , y 轴表示精确率 P ; (b) F_β 曲线图: x 轴表示固定分割阈值 T_f , y 轴表示 F_β ; (c) 使用自适应分割阈值 T_a [32] 计算得到的 $P-R-F_\beta$ 柱状图: 以 F_β 值升序排列

表 2.6 与 24 个显著性方法的性能比较

序号	方法	AvgF	MaxF	AdpF	平均
1	AC ^[108]	.2139	.5107	.5174	.4140
2	AIM ^[109]	.3201	.4784	.4370	.4118
3	CA ^[70]	.4043	.5615	.5569	.5076
4	COV ^[110]	.3413	.6305	.6264	.5327
5	FES ^[111]	.4484	.6859	.6840	.6061
6	FT ^[32]	.4342	.6681	.6677	.5900
7	GB ^[91]	.2478	.5041	.5207	.4242
8	HC ^[29]	.6113	.7255	.7009	.6793
9	HFT ^[46]	.4526	.6839	.6806	.6057
10	IT ^[7]	.1701	.4357	.4118	.3392
11	LC ^[92]	.3678	.5268	.5272	.4739
12	MNP ^[17]	.4734	.6246	.5919	.5633
13	MSS ^[112]	.4116	.7321	.7369	.6269
14	MZ ^[55]	.2229	.4354	.4318	.3633
15	OBJ ^[49]	.5139	.7095	.6367	.6200
16	RC ^[29]	.5192	.7570	.6809	.6523
17	SEG ^[113]	.4305	.6485	.5288	.5359
18	SeR ^[114]	.3975	.5037	.4300	.4437
19	SIM ^[115]	.3162	.4384	.2002	.3183
20	SR ^[37]	.1435	.3964	.3964	.3121
21	SS ^[116]	.3581	.5308	.5062	.4651
22	SUN ^[117]	.2916	.4402	.3803	.3707
23	SVO ^[118]	.4966	.7760	.4227	.5651
24	SWD ^[119]	.4399	.6434	.6033	.5622
25	RPC	.5762	.8002	.7880	.7215
平均		.3841	.5939	.5466	.5082

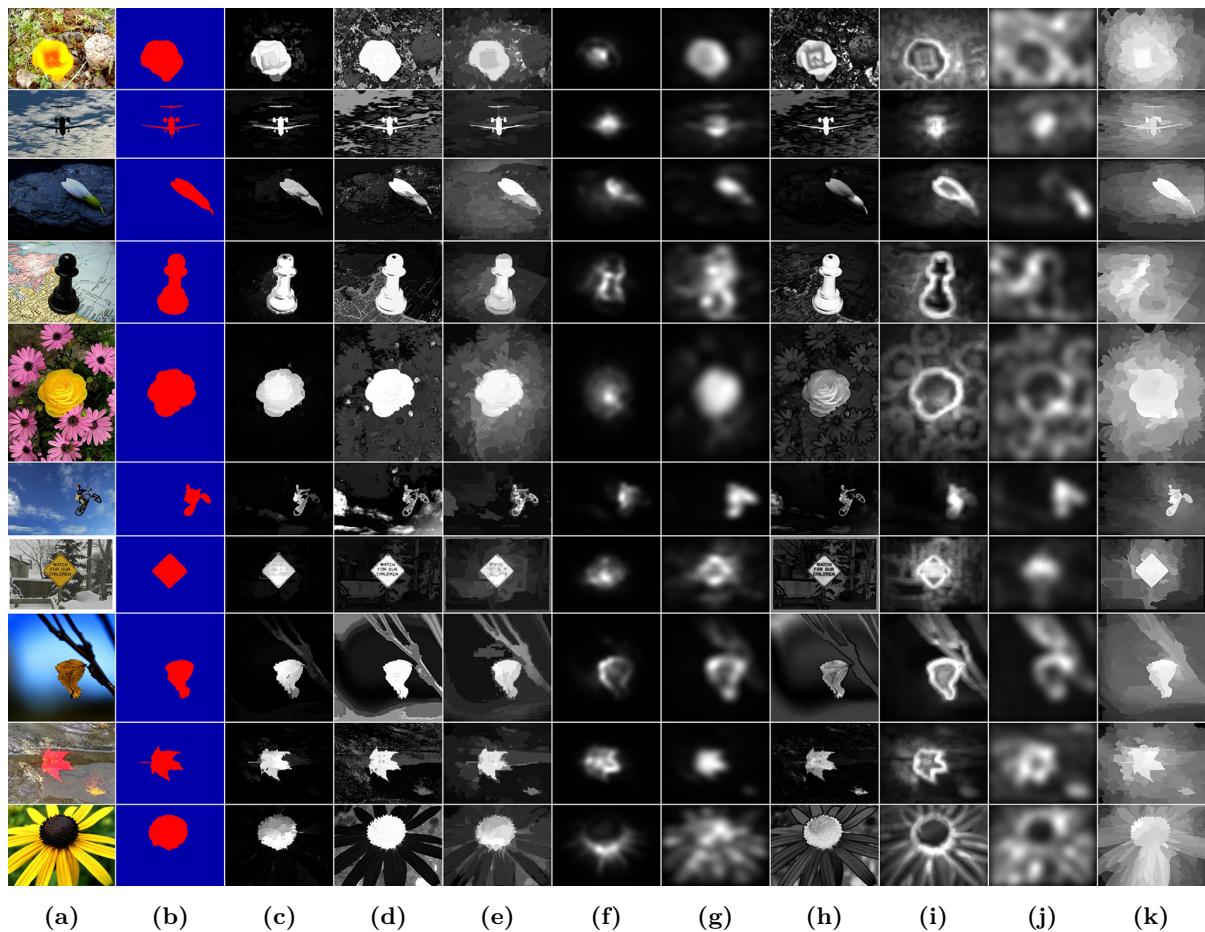


图 2.16 与 8 个显著性方法的视觉比较。(a) 输入图像^[9]; (b) 人工标注结果^[32]; (c)–(k) 是分别使用不同方法得到的显著图: (c) 本章提出的 RPC 方法; (d) HC^[29]; (e) RC^[29]; (f) COV^[110]; (g) HFT^[46]; (h) FT^[32]; (i) MNP^[17]; (j) SS^[116]; (k) SVO^[118]

2.5.5 实验结果讨论

虽然通过定量的性能评估和对比分析, 表明了本章提出的 RPC 方法对显著物体检测的有效性, 但该方法仍然存有一定缺陷。为了实际展示失败的检测结果, 对每一幅由 RPC 方法产生的显著图计算 MAE 指标 (定义见第 1.2.4.2 节), 并对所有 MAE 值从大到小排序, 从中挑选出排在前列的具有代表性的图像。显然, MAE 指标值越大, 表明 RPC 方法产生的显著图越偏离人工标注结果。

图 2.17 中, 第 1 行和第 2 行是示例图像及人工标注图, 从第 3 行往下, 依次是最小方差量化为 32 色的颜色直方图、区域主色填充图、全局显著图、以及 RPC 方法的检测结果。可以看到, RPC 方法要么得到了错误的显著物体区域 (图 2.17(a) 和 2.17(b)), 要么仅检测出显著物体内的部分区域 (图 2.17(c) – 2.17(e))。由于 RPC 方法是在全局显著性基础上计算局部显著性, 因此当全局显著性产生错误结果时, 将直接导致 RPC 方法检测的失效。

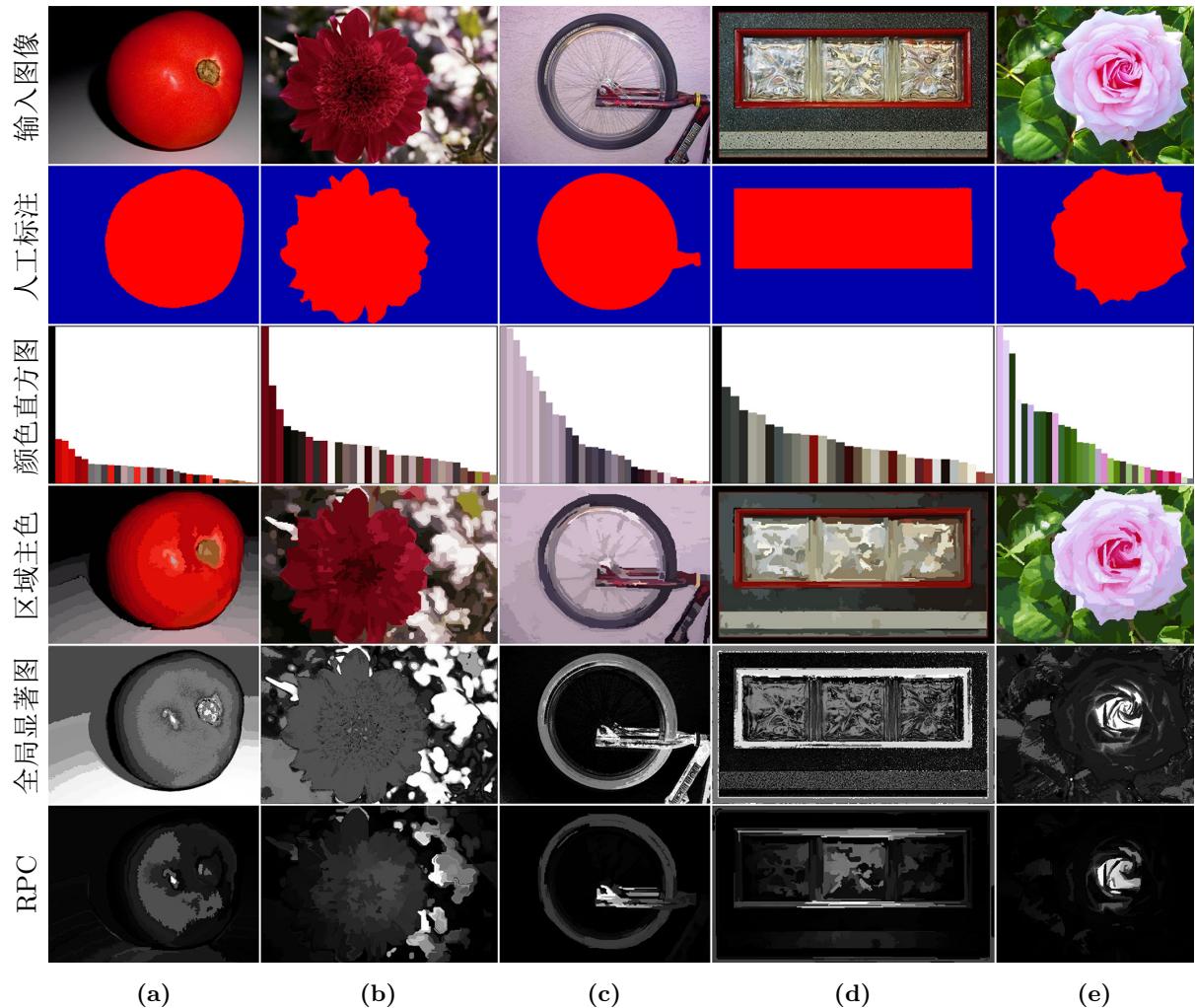


图 2.17 检测失败示例

实际上，要使基于颜色稀缺性的全局显著性方法得到理想的检测结果，显著物体所包含的颜色应是场景中的低频数量颜色，同时与高频数量颜色之间应具有较大的颜色差异。然而从图 2.17 第 3 行的颜色直方图中可以看到，上述检测失败的示例中，显著物体所包含的颜色显然不属于场景中的低频数量颜色。例如，图 2.17(a) 中的显著区域属于次高频率颜色，图 2.17(b)、2.17(c) 和 2.17(e) 中的显著区域则都属于高频率颜色，这使得图像中的背景因颜色的低频数量特性被凸显出来，从而抑制了物体区域的显著性（仅有图 2.17(a) 和 2.17(e) 的物体内部中心区域因属于低频数量颜色而获得了较高的显著值）。此外，图 2.17(c) 和 2.17(d) 中的黑色轮胎和红色边框都属于低频数量颜色，从全局显著图中也可以看到，虽然全局显著性方法已将这两个区域正确地检测了出来，但由于人工标注时将它们的内部区域也标记为显著区域，故同样导致了检测的失效。

虽然上述示例表明，当显著物体区域包含的是图像中的高频率颜色时，基于全局颜色稀缺性的 RPC 方法会错误地定位显著区域。但一个值得注意的现象是，无论物体内部的颜色在场景中的比重如何，这些显著物体区域本身都具有完整闭合的外部轮廓，

使得它们能与背景区域相互分离。而人类在观察此类图像时，往往也是通过这种闭合性将物体从场景中分离出来。因此，这意味着可以在颜色线索的基础上引入新的相关线索，引导显著性检测方法重新回到对“物体”本身的审视上来。在下一章中，将提出一种融合颜色线索和环绕性线索的显著物体检测方法，实验结果也表明了该方法对比 RPC 方法在检测性能上的提升。

2.6 本章小结

本章采用区域主色以及区域空间关系，提出了用于单张图像显著物体检测的 RPC 方法。为了降低计算全局显著性时的算法复杂度，首先对原始图像采用最小方差量化将颜色数减少为 32 色，并基于全局颜色差异计算颜色全局显著值；然后对量化图像采用超像素分割，并对分割得到的每个局部区域仅保留区域主色，进一步减少局部显著性计算时区域之间的颜色比较次数；最后，计算区域间距离以及区域与图像中心间距离，并利用这两种空间关系对局部显著性加权，得到最终的显著图。实验结果表明，采用最小方差量化可在更少的颜色数下获得更好的检测性能并降低了算法复杂度，而采用的区域空间关系可在抑制非显著区域的同时，提升中心区域的显著值。

由于本章提出的 RPC 方法是一种自底向上的计算显著性模型，且仅利用了底层的颜色特征，因此当显著物体包含的是场景中的高频数量颜色时，将导致 RPC 方法的失效。同时，显著物体区域的显著值不够突出，也限制了后续对显著物体的分割操作。针对上述问题，采取的解决措施如下：

- (1) 拟在颜色对比模型的基础上，引入物体区域的环绕性特征，通过图形–背景分离实现显著物体检测。在第 3 章中，将具体介绍这一工作。
- (2) 拟引入包含亮度变换在内的后处理操作，以突出显著物体区域，使输出的显著图更接近于二值结果，第 3 章就包含了后处理操作的具体实现。
- (3) 由于 RPC 方法仅在单个尺度上进行检测，故对于较小尺寸的显著区域，可能因在超像素分割时被并入邻近区域而导致计算区域主色时被直接忽略。解决这一问题的常见手段是采用多尺度或分层结构，在第 4 章中，将采用分层结构实现协同显著物体检测中对单图显著图的计算。

3 基于颜色名空间的显著物体检测

3.1 引言

第2章提出的RPC方法围绕图像底层颜色特征实现显著物体检测，其中的颜色量化以及颜色间的差异度量，分别在RGB和LAB空间中完成。从第2.5.5节的实验结果讨论中可以看到，当显著物体颜色是场景中的高频数量颜色时，将导致RPC方法的失效。但是，图2.17显示出这些检测失败的显著物体都具有完整封闭的外部轮廓，即具有环绕性特征。因此，本章从图像底层颜色线索出发，研究自底向上的、融合环绕性线索的显著物体检测方法，第2章中采用的基于稀缺性的全局颜色比较，在本章工作中同样有所运用。

事实上，如果重新审视显著物体检测任务，从其字面中很容易解读出两层含义：一是对显著性的检测，即检测出的区域应该是场景中的显著区域；二是对物体的检测，即检测出的显著区域应该包含潜在的任意类别的物体。格式塔(Gestalt)心理学的研究结果表明，位于前景区域的物体(图形)相较于背景会更加显著^[123,124]。虽然有心理学家认为图形与背景中存在着一些外源性的因素，会影响到人类对场景中图形或背景区域的分配^[125]，但也有学者认为这种下意识的过程往往处于前注意阶段^[126]，而图形与背景的分离可以不需要视觉注意的介入^[127]。

既然显著物体更有可能位于前景区域中，随之而来的两个问题就是：1) 如何能从整个场景中抽取出前景区域？因为其可能是潜在的物体区域；2) 采用何种技术手段来计算这些前景区域的显著性？因为其确定了所抽取出的前景区域是否显著。显然，这两个问题与显著物体检测任务的两层字面含义紧密相关。

对于第一个问题，一种有效的解决方法是基于环绕性线索的图形–背景分离^[75,128]。然而，在下一节的讨论中将会看到，基于环绕性线索的检测方法只考虑了场景结构信息，却忽略了颜色特征在计算显著性时所起的重要作用。因此，本章采用将环绕性线索与两个全局的颜色名线索相融合，在图形–背景分离的同时，解决上述第二个对显著性的计算问题。与第2章在RGB空间进行量化、在LAB空间中进行度量不同，本章采用的颜色空间是基于颜色命名模型^[129,130]生成的颜色名空间。在对颜色名空间中各颜色名通道进行图形–背景分离计算出显著图后，进一步采用全局的颜色名线索计算加权显著图，最后采用高亮截断的融合操作和后处理操作，得到显著物体检测结果。本章工作的创新点为：在颜色名空间的基础上，提出了一种融合环绕性线索和全局颜色名线索的集成框架，可有效弥补仅基于环绕性线索检测显著性的不足，同时提出了基于统计直方图的后处理算法，可有效抑制图像背景，并在突出显著物体的同时，使物体区域获得均匀且平滑的显著值。

为方便起见，本章提出的基于颜色名空间 (Color Name Space) 的显著物体检测方法，简记为“CNS”方法^①。本章内容安排如下：第 3.2 节介绍研究动机与出发点；第 3.3 节介绍 CNS 方法框架；第 3.4 节详细介绍 CNS 方法，包括基于颜色名通道的显著性算法、基于全局颜色名线索的显著性加权算法、以及基于高亮截断和后处理的显著性融合算法；第 3.5 节是数据集介绍、参数分析、以及实验结果和结果分析；最后是本章小结和未来的工作展望。

3.2 研究动机与出发点

格式塔心理学对图形–背景感知的相关研究表明，场景中具有特定特征的区域将更可能以图形的方式呈现，如底部区域^[131]、小区域^[123]、具有较高空间频率的区域^[132]、凸区域^[133] 和对称区域^[134] 等。遵循这些建议，文献 [75] 利用环绕性 (Surroundedness) 特征^[128] 来解决前景图形的抽取问题。由于使用了一组序列化采样的布尔图对图形区域进行建模，该方法也被称为基于布尔图的显著性检测方法 (Boolean Map based Saliency，以下简记为 BMS)。BMS 方法首先示例了基于稀缺特征对比的显著性方法的缺点，指出这类方法往往忽略了全局结构信息，并通常将高显著值错误地定位在图像中具有高对比度的区域 (如两个特征差异明显的区域交界处)。在把输入图像 I 从 RGB 空间转换到 LAB 空间后，BMS 方法将包括三个通道以及各自的反相图在内的所有通道图作为 I 的特征图 $\phi(I)$ ，并使用一组序列化的阈值分割得到布尔图

$$B_i = \text{THRESH}(\phi(I), \theta), \quad (3.1)$$

其中， $\phi(I)$ 是被归一化到 $[0, 255]$ 整数区间的特征图， $\theta \in [0, 255]$ 是分割阈值。对于 $\phi(I)$ 中的任意元素，如果值大于 θ ，**THRESH** 函数将把 B_i 中对应位置的值设为 1，否则设为 0。随后，BMS 方法利用一系列的形态学操作将布尔图 B_i 中的贴边区域 (即潜在的背景区域) 去除，生成对应的注意图 (Attention Map)。最后，所有的注意图被线性平均，并经过后处理得到结果显著图。此外，使用不同的参数设置，BMS 方法可同时适用于显著物体检测和人眼视点预测两项任务。

虽然 BMS 方法简单直接，也获得了很好的检测效果，但还是有一定缺陷，图 3.1 给出了相关示例。其中，图 3.1(a)、3.1(c)、3.1(e) 和 3.1(f) 的第一行，都是来自于 ECSSD 数据集^[60,61] 的原始图像，第二行和第三行分别是 BMS 方法和本章提出的 CNS 方法的检测结果。首先，对于图 3.1(a)，相较于 CNS，虽然 BMS 方法在显著区域的整体平滑性和对背景的抑制上略有不足，但检测出了正确的显著物体，其判断最显著的区域是在斗牛犬的身体部分。有趣的是，如果把图 3.1(a) 转换到 LAB 颜色空间，且仅提取 L 通道并归一化到 $[0, 255]$ 区间 (见图 3.1(b))，BMS 方法检测出的最显著区域仍然是斗牛犬的身体部分。实际上，在抛弃颜色信息后， L 通道虽然只保留了原图像中的亮度信息，但由于斗牛犬身体区域与周围背景之间存在一定的亮度差异，故基于环绕性线索的

^① CNS 方法的 MATLAB 源代码和显著图等资源，将发布在论文主页：<http://www.loujing.com/cns-sod/>。

BMS 方法检测出了较为一致的显著区域。但是，仅使用 L 通道却给依赖于颜色特征的 CNS 方法带来了很大困扰，当斗牛犬与背景之间不存在颜色差异时，CNS 方法产生了截然不同、甚至完全错误的检测结果。

另一个是与上面相反的示例。对于图 3.1(c)，由于负鼠颈部较偏棕色，在引入颜色名空间时与底部的鹅卵石颜色相近，导致 CNS 方法检测结果中对颈部显著区域的丢失，而 BMS 方法在该图上获得了更好的检测结果。但如果同样取 L 通道（见图 3.1(d)），则 BMS 获得了近乎一致的检测结果，其判断出的最显著区域同样是具有更高亮度的负鼠头部。相反，由于 L 通道中整个负鼠区域与深色背景之间有着明显的亮度差异，使得颜色名空间中大致只有黑、白、灰三种颜色，区分度较高，CNS 方法反而取得了更好的检测结果。

综上，虽然 BMS 方法同时使用了 LAB 颜色空间的三个通道，但更多的是利用了亮度通道的环绕性特征，关键的颜色线索特别是颜色之间的差异线索，并没有过多地参与到计算中。当显著物体区域与邻域背景之间具有相似的亮度值时，BMS 方法容易对显著区域的定位产生混淆（见图 3.1(e)）、甚至是错误的检测结果（见图 3.1(f)）。此外，BMS 方法存在的另一问题是针对如图 3.1(g) 所示的棋盘格图像，其检测结果中仅将贴边的非环绕的方格去除，而留下了所有带有环绕性特征的内部方格。但显然，人在观察该图像时，通常会随机地在棋盘格间扫视，而视点并不会停留在特定的方格上。可以看到，同样是在使用默认参数的情况下，CNS 方法得到了一张纯黑的显著图，也就是说，CNS 方法认为原图中不存在任何的显著区域。

针对上述问题，本章提出的 CNS 方法将在环绕性线索的基础上引入更多的颜色线索，并由颜色命名模型^[129,130]生成的颜色名空间来取代 BMS 方法所使用的 LAB 空间。除了直接利用颜色名空间来产生布尔图外，CNS 方法还同时采用两个与颜色名相关的全局线索，并与环绕性线索相融合对显著性进行加权。此外，CNS 方法还基于统计直方图进行后处理操作，以有效抑制图像背景并突出显著物体区域。

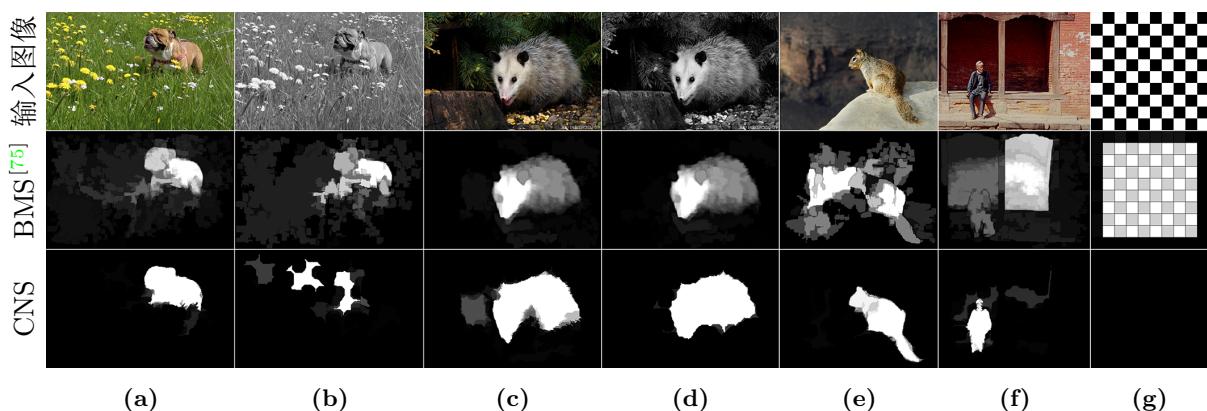


图 3.1 基于环绕性线索的检测示例。(a) 示例图像 1；(b) 示例图像 1 的 L 通道；(c) 示例图像 2；(d) 示例图像 2 的 L 通道；(e) 示例图像 3；(f) 示例图像 4；(g) 棋盘格图像

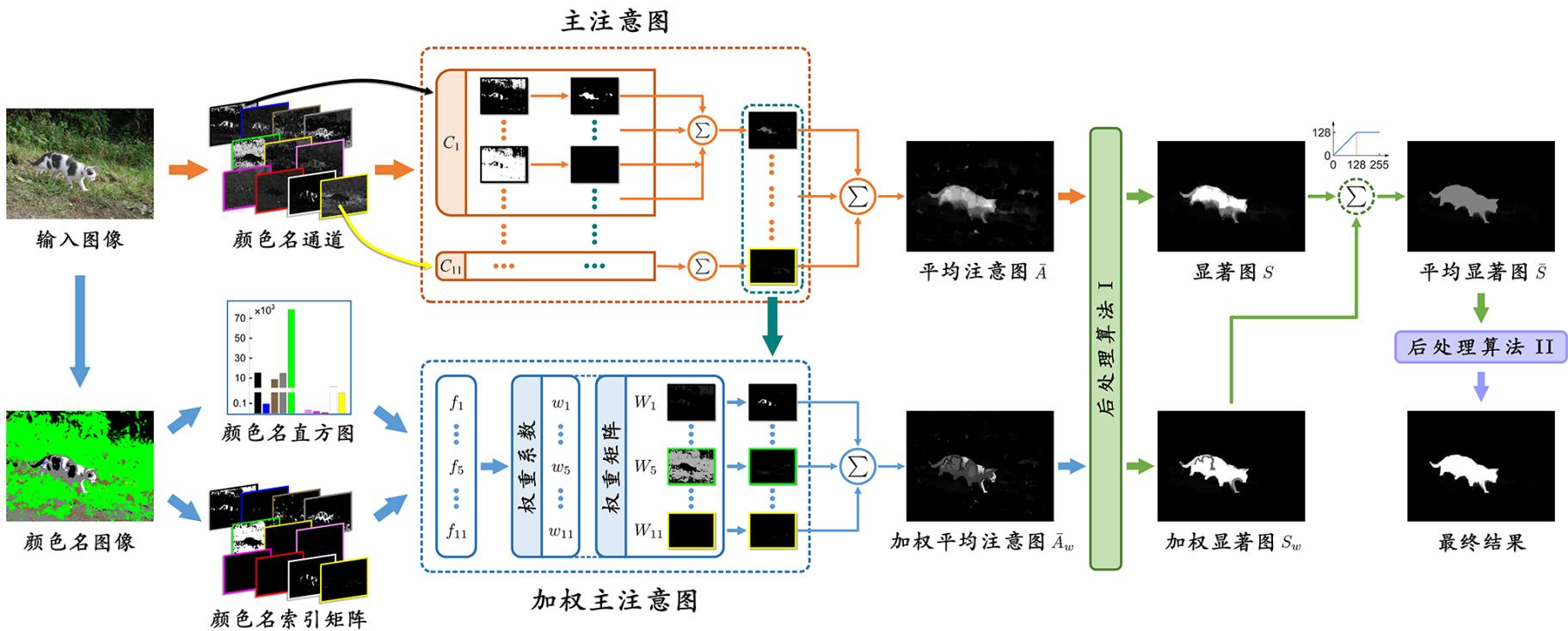


图 3.2 CNS 方法框架

3.3 方法框架

CNS 方法框架如图 3.2 所示，其中的两条流水线分别产生显著图 S 和加权显著图 S_w ，融合模块对这两个显著图采用高亮截断的融合操作与后处理操作，得到最终检测结果，具体如下：

- **流水线 I—显著性计算**

在流水线 I 中，每张输入图像被统一缩放到 400 个像素宽度，并从 RGB 颜色空间转换到颜色名空间。这里的颜色名空间，是由 11 个单色的强度图所构成，本文称为“颜色名通道”(Color Name Channel)，各通道的元素值被归一化到 [0, 255] 区间。随后基于环绕性线索对每个颜色名通道生成一组注意图，并对每个通道的注意图采用线性融合得到该通道对应的“主注意图”(Master Attention Map)。最后，将 11 个主注意图线性融合，得到平均注意图 \bar{A} ，并采用后处理算法 I 得到第一条流水线的显著图 S 。

- **流水线 II—显著性加权**

在流水线 II 中，首先将缩放到 400 个像素宽度的 RGB 图像转换为颜色名图像(Color Name Image)，并从中计算出颜色名直方图和颜色名索引矩阵。随后，基于颜色名直方图和索引矩阵计算全局颜色名统计特征以及对比特征，并与流水线 I 中的 11 个主注意图相融合，得到 11 个“加权主注意图”(Weighted Master Attention Map)。最后，采用与流水线 I 中相同的线性融合和后处理算法，在加权平均注意图 \bar{A}_w 的基础上得到第二条流水线的加权显著图 S_w 。

- **显著性融合**

在融合阶段，对上述两条流水线产生的显著图 S 和加权显著图 S_w ，采用高亮截断的融合操作得到平均显著图 \bar{S} ，并对 \bar{S} 采用后处理算法 II 得到最终检测结果。可以看到，结果中的显著物体具有近似均匀的显著值，同时背景部分的显著值几乎被抑制为 0，这种形式的显著图会给后续的显著物体分割带来极大的便利。

3.4 基于颜色名空间的显著物体检测方法

3.4.1 基于颜色名通道和环绕性线索的注意图生成

本章提出的 CNS 方法使用的颜色名空间，源自基于概率潜语义分析(Probabilistic Latent Semantic Analysis，简记为 PLSA)的颜色命名模型^[129,130]。从语言学的角度，人类对真实世界中的颜色进行描述时，并不会使用 RGB 颜色空间中的数值，而是换以一些语言学上的颜色术语，如亮黑色、黑色、银色、金色、玫瑰金色等，这些颜色术语被称为颜色名(Color Name)。有语言学的研究指出^[135]，英语语言中共包含 11 个基本的颜色名(见表 3.1)，颜色命名模型的任务就是建立起 RGB 颜色值与颜色名之间的对应映射关系。

表 3.1 英语中 11 个基本的颜色名

i	颜色名 (t_i)	中文名	RGB 颜色值 (c_i)		
			R	G	B
1	Black	黑	0	0	0
2	Blue	蓝	0	0	1
3	Brown	棕	.5	.4	.25
4	Grey	灰	.5	.5	.5
5	Green	绿	0	1	0
6	Orange	橙	1	.8	0
7	Pink	粉红	1	.5	1
8	Purple	紫	1	0	1
9	Red	红	1	0	0
10	White	白	1	1	1
11	Yellow	黄	1	1	0

在文档分析领域, PLSA 模型^[136]常被用于对一组文档中所包含的潜在主题进行建模。假设有一组文档 $D = \{d_1, \dots, d_N\}$, 包含的词表 $W = \{w_1, \dots, w_M\}$, 这些词由潜在的主题集合 $Z = \{z_1, \dots, z_K\}$ 生成, PLSA 模型将一个单词 w 出现在一个文档 d 中的条件概率建模为

$$p(w|d) = \sum_{z \in Z} p(w|z)p(z|d), \quad (3.2)$$

其中, 条件概率 $p(w|z)$ 和 $p(z|d)$ 可以由期望最大化 (Expectation-Maximization, 简记为 EM) 算法估计得到^[136]。如果把图像看作文档, 像素或小的图像区域看作是词, 那么该模型就可以被应用于诸如场景建模^[137] 和区域分类^[138] 等计算机视觉任务中。

不同于基于颜色卡的颜色命名方法^[139-141], J. van de Weijer 等人^[129,130]通过在 Google 搜索引擎中输入 11 个颜色名作为关键词, 然后利用改进的 PLSA 模型从这些采集到的带标签的样本图像中学习出颜色名。在早期的工作中^[129], 他们为每个颜色名各采集 100 张图像 (即 $N = 11 \times 100 = 1100$), 之后这一数量被扩充到每个颜色名 200 张图像^[130]①。为了利用 PLSA 模型解决颜色命名问题, 每一幅参与训练的图像被视作一个文档 d , 每个颜色名被视作一个主题 z , 主题集合中共包含 11 个主题即 $K = 11$ 。而对于词 w , 在将原图像经过系数为 2.4 的 Gamma 校正并转换到 LAB 空间后, 三个通道分别被等间隔地划分为 10、20、20 的大小, 然后分别统计 $10 \times 20 \times 20$ 个单元格内的像素个数构造出颜色直方图, 每个颜色柱为一个词, 即词表大小 $M = 4000$ 。

式 (3.2) 可以理解为将矩阵 $p(w|d)$ 分解为 $p(w|z)$ 和 $p(z|d)$, 矩阵 $p(w|d)$ 的大小为 $M \times N$ (行数 \times 列数), 其中每一列是一幅样本图像 d 的颜色直方图。矩阵 $p(w|z)$ 的

① 该文献中, 作者评价了对每个颜色名分别使用 10、25、50、100、150、200 幅图像训练后的模型性能, 结果表明对每个颜色名使用 200 幅图像时性能最优。

大小为 $M \times K$, 其中每一列是对应颜色名 d 的颜色空间分布, 也是模型需要最终估计的结果。矩阵 $p(z|d)$ 的大小为 $K \times N$, 其中每一列是一个与 K 维参数 α_{l_d} 相关的 Dirichlet 分布。对于一幅带有颜色名标签 l_d 的样本图像 d , 如果 $z = l_d$, 则 $\alpha_{l_d}(z) = c$ ($c \geq 1$, 具体值通过学习得到), 否则 $\alpha_{l_d}(z) = 1$ 。参考 PLSA 模型, 由于给定 N 幅带有颜色名标签的样本图像时 $p(z|d)$ 已知, $p(w|z)$ 被初始化为所有颜色名标签为 z 的图像中词 w 分布的均值, 则初始 E 步可以得到 $p(z|w, d) \propto p(z|d)p(w|z)$, M 步可以利用 $p(z|w, d)$ 得到新的 $p(z|d)$ 和 $p(w|z)$ 。迭代 E 步和 M 步, 直至得到每个颜色名 z 下的颜色空间分布 $p(w|z)$, 便可以计算出给定 w 在每个颜色名 z 下的概率

$$p(z|w) \propto p(z)p(w|z). \quad (3.3)$$

考虑到 w 不属于特定的颜色名, 对各颜色名的先验分布 $p(z)$ 取相同值。

3.4.1.1 基于颜色名通道的布尔图生成

CNS 方法中包含部分形态学图像处理算法, 如腐蚀、闭运算、形态学重构等, 故结构元素的选择会对检测结果产生实质性影响。对于不同大小的输入图像而言, 为了能够在统一的尺度下获得结构元素的最优参数, 首先将原始输入图像统一缩放为固定的 400 个像素宽度。随后利用颜色命名模型^[129,130] 提供的 `im2c` 函数以及大小为 32768×11 的映射矩阵 `w2c`, 将缩放尺寸后的图像 I 从 RGB 空间转换到颜色名空间 \mathbf{C} 。对于像素值范围在 $[0, 255]$ 整数区间的 RGB 图像, `im2c` 函数利用图像坐标 (x, y) 处的 RGB 颜色值 $(R_{x,y}, G_{x,y}, B_{x,y})$ 计算映射索引 $Ind_{x,y}$

$$Ind_{x,y} = 1 + \left\lfloor \frac{R_{x,y}}{8} \right\rfloor + 32 \times \left\lfloor \frac{G_{x,y}}{8} \right\rfloor + 32^2 \times \left\lfloor \frac{B_{x,y}}{8} \right\rfloor, \quad (3.4)$$

其中符号 $\lfloor \cdot \rfloor$ 表示向下取整, 计算得到的映射索引 $Ind_{x,y} \in [1, 32768]$ 。随后 `im2c` 函数从映射矩阵 `w2c` 中取出索引 $Ind_{x,y}$ 对应的行, 得到一个 11 维的向量。

通过上述转换, I 中每个像素的 RGB 颜色值 $(R_{x,y}, G_{x,y}, B_{x,y})$ 被映射为一个 11 维的颜色名向量, 颜色名向量中的每个元素是一个 $[0, 1]$ 区间内的浮点数, 表示 I 中该像素属于某个颜色名的概率, 向量中 11 个元素之和等于 1。由于每次调用 `im2c` 函数只能得到一个颜色名的概率图(图中每个元素值表示 I 中对应位置属于该颜色名的概率), 因此 CNS 方法分 11 次调用 `im2c` 函数, 共得到 11 个颜色名概率图。形式化起见, 将每个颜色名概率图称为一个颜色名通道, 整个颜色名空间 \mathbf{C} 是由 N 个颜色名通道所构成, 即 $\mathbf{C} = \{C_1, C_2, \dots, C_N\}$, 其中 $N = 11$, 下标 $1, 2, \dots, N$ 分别对应表 3.1 中的 11 个颜色名。为了能够使用一组序列化的阈值分割得到布尔图, 每个颜色名通道被归一化到 $[0, 255]$ 的整数区间。

图 3.3(b) – 3.3(l) 展示了对缩放后图像 I 生成的颜色名通道。虽然每个通道是一个概率图, 但其本质上反映了该颜色名的强度信息, 也就是说, 由这些通道所构成的颜色名空间中包含了语言学描述下的感知颜色特征。考虑到检测方法对场景中显著物体区

域所包含的颜色没有任何的先验知识，在随后的序列化分割时，所有的颜色名通道被等同对待。

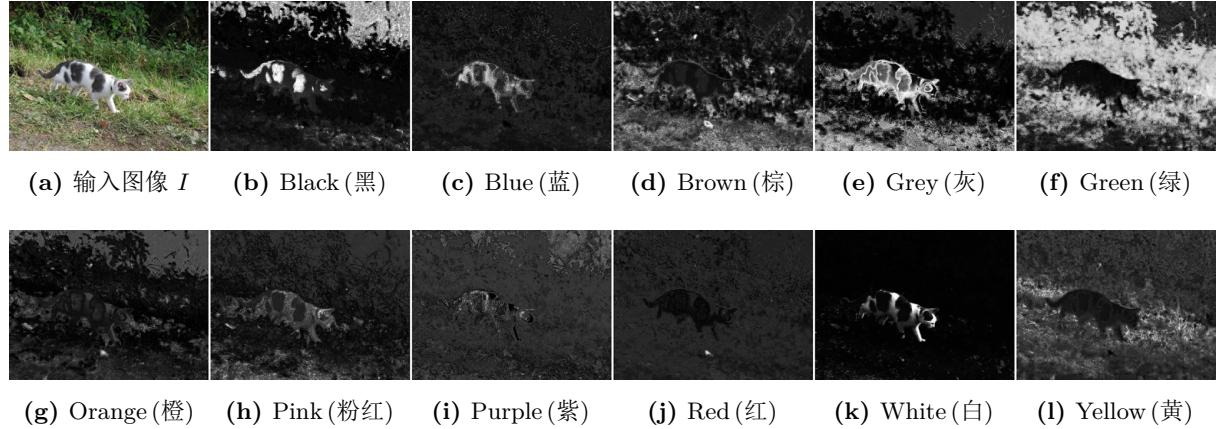


图 3.3 颜色名空间的 11 个颜色名通道。(a) 经过尺寸缩放的输入图像 I ; (b)–(l) 归一化到 $[0, 255]$ 区间的 11 个颜色名通道

利用颜色名通道生成布尔图的步骤为：首先在 $[0, 255]$ 区间内以 δ 为采样间隔，得到一组序列化的分割阈值。假设这组分割阈值的总数为 n ，其中第 j 个阈值记为 θ_j ，利用这 n 个阈值对每个颜色名通道 $C_i \in \mathbf{C}$ 进行如下的序列化分割

$$B_i^j = \text{THRESH}(C_i, \theta_j), \quad (3.5)$$

其中， B_i^j 是对颜色名通道 C_i 使用阈值 θ_j 分割得到的布尔图，下标 $i \in \{1, 2, \dots, N\}$ 表示颜色名通道号，上标 $j \in \{1, 2, \dots, n\}$ 表示分割阈值的序号。**THRESH** 函数的功能与式(3.1)相同，即当 C_i 中坐标 (x, y) 处的元素值 $C_i(x, y) \geq \theta_j$ 时， $B_i^j(x, y) = 1$ ，否则 $B_i^j(x, y) = 0$ 。相同的序列化阈值分割操作也对各颜色名通道的反相图执行，引入反相图的目的，是考虑到显著物体自身颜色强度的不同。假设显著物体区域具有较大的颜色强度，则对正相图的序列化分割可有效得到物体区域；反之，如果显著物体区域颜色强度较小，则仅对正相图分割会使得物体区域成为背景，这就需要依赖于反相图。在坐标 (x, y) 处，当 $B_i^j(x, y) = 1$ 时则 $\tilde{B}_i^j(x, y) = 0$ ，反之 $\tilde{B}_i^j(x, y) = 1$ 。形式为

$$\tilde{B}_i^j = \text{INVERT}(B_i^j). \quad (3.6)$$

至此，对 N 个颜色名通道包括其正/反相在内的共计 $N \times 2 \times n$ 个布尔图生成完毕。

图 3.4 展示了对黑色通道 C_1 进行序列化分割得到的部分布尔图，其中第 1 行是正相布尔图，第 2 行是反相布尔图。这些布尔图本身都是二值图像，白色的为前景像素，黑色的为背景像素。对于输入图像 I 中的显著物体——猫而言，其身上的黑色区域在正相布尔图中清晰可见，这意味着基于环绕性线索，这些区域可以很容易地从周围背景中分离出来。然而，正相布尔图中除了贴边的区域之外，还有一些靠近图像边界、面积稍大的背景区域存在，为了抑制掉这些区域，下一节中将引入形态学操作将它们连接起

来后作为非环绕区域一并去除。此外，虽然反相布尔图中看起来包含了猫身上的白色区域，但这些区域基本上都与图像边界相连接，因此对于白色区域的显著性检测，实际上主要依赖于由白色通道（即 C_{10} ）产生的布尔图。

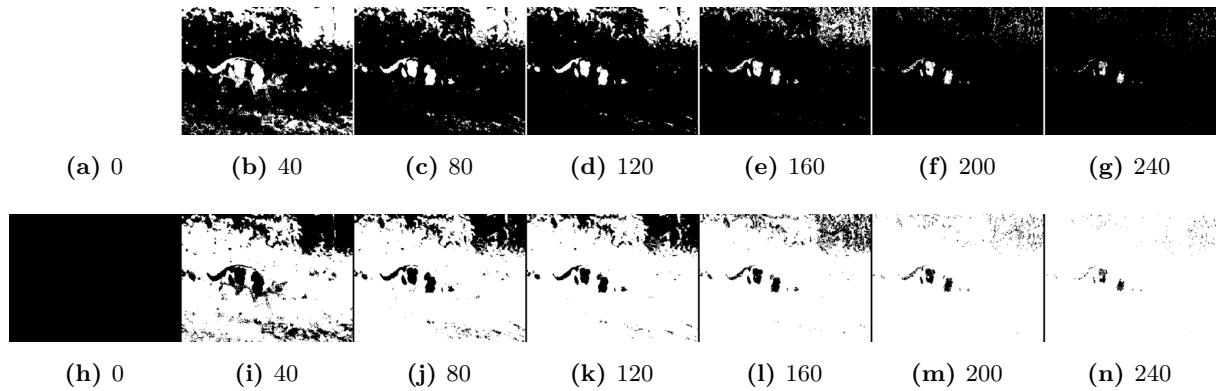


图 3.4 对黑色通道 C_1 序列化分割得到的布尔图：第 1 行为正相布尔图，第 2 行为反相布尔图，各子图标题为分割阈值

3.4.1.2 基于环绕性线索的注意图生成

在本章之前已讨论过：前景图形相较于背景会更加显著^[123,124]，即前景图形更有可能是显著物体；而基于图形–背景分离原则，环绕性区域更有可能是前景图形^[75,128]，即相较于具有完整闭合轮廓的前景图形，与图像边界相连接的区域因其轮廓的不闭合性，更有可能是场景中的背景部分。因此，在布尔图中利用环绕性线索抽取前景图形的最简单直接的方式，就是将与图像边界相连接的区域清除。在 CNS 方法中，该过程是通过序列化地对布尔图执行三个形态学操作来实现，包括闭运算、孔洞填充和边界对象清除^[142]，所得到的结果被称为注意图。

从图像 I 中生成布尔图和注意图的伪代码如算法 3.1 所示，其中：

- 闭运算 **CLOSE**: 闭运算是对图像先膨胀再腐蚀的操作，它可以将因阈值分割导致的前景图形内部的狭小缺口连接起来，使得这些区域尽可能地连通为一个整体。同时，闭运算也可以将靠近图像边界的环绕性区域连接起来，以与图像边界相连通，使这些区域之后作为背景被清除。考虑到前景区域轮廓的顺滑性，这里使用一个半径为 ω_c 的圆形结构元素， ω_c 为待定参数，其参数分析过程见第 3.5.2 节。

- 孔洞填充 **FILL**: 虽然闭运算可以将前景区域内部比结构元素小的孔洞自动补全，但无法处理比结构元素大的区域，而孔洞填充则可以处理这种白色前景区域内部的黑色孔洞，使前景区域更为完整。除可作用于布尔图外，第 3.4.2.2 节将把孔洞填充应用于显著图，其结果是将亮区域内部的暗区域填充为更高的灰度值，使得显著物体内部具有更为一致的显著值。

- 边界对象清除 **CLEAR-BORDER**: 边界对象清除属于形态学重构处理的一种，作用是将图像中与图像边界相连接的、比周围更亮的区域去除。在布尔图中，被去除的边

界对象实际上是白色的前景区域，但因与图像边界连接，这些区域更有可能是非环绕的背景，经过处理后注意图中保留的都是前景图形。在算法实现中，标记与图像边界连接的对象时采用 8 连接。

算法 3.1 布尔图和注意图生成

输入: 经过尺寸缩放的输入图像 I

输出: 各颜色名通道的正相注意图 A_i^j 和反相注意图 \tilde{A}_i^j

```

1: 将  $I$  转换到颜色名空间  $C$ 
2: for each  $C_i \in C$  do
3:   for  $\theta_j = 0 : \delta : 255$  do
4:      $B_i^j = \text{THRESH}(C_i, \theta_j)$                                 ▷ 正相布尔图
5:      $B_i^j = \text{CLOSE}(B_i^j, \omega_c)$                             ▷ 闭运算
6:      $B_i^j = \text{FILL}(B_i^j)$                                      ▷ 孔洞填充
7:      $A_i^j = \text{CLEAR-BORDER}(B_i^j)$                            ▷ 边界对象清除
8:      $\tilde{B}_i^j = \text{INVERT}(B_i^j)$                                 ▷ 反相布尔图
9:      $\tilde{B}_i^j = \text{CLOSE}(\tilde{B}_i^j, \omega_c)$ 
10:     $\tilde{B}_i^j = \text{FILL}(\tilde{B}_i^j)$ 
11:     $\tilde{A}_i^j = \text{CLEAR-BORDER}(\tilde{B}_i^j)$ 
12:  end for
13: end for

```

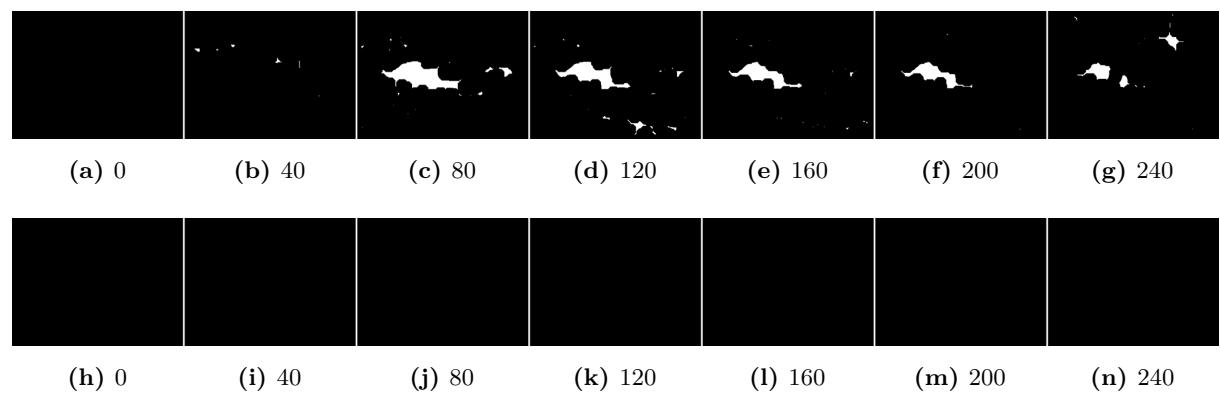


图 3.5 注意图：第 1 行为正相注意图，第 2 行为反相注意图，各子图标题为阈值

图 3.5 中给出了由图 3.4 所得到的注意图，同样来自于黑色通道 C_1 ，其中第 1 行是正相注意图，第 2 行是反相注意图。可以看到，在正相注意图中，原正相布尔图里那些与图像边界相连接的前景区域已被去除，仅留下了布尔图内部的环绕性区域；而反相注意图中，由于前景区域大面积与图像边界连接，导致最终得到的结果中只包含全黑的背

景。对照输入图像 I , 虽然正相注意图中还有零星的非显著物体区域存在, 但这些区域极少有在空间上的重复, 重复区域主要发生在显著物体内部, 这意味着如果将这些重复区域叠加, 就可以得到大致的显著物体。至此, 对 N 个颜色名通道包括其正/反相在内的共计 $N \times 2 \times n$ 个注意图生成完毕。

3.4.2 基于线性融合和后处理的显著性计算

3.4.2.1 基于线性融合的主注意图与平均注意图生成

在得到所有颜色名通道的注意图后, 参考 BMS 方法^[75], 将这些注意图采用线性平均的方式加以融合, 得到重复的前景图形区域。对于第 i 个颜色名通道, 由于产生了 n 个正相注意图 A_i^j 和 n 个反相注意图 \tilde{A}_i^j , 故该通道的平均注意图 A_i 为

$$A_i = \frac{1}{2n} \sum_{j=1}^n (A_i^j + \tilde{A}_i^j), \quad (3.7)$$

N 个颜色名通道的平均注意图 \bar{A} 为

$$\bar{A} = \frac{1}{N} \sum_{i=1}^N A_i. \quad (3.8)$$

需要特别指出的是, 不同于 BMS 方法将 LAB 空间三个通道所有的 (即 $3 \times 2 \times n$ 个) 注意图线性融合, 本章的 CNS 方法是先将各颜色名通道的注意图融合得到 A_i , 再将所有颜色名通道的平均注意图融合得到总的平均注意图 \bar{A} 。虽然两种方式本质上没有什么区别, 但这是为了 CNS 方法中的第二条流水线所考虑。在第 3.4.3 节中, 各颜色名通道的平均注意图将与两个全局的颜色名线索结合, 生成加权显著图。为避免混淆, 本章将每个颜色名通道得到的平均注意图 A_i 称为第 i 个通道的主注意图, 将所有颜色名通道的线性融合结果 \bar{A} 称为平均注意图。

图 3.6(b) – 3.6(l) 展示了不同颜色名通道的主注意图, 它们的平均注意图 \bar{A} 如图 3.6(a) 所示。出于展示的需要, 这里对主注意图和平均注意图以热图方式显示, 从中可以比较各注意图之间的差异。首先, 注意图中的像素值为浮点数, 各主注意图中的像素最小值为 0, 最大值约在 0.1563 至 0.4219 之间; 平均注意图的像素最小值也为 0, 最大值约为 0.1307。其次, 所有主注意图中, 最为显著的是由黑色通道和白色通道所产生, 两者可以大致拼接出猫的区域; 次为显著的是蓝色通道, 特别是在后肢位置, 这与颜色名图像 3.8(a) 中将这块区域判定为含部分蓝色有关。第三, 颜色名图像中猫身上还有大块的灰色区域, 但由于背景中灰色区域的存在, 使得猫身上的灰色区域与图像边界连接而被清除, 灰色通道产生的主注意图并不太明显。最后, 对于平均注意图 \bar{A} 而言, 虽然猫整体上显著值不高, 但肉眼仍然能够分辨出其与背景之间的差异。在下一节中, 将引入一个后处理算法, 以凸显整个物体区域。

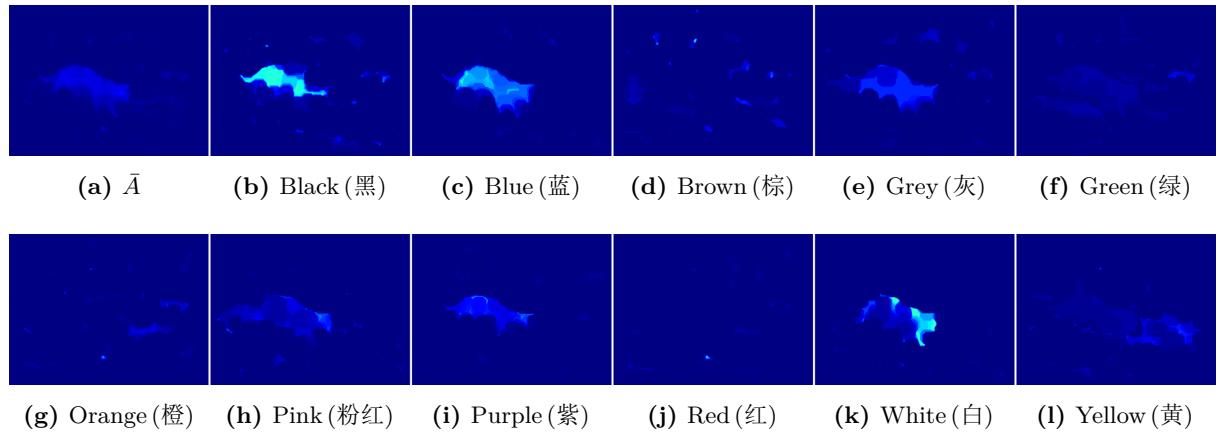


图 3.6 主注意图和平均注意图。(a) 平均注意图 \bar{A} ; (b)–(l) 11 个颜色名通道的主注意图 $A_i, i \in \{1, 2, \dots, 11\}$

3.4.2.2 后处理算法 I

将平均注意图 \bar{A} (见图 3.6(a)) 进行归一化操作, 得到的结果如图 3.7(a) 所示。平均注意图经过归一化后, 清晰地显示了其中的显著物体区域。但是, 该结果仍然存在两个问题: 1) 在显著物体区域之外, 还包括有一些面积较小的显著区域, 其中部分区域的显著值并不低, 而这些区域显然应属于背景被抑制掉; 2) 显著物体本身的显著值还不够突出, 最终的检测结果应在凸显物体区域的同时, 使其获得更为平滑一致的显著性。为了解决上述问题, 本节的后处理算法中包括两个主要步骤: 一是利用形态学重构做开运算, 用于解决第 1 个问题; 二是通过亮度变换处理, 来解决第 2 个问题。完整的后处理步骤伪代码如算法 3.2 所示。

算法 3.2 后处理算法 I

输入: 归一化的平均注意图 \bar{A}

输出: 显著图 S

- | | |
|---|--------------------|
| 1: $S = \text{RECONSTRUCT}(\bar{A}, \omega_r)$ | ▷ 由形态学重构做开运算 |
| 2: $S = \text{NORMALIZE}(S, [0, 255])$ | ▷ 归一化到 [0, 255] 区间 |
| 3: $S = \text{ADJUST}(S, \vartheta_r, \vartheta_g)$ | ▷ 亮度变换 |
| 4: $S = \text{FILL}(S)$ | ▷ 孔洞填充 |
-

为了将小的显著区域去除, 一种常见的处理是采用形态学开运算, 即首先利用腐蚀操作去除小的显著区域, 再利用随后的膨胀操作将大的显著区域 (即猫) 还原。但由于这一过程中所使用的结构元素与猫的轮廓形状不能完全契合, 导致膨胀后对猫形状的还原不够准确。而利用形态学重构做开运算则可以有效解决这一问题, 使得被腐蚀之后的猫的形状得到准确的还原。在算法实现中, 函数 **RECONSTRUCT** 首先构造一个半径为 ω_r 的圆形结构元素 (参数分析见第 3.5.2 节), 对归一化的平均注意图 \bar{A} 进行腐

蚀，然后将腐蚀结果作为标记图像 (marker)、 \bar{A} 作为掩模图像 (mask)，来约束重构过程，重构过程使用快速混合重构算法[143] 实现。之后，这一过程再被反相执行一次，即先利用相同的结构元素对重构结果膨胀，然后将膨胀结果的补图像作为标记图像、将前次重构结果的补图像作为掩模图像，再次进行重构，最后求补得到结果。图 3.7(b) 是对归一化后的平均注意图 \bar{A} 进行上述处理得到的结果，可以看到，背景中那些小的显著区域已被去除，猫的区域在保持原有形状的同时获得了更为平滑的显著值。

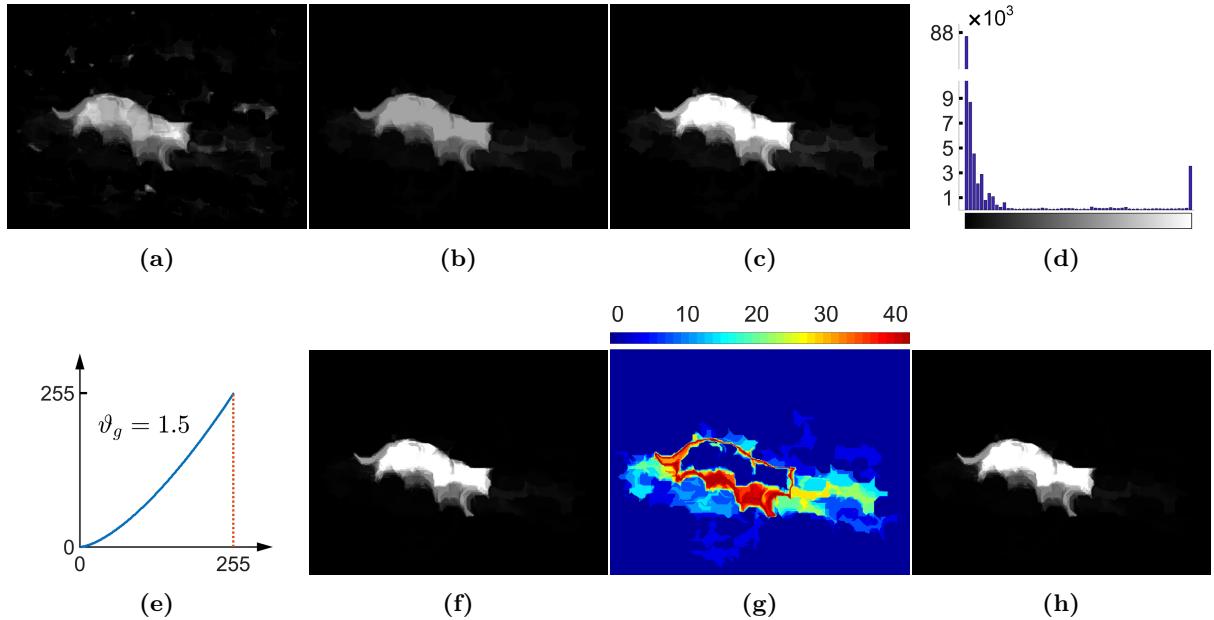


图 3.7 后处理算法 I。(a) 将图 3.6(a) 归一化；(b) 对图 (a) 重构的结果；(c) 将图 (b) 归一化到 $[0, 255]$ 区间；(d) 图 (c) 的直方图；(e) 亮度映射曲线；(f) 对图 (c) 使用参数值 $\vartheta_r = 0.02$ 、 $\vartheta_g = 1.5$ 亮度变换的结果；(g) 图 (c) 与图 (f) 的差图像；(h) 显著图 S

如果直接将重构结果 3.7(b) 归一化，会得到如图 3.7(c) 所示的结果，该结果看起来很好地凸显了显著物体区域。然而，在猫头前方，有一大块不引人注意的暗色区域存在，这将直接影响检测结果的精确率。为了能将其抑制为背景，这里采用一个自适应的亮度变换操作进行处理。假设亮度变换前的图像为 F ，变换后的结果为 G ，亮度变换函数 **ADJUST** 的形式化定义如下

$$G = \text{ADJUST}(F, [0, T_F], \vartheta_g), \quad (3.9)$$

其中， T_F 是一个自适应阈值，参数 ϑ_g 用来指定亮度变换曲线的形状，亮度变换函数的作用是通过参数 ϑ_g 控制的曲线来映射 F 的亮度值，生成新的图像 G 。在此过程中， F 中亮度值超过阈值 T_F 的像素将被映射为最大亮度 (即 255)，而区间 $[0, T_F]$ 内的像素的映射值则由映射曲线所确定。

引入亮度变换操作的目的，与显著物体检测任务密切相关。第 1.2.1 节中已指出，显著物体检测本质上是一个图形–背景分割问题[28]，理想的显著图输出中，显著物体区

域的显著值应为 1 (对于灰度图而言是 255), 背景的显著值应为 0。但现有的大部分工作并没有朝这个方向所努力, 而亮度变换正是希望得到类似于二值的映射结果。在亮度变换前的图像 F 中, 理想的显著物体区域内部像素的亮度值应大于 T_F , 这样亮度变换函数可将这些像素的新值全部映射为最大亮度, 即 255。而亮度值小于 T_F 的像素最有可能是背景像素, 故参数 ϑ_g 理论上应大于 1, 这样可以抑制区间 $[0, T_F]$ 内的像素。图 3.7(e) 给出了 $\vartheta_g = 1.5$ 时的映射曲线, 其中 x 轴表示输入图像 F 的亮度值, y 轴表示输出图像 G 的映射值, 曲线显示出映射过程对更暗输出值的加强。

对于亮度变换函数中的自适应阈值 T_F , 这里希望通过重构结果的统计特征自动获得, 一种朴素的想法就是利用其直方图。假设 H 是归一化后的重构结果 (见图 3.7(c)) 的直方图, 在区间 $[0, 255]$ 内共有 256 个灰度级, 其中第 k 个灰度级 ($k \in [0, 255]$) 的像素数记为 H_k 。从灰度级 0 的像素数 H_0 开始累计, 直至找到满足以下条件的最小灰度级 p 时停止

$$(1 - \vartheta_r) \sum_{k=0}^{255} H_k \leq \sum_{k=0}^p H_k, \quad (3.10)$$

其中 ϑ_r 是一个待定参数。此时得到的 p , 即为亮度变换的自适应阈值 T_F 。

上述寻找自适应阈值 T_F 的过程, 实际上是通过参数 ϑ_r 来控制显著物体区域在整个图像中所占的比例。理想情况下, 如果重构过程得到了较好的检测结果, 其中占有较高亮度值的像素数占图像总像素数的比例为 ϑ_r , 那么亮度变换过程可以将这些亮度值大于 T_F 的像素全部映射到最大亮度 255, 剩余像素则被映射曲线所抑制。然而, 从第 2.5.1 和 3.5.1 节的数据集特性统计中可以看到, 不同数据集的平均显著物体区域面积占图像的比例并不相同, 且不同图像之间的这一比例也有较大差异。例如, 直方图 3.7(d) 中共有 60 个像素数非 0 的灰度级, 像素数范围是 [57, 87898], 在背景与目标之间有一段间隔较大的低频数灰度值区间存在。如果以 20% 即 $\vartheta_r = 0.2$ 来控制显著物体区域所占的比例, 那么灰度值 4 (即 $T_F = 4$) 以上的像素都将被映射为 255; 如果这一比例下降到 10% 即 $\vartheta_r = 0.1$, 则灰度值 26 (即 $T_F = 26$) 以上的像素被映射为 255。显然, 这都不是理想的映射结果。因此, 对最优 ϑ_r 值的确定, 是对所有实验图像共同调参得到。在第 3.5.2 节的参数分析中, 得到的最优 ϑ_r 为 0.02, 表明平均约只有占图像面积 2% 的高亮像素受到了这一变换过程的影响。

图 3.7(f) 是经过亮度变换得到的映射结果, 当 $\vartheta_r = 0.02$ 时自适应阈值 $T_F = 255$, 表明原高亮区间段没有变化。但如果对变换前后的图像计算差值, 就可以清楚地看到该过程对较暗像素的抑制。如图 3.7(g) 所示, 猫的躯干部分没有变化 (这部分主要是原亮度值为 255 的像素), 而猫头前部的大块区域以及猫尾后部的区域则已被完全抑制为背景, 这就在一定程度上提高了检测结果的精确率。但新的问题是, 猫腹部的显著值也被一并抑制 (变换前后的最大亮度值差异为 38), 导致检测结果的召回率又有所降低。对于这一问题, 将由下一节介绍的基于全局颜色名线索的加权显著图进行解决。

由于上述整个亮度变换过程可以藉由两个待定参数 ϑ_r 和 ϑ_g 确定 (对 ϑ_r 和 ϑ_g 的

参数分析详见第 3.5.2 节), 为方便起见, 将式(3.9)重新定义为如下形式

$$G = \text{ADJUST}(F, \vartheta_r, \vartheta_g). \quad (3.11)$$

最后, 对亮度变换结果进行孔洞填充^[142]操作, 就得到了 CNS 方法第一条流水线的结果——显著图 S , 如图 3.7(h) 所示。对于灰度图而言, 孔洞是指那些被较亮区域完整包围的较暗区域, 但由于亮度变换结果中没有该类区域存在(见图 3.7(f)), 孔洞填充前后得到了完全相同的显著图。在第 3.4.3.2 节中, 将看到对亮度变换结果进行孔洞填充的效果。

3.4.3 基于全局颜色名线索的显著性加权

在将图像 I 转换到颜色名空间 C 的过程中, 每个像素被映射为一个 $N = 11$ 维的颜色名向量, 向量中的每个元素表示该像素隶属于某个颜色名的概率。显然, 概率值越大, 表明该像素越接近于人眼视觉感知所对应的颜色名。如果对每个像素取颜色名向量中概率值最大的那个分量, 并用表 3.1 中的 RGB 值来表示该分量的颜色值, 那么图像 I 就可以转换成一幅完全由颜色名像素所构成的图像, 该图像被称为颜色名图像。

上述转换过程, 可以看作是对图像 I 的颜色量化过程, 即将 I 量化为语言学上的颜色。在第 2 章中, 已经验证了利用有效的颜色量化, 可以在较少的量化颜色数下同样取得不错的检测结果, 同时也验证了基于量化图像的全局颜色特征在显著物体检测时的有效性。因此, 本节将利用颜色名图像的两个全局的颜色名线索, 与第 3.4.2.1 节产生的主注意图相结合, 得到 CNS 方法第二条流水线的加权显著图。

3.4.3.1 基于颜色名统计的显著性加权

显著性加权中的第一个全局颜色名线索, 是基于颜色名图像的统计特征和分布特征, 通过构造其颜色名直方图和颜色名索引矩阵实现。图 3.8(a) 是对应图 3.3(a) 的颜色名图像, 从中可以看到颜色命名模型^[129,130]对输入图像的量化过程。显著物体猫主要有黑、白、灰三色组成, 其中还包括有少量的蓝色像素, 而背景中则包括大量的绿色区域, 其次是黑色和棕色, 整幅图像中不含有橙色像素。颜色名图像的颜色分布如颜色直方图 3.8(b) 所示, 通过该直方图可以得到第一个特征, 即各颜色名的概率, 其计算方式是颜色名像素数与图像总像素数的比值。表 3.2 给出了对应图 3.8(a) 的不同颜色名的像素数和概率值 ($[0, 1]$ 区间), 其中 i 表示颜色名通道号, 颜色名 t_i 与表 3.1 相一致, 对于第 i 个颜色名通道, 其概率值记为 f_i 。

为了将概率值与第 3.4.2.1 节的主注意图结合, 这里再次利用颜色名图像, 得到第二个特征, 即颜色名索引矩阵。生成过程如下: 假设把颜色名图像记为 M , 依次扫描 M 中每个空间位置 (x, y) 处的颜色名, 如果其属于第 i 个颜色名, 则将第 i 个颜色名索引矩阵 M_i 中对应位置 (x, y) 处的元素值标记为 1, 否则标记为 0, 即

$$M_i(x, y) = \begin{cases} 1, & \text{if } M(x, y) = i; \\ 0, & \text{otherwise.} \end{cases} \quad (3.12)$$

可以看到，该过程构建的颜色名索引矩阵，实际上反映了各颜色名在图像空间位置上的分布，而这种分布正好与基于颜色名通道得到的主注意图中的环绕性区域在空间上相一致。因此，数量上的统计特征就可以通过空间上的分布特征对主注意图施以影响，将基于全局统计得到的权重信息带入主注意图。利用概率值 f_i 和对应的颜色名索引矩阵 M_i ，对各颜色名通道的权重矩阵构造如下

$$W_i = f_i M_i. \quad (3.13)$$

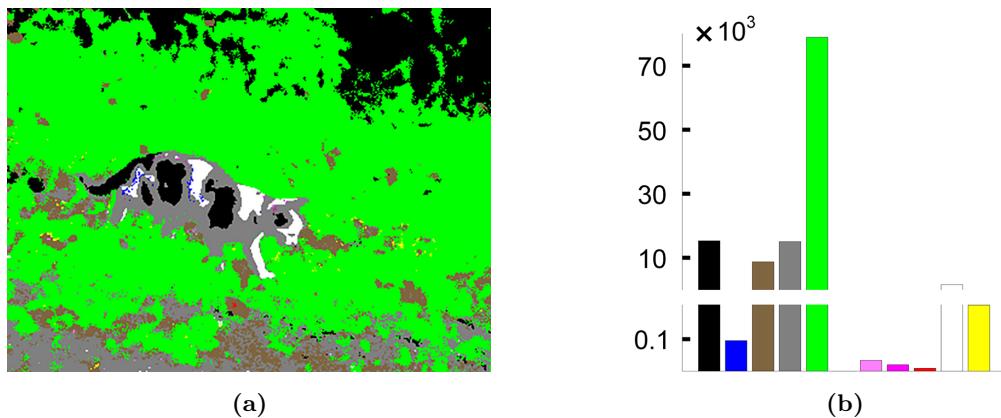


图 3.8 颜色名图像和颜色名直方图。(a) 图 3.3(a) 的颜色名图像；(b) 图 (a) 的颜色名直方图

表 3.2 基于统计的颜色名概率

i	1	2	3	4	5
颜色名 (t_i)	Black (黑)	Blue (蓝)	Brown (棕)	Grey (灰)	Green (绿)
像素数	15,321	96	8,746	15,078	78,905
概率 (f_i)	0.1277	0.0008	0.07288	0.12565	0.6575
6	7	8	9	10	11
Orange (橙)	Pink (粉红)	Purple (紫)	Red (红)	White (白)	Yellow (黄)
0	33	20	9	1,586	206
0	0.000275	0.0001667	0.000075	0.01322	0.001717

图 3.9 是不同颜色名通道的颜色名索引矩阵和对应的权重矩阵。对于任意一个权重矩阵 W_i ，0 值元素表示对应空间位置未被量化为颜色名 t_i ，而非 0 值位置的元素为 t_i 的概率值 f_i 。显然，通过这种方式构造的权重矩阵，将导致像素数多的颜色名获得更

高的权值，例如图 3.9(e)。这就带来了一个困惑之处，这是否意味着将权重矩阵带入主注意图后，绿色通道的区域将更加显著？

事实上，拥有最高概率值的绿色通道中基本是贴边的背景区域，在注意图生成阶段，这些区域将被作为边界对象清除。从图 3.6(f) 也可以看到，该通道主注意图中并不包含显著的前景图形，从而对最终的加权结果几乎不会产生影响。相反，主注意图中的黑色通道产生了高亮的显著性区域（见图 3.6(b）），同时其概率值排名第二，使得在引入权重矩阵后显著性区域将更加突出。

由此可见，构造权重矩阵的目的，是希望对面积较大的环绕性区域赋予更大的权重，以使它们获得更大的显著值。蓝色通道的主注意图中虽然也有高亮的显著性区域存在（见图 3.6(c)），但由于概率值较低，这些区域将被极大地抑制。灰色通道的概率值排名第三，其主注意图的显著性处于适中地位（见图 3.6(e)），两者结合将产生中和的结果。较为特殊的是白色通道，图 3.6(k) 显示其也包含高亮的显著性区域，但较低的概率值影响了权重矩阵与主注意图的结合，这将通过下一节的全局颜色名对比特征予以解决。当然，上述处理方式也面临着一定的问题，就是当背景颜色并未与图像边界连接时，将不能通过环绕性算法有效清除，从而增加了干扰信息。例如图像存在边框，或是显著物体被多层颜色的背景包围。对于前者，第 4 章将引入一种边框裁剪算法予以解决；而对于后者，仅基于环绕性线索很难解决，一种可行的处理是融入似物性采样，通过对似物性区域的检测来排除背景。

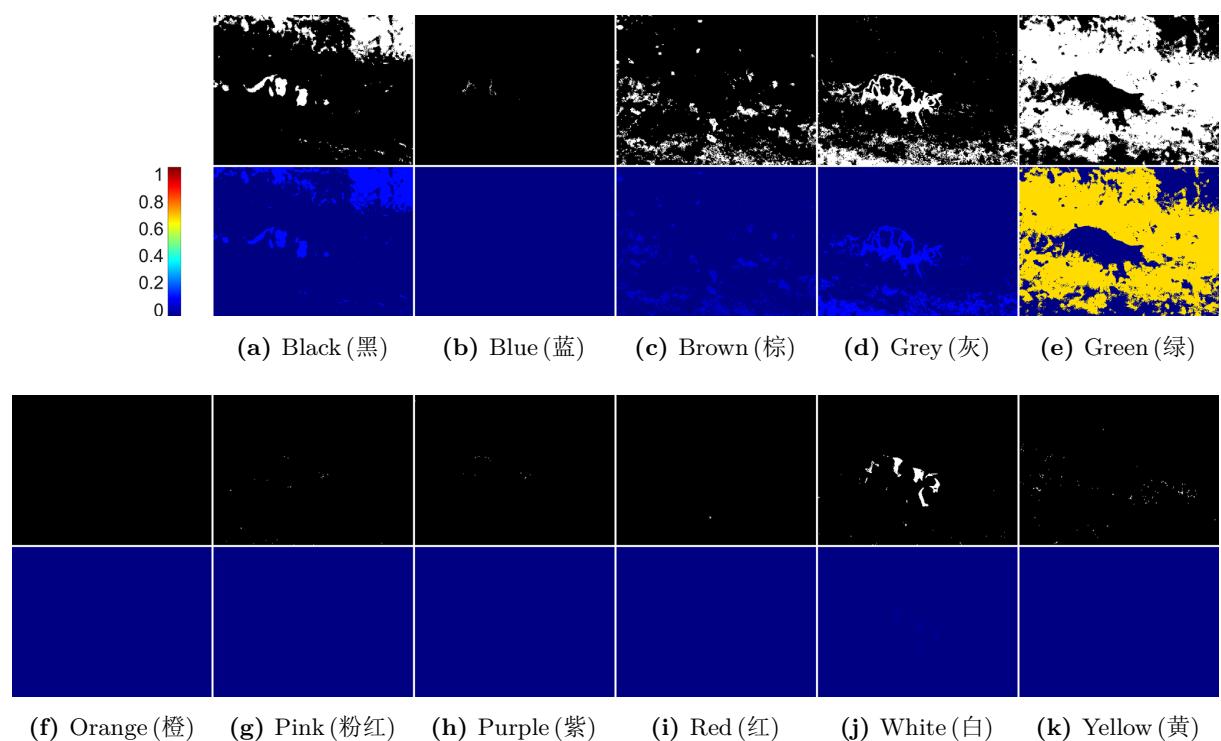


图 3.9 基于统计与分布的颜色名索引矩阵和权重矩阵。（a）–（k）对应 11 个颜色名通道，各子图中第 1 行为颜色名索引矩阵，第 2 行为权重矩阵

3.4.3.2 基于颜色名对比的显著性加权

显著性加权中的第二个全局颜色名线索，是基于颜色名直方图的对比特征。对于第 i 个颜色名 t_i , $i \in \{1, 2, \dots, N\}$, 其在颜色名图像中的概率为 f_i , RGB 颜色值为 c_i , 则 t_i 对应的权重系数为

$$w_i = \begin{cases} \sum_{j=1}^N f_j \|c_i - c_j\|_2^2, & \text{if } f_i \neq 0; \\ 0, & \text{otherwise.} \end{cases} \quad (3.14)$$

其中, 符号 $\|\cdot\|_2$ 表示 ℓ_2 范数, f_j 是上一节计算得到的颜色名 t_j 的概率, c_i 和 c_j 是表 3.1 中对应颜色名 t_i 和 t_j 的 RGB 颜色值。

上述对权重系数 w_i 的计算, 本质上是利用了颜色名在整个图像场景中的全局稀缺性, 即占图像比例低的、且与其它差异大的颜色名将获得更高的权重, 这与第 2.4.1.2 节中的式 (2.1) 作用相同。为了避免颜色名图像中不存在的量化颜色产生非 0 的权重值, 这里将概率 $f_i = 0$ 的颜色名的权重 w_i 设为 0。同时, 考虑到颜色命名模型^[129,130] 得到的颜色名已具有很高的感知区分度, 不同于式 (2.1) 使用的 LAB 颜色空间, 而是改为直接利用颜色名的 RGB 值来度量颜色之间的差异。此外, 与第 2 章提出的 RPC 方法不同, 这里并非直接利用全局稀缺性来定义各颜色名的显著值, 而是将其作为权重系数, 去提高主注意图中具有稀缺颜色的环绕性区域的显著值。

至此, 基于全局颜色名线索的权重矩阵和权重系数都已构造完毕, 两者的联合作用是对主注意图中面积较大的、且与其它颜色名有较大颜色差异的环绕性区域赋予更高的显著值, 带入第 3.4.2.1 节生成的主注意图, 就得到了加权主注意图。对于第 i 个颜色名通道, 加权主注意图的计算公式为 $w_i \cdot \mathbf{N}(W_i \circ A_i)$ 。然后将所有加权主注意图线性融合, 得到加权平均注意图

$$\bar{A}_w = \sum_{i=1}^N w_i \cdot \mathbf{N}(W_i \circ A_i), \quad (3.15)$$

其中, 符号 \circ 表示 Hadamard 积运算^①, 函数 \mathbf{N} 表示归一化到 $[0, 1]$ 区间。

这里需要对式(3.14)中的“otherwise”分支做一点额外说明: 事实上, 当 $f_i = 0$ 时权重矩阵 W_i 是一个全 0 矩阵, $W_i \circ A_i$ 也是一个全 0 矩阵。此时无论 w_i 是否为 0, 都会得到全 0 的加权主注意图。因此, 式(3.14)可去除“otherwise”分支, 进一步简化为

$$w_i = \sum_{j=1}^N f_j \|c_i - c_j\|_2^2. \quad (3.16)$$

图 3.10(b) – 3.10(l) 是由上述方法产生的加权主注意图, 线性融合得到的加权平均注意图 \bar{A}_w 如图 3.10(a) 所示。可以看到, 显著值最高的区域出现在白色通道中, 其环绕性区域所具有的白色也是颜色名图像中最为稀缺的颜色, 表明基于颜色名对比加权

^① $m \times n$ 的矩阵 $A = [a_{x,y}]$ 与 $m \times n$ 的矩阵 $B = [b_{x,y}]$ 的 Hadamard 积记作 $A \circ B$, 它仍然是一个 $m \times n$ 矩阵, 其元素定义为两个矩阵对应元素的乘积, 即 $(A \circ B)_{x,y} = a_{x,y} b_{x,y}$, 其中 x 和 y 表示元素坐标。

的有效性。紧随其后的是黑色通道和灰色通道，两者具有接近的概率值，但由于颜色名图像中黑色比灰色更为稀缺（两者的权重系数相差约 0.2248），加之其主注意图更为显著，故相对于灰色通道，黑色通道产生了更为显著的物体区域。

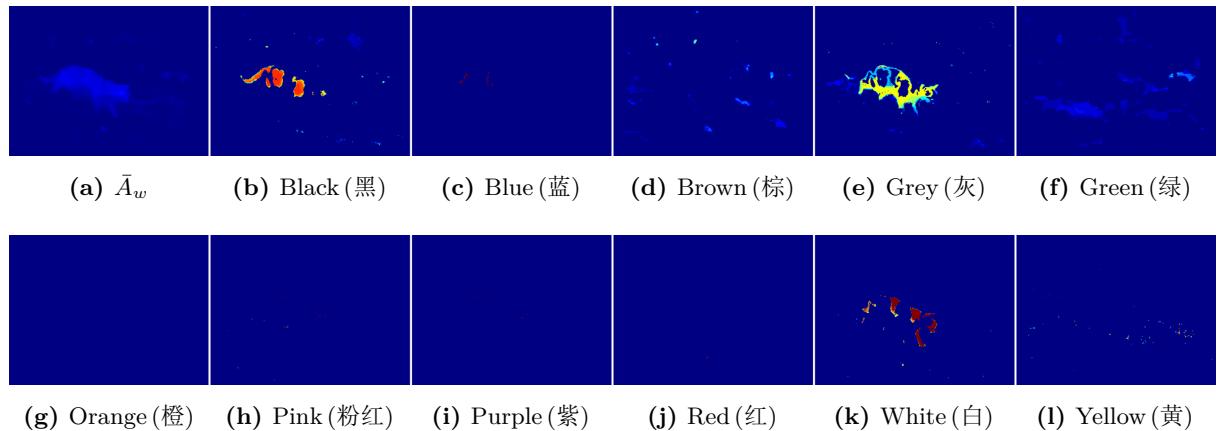


图 3.10 加权主注意图和加权平均注意图。(a) 加权平均注意图 \bar{A}_w ; (b)–(l) 11 个颜色名通道的加权主注意图

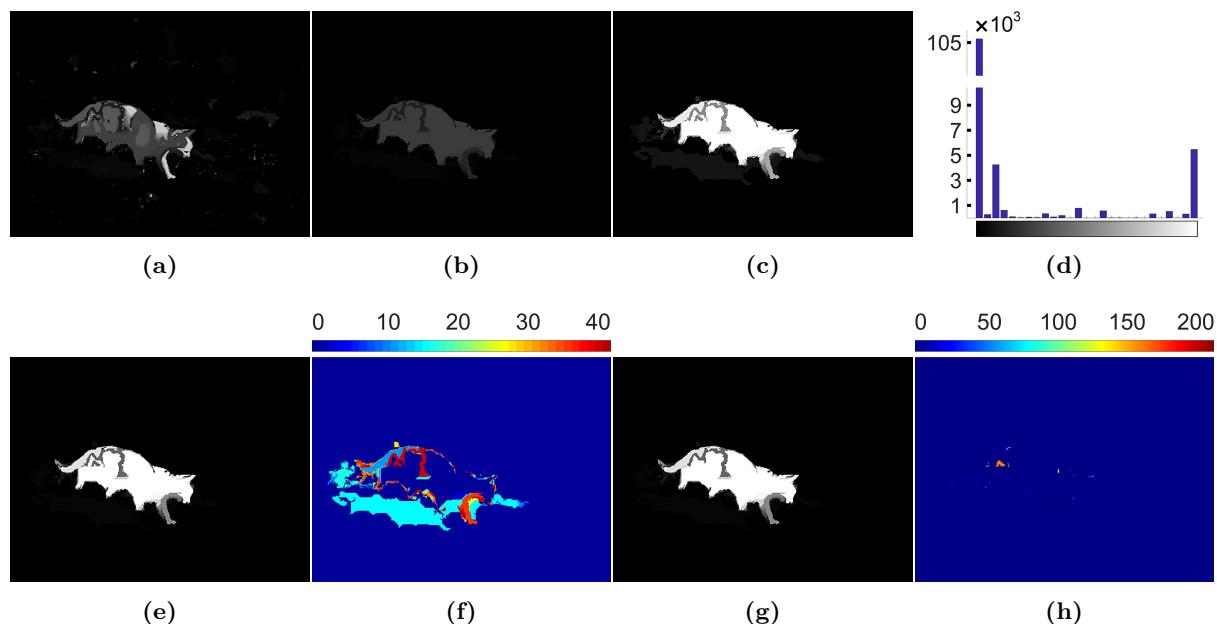


图 3.11 将后处理算法 I 应用于加权平均注意图 \bar{A}_w 。(a) 将图 3.10(a) 归一化; (b) 对图 (a) 重构的结果; (c) 将图 (b) 归一化到 $[0, 255]$ 区间; (d) 图 (c) 的直方图; (e) 对图 (c) 使用参数值 $\vartheta_r = 0.02$ 、 $\vartheta_g = 1.5$ 亮度变换的结果; (f) 图 (c) 与图 (e) 的差图像; (g) 对图 (e) 孔洞填充后得到的加权显著图 S_w ; (h) 图 (e) 与图 (g) 的差图像

最后，再次使用第 3.4.2.2 节介绍的算法 3.2 对加权平均注意图 \bar{A}_w 执行后处理操作，就得到了 CNS 方法第二条流水线的结果——加权显著图 S_w 。详细过程这里不再赘述，其形态学重构、归一化、亮度变换和孔洞填充的对应结果，分别如图 3.11 中所示。需要补充说明的是孔洞填充操作，填充前后的差异如图 3.11(h) 所示，除了零星的像素点被

填充之外，由于猫尾部位的一小块较暗区域完全被较亮的区域所包围，使得填充之后这块区域已成为较亮的前景。图 3.11(g) 是基于颜色名线索得到的加权显著图 S_w ，与上一节的显著图 S 相比，猫的轮廓更为清晰、猫腹部的显著值也得到了很好的提升，缺点在于猫尾部位存在两条明显的较暗裂纹，而显著图 S 则没有这个问题。

3.4.4 基于高亮截断和后处理的显著性融合

在分别得到两条流水线产生的显著图 S 和 S_w 后，一种朴素的想法是直接将两者做线性平均融合，以将环绕性线索与颜色名线索集成到一个统一的框架中，结果如图 3.12(a) 所示。可以看到，采用这种融合方式，很好地互补了两个显著图各自的不足，显著图 S 中猫腹部的显著值有所增强，而显著图 S_w 中猫尾部位深色裂痕的显著值也有所提升。但这仍然不是理想的显著物体检测结果，如前所述，最终希望得到的是接近于二值标注结果的显著图。因此，本节将分别基于高亮截断和后处理操作实现融合过程，以输出理想的显著图。

3.4.4.1 基于高亮截断的融合算法

在第 3.4.2.2 和 3.4.3.2 节中，后处理算法 3.2 很好地抑制了 S 和 S_w 中的背景，并且凸显了显著的前景物体区域。如图 3.7(h) 和 3.11(g) 所示，相对而言，两者的显著物体区域在较暗的灰度段上趋于一致，差别较大的地方是在较亮的灰度段上。因此，高亮截断的目的，就是保留共同的较暗灰度段，同时将较亮的灰度段截断。由于 S 与 S_w 中的像素值都在 $[0, 255]$ 区间，高亮截断可形式化定义为

$$\bar{S} = \frac{[S + S_w]_0^{255}}{2}, \quad (3.17)$$

其中 $[\cdot]_0^{255}$ 表示高亮截断运算符，其作用是当符号内部结果处于 $[0, 255]$ 区间时保持输出不变，而超过 255 的部分将被强制截断为固定的输出值 255。

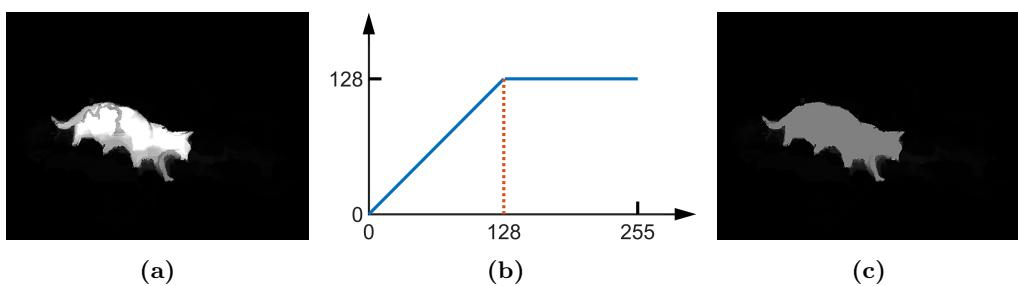


图 3.12 基于高亮截断的线性融合。(a) 对图 3.7(h) 和 3.11(g) 直接线性平均的结果；(b) 高亮截断的亮度映射曲线；(c) 平均显著图 \bar{S}

式 (3.17) 相当于是对原始的线性平均结果 $(S + S_w)/2$ 再次执行一个亮度变换操作，等价的映射曲线如图 3.12(b) 所示，其输出映射值介于 $[0, 128]$ ，而原线性平均结果中区间 $[128, 255]$ 内的像素值被映射为 128。基于高亮截断得到的平均显著图 \bar{S} 如

图 3.12(c) 所示, 可以看到, 与预期一致, \bar{S} 中保留了两个显著图 S 和 S_w 中共同的较暗灰度段, 得到了清晰平滑的物体区域。唯一的缺点是物体区域的显著值还不够突出, 这将通过后处理算法 II 加以改进。

3.4.4.2 后处理算法 II

对平均显著图 \bar{S} 执行后处理操作的目的, 与第 3.4.2.2 节介绍的后处理算法 I (即算法 3.2) 相同, 伪代码如算法 3.3 所示。由于 \bar{S} 中背景部分已被足够抑制, 且没有非目标的显著性区域存在, 因此后处理算法 II 去除了形态学重构步骤, 主要通过对 \bar{S} 进行亮度变换来凸显其中的显著物体区域。

算法 3.3 后处理算法 II

输入: 平均显著图 \bar{S}

输出: 最终显著图 \hat{S}

- 1: $\hat{S} = \text{ADJUST}(\bar{S}, \vartheta_r, \vartheta_g)$ ▷ 亮度变换
 - 2: $\hat{S} = \text{FILL}(\hat{S})$ ▷ 孔洞填充
 - 3: $\hat{S} = \text{RESIZE}(\hat{S})$ ▷ 缩放回原图大小
-

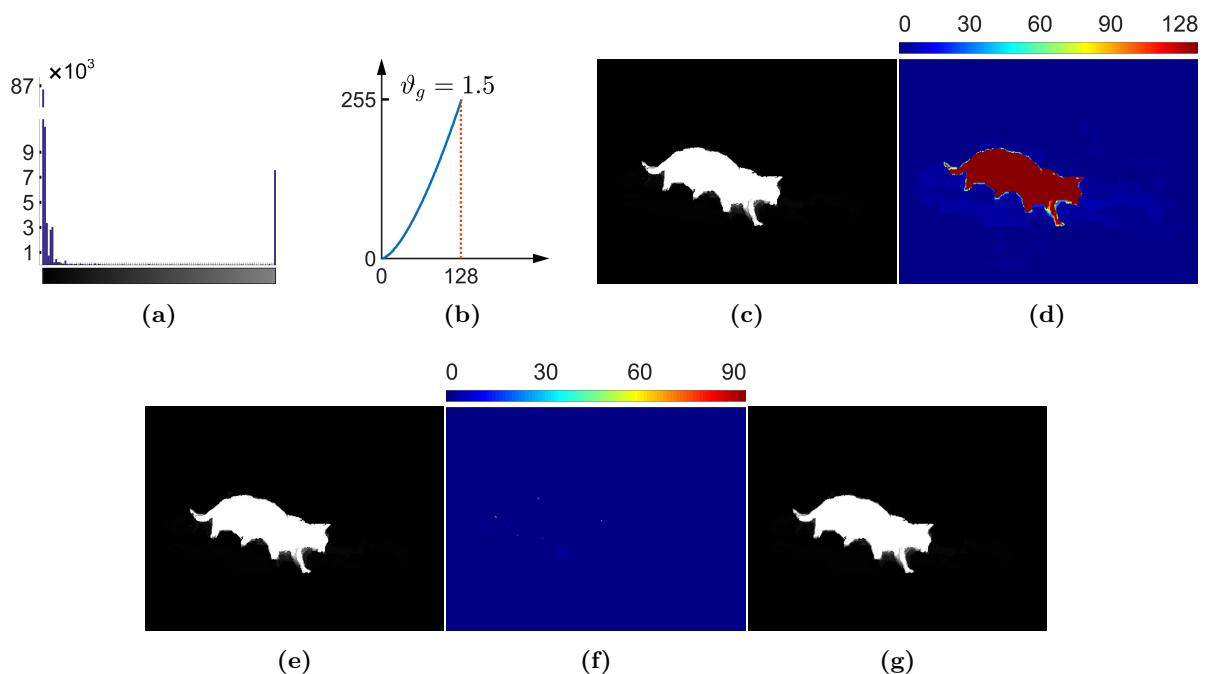


图 3.13 后处理算法 II。(a) 图 3.12(c) 的直方图; (b) 亮度映射曲线; (c) 对图 3.12(c) 使用参数值 $\vartheta_r = 0.02$ 、 $\vartheta_g = 1.5$ 亮度变换的结果; (d) 图 3.12(c) 与图 (c) 的差图像; (e) 对图 (c) 孔洞填充的结果; (f) 图 (c) 与图 (e) 的差图像; (g) 缩放回原图大小后得到的最终显著图 \hat{S}

图 3.13(b) 是亮度变换的映射曲线, 与后处理算法 I 稍有不同的是, 其效果相当于将 $[0, 128]$ 的输入区间映射到 $[0, 255]$ 的输出区间, 这就将 \bar{S} 中物体区域的显著值提

升至高亮的灰度级。同时为了减少参数数量, **ADJUST** 函数中两个参数 ϑ_r 与 ϑ_g 的取值与后处理算法 I 保持相同。亮度变换的结果如图 3.13(c) 所示, 从与 \bar{S} 的差异图 (见图 3.13(d)) 中可以看到变换过程对显著物体区域提升的效果。随后执行的孔洞填充如图 3.13(e) 所示, 填充前后的差异图 3.13(f) 中, 物体区域内部只有零星的像素点被填充 (填充前后的灰度值最大相差 91), 这表明基于高亮截断的融合得到了非常平滑的物体区域。最后, 利用 **RESIZE** 函数缩放回原图大小, 就是 CNS 方法的最终检测结果 \hat{S} 。从图 3.13(g) 中可以看到, 结果显著图已非常接近于二值的人工标注图。

3.5 实验与讨论

为了实际评估本章提出的 CNS 方法的有效性, 在三个数据集上与 21 个显著性方法进行了性能比较。除了在第 2 章已使用过的 ASD 数据集^[9,32] 外, 还分别在 ECSSD 数据集^[60,61] 和 ImgSal 数据集^[45,46] 上进行了测试。实验中所比较的方法包括: AC^[108]、BMS^[75]、CA^[70]、COV^[110]、FES^[111]、FT^[32]、GC^[121]、GU^[121]、HC^[29]、HFT^[46]、MSS^[112]、PCA^[71]、RC^[29]、SEG^[113]、SeR^[114]、SIM^[115]、SR^[37]、SUN^[117]、SWD^[119]、TLLT^[59]、以及第 2 章提出的 RPC 方法。这些方法名的英文缩写, 与文献 [28] 或原作者论文保持一致。

3.5.1 数据集简介

(1) ECSSD 数据集简介

ECSSD 数据集^[60,61] 扩展自 CSSD 数据集^[61]^①。严琼等人考虑到虽然 ASD 数据集^[9,32] 中的图像在内容上有较大变化, 但图像背景普遍较为简单。于是从 BSD300^[144]、PASCAL VOC 2012^[145] 以及互联网上收集了 200 张图像构成 CSSD 数据集, 并通过 5 名志愿者对显著物体区域进行了标注, 使测试图像具有更为丰富的前景与背景, 便于将显著物体检测技术更实际地应用到真实的自然场景中。随后, 石建萍等人^[60] 进一步将 CSSD 数据集从 200 张图像扩展为包含 1,000 张图像的 ECSSD 数据集, 新加入的图像具有更为复杂的结构特征, 如背景不再均匀平滑、部分物体区域的外部轮廓模糊、物体颜色与背景颜色相互干扰等。图 3.14 的第 1 行示例了 ECSSD 数据集中的部分图像, 第 2 行是对其中显著物体区域人工标注的结果。

ECSSD 数据集包含的 1,000 张图像中, 共有 1,123 个人工标注的显著物体区域, 其中 909 张图像中仅包含有一个显著物体区域, 剩余的 91 张图像中则至少存在两个显著物体区域。图 3.15(a) 是所有人工标注显著物体的平均标注图, 与 ASD 数据集^[9,32] 类似, ECSSD 也带有中心偏置现象。平均标注图中心点的显著值约为 0.7765 (区间 [0, 1]), 要高于 ASD 数据集, 表明 ECSSD 中有更多的显著物体区域包含图像的中心点。在 ECSSD 数据集的 1,123 个人工标注区域中, 有 286 个区域与图像边界相连接, 数量上

^① CSSD^[61] 和 ECSSD^[60,61] 数据集下载网址: <http://www.cse.cuhk.edu.hk/leojia/projects/hsaliency/dataset.html>。

要远远超过 ASD，它们的平均标注图如图 3.15(b) 所示。可以看到，与整个数据集相比，贴边区域平均上具有更大的尺寸，大部分区域也都包含图像的中心点，且基本上与图像的下边界连接。此外，ECSSD 数据集的图像平均分辨率为 375×311 像素（宽度 \times 高度），接近于 ASD。如果对所有人工标注的显著物体区域的面积大小进行归一化处理，概率密度分布如图 3.15(c) 所示。平均而言，ECSSD 的显著物体区域面积大小约占整个图像面积大小的 23.51%（绿色虚线所示）。概率密度曲线峰值处所对应的 x 轴坐标约为 0.2015（红色虚线所示），意味着在所有人工标注的显著物体区域中，占图像面积大小 20.15% 的区域数量最多，同样接近于 ASD 数据集。



图 3.14 ECSSD 数据集

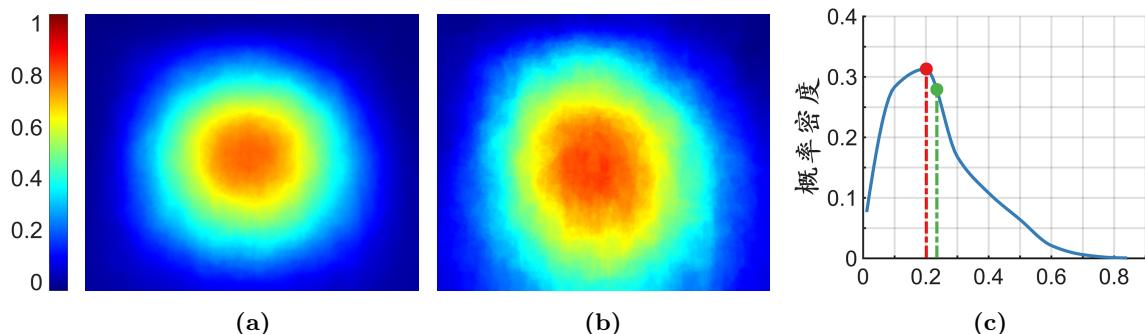


图 3.15 ECSSD 数据集特性统计。(a) 数据集的平均标注图 (归一化到 $[0, 1]$)；(b) 与图像边界连接的显著区域的平均标注图 (归一化到 $[0, 1]$)；(c) 显著区域大小的概率密度曲线： x 轴表示显著区域占图像面积大小的比例

(2) ImgSal 数据集简介

ImgSal 数据集^①是由李健等人所建立^①，他们认为一个性能优秀的显著性检测方法，应该能够同时适用于较大或较小的显著性区域，并且能够应对具有杂乱或重复模式背景的图像，于是利用 Google 收集了 235 张图像构成 ImgSal 数据集。为了体现出显著区域尺寸大小以及图像背景模式的多样性，数据集中包括了 50 张大尺寸的、80 张中等尺寸的、60 张小尺寸的、以及 15 张同时具有大尺寸和小尺寸显著区域的图像，此外另有 15 张图像中包含杂乱背景，15 张图像的背景中带有干扰性的重复模式。可以看到，相较于 ASD^[9,32] 和 ECSSD^[60,61]，该数据集更具有一定的挑战性。19 名志愿者参

^① ImgSal 数据集^[45,46] 下载网址：<http://www.escience.cn/people/jianli/DataBase.html>。

与了 ImgSal 数据集的人工标注工作，除了提供用于显著物体检测任务的二值标注图外，还采集了人眼视点数据用于视点预测任务。图 3.16 的第 1 行示例了 ImgSal 数据集中的部分图像，第 2 行是人工标注的显著物体区域，视点数据的密度图如图 1.4 所示。本章实验中，仅使用二值标注图。



图 3.16 ImgSal 数据集

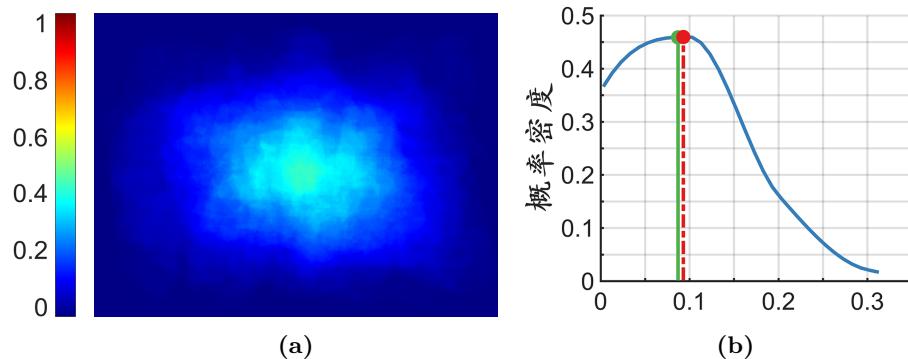


图 3.17 ImgSal 数据集特性统计。(a) 数据集的平均标注图 (归一化到 $[0, 1]$); (b) 显著区域大小的概率密度曲线: x 轴表示显著区域占图像面积大小的比例

ImgSal 数据集包含的 235 张图像中，共有 491 个人工标注的显著物体区域，在数量上要超过图像数的 2 倍，表明平均每幅图像包含约两个显著区域，而 ASD 和 ECSSD 数据集在这一比例上接近于 1.2。在 235 张图像中，有 112 张图像仅包含一个显著物体区域，剩余的 123 张图像中有 62 张至少包含 3 个显著物体区域，约占到数据集的四分之一。图 3.17(a) 是所有人工标注显著物体的平均标注图，ImgSal 同样也带有一定的中心偏置现象。但，其平均标注图中心点的显著值仅为 0.4078，这意味着有 59.22% 的显著物体区域不包含图像的中心点，这一比例要远远高于 ASD 和 ECSSD。但与 ASD 和 ECSSD 不同的是，ImgSal 所有显著区域中并不存在贴边区域，因此这里没有展示平均标注图。此外，ImgSal 数据集中所有图像大小都统一为 640×480 像素 (宽度 \times 高度)。如果对所有人工标注的显著物体区域的面积大小进行归一化处理，概率密度分布如图 3.17(b) 所示。平均而言，ImgSal 的显著物体区域面积大小约占整个图像面积大小的 8.70% (绿色实线所示)，这一比例还不到 ASD 或 ECSSD 的一半，表明 ImgSal 中包含更多的小尺寸显著物体。概率密度曲线峰值处所对应的 x 轴坐标约为 0.0930 (红色虚

线所示), 意味着在所有人工标注的显著物体区域中, 占图像面积大小 9.30% 的区域数量最多, 同样也不到 ASD 或 ECSSD 数据集的一半。

3.5.2 参数分析

CNS 方法共包括 5 个参数, 分别是生成布尔图的采样间隔 δ , 生成注意图的闭运算结构元素半径 ω_c , 后处理算法 I 中重构结构元素半径 ω_r , 以及亮度变换中自适应阈值参数 ϑ_r 和映射曲线参数 ϑ_g 。最优参数的选取过程与第 2 章相同, 即先预设参数取值范围, 调参过程中每次仅对单个参数调参, 而其它参数取值保持固定不变。然后通过分析 MaxF 参数曲线得到最优参数。5 个参数的预设取值范围分别为: $\delta \in [4 : 4 : 40]$ 、 $\omega_c \in [1, 20]$ 、 $\omega_r \in [1, 20]$ 、 $\vartheta_r \in [0.001 : 0.001 : 0.009] \cup [0.01 : 0.01 : 0.1]$ 、 $\vartheta_g \in [1.0 : 0.1 : 3.0]$ 。同时, 考虑到实验包含三个数据集, 为公平起见, 这里采用混合调参方式, 即在分别得到三个数据集的 MaxF 曲线后, 以三条曲线的均值曲线峰值所在位置确定参数的最优取值。虽然这样得到的最终参数值对任意一个数据集而言都非最优参数, 但由于不同数据集之间存在的特性差异, 通过混合调参得到的参数会具有较好的适应性, 便于推广到新的数据集中^①。

图 3.18(a) – 3.18(e) 是对应 5 个参数的参数曲线。各参数曲线中, x 轴表示预设的参数范围, y 轴表示不同参数取值对应的 MaxF 指标, 蓝色、红色和绿色分别代表数据集 ASD^[9,32]、ECSSD^[60,61] 和 ImgSal^[45,46], 黑色为均值曲线, 均值曲线峰值点采用黑色圆点标记, 所有参数曲线的具体指标值见表 3.3。相关分析如下:

(1) 采样间隔参数 δ 引起的性能变化不大, 三个数据集的 MaxF 指标都在一个较小的范围内波动。但由于 δ 决定了布尔图的生成数量 (即 $256/\delta$), 取值大小将直接影响到 CNS 方法的执行效率。当 δ 是一个较大的间隔值时, 各颜色名通道得到的布尔图较少, 可以加快注意图和主注意图的计算速度, 但较少的布尔图又意味着对场景结构刻画的不精确。从表 3.3(a) 中可以看到, 随着 δ 的不断增大, 在到达峰值点 $\delta = 8$ 后, 方法的平均性能开始缓慢下降。

(2) 结构元素半径 ω_c 和 ω_r 具有相似的曲线变化, 随着参数值的不断增大, 方法性能逐步上升, 在到达峰值点之后性能开始缓慢下降。相对而言, 三个数据集中, ω_c 和 ω_r 对 ImgSal 数据集的影响较大, 这与原作者在构造数据集时特意放入尺寸差异较大的显著物体区域有关。此外, 从表 3.3(b) 和 3.3(c) 中可以看到, ω_c 和 ω_r 都在 14 时达到性能峰值, 这也为将来算法中可以省去一个参数提供了实验依据。

(3) 三个数据集的 ϑ_r 参数曲线有一些相似之处, 从表 3.3(d) 中可以看到, 在 $[0.001, 0.01]$ 区间内方法性能几乎没有变化。均值曲线在峰值点 $\vartheta_r = 0.02$ 之后开始下降, 特别是 ImgSal 数据集的下降趋势较为明显。由于参数 ϑ_r 决定了亮度变换的映射

^①需要特别指出的是, 本章已公开的 arXiv 预印本^[46] 中, 部分结果与本章稍有差异, 这是因为之前工作是在多台计算机上运行不同版本的 MATLAB 所致。但笔者发现, 使用不同版本的 MATLAB 确实会对 CNS 方法产生一定的影响。为解决 arXiv 论文中存在的这个问题, 本章统一使用 MATLAB R2014b (版本 8.4) 完成数据集的调参工作, 并修正了部分结果。

阈值 T_F , 反映了显著物体区域面积在整幅图像中的占比。当 ϑ_r 取值越大, 就越可能有更多的背景被提升为高亮的显著区域, 从而拉低了精确率进而影响到 MaxF 指标。因此, 平均显著物体面积最小的 ImgSal 数据集受到参数 ϑ_r 的影响最大。

(4) CNS 方法对参数 ϑ_g 的变化最不敏感, 所有数据集的参数曲线近乎都是一条水平线, 这表明亮度变换过程更多的是依赖映射阈值 T_F , 而受参数 ϑ_g 的影响较小。由于参数 ϑ_g 主要用于在亮度变换过程中抑制较暗的背景区域, 理论取值应 > 1 , 从表 3.3(e) 得到的经验来看, 该参数可在 [1, 2] 区间内取值。

(5) 此外, 参数曲线也可以从侧面反映不同数据集的检测难易程度。可以看到, 三条 MaxF 曲线从高到低排序依次为 ASD、ECSSD、ImgSal, 这说明 ASD 数据集的检测难度最低, 而 ImgSal 数据集最为困难, 这与表 3.5 中报告的结果一致。从表 3.5 中不同显著性方法得到的结果来看, 所有方法平均上在 ImgSal 数据集上得到的指标性能最低, 要低于最高的 ASD 数据集 21.22%, 这也进一步验证了 ImgSal 数据集被经过精心设计, 具有更大的挑战性。通过分析参数曲线, 最终确定 CNS 方法 5 个参数的最优取值如表 3.4 所示。

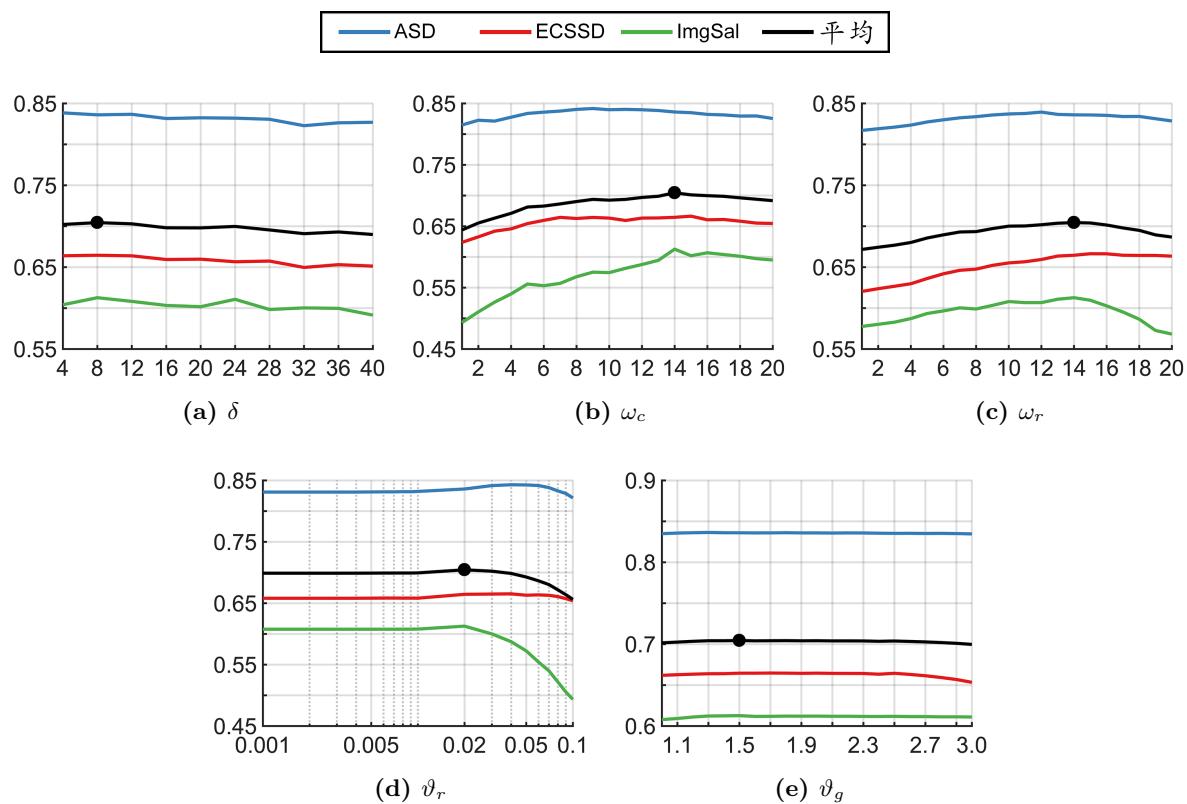


图 3.18 CNS 方法参数分析: x 轴表示参数取值范围, y 轴表示 MaxF 的值

表 3.3 CNS 方法参数分析 (#1 – ASD^[9,32] #2 – ECSSD^[60,61] #3 – ImgSal^[45,46])

(a) δ			(b) ω_c			(c) ω_r			(d) ϑ_r			(e) ϑ_g												
δ	MaxF		平均	ω_c	MaxF		平均	ω_r	MaxF		平均	ϑ_r	MaxF		平均	ϑ_g	MaxF		平均					
	#1	#2			#1	#2			#1	#2			#1	#2			#1	#2						
4	.8385	.6639	.6041	.7022	1	.8149	.6238	.4931	.6439	1	.8171	.6204	.5776	.6717	0.001	.8310	.6581	.6077	.6989	1.0	.8350	.6619	.6079	.7016
8	.8361	.6645	.6127	.7044	2	.8227	.6327	.5105	.6553	2	.8191	.6237	.5802	.6744	0.002	.8310	.6581	.6077	.6989	1.1	.8357	.6627	.6092	.7025
12	.8368	.6639	.6082	.7029	3	.8213	.6420	.5265	.6633	3	.8211	.6267	.5828	.6769	0.003	.8310	.6581	.6077	.6989	1.2	.8361	.6633	.6110	.7035
16	.8316	.6594	.6033	.6981	4	.8276	.6457	.5398	.6710	4	.8236	.6298	.5872	.6802	0.004	.8311	.6581	.6077	.6990	1.3	.8365	.6639	.6123	.7042
20	.8325	.6598	.6017	.6980	5	.8337	.6543	.5559	.6813	5	.8274	.6361	.5934	.6856	0.005	.8313	.6583	.6077	.6991	1.4	.8361	.6640	.6124	.7042
24	.8320	.6566	.6107	.6998	6	.8358	.6596	.5531	.6829	6	.8300	.6419	.5966	.6895	0.006	.8315	.6584	.6077	.6992	1.5	.8361	.6645	.6127	.7044
28	.8307	.6575	.5983	.6955	7	.8376	.6645	.5570	.6864	7	.8325	.6462	.6003	.6930	0.007	.8315	.6585	.6077	.6992	1.6	.8359	.6646	.6118	.7041
32	.8228	.6497	.6003	.6909	8	.8403	.6626	.5678	.6902	8	.8338	.6476	.5989	.6935	0.008	.8317	.6583	.6077	.6993	1.7	.8360	.6647	.6118	.7042
36	.8263	.6531	.5997	.6930	9	.8418	.6643	.5752	.6938	9	.8359	.6522	.6033	.6971	0.009	.8316	.6584	.6078	.6993	1.8	.8362	.6646	.6121	.7043
40	.8270	.6513	.5915	.6899	10	.8398	.6633	.5745	.6925	10	.8371	.6552	.6080	.7001	0.01	.8321	.6583	.6079	.6994	1.9	.8359	.6644	.6120	.7041
					11	.8404	.6594	.5815	.6937	11	.8377	.6566	.6067	.7004	0.02	.8361	.6645	.6127	.7044	2.0	.8359	.6646	.6121	.7042
					12	.8397	.6633	.5876	.6969	12	.8394	.6594	.6067	.7018	0.03	.8416	.6650	.6001	.7022	2.1	.8358	.6643	.6119	.7040
					13	.8385	.6635	.5945	.6988	13	.8366	.6636	.6109	.7037	0.04	.8430	.6653	.5873	.6985	2.2	.8359	.6643	.6119	.7040
					14	.8361	.6645	.6127	.7044	14	.8361	.6645	.6127	.7044	0.05	.8427	.6631	.5724	.6927	2.3	.8358	.6642	.6117	.7039
					15	.8350	.6664	.6019	.7011	15	.8359	.6664	.6096	.7040	0.06	.8417	.6636	.5542	.6865	2.4	.8356	.6633	.6117	.7035
					16	.8322	.6605	.6068	.6998	16	.8355	.6663	.6027	.7015	0.07	.8383	.6629	.5400	.6804	2.5	.8354	.6644	.6118	.7039
					17	.8313	.6610	.6039	.6987	17	.8340	.6646	.5952	.6979	0.08	.8329	.6608	.5217	.6718	2.6	.8354	.6631	.6115	.7033
					18	.8295	.6582	.6011	.6963	18	.8342	.6643	.5864	.6950	0.09	.8288	.6575	.5055	.6640	2.7	.8352	.6614	.6115	.7027
					19	.8296	.6551	.5972	.6940	19	.8313	.6643	.5727	.6894	0.1	.8216	.6540	.4934	.6563	2.8	.8354	.6593	.6112	.7020
					20	.8255	.6545	.5950	.6917	20	.8286	.6635	.5683	.6868						2.9	.8351	.6568	.6112	.7011
																		3.0	.8347	.6534	.6110	.6997		

表 3.4 CNS 方法最优参数

参数名	最优参数值
δ	8
ω_c	14
ω_r	14
ϑ_r	0.02
ϑ_g	1.5

3.5.3 与其它显著性方法的性能比较

CNS 方法与其它 21 个显著性方法性能比较时，利用了已在第 1.2.4.3 节介绍过的性能度量指标。计算各方法在 ASD^[9,32]、ECSSD^[60,61] 和 ImgSal^[45,46] 三个数据集上的精确率 P 、召回率 R 以及 F_β 度量之前，所有显著图都归一化到 [0, 255] 区间。实验中各方法的结果显著图来源如下：

- 对于第 2 章提出的 RPC 方法，利用第 2.5.2 节讨论的最优参数计算得到显著图；
- 对于 BMS^[75]、HFT^[46]、TLLT^[59] 这 3 个方法，通过分别运行原作者发布的 MATLAB 代码生成显著图①；
- 对于剩余的 17 个方法：
 - ASD 数据集^[9,32] 与第 2 章相同，其中 AC^[108]、CA^[70]、FT^[32]、HC^[29]、RC^[29]、SR^[37]6 个方法，直接使用文献^[29]提供的显著图②；对于其余 11 个方法，由于 ASD 数据集是 MSRA10K 数据集^[8,28,121,122] 的子集，因此从 MSRA10K 中检索出所需的 1,000 张显著图③；
 - 对于 ECSSD 数据集^[60,61]，直接使用 A. Borji 等人在显著性基准数据库^[28]中提供的显著图；对于 ImgSal 数据集^[45,46]，通过运行随基准数据库一同发布的 C++ 或 MATLAB 代码生成显著图④。

图 3.19 给出了基于固定分割阈值 T_f 得到的 $P-R$ 曲线图和 F_β 曲线图、以及基于自适应分割阈值 T_a ^[32] 得到的 $P-R-F_\beta$ 柱状图，性能分析如下：

(1) 图 3.19(a) – 3.19(c) 是不同方法的 $P-R$ 曲线图，图中 x 轴表示召回率 R ， y 轴表示精确率 P 。由于 CNS 方法的结果显著图更接近于二值图，使得 $P-R$ 曲线被压缩在一个较窄的范围内。具体到召回率 R ，三个数据集上的范围分别是 [0.6917, 1]、[0.3858, 1]、[0.4832, 1]，这样 CNS 方法的 $P-R$ 曲线就不能够完整地覆盖任何一个方

① BMS^[75]: <http://cs-people.bu.edu/jmzhang/BMS/BMS.html>; HFT^[46]: <http://www.escience.cn/people/jianli/publications.html>; TLLT^[59]: <http://www.escience.cn/people/chengong/Codes.html>。

② 网址: <http://cg.cs.tsinghua.edu.cn/people/~cmm/Saliency/Index.htm>。

③ MSRA10K 数据集^[8,28,121,122] 网址: <http://mmcheng.net/msra10k/>。

④ 显著性基准数据库^[28] 网址: <http://mmcheng.net/zh/salobjbenchmark/>。

法。由于随着 R 值的增大, $P-R$ 曲线逐渐下降, 而较高的 P 值往往伴随着较低的 R 值, 因此, 需进一步借助 F_β 曲线进行评估。

(2) 图 3.19(d) – 3.19(f) 是不同方法的 F_β 曲线图, 其中 x 轴表示固定分割阈值 T_f , y 轴表示 F_β 值。可以看到, CNS 方法的 F_β 曲线与其它方法有很大不同, 其在极窄的阈值范围内快速到达峰值, 而后处于相对平缓的变化中。显然, 这也与结果显著图接近于二值图有关。可以想见, 对 CNS 方法显著图的分割将在很长一段阈值区间内得到相近的分割结果, 使得 F_β 值不会产生太大变化。此外, F_β 曲线峰值代表了方法的最优性能, CNS 方法在 ASD 数据集上, 要低于 TLLT 和 BMS 方法, 排名第三; 在 ECSSD 数据集上, 要低于 RC 方法, 排名第二; 而在 ImgSal 数据集上, 则排名第一。

(3) 图 3.19(g) – 3.19(i) 是不同方法的 $P-R-F_\beta$ 柱状图, 图中以 F_β 值的升序排列:

- 在 ASD 数据集上, CNS 方法的 F_β 值排名第三, 要低于第一名的 TLLT 方法 4.01%、低于第二名的 BMS 方法 1.17%;
- 在 ECSSD 数据集上, CNS 方法的 F_β 值依然排名第三, 低于第一名的 RC 方法 2.08%、低于第二名的 TLLT 方法 0.78%, 但 CNS 方法的 P 值要高出 RC 方法 1.45%;
- 在 ImgSal 数据集上, CNS 方法的 P 值和 F_β 值都排名第一, F_β 值比第二名的 HFT 方法和第三名的 TLLT 方法分别高出 5.73% 和 8.28%, 虽然 HFT 方法在 R 值上有较大性能优势 (高于 CNS 方法 13.08%), 但由于 CNS 方法在 P 值上要高出 8.87%, 因此 CNS 方法的 F_β 值仍然领先于 HFT 方法;
- 三个数据集平均而言, CNS 方法的 F_β 值排名第一, 排名第二和第三的分别是 TLLT 方法和 BMS 方法, CNS 方法的 F_β 值要分别高出 TLLT 和 BMS 方法 1.17% 和 3.88%。

对性能指标定量的评估结果见表 3.5, 表中包括三个数据集上的 AvgF、MaxF、AdpF 三项指标及均值, 各项指标的前三名分别使用红色、绿色、蓝色标记:

(1) 在 ASD 数据集上, TLLT 方法获得了最高指标, CNS 方法在 AvgF、MaxF、AdpF 三项指标上分别排名第二、第三和第三。对于关键的 MaxF 指标, CNS 方法分别低于前两名的 TLLT 和 BMS 方法 3.38% 和 1.94%。三项指标平均而言, CNS 方法排名第二, 低于排名第一的 TLLT 方法 2.68%, 但要高出排名第三的 BMS 方法 2.03%, 这主要是由 BMS 方法较低的 AvgF 指标所致。

(2) 在 ECSSD 数据集上, CNS 方法在 AvgF 指标上获得第一, 而 MaxF 和 AdpF 两项指标的第一名都由 RC 方法获得, CNS 方法分别排名第二和第三。对于关键的 MaxF 指标, CNS 方法要低于排名第一的 RC 方法 2.15%, 但超过排名第三的 TLLT 方法 1.02%。三项指标平均而言, CNS 方法排名第一, 排名第二和第三的分别是 RC 方法和 TLLT 方法, RC 方法略低于 CNS 方法同样也是由其稍低的 AvgF 指标所致。

(3) 在 ImgSal 数据集上, CNS 方法的三项指标都排名第一。对于关键的 MaxF 指标, CNS 方法分别高出排名第二的 HFT 方法和排名第三的 CA 方法 0.48% 和 2.17%。

在 AvgF 指标上, CNS 方法有较大性能优势, 分别高出排名第二的 BMS 方法和排名第三的 TLLT 方法 12.97% 和 13.9%。CNS 方法的性能优势同样也体现在 AdpF 指标上, 分别高出排名第二的 HFT 方法和排名第三的 TLLT 方法 5.73% 和 8.28%。

(4) 整体上, 三个数据集平均而言, CNS 方法排名第一, 三项指标均值分别高出排名第二的 TLLT 方法和排名第三的 BMS 方法 3.38% 和 5.81%。由于本章主要受 BMS 方法^[75] 的启发, 与 BMS 方法比较时, CNS 方法的三项指标分别高出 BMS 方法了 10.64%、2.91% 和 3.88%, 这表明在环绕性线索的基础上引入颜色名线索的有效性。此外, 从表中也可以明显看出本章的 CNS 方法对比第 2 章提出的 RPC 方法(序号 #14)的性能优势, CNS 方法三项指标的均值比 RPC 方法高出 15.38%。

(5) 此外, 从单个方法的横向对比来看, 大部分方法在三个数据集上的性能排序依次为 ASD、ECSSD、ImgSal。从表中最后一行的“平均”项也可以看到, ImgSal 数据集整体上获得了最低性能指标。正如之前所述, ImgSal 数据集具有更大的检测挑战性, 这与发布 ImgSal 的作者精心设计了数据集中包含不同类别的图像有关(详见第 3.5.1 节 ImgSal 数据集简介)。

(6) CNS 方法之所以能在 ASD 和 ECSSD 两个数据集并不明显占优的情况下, 仍然整体上排名第一, 主要得益于在 ImgSal 数据集上的性能优势。对比第 2.5.1 节和第 3.5.1 节三个数据集的特性统计可以看到, 这实际上与 ImgSal 数据集的自身特点有关。ImgSal 数据集中显著物体区域的面积在图像中的占比, 还不到 ASD 或 ECSSD 数据集的一半, 同时 ImgSal 数据集中没有显著物体区域与图像边界相连接(即没有边界对象), 这就在一定程度上反映了本章 CNS 方法所适用的场景。在默认参数下, CNS 方法更适合于检测那些具有中小尺寸显著物体、且这些显著物体都不是边界对象的图像, 而当被检测图像不满足上述条件时, 就有可能导致 CNS 方法检测的失效。在下一节中, 将给出一些具体的检测失败示例, 并展开更为详细的讨论。

最后是不同显著性方法检测结果在视觉上的比较。这里从 21 个横向对比的显著性方法中选取 8 个方法, 图 3.20 给出了 CNS 方法与这 8 个方法对 9 幅输入图像检测得到的显著图。其中, 图 3.20(c) 是 CNS 方法的检测结果, 图 3.20(d) 是第 2 章提出的 RPC 方法的结果。BMS 方法(图 3.20(e))在 ASD 数据集上同样有不错的性能表现, 但在其它两个数据集特别是 ImgSal 数据集上, BMS 方法对显著区域出现了错误的定位。相对而言, HFT 方法(图 3.20(h))在三个数据集上都能大致定位到正确的显著物体区域, 但由于其得到的显著图更偏向于对显著区域的视点预测, 导致对显著物体的轮廓描绘不够精确。值得注意的是 CNS 方法得到的结果显著图, 从中可以很明显地看到更接近于二值图(图 3.20(b)), 这意味着利用这些显著图可以很方便地从场景中分割出显著物体区域。显然, 具有这种形式的显著图会更贴近于显著物体检测的最终任务, 即本质上显著物体检测是用于解决对场景的图形-背景分割^[28]。

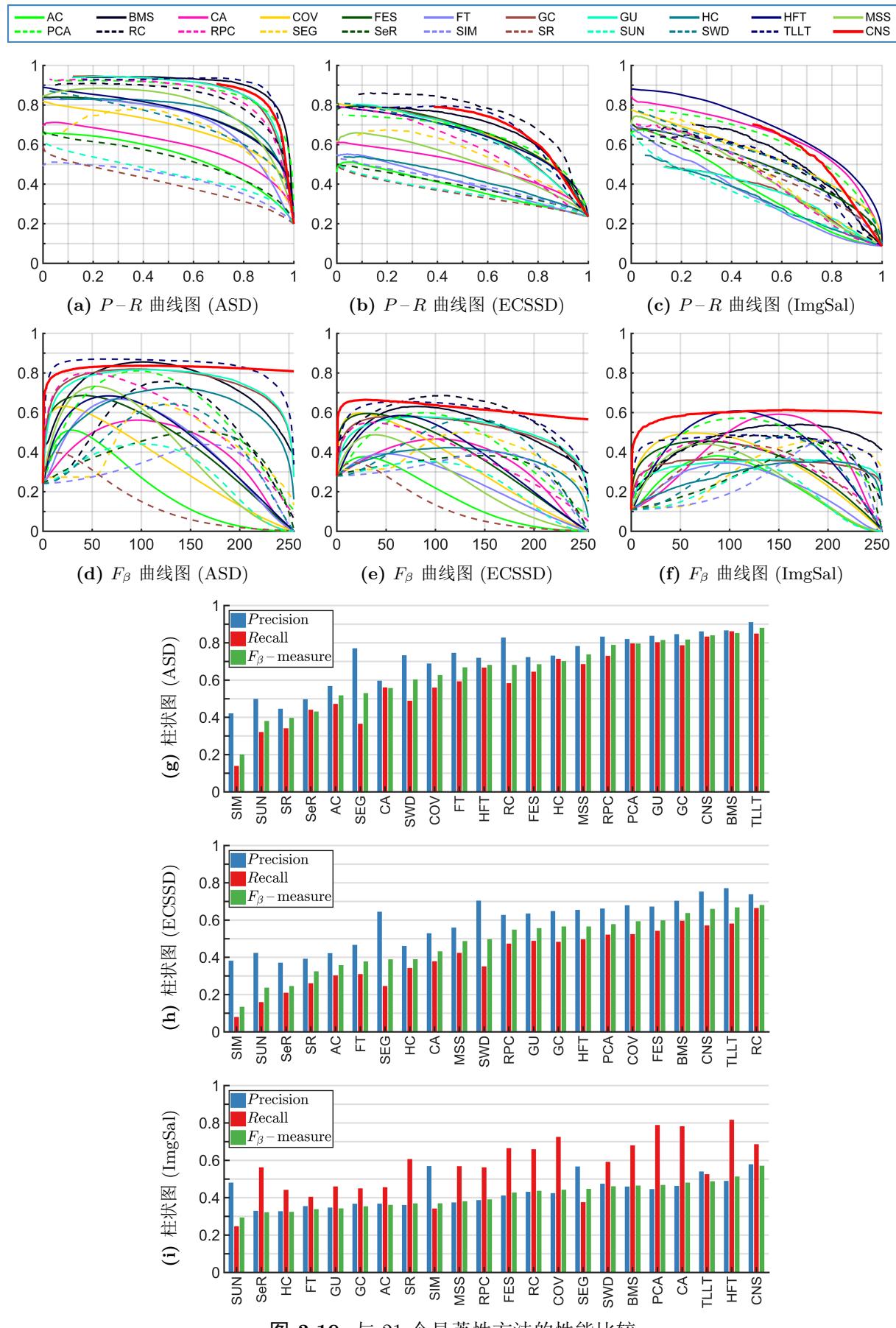


图 3.19 与 21 个显著性方法的性能比较

表 3.5 与 21 个显著性方法的性能比较

序号	方法	ASD [9,32]			ECSSD [60,61]			ImgSal [45,46]			平均		
		AvgF	MaxF	AdpF	AvgF	MaxF	AdpF	AvgF	MaxF	AdpF	AvgF	MaxF	AdpF
1	AC [108]	.2139	.5107	.5174	.1688	.3766	.3575	.2298	.3807	.3611	.2042	.4227	.4120
2	BMS [75]	.7285	.8555	.8515	.5214	.6302	.6370	.4605	.5401	.4646	.5701	.6753	.6510
3	CA [70]	.4043	.5615	.5569	.3403	.4661	.4314	.3913	.5910	.4801	.3786	.5395	.4895
4	COV [110]	.3413	.6305	.6264	.3347	.5973	.5931	.3485	.4960	.4419	.3415	.5746	.5538
5	FES [111]	.4484	.6859	.6840	.3762	.5951	.5976	.3371	.4557	.4268	.3872	.5789	.5695
6	FT [32]	.4342	.6681	.6677	.2419	.3915	.3775	.2234	.3451	.3380	.2998	.4682	.4611
7	GC [121]	.7474	.8193	.8169	.5118	.5814	.5652	.3381	.3642	.3531	.5324	.5883	.5784
8	GU [121]	.7454	.8164	.8141	.5103	.5774	.5558	.3339	.3646	.3419	.5299	.5862	.5706
9	HC [29]	.6113	.7255	.7009	.3642	.4224	.3894	.2849	.3561	.3238	.4202	.5013	.4714
10	HFT [46]	.4526	.6839	.6806	.3739	.5849	.5652	.4254	.6079	.5129	.4173	.6255	.5862
11	MSS [112]	.4116	.7321	.7369	.2543	.4873	.4864	.2656	.4415	.3807	.3105	.5536	.5347
12	PCA [71]	.5884	.8101	.7953	.4252	.5987	.5778	.4415	.5718	.4679	.4850	.6602	.6137
13	RC [29]	.5192	.7570	.6809	.5766	.6860	.6801	.4048	.4871	.4365	.5002	.6434	.5992
14	SEG [113]	.4305	.6485	.5288	.3840	.4990	.3883	.3096	.4569	.4470	.3747	.5348	.4547
15	SeR [114]	.3975	.5037	.4300	.3179	.3818	.2452	.2855	.4513	.3216	.3336	.4456	.3323
16	SIM [115]	.3162	.4384	.2002	.3080	.3998	.1342	.2497	.4626	.3698	.2913	.4336	.2347
17	SR [37]	.1435	.3964	.3964	.1275	.3469	.3246	.3006	.4324	.3687	.1905	.3919	.3632
18	SUN [117]	.2916	.4402	.3803	.2442	.3522	.2365	.1764	.3198	.2937	.2374	.3708	.3035
19	SWD [119]	.4399	.6434	.6033	.4074	.5700	.4971	.3016	.4787	.4605	.3830	.5640	.5203
20	TLLT [59]	.8270	.8699	.8799	.5832	.6543	.6671	.4512	.4878	.4874	.6205	.6707	.6781
21	RPC	.5762	.8002	.7880	.3757	.5499	.5479	.3400	.4598	.3907	.4306	.6033	.5755
22	CNS	.8204	.8361	.8398	.6191	.6645	.6593	.5902	.6127	.5702	.6765	.7044	.6898
平均		.4950	.6742	.6444	.3803	.5188	.4779	.3404	.4620	.4109	.4052	.5517	.5111

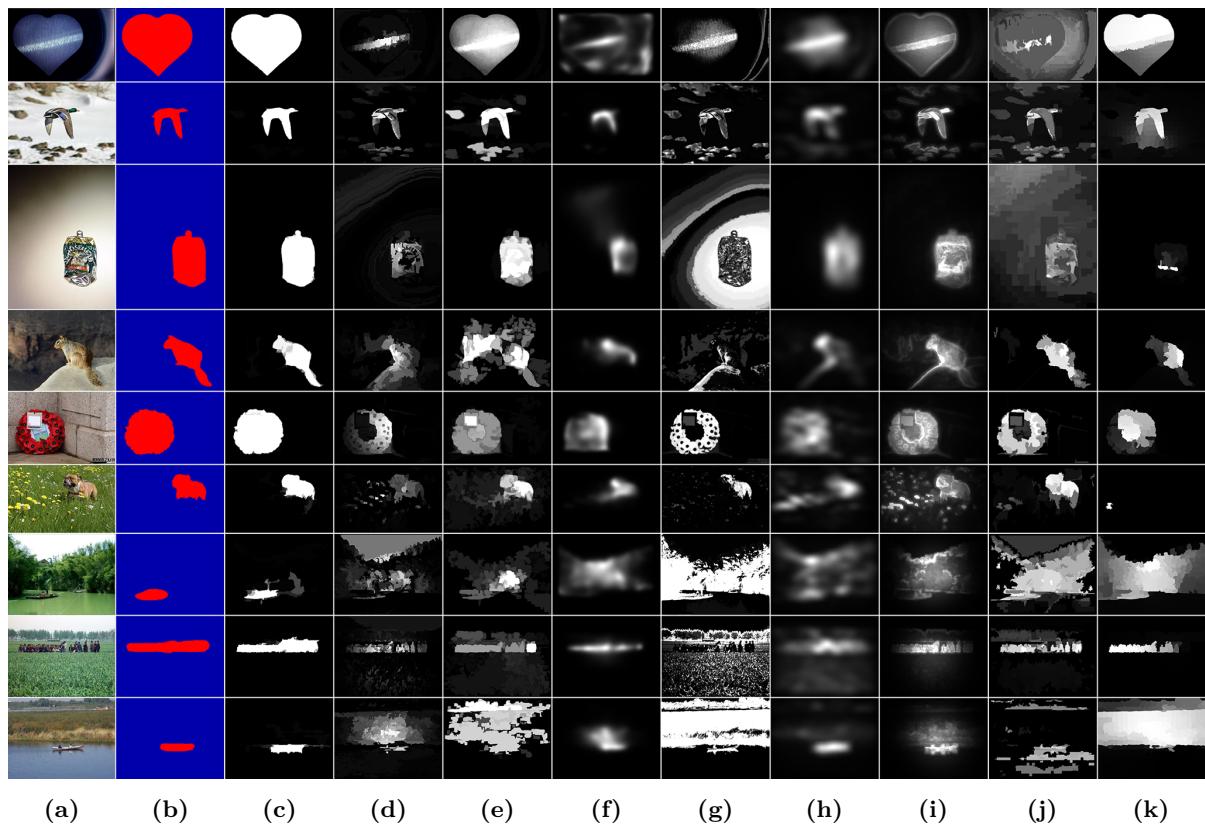


图 3.20 与 8 个显著性方法的视觉比较。(a) 和 (b) 分别是输入图像和人工标注结果, 其中前 3 行、中间 3 行、后 3 行分别来自于 ASD^[9,32]、ECSSD^[60,61]、ImgSal^[45,46] 数据集; (c)–(k) 是分别使用不同方法得到的显著图: (c) 本章提出的 CNS 方法; (d) 第 2 章提出的 RPC 方法; (e) BMS^[75]; (f) FES^[111]; (g) GC^[121]; (h) HFT^[46]; (i) PCA^[71]; (j) RC^[29]; (k) TLLT^[59]

3.5.4 实验结果讨论

虽然通过定量的性能评估和对比分析, 表明本章提出的 CNS 方法在多个数据集上的有效性, 但是该方法仍然存在一定的缺陷。与第 2 章相同, 这里同样利用第 1.2.4.2 节介绍的 MAE 指标, 挑选出六个 MAE 值较大且具有代表性的失败示例, 如图 3.21 所示。图中, 从上至下的各行依次为示例图像、人工标注结果、基于颜色命名模型^[129,130]得到的颜色名图像, 以及 CNS 方法的检测结果。

总体上, CNS 方法的检测失效主要是由以下几个原因所致:

(1) 位置: 在基于环绕性线索生成注意图时, 边界对象将作为非环绕的背景区域被去除。如图 3.21(b) 所示, 虽然颜色名图像显示出显著物体区域与背景之间存在明显的颜色差异, 但由于显著物体恰好是一个边界对象, 使得被去除后产生了错误的注意图, 从而直接导致了基于颜色名进行显著性加权的失效。

(2) 颜色: CNS 方法主要基于颜色名空间实现显著物体检测, 因此由颜色命名模型生成的颜色名通道在很大程度上决定了 CNS 方法的可靠性。图 3.21(c) 和 3.21(d) 中, 虽然显著物体区域本身并非边界对象, 但从颜色名图像可以看出, 与背景的颜色混淆使

得显著物体间接地通过背景区域与图像边界连接，从而成为边界对象在注意图生成阶段被去除。同时，这种颜色的相似性也阻止了显著物体区域从背景中被分离出来。

(3) 大小：虽然在调参阶段，CNS 方法采用混合调参以适应不同的输入图像，但选取出来的最优参数只能照顾到大多数的情况。如图 3.21(e) 和 3.21(f) 所示，对于场景中较小的显著物体区域，形态学处理中采用过大的结构元素会使得物体区域被直接腐蚀，或者与其它环绕性区域连成一片。整体而言，当数据集中大部分是较大的显著物体时，单一的结构元素半径在遇到小区域场景时将面临失效。

(4) 边框：最后是由于数据集中的部分图像含有边框，因边框内部所有区域无法与图像边界相连接，导致了图像背景也无法成为边界对象被有效去除。如图 3.21(a) 所示，CNS 方法生成的显著图将图像边框内的所有部分都判定为显著区域。虽然颜色名图像表明前景图形可以很容易地从背景中分离出来，但边框的存在导致背景也被赋予了极高的显著值，因而阻止了花朵从整个场景中凸显出来。

综上可见，由于本章对显著物体检测问题的解决，主要聚焦于自底向上的图像处理方法。除了场景的结构特征外，更多得依赖于颜色名通道的强度特征和颜色特征，而由颜色命名模型引起的图形与背景间的颜色混淆，在很大程度上限制了 CNS 方法的检测能力。显然，由于颜色名在本质上相当于对输入图像的颜色量化过程，而 11 个颜色名无法应对场景中丰富的颜色信息。近来，有研究人员将 11 个颜色名扩展到 39 个颜色名^[147]，这为更好地解决颜色混淆问题带来了一种新的尝试可能。此外，针对显著物体是边界对象、单一结构元素、以及边框这些容易引起 CNS 方法失效的问题，在第 4 章中，将分别采用背景先验、分层结构和图像边框自动裁剪等多种技术手段予以解决。

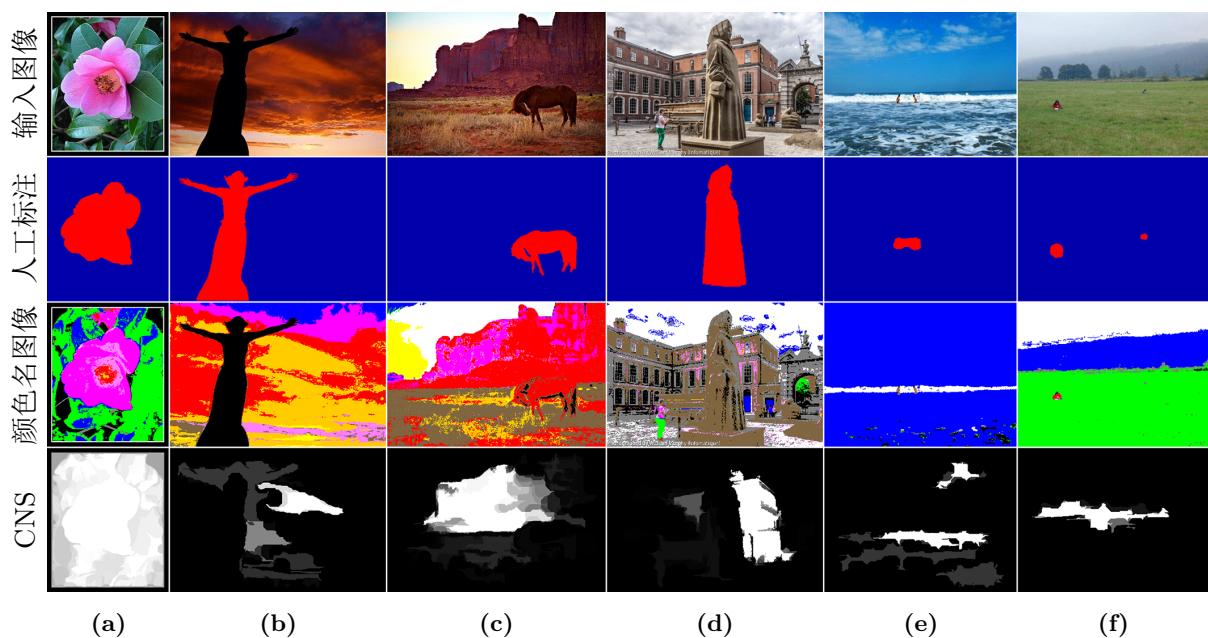


图 3.21 检测失败示例。前 2 列、中间 2 列、后 2 列中的输入图像和人工标注结果，分别来自于 ASD^[9,32]、ECSSD^[60,61] 和 ImgSal^[45,46] 数据集

3.6 本章小结

本章采用环绕性线索和全局颜色名线索，提出了用于单张图像显著物体检测的 CNS 方法。首先，将原始图像转换到包含有 11 个颜色名通道的颜色名空间，对每个颜色名通道采用序列化的阈值分割以及图形–背景分离获得注意图，并对注意图采用线性融合和后处理操作生成显著图；然后将原始图像转换为颜色名图像并计算全局的颜色名统计特征和对比特征，并利用这两种全局颜色名线索对注意图进行加权，生成加权显著图；最后，对显著图和加权显著图采用高亮截断的融合以及后处理操作，得到接近于二值图的显著物体检测结果。在三个数据集上的评估实验结果表明，CNS 方法可有效抑制非显著的背景区域，并在突出显著物体的同时，使其获得均匀且平滑的显著值。

由于本章提出的 CNS 方法是一种自底向上的，由底层的图形–背景结构特征和颜色名特征驱动的计算方法，因此所采用的环绕性线索和颜色名线索从根本上决定了 CNS 方法是否可靠。针对第 3.5.4 节讨论的导致 CNS 方法失效的几个问题，下一步以下几个方面开展相关研究：

(1) 对于显著物体恰好是边界对象的情况，拟采用背景度量机制解决显著物体区域与图像边界重度连接的问题。本文第 4 章将提出一种将 CNS 方法与背景连通性线索相融合的协同显著物体检测方法。

(2) CNS 方法中采用了若干形态学操作，但单一的结构元素不能很好得适应不同尺寸的显著物体，下一步考虑采用分层结构或多尺度方式解决这一问题。在第 4 章中，将采用分层结构并对分层显著图加以融合，生成单图显著图。

(3) 下一步考虑采用颜色名特征以外的其它自顶向下的先验信息，或者将颜色名空间扩展为包含更多的颜色名通道^[147]，以有效解决显著物体区域与图像背景之间的颜色混淆问题。

(4) CNS 方法使用 MATLAB 语言实现具体算法，由于颜色名空间中包含 11 个颜色名通道，加上 MATLAB 语言本身的效率问题，导致算法执行的时间消耗较高。下一步计划优化代码实现或采用混合编程，解决 CNS 方法的执行效率问题。

4 基于分层结构与颜色名的协同显著物体检测

4.1 引言

在第 2 和第 3 章中，分别介绍了两种显著物体检测方法，待检测对象都是单张彩色图像。如第 1.2.1 节所述，不同于针对单张图像的显著物体检测，协同显著物体检测要解决的是如何从一个包含两张或多张图像的图像组中检测出共同显著的前景区域。作为初步的探索性研究，本章先尝试解决针对两张图像（或称为一个图像对）的协同显著物体检测问题。与被广泛采用的在图像内部检测显著性、在图像之间寻找相似性的技术手段相同^[33]，本章也是在单张图像显著性的基础上，通过寻找相似特征模式，来检测两张图像间的协同显著性。因此，本章主要包括以下两个方面内容：

- 单图显著性检测：为了解决第 2 章的 RPC 方法因显著物体颜色是图像中的高频数量颜色而导致的检测失效问题，第 3 章提出了融合环绕性线索^[128] 和颜色名空间的 CNS 方法。然而，从图 3.21 的示例中可以看到，显著物体的大小、位置、以及图像边框等多种因素，都可能导致 CNS 方法的失效。为了解决这些问题，本章引入多种技术手段：针对单一结构元素，受多尺度显著性检测方法^[7,61,70] 的启发，本章设计了一个多层次融合模型来适应不同大小的显著物体区域。在分层结构的每一层中，基于边缘图的自动裁剪算法处理图像边框，并在 CNS 方法的基础上融合基于背景度量的边界连通性线索^[58] 处理边界对象问题。除了用于生成单层显著图外，颜色名线索也被用于对单层显著图的改进以及对多层次显著图间的融合。

- 多图显著性协同：第 2 章提出的 RPC 方法和第 3 章提出的 CNS 方法已经充分验证了颜色特征在显著性检测中起到的重要作用，本章同样基于颜色特征解决显著性的协同问题。从人眼视觉的角度，协同显著区域通常具有语言学上^[135] 高度的颜色相似性，因此本章基于颜色名的比较，剔除单图显著图中的非协同显著区域，得到协同显著图。得益于较好的单图显著图输出，对显著物体的分割直接利用自适应阈值^[32] 完成。

本章提出的协同显著物体检测方法，是一种自底向上的、由底层数据驱动的、基于图像处理技术的方法，可同时获得单张图像的显著图以及两张图像的协同显著图。与 6 个协同显著性方法在 Image Pair 数据集^[34] 上的性能比较，表明了提出方法的有效性。此外，由于本章方法的中间过程可以产生单张图像的显著物体检测结果，因此本章也在 ASD^[9,32]、ECSSD^[60,61] 和 ImgSal^[45,46] 三个数据集上，与包括第 2 章的 RPC 方法以及第 3 章的 CNS 方法在内的 23 个显著性方法进行了性能比较。本章工作的创新点为：在采用分层结构的基础上，提出了一种融合环绕性、颜色名和边界连通性多种线索在内的协同显著物体检测方法，该方法同样适用于解决单张图像的显著物体检测问题，并易于推广到对现有显著性方法的融合操作。

为方便起见，本章将提出的基于分层结构 (Hierarchical Structure) 与颜色名 (Color Name) 的协同显著物体检测方法，简记为“HCN”方法^①。本章内容安排如下：第 4.2 节介绍研究动机与出发点；第 4.3 节介绍 HCN 方法框架；第 4.4 节详细介绍 HCN 方法，包括基于显著图一致性的单层融合与改进算法、基于颜色名稀缺性与单层显著图一致性的多层融合与改进算法、以及基于颜色名一致性的协同显著物体检测算法；第 4.5 节是数据集介绍、参数分析、融合算法的性能评估，以及实验结果和结果分析；最后是本章小结和未来的工作展望。

4.2 研究动机与出发点

人类视觉系统在观察场景中不同尺度的物体时，可以快速感知到它们在视觉特征上的差异，这使得人类对于复杂场景具有很强的实时处理能力。受这一生物学现象的影响，一些显著性方法基于分层或多尺度结构设计检测模型^[7,61,70]，目的是在不同尺度上描述并建模物体或场景的结构性与复杂性。对于大尺度的图像层，虽然可以得到均匀的图像背景和完整的显著区域，但会导致小的显著区域因与背景合并而丢失；而对于小尺度的图像层，虽然可以保留更多的图像细节，但会导致过分割的出现。

针对上述问题，HS 方法^[61] 使用类分水岭算法^[148] 对图像做初始分割，利用三层模型对超像素合并构建分层结构。每一层将区域尺寸设计为区域所能包含最大正方形的边长，当某个区域小于这个边长阈值时，则在 LUV 颜色空间上计算区域的平均颜色，将其与平均颜色最近的区域合并，然后更新合并后区域的平均 LUV 颜色，直至完成整个合并过程。受此启发，本章提出的 HCN 方法同样采用三层结构，但与 HS 方法不同，HCN 方法采用直接给定图像层大小实现图像的多层表示，以简化模型结构设计。实验结果表明，分层结构在多层显著图融合时，可有效提升检测性能。

构建好分层结构后，本章引入第 3 章提出的 CNS 方法，计算分层结构中的各单层显著图。CNS 方法在颜色名空间中将两个全局颜色名线索与环绕性线索结合，可有效弥补仅基于环绕性特征检测的不足。但由于在注意图生成阶段，CNS 方法采用形态学重构去除非环绕的边界区域^[142]，因此当显著物体区域与图像边界相连接时，会导致检测的失效。为解决这一问题，本章将 CNS 方法与基于鲁棒背景检测 (Robust Background Detection) 的 RBD 方法^[58] 相融合。

基于背景先验检测显著区域，因与人类对场景整体把握的感知机制紧密相关，已被诸多显著性方法所采用^[57,149]。RBD 方法首先提出边界连通性 (Boundary Connectivity) 的概念，通过区域落在图像边界上的像素个数与区域面积的比值，来度量不同区域的贴边程度。直观上，物体区域的这一比值会比较小 (通常都接近于 0)，而背景区域的这一比值会比较大 (通常都大于 1)。为了简化背景连通性的计算过程，RBD 方法在对图像做 SLIC 超像素分割^[150] 构造出带权重的无向图后，利用 Johnson 算法^[151] 计算任意两

^① HCN 方法的 MATLAB 源代码、显著图/协同显著图、性能指标结果和幻灯片等资源，已发布在论文主页：<http://www.loujing.com/hcn-co-sod/>。

个相邻超像素之间最短路径的权重之和，从而将区域的边界连通性转换为计算每个超像素的边界连通性。最后通过对比模型计算出背景概率，并结合前景概率以及相邻平滑性，构造一个二次目标函数进行优化，以此得到超像素的显著值。

由于 RBD 方法在任意两个贴边的超像素之间增加了一条边，可改善 CNS 方法因边界对象而导致的检测失效问题。因此本章将其作为一种有效的互补手段，融合 CNS 与 RBD 方法计算单层显著图，所提出的融合与改进算法也可推广到对其他现有显著性方法的融合中。此外，颜色名特征也在本章中被多次采用，不仅与全局对比模型结合，完成对单层/多层融合结果的改进，也被用于解决多图显著性之间的协同问题。

4.3 方法框架

HCN 方法框架如图 4.1 所示，共包含五个模块，其中前四个模块用于计算单张图像的显著图，最后一个模块用于计算协同显著图：

- 模块 I – 单层融合

对于一对图像中的任意一幅（编号分别为 #1 和 #2），在生成三个图像层（记为 $\mathcal{L}^i, i \in \{1, 2, 3\}$ ）后分别调用第 3 章的 CNS 方法以及 RBD 方法^[58]，得到显著图 $\mathcal{L}_{\text{CNS}}^i$ 和 $\mathcal{L}_{\text{RBD}}^i$ ，然后采用显著图一致性实现层内融合，输出“单层融合图”，记为 $\mathcal{L}_{\text{HCN}}^i$ 。

- 模块 II – 单层融合图改进

为了更好地完成多层融合，在将三个单层融合图 $\mathcal{L}_{\text{HCN}}^i$ 送入多层融合模块之前，采用一个改进模块对单层融合图加以改进。改进得到的结果在本章中被称为“单层显著图”（Single-Layer Saliency Map），记为 $\tilde{\mathcal{L}}_{\text{HCN}}^i$ 。

- 模块 III – 多层融合

在将各单层显著图缩放至统一大小后（记为 $\hat{\mathcal{L}}_{\text{HCN}}^i$ ），采用计算平均显著图的方式完成多层显著图之间的融合操作。输出的结果在本章中被称为“多层融合图”，记为 $\hat{\mathcal{L}}_{\text{HCN}}$ 。

- 模块 IV – 多层融合图改进

为了便于协同显著物体检测前的显著物体分割，多层融合图改进模块基于颜色名图像对 $\hat{\mathcal{L}}_{\text{HCN}}$ 加以改进。改进后的结果被称为“单图显著图”（Single-Image Saliency Map），该图也是 HCN 方法针对单张图像检测得到的“显著图”（Saliency Map）。

- 模块 V – 协同显著性检测

在采用自适应阈值分割得到各单图显著图中的显著区域后，协同显著性检测模块采用语言学上的平均颜色作为显著区域的特征表示，通过度量平均颜色差异，定义不同显著区域的相似性。作为协同显著重复模式的判定依据，在采用相似性度量去除非协同显著区域后，得到“协同显著图”（Co-saliency Map）。

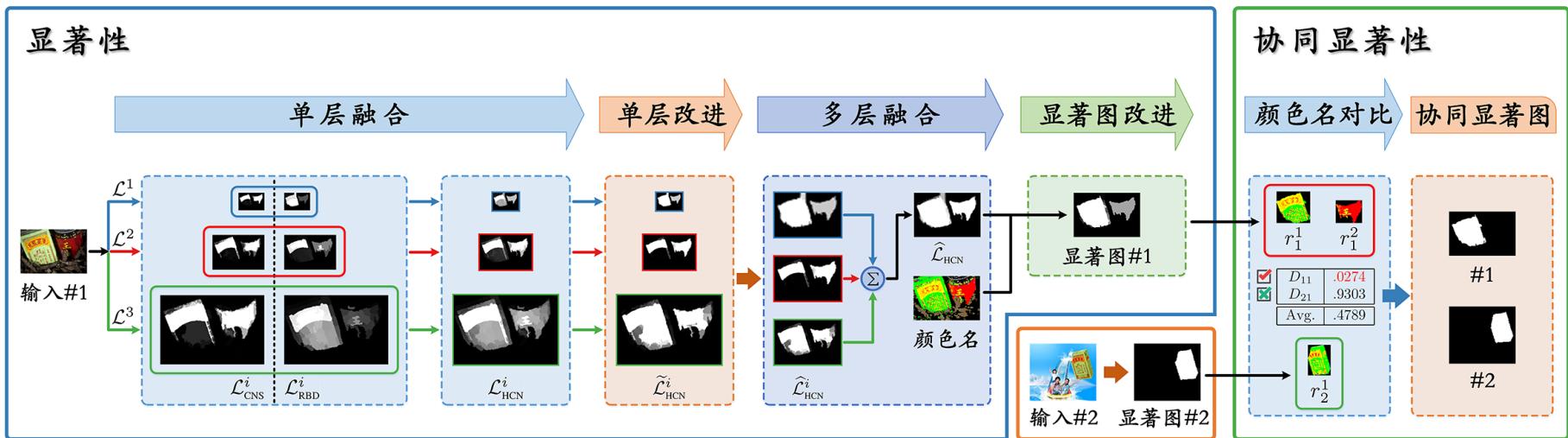


图 4.1 HCN 方法框架

4.4 基于分层结构与颜色名的协同显著物体检测方法

4.4.1 基于显著图一致性的单层融合

第3章提出的CNS方法中，原始图像被统一缩放至400个像素宽度，这在客观上决定了形态学处理中相关结构元素的大小。然而第3.5.3节的示例表明，单一结构元素大小会导致对部分小物体区域的检测失效。从框架图4.1中也可以看到，对于输入图像#1，CNS方法在三层上得到了较为接近的显著图，但RBD方法^[58]明显在第1层上结果更优（见 $\mathcal{L}_{\text{RBD}}^i$ 列）。因此，考虑到对不同显著物体大小的适应能力，本章提出的HCN方法采用三层结构设计，并在每一层上对CNS方法和RBD方法生成的显著图进行融合，得到单层融合图。

4.4.1.1 分层结构设计

HCN方法的分层结构采用一种非常直接的设计，即通过简单得对原始图像进行等比例的倍数采样。三个图像层的像素宽度为 $100 \times 2^i, i \in \{0, 1, 2\}$ ，即：对于最大的图像层 \mathcal{L}^3 ，与CNS方法一样被设置为400个像素宽度；由于实验数据集中图像的平均宽度约为131个像素（见第4.5.1节），因此将最小的图像层 \mathcal{L}^1 设置为100个像素宽度；而中间图像层 \mathcal{L}^2 则被设置为200个像素宽度。这里将 \mathcal{L}^1 设置为小于数据集的平均图像宽度，在一定程度上类似于一个下采样操作，而 \mathcal{L}^2 和 \mathcal{L}^3 则可以看作是连续的两次上采样操作。但这种上/下采样并没有依据原始图像的大小，而是将各层固定为统一宽度，这是为了便于形态学处理时对结构元素大小的确定。

尽管HCN方法通过分层设计避免了仅使用单一结构元素的缺点，也基于参数分析得到了各图像层的最优结构元素大小（详见第4.5.2节），但这种结构设计方式本身还是存在一定的限制。就单个图像层而言，由于没有物体区域大小先验信息，参数分析所得到的层内最优参数，只能兼顾大部分的显著物体区域情况。但由于不同图像内物体区域大小客观上存在差异，导致即便是使用层内最优参数，也无法完全适应每一个显著物体区域。事实上，HCN方法没有试图去寻找针对每一个显著物体的自适应结构元素大小，而是希望通过在不同图像层中发掘出有用的显著性信息，能够为后续的多层融合模块所使用，从而在整体上提升方法的检测性能。在第4.5.3节的实验评估中，可以看到采用多层结构设计，对比单一图像层的性能优势^①。生成图像层的更多细节，将在第4.4.2节中予以介绍。

4.4.1.2 单层融合算法

构建好 \mathcal{L}^1 、 \mathcal{L}^2 和 \mathcal{L}^3 三个图像层后，在每一层上分别调用CNS方法和RBD方法^[58]生成显著图，并归一化到 $[0, 1]$ 区间。在第*i*层上，两个方法得到的显著图分别记

^①在HCN方法的算法实现中，也尝试了其它图像层大小，如下采样到25或50个像素宽度、上采样到800个像素宽度，但得到的结果并不理想。在较小的图像层上，一部分输入得不到任何的显著区域；而在较大的图像层，运行效率明显下降。因此，出于性能和效率的综合考虑，最终确定如上所述的三层结构设计。

为 $\mathcal{L}_{\text{CNS}}^i$ 和 $\mathcal{L}_{\text{RBD}}^i$ ($i \in \{1, 2, 3\}$)。随后，将每层的两个显著图在对应空间坐标上进行逐像素融合，输出单层融合图 $\mathcal{L}_{\text{HCN}}^i$ ，其在坐标空间 (x, y) 处的融合值定义如下

$$\mathcal{L}_{\text{HCN}}^i(x, y) = \underbrace{\left(w_f \mathcal{L}_{\text{CNS}}^i(x, y) + (1 - w_f) \mathcal{L}_{\text{RBD}}^i(x, y) \right)}_{\text{加权融合}} \times \underbrace{\left(2e^{-|\mathcal{L}_{\text{CNS}}^i(x, y) - \mathcal{L}_{\text{RBD}}^i(x, y)|} - 2e^{-1} \right)}_{\text{显著图一致性}} + 1, \quad (4.1)$$

其中，指数函数中的符号 $|\cdot|$ 表示绝对值运算， $w_f \in (0, 1)$ 是一个权重系数。

在上式中， $w_f \mathcal{L}_{\text{CNS}}^i(x, y) + (1 - w_f) \mathcal{L}_{\text{RBD}}^i(x, y)$ 是对两个显著图的加权融合，通过权重系数 w_f 控制各自在融合结果中所占的比重 (w_f 的参数分析见第 4.5.2 节)。考虑到两个方法针对同一个图像层可能检测出不同的显著物体区域，因此增加了一个“显著图一致性项”(为方便起见，以下该项被简称为“一致性项”)，即 $2e^{-|\mathcal{L}_{\text{CNS}}^i(x, y) - \mathcal{L}_{\text{RBD}}^i(x, y)|} - 2e^{-1}$ 。

对一致性项的解释如下：

(1) 在空间坐标 (x, y) 处，如果 CNS 方法和 RBD 方法产生了完全相同的显著性检测结果，则一致性项的值约为 1.26，那么：

- 如果两个方法都认为该像素属于最为显著的前景物体(值为 1)，则一致性项进一步增强了加权融合项得到的显著值；
- 反之，如果两个方法都认为该像素属于最不显著的背景(值为 0)，则由于加权融合项的值为 0，一致性项与之相乘后的结果依然为 0。

(2) 在空间坐标 (x, y) 处，如果 CNS 方法和 RBD 方法产生了截然不同的检测结果，如 CNS 方法认为该像素属于显著的前景物体(值为 1)、但 RBD 方法却认为该像素属于背景(值为 0)，则一致性项的值为 0，该项与加权融合项相乘结果仍然为 0。

(3) 其它情况下，通过一致性项的引入可以进一步修正加权融合项的显著值。显然，一致性项会更倾向于两个显著图中显著值更为一致的像素。对任一像素而言，两个方法的显著性结果越相似，一致性项的值越大，反之，一致性项的值越小。

综上，引入一致性项可在两个方法加权融合的基础上进一步修正融合结果，即可以将一致性项本身看作是一个修正项，避免直接融合时带来的显著性不一致问题。

此外，对式(4.1)的补充说明如下：

(1) 一致性项中采用的是以自然常数 e 为底的指数函数，这种形式会使修正系数属于 $[0, 1.26]$ 区间，从而避免因系数区间过大导致的过修正问题。可以想见，如果修正系数属于一个较大的区间范围，那么两个参与融合的方法在显著性上的细微差异将被修正系数急剧放大，而像素显著值之间过大的差异会使得单层融合结果 $\mathcal{L}_{\text{HCN}}^i$ 不利于之后的多层融合操作。

(2) 在一致性项的基础上叠加了一个“+1”项，该项的引入会使修正系数范围从 $[0, 1.26]$ 变为 $[1, 2.26]$ ，这种处理带来两个好处：

- 考虑没有“+1”项的情况：假设两个参与融合的方法在坐标 (x, y) 处得到了截然不同的显著值，即一个为 0、另一个为 1，则 $\mathcal{L}_{\text{HCN}}^i(x, y) = 0$ ，这意味着融合结果中该位置被判定为不含显著性的背景像素。但实际上，由于无法确定究竟

是哪个方法产生了正确或错误的检测结果，因此不能直接将该像素武断地判定为背景像素。而“+1”项的引入，可以确保在融合方法产生严重分歧的情况下，并不将 $\mathcal{L}_{\text{HCN}}^i(x, y)$ 置为 0，而是赋予一个较小的融合值；

- 如果修正系数属于 $[0, 1.26]$ 区间，则当修正系数 < 1 时，与加权融合项相乘会使得融合结果 $\mathcal{L}_{\text{HCN}}^i(x, y)$ 被反向抑制，从而压缩了单层融合图的动态范围。而“+1”项的引入，会使得修正系数 ≥ 1 ，这样可以确保加权融合项被正向修正，以保留一定的动态范围便于之后的单层融合图改进。

图 4.2 示例了一个单层融合算法的结果。一方面，在 CNS 方法的结果显著图 $\mathcal{L}_{\text{CNS}}^1$ 中，人脸区域被很好地检测了出来，得到了近似于二值的检测结果，但却丢失了属于边界对象的身体区域。另一方面，RBD 方法的结果显著图 $\mathcal{L}_{\text{RBD}}^1$ 中，身体区域被很好地保留，但人脸区域的显著值却较低，且人脸中心有被判定为不显著的背景区域存在。从两个方法的单层融合图 4.2(d) 中可以看到，与 $\mathcal{L}_{\text{CNS}}^1$ 或 $\mathcal{L}_{\text{RBD}}^1$ 相比， $\mathcal{L}_{\text{HCN}}^1$ 有效地弥补了各自的不足，输出了更为理想的融合结果^①。

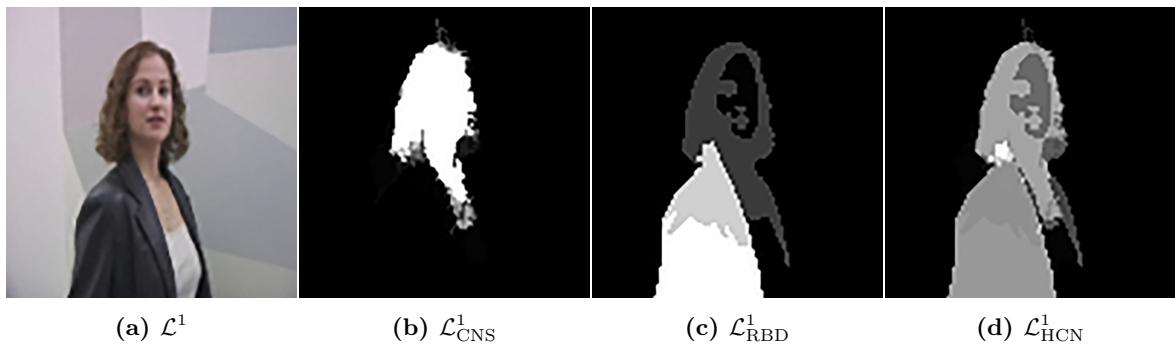


图 4.2 单层融合算法。(a) 某输入图像的 \mathcal{L}^1 图像层；(b) CNS 方法的显著图；(c) RBD 方法^[58] 的显著图；(d) 单层融合图

4.4.1.3 图像边框的影响

在 Image Pair 数据集^[34]中，部分样本图像存在宽度不一的图像边框，如果直接使用来生成三个图像层的话，得到的图像层中仍然带有边框，从而直接影响到 CNS 方法的输出结果（详见第 3.5.4 节的讨论）。图 4.3 给出了这样一个示例，其中图 4.3(a) 是对某输入图像构造的 \mathcal{L}^3 图像层，由于带有黑色边框，边框内部都被认为是非边界对象，从而导致 CNS 方法认为内部所有区域都是显著的，得到图 4.3(b) 所示的显著图。为解决这一问题，这里利用原始图像的边缘图计算边框宽度，完成图像边框的自动裁剪。

具体裁剪过程如下：首先将原始图像转换为灰度图，再利用 Canny 算法^[152] 计算

^① 在单层融合算法的具体实现中，也尝试了其它多种融合技术（如将 CNS 方法和 RBD 方法进行最简单的算术平均、加权平均、或构造不同的加权方式和加权矩阵等），并基于三个图像层的 MaxF 指标均值度量不同融合技术的性能，最终确定采用式 (4.1) 的融合方式。

边缘图，这个过程通过直接调用 MATLAB 内置的 `edge` 函数实现^①。Canny 算法会计算出局部梯度最大值，并基于非极大值抑制及两个动态阈值输出二值边缘图。利用该边缘图，就可以在行/列两个方向上计算出边缘密度。以上边框为例：从第一行开始，计算行内属于边缘像素点的个数（即值为 1 的像素个数），再除以图像宽度就得到了第一行的边缘密度。显然，边缘密度越大，表明行内的边缘像素点数量越多，该行越有可能位于边框与图像内部的交界处。

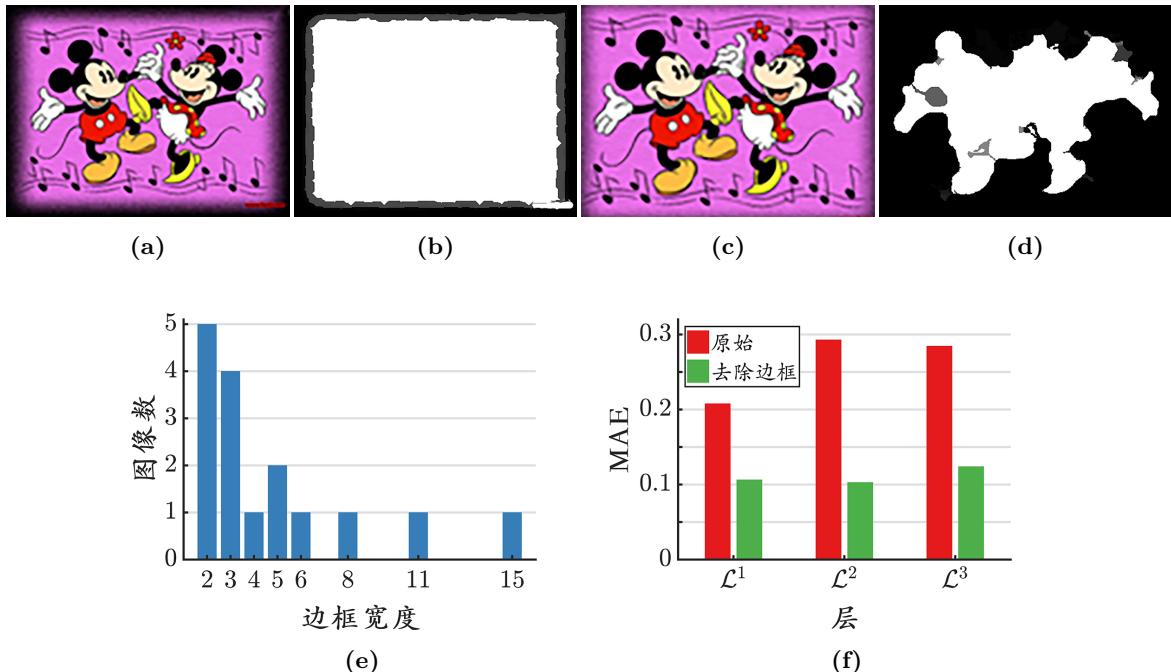


图 4.3 图像边框的影响。(a) 某输入图像的 \mathcal{L}^3 图像层；(b) CNS 方法对图(a)检测得到的显著图；(c) 去除图(a)中的图像边框，(d) CNS 方法对图(c)检测得到的显著图；(e) Image Pair 数据集中边框图像的边框宽度分布；(f) MAE：平均绝对误差（定义详见第 1.2.4.2 节）

在预设最大边框宽度范围内（算法中设为 15 个像素），找到边缘密度大于给定阈值的最下一行（算法中将边缘密度阈值设为 0.7），则该行与第一行的行差即为上边框的宽度；若在预设的最大边框范围内，所有行的边缘密度都小于边缘密度阈值，则表明原始图像中不含有上边框。采用同样的方式，依次从下至上、从左至右、从右至左分别计算出下边框、左边框、右边框的宽度，然后取四个边框宽度的最大值作为图像边框宽度。最后，以边框宽度裁剪掉图像四周对应行/列数的像素，就得到不含边框的裁剪结果。

图 4.3(c) 为利用上述算法对图 4.3(a) 裁剪后的结果，采用 CNS 方法检测得到的显著图如图 4.3(d) 所示。由于裁剪掉了图像边框，CNS 方法产生了更为准确的检测结果。在 Image Pair 数据集中，共有 16 幅图像被检测出含有边框，边框宽度分布如图 4.3(e) 所

^① 值得注意的是，调用不同版本 MATLAB 中的 `edge` 函数得到的边缘图会有所差别，进而影响图像边框宽度大小的检测结果。为统一期间，本章所有方法，包括 CNS、RBD^[58] 和提出的 HCN 方法，都使用 MATLAB R2017a（版本 9.2）运行。

示。如果使用 CNS 方法分别对裁剪前/后的图像进行检测，三个图像层的 MAE 差异如图 4.3(f) 所示。平均而言，裁剪后的 MAE 较裁剪前下降了 57.56%。

回到第 4.4.1.1 节中的分层结构设计，其完整过程如下：原始图像在送入单层融合模块之前，先利用裁剪算法检测并去除图像中可能包含的边框，再生成三个图像层。对于 RBD 方法，关闭其中的去边框操作（即将 *doFrameRemoving* 参数选项设置为“false”），以使 CNS 和 RBD 两个方法在同一个图像层上获得一致的输入。

4.4.2 基于稀缺性与一致性的单层融合图改进

在得到三个图像层的单层融合图后，HCN 方法并没有直接将它们进行多层融合，而是增加了一个改进模块，对单层融合图进行改进。这主要出于两个方面的考虑：一方面，单层融合图本质上也是一个显著图，如本文此前多次所述，显著图最终要为图形-背景分割所服务，因此这里希望通过改进模块产生更为理想的显著性输出；另一方面，三个图像层的融合结果客观上存在一定差异，因此这里也希望通过改进模块，产生较为一致的显著性输出，便于后期的多层融合。

虽然送入单层融合模块的三个图像层大小不同，但具有与原始图像相同颜色分布，因此改进单层融合图的手段同样是引入颜色命名模型^[129,130]。在第 i 层上，基于颜色名的全局对比模型生成对应的权重矩阵，然后在单层融合图 $\mathcal{L}_{\text{HCN}}^i$ 的基础上增加一个“显著图一致性”项，得到改进后的融合结果 J^i ，定义如下

$$J^i = \underbrace{\left(W^i \circ (\mathcal{L}_{\text{CNS}}^i)^{\circ 2} \right)}_{\text{显著图一致性}} \circ \underbrace{\left(W^i \circ (\mathcal{L}_{\text{RBD}}^i)^{\circ 2} \right)}_{\text{颜色名稀缺性}} + (\mathcal{L}_{\text{HCN}}^i)^{\circ 2}, \quad (4.2)$$

其中， W^i 表示第 i 层的权重矩阵，该矩阵与图像层 \mathcal{L}^i 具有相同大小，符号 \circ 和 $\circ 2$ 分别表示 Hadamard 积运算（定义见第 3.4.3.2 节）和二次 Hadamard 幂运算^①。下面首先介绍 W^i 的计算过程，之后解释颜色名稀缺性和上式的具体含义。

式 (4.2) 中，权重矩阵 W^i 是通过图像层 \mathcal{L}^i 的颜色名图像计算得到。由于 \mathcal{L}^i 中已经去除了图像边框，将其直接转换为颜色名图像后，统计各颜色名在图像中的概率，这里将第 j 个颜色名的概率值记为 f_j , $j \in \{1, 2, \dots, N\}$ ，其中 $N = 11$ 为颜色名的总数。权重矩阵 W^i 中任意坐标 (x, y) 处的值，就可以利用颜色名图像中该坐标处隶属的颜色名与其它颜色名之间的全局对比计算得到。假设 $\mathcal{L}^i(x, y)$ 属于第 $k \in \{1, 2, \dots, N\}$ 个颜色名，对应的 RGB 颜色值为 c_k （参见表 3.1），则权重矩阵 W^i 在坐标 (x, y) 处的值为

$$W^i(x, y) = \sum_{j=1}^N f_j \|c_k - c_j\|_2^2, \quad (4.3)$$

其中，符号 $\|\cdot\|_2$ 表示 ℓ_2 范数， c_j 表示第 j 个颜色名的 RGB 颜色值。

^① 给定一个 $m \times n$ 的矩阵 $A = [a_{x,y}]$ 和一个整数 k ，称 $A^{\circ k}$ 是矩阵 A 的 k 次 Hadamard 幂。它仍然是一个 $m \times n$ 的矩阵，其元素定义为矩阵 A 中对应元素的 k 次幂，即 $(A^{\circ k})_{x,y} = (a_{x,y})^k$ ，其中 x 和 y 表示元素坐标。当 $k = 2$ 时， $A^{\circ 2}$ 中各元素为矩阵 A 中对应元素的平方。

初看起来，权重矩阵 W^i 与第 3.4.3.2 节 CNS 方法中权重系数 w_i 的计算方式（即式(3.16)）相同，但这里得到的 W^i 应等价于 CNS 方法中的权重系数 w_i 与颜色名索引矩阵 M_i （见第 3.4.3.1 节）的乘积和，即 $\sum_{i=1}^N w_i M_i$ 。而权重矩阵 W^i 的作用却与 CNS 方法中权重系数 w_i 的作用一致，同样是利用颜色名图像，在场景中发现具有全局稀缺性的颜色名，并对隶属于稀缺颜色名的像素赋予较大的权重，而对于那些隶属于非稀缺颜色名的像素，则赋予相对较小的权重。

为了与多层融合改进模块在形式上统一，式(4.2)等价地写为如下形式

$$J^i = (W^i)^{\circ 2} \circ (\mathcal{L}_{\text{CNS}}^i)^{\circ 2} \circ (\mathcal{L}_{\text{RBD}}^i)^{\circ 2} + (\mathcal{L}_{\text{HCN}}^i)^{\circ 2}, \quad (4.4)$$

如果令

$$C^i = \mathcal{L}_{\text{CNS}}^i \circ \mathcal{L}_{\text{RBD}}^i, \quad (4.5)$$

则式(4.4)可进一步简化为

$$J^i = (W^i \circ C^i)^{\circ 2} + (\mathcal{L}_{\text{HCN}}^i)^{\circ 2}. \quad (4.6)$$

结合以上几个式子，对式(4.2)的解释说明如下：

(1) 二次 Hadamard 幂运算（符号 $\circ 2$ ）的作用是抑制背景。对于一幅 $[0, 1]$ 区间的显著图，在经过 Hadamard 幂运算后，物体前景像素的显著值仍然为 1，背景像素的显著值依然为 0，但较低的显著值被进一步抑制至更低的输出（例如 $0.2 \rightarrow 0.04$ ），因为这部分像素更有可能是背景像素。当然，在此过程中，具有较高显著值的前景像素同样会受到一定程度上的抑制（例如 $0.9 \rightarrow 0.81$ ），但这导致的影响并不大：一是前景像素受抑制的程度，相对于背景像素要小得多；二是对前景的凸显改进，还可以依赖于后处理阶段中的亮度变换操作。

(2) 式(4.2)中，对 $\mathcal{L}_{\text{CNS}}^i$ 和 $\mathcal{L}_{\text{RBD}}^i$ 分别引入权重矩阵 W^i 的目的，是为了与颜色名的全局稀缺性相结合。与第 2 和第 3 章中基于颜色或颜色名的全局稀缺性检测显著性相同，“颜色名稀缺性”项会突出显著图 $\mathcal{L}_{\text{CNS}}^i$ 和 $\mathcal{L}_{\text{RBD}}^i$ 中稀缺的颜色名，使得占图像比例低的、与其它颜色名差异大的颜色名更为显著。在后处理阶段，形态学重构的掩模部分将更多地依赖于此部分像素，并约束形态学变换过程。

(3) 式(4.5)可以看作是 CNS 和 RBD 两个方法在另一种形式上的融合，本质上同样利用了两者的显著图一致性信息。在 $\mathcal{L}_{\text{CNS}}^i \circ \mathcal{L}_{\text{RBD}}^i$ 中，两个方法达成显著性一致的前景像素（值为 1）或背景像素（值为 0）仍然保持原有的输出，即 $(1 \times 1 \rightarrow 1, 0 \times 0 \rightarrow 0)$ ，而在分歧较大的像素上产生较低的输出值（例如 $0.9 \times 0.1 \rightarrow 0.09$ ）。这种融合方式的缺点是会导致显著性较为一致的前景像素的动态范围有所压缩（例如 $0.9 \times 0.8 \rightarrow 0.72$ ），而这同样可以通过亮度变换解决。

(4) 与式(4.1)的融合方式不同，式(4.5)的 $\mathcal{L}_{\text{CNS}}^i \circ \mathcal{L}_{\text{RBD}}^i$ 是一种强一致性融合，着重强调的是两个方法检测一致的前景像素。因此，联合上述第(2)和第(3)点可以看到，式(4.2)中的“显著图一致性”项，实际上是在颜色名全局稀缺性的基础上，寻求两个方法检测得到的共同显著的前景像素。

(5) 最后, 式(4.2)中的 J^i 是在原有融合结果 $\mathcal{L}_{\text{HCN}}^i$ 的基础上叠加了“显著图一致性”项, 其直接效果是将 $\mathcal{L}_{\text{HCN}}^i$ 中同时具有稀缺性和一致性的像素更加突出。对于这部分像素, 稀缺性表征了全局场景中的显著性, 如果两个融合方法又在显著性上达成了高度一致, 那么这部分像素就被视作正确的检测结果。对这部分像素执行后处理操作, 就得到了三个图像层的单层显著图。

对 J^i 执行后处理的伪代码如算法4.1所示, 后处理的目的与第3.4.2.2节的算法3.2相同, 主要是为了在抑制背景的同时、能够凸显物体区域, 并且使物体区域具有平滑一致的显著值。虽然算法中使用了相同的函数名, 但形态学重构和亮度变换两个步骤, 与算法3.2略有不同。

算法 4.1 单层融合图改进的后处理算法

输入: C^i 和 J^i

输出: 单层显著图 $\tilde{\mathcal{L}}_{\text{HCN}}^i$

- | | |
|--|---------|
| 1: $\tilde{\mathcal{L}}_{\text{HCN}}^i = \text{RECONSTRUCT}(C^i, J^i)$ | ▷ 形态学重构 |
| 2: $\tilde{\mathcal{L}}_{\text{HCN}}^i = (\tilde{\mathcal{L}}_{\text{HCN}}^i)^{\circ 2}$ | ▷ 抑制背景 |
| 3: $\tilde{\mathcal{L}}_{\text{HCN}}^i = \text{ADJUST}(\tilde{\mathcal{L}}_{\text{HCN}}^i, t_a)$ | ▷ 突出前景 |
| 4: $\tilde{\mathcal{L}}_{\text{HCN}}^i = \text{FILL}(\tilde{\mathcal{L}}_{\text{HCN}}^i)$ | ▷ 孔洞填充 |
-

- 形态学重构 **RECONSTRUCT**: 形态学重构的作用是去除 J^i 中那些显著值较高的非物体区域。在算法3.2中, 利用一个圆形结构对平均注意图腐蚀得到标记图像, 而后由形态学重构对平均注意图执行开运算操作。但在这里, 由于强一致性融合图 C^i 已经标记出了共同显著的前景区域, 可以直接将其作为标记图像(marker), 由 J^i 作为掩模图像(mask)完成重构过程, 该过程同样使用了快速混合重构算法^[143]。这样, 一方面可以省去为腐蚀操作单独设计结构元素; 另一方面去掉非物体区域的重构结果, 恰好是CNS和RBD达成一致的显著物体区域。

- 亮度变换 **ADJUST**: 算法4.1中第#2行的二次Hadamard幂运算在抑制背景的同时也抑制了前景像素的显著值, 因此采用亮度变换来突出前景像素, 便于之后的多层融合。在算法3.2中, 亮度变换函数通过两个参数确定映射过程, 其中参数 ϑ_r 控制映射阈值上限 T_F , 参数 ϑ_g 控制映射曲线的形状。为了省去这两个参数, 这里直接将经过二次Hadamard幂运算得到的 $\tilde{\mathcal{L}}_{\text{HCN}}^i$ 的均值作为阈值上限(记为 t_a), 将参数 ϑ_g 设为1。这样, $\tilde{\mathcal{L}}_{\text{HCN}}^i$ 中超过 t_a 的像素被直接映射为1, 而区间 $[0, t_a]$ 内的像素则保持输出不变。这是出于两个方面考虑: 一方面, 得益于单层融合改进以及形态学重构过程, $\tilde{\mathcal{L}}_{\text{HCN}}^i$ 中的背景已被足够抑制、且前景物体与背景之间已很好地分离, 但由于此时物体区域的显著值较低, 将映射阈值上限 T_F 设为常见的两倍均值^[32]明显过大, 相对而言单倍均值 t_a 更为合适, 同时由于 t_a 是自适应的, 可以省去参数 ϑ_r ; 另一方面, 从第3.5.2节的参数分析可以看到, 映射过程对参数 ϑ_g 的变化并不敏感, 映射过程主要由 T_F 所决定, 因此这里直接改为线性映射, 省去参数 ϑ_g 。

图 4.4(e) 是对单层融合图 4.2(d) 改进得到的结果，即单层显著图，其中间过程如图 4.4(a) – 4.4(d) 所示。与改进前的单层融合图 $\mathcal{L}_{\text{HCN}}^1$ 相比，单层显著图 $\tilde{\mathcal{L}}_{\text{HCN}}^1$ 更趋向于一个二值输出，其中显著物体的显著值被提升至 1、图像背景的显著值被抑制为 0，这与引入单层融合图改进模块的初衷相一致。

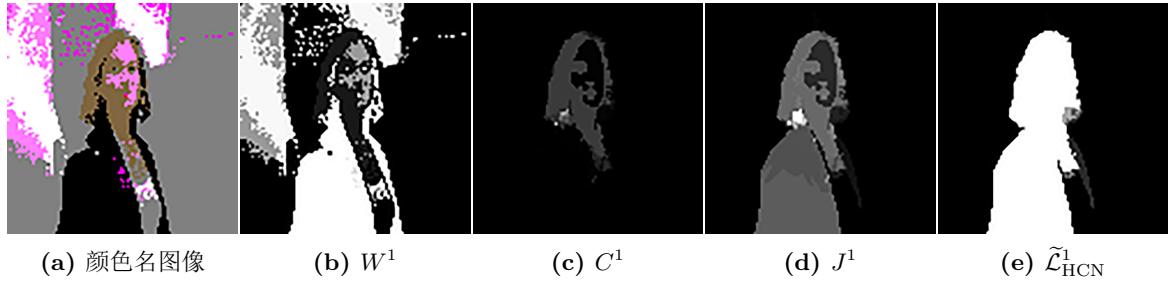


图 4.4 单层融合图改进。(a) 图 4.2(a) 的颜色名图像; (b) 利用图(a)得到的权重矩阵; (c) 式(4.5)的输出结果; (d) 式(4.6)的输出结果; (e) 单层显著图

4.4.3 基于显著图一致性的多层融合

多层处理流程中同样包括两个顺序执行的模块：多层次融合模块和多层次融合图改进模块。在将三个单层显著图 $\tilde{\mathcal{L}}_{\text{HCN}}^i$ 送入多层次融合模块之前，需要统一到相同大小，这个过程依据分层时原始图像是否被检测到含有图像边框来实现。如果原始图像不含边框，则直接将每个 $\tilde{\mathcal{L}}_{\text{HCN}}^i$ 缩放回原图大小；否则，先将 $\tilde{\mathcal{L}}_{\text{HCN}}^i$ 缩放到经过边框裁剪后的图像大小，再在图像四边补全相同边框宽度的像素，由于边框为背景，这部分补全像素的显著值被设为 0。经过上述处理得到的新的单层显著图，记为 $\hat{\mathcal{L}}_{\text{HCN}}^i$ 。

三个单层显著图 $\hat{\mathcal{L}}_{\text{HCN}}^i$ 中，已包含了不同图像层上检测到的显著性信息，而多层次融合算法就是要从这些信息中发掘出跨层的显著一致性，得到理想的多层次融合图。从融合过程要实现的目标来看：对于不同图像层上检测较为一致的共同显著区域，仍保持高显著值或进一步凸显；对于检测较为一致的共同背景区域，则继续保持较低的显著值；而对于检测不一致的显著区域，应在适当兼顾的基础上，通过后续的改进算法进行修正。基于上述考虑，这里提出一种基于平均显著图的加权融合算法来实现多层次融合过程，其中的权重系数是通过各单层显著图与平均显著图之间的差异计算得到。为此，首先定义三个图像层的平均显著图为

$$\bar{\mathcal{L}}_{\text{HCN}} = \frac{1}{3} \sum_{i=1}^3 \hat{\mathcal{L}}_{\text{HCN}}^i. \quad (4.7)$$

第 i 层的单层显著图 $\hat{\mathcal{L}}_{\text{HCN}}^i$ 与平均显著图 $\bar{\mathcal{L}}_{\text{HCN}}$ 的差异值 d_i ，通过逐像素累计的方式得到，即

$$d_i = \frac{1}{M \times N} \sum_{x=1}^M \sum_{y=1}^N |\hat{\mathcal{L}}_{\text{HCN}}^i(x, y) - \bar{\mathcal{L}}_{\text{HCN}}(x, y)|, \quad (4.8)$$

其中， $|\cdot|$ 表示绝对值运算符， M 和 N 分别表示原始图像的宽度和高度。

实际上，式(4.8)计算的是单层显著图 $\widehat{\mathcal{L}}_{\text{HCN}}^i$ 与平均显著图 $\overline{\mathcal{L}}_{\text{HCN}}$ 之间的 MAE 值（定义见第 1.2.4.2 节），即两者整体上的显著性偏差。利用这种偏差关系，可以构造出各层对应的权重系数，并据此得到如下的多层融合结果

$$\widehat{\mathcal{L}}_{\text{HCN}} = \sum_{i=1}^3 \exp \left(-\frac{d_i}{2 \sum_{i=1}^3 d_i} \right) \widehat{\mathcal{L}}_{\text{HCN}}^i, \quad (4.9)$$

其中，由 d_i 构造的指数函数 $\exp(\cdot)$ 的计算结果即为权重系数。

采用上式进行多层融合，主要出于如下考虑：

(1) 平均显著图 $\overline{\mathcal{L}}_{\text{HCN}}$ 体现的是所有单层显著图间的显著一致性，在达成一致的显著区域上会产生较高的显著值。当单层显著图 $\widehat{\mathcal{L}}_{\text{HCN}}^i$ 与 $\overline{\mathcal{L}}_{\text{HCN}}$ 间的显著性差异较小时，会得到一个较小的 d_i ，而利用 d_i 构造的指数函数可以产生一个较大的权重，多层融合结果最终将倾向于与 $\overline{\mathcal{L}}_{\text{HCN}}$ 更为一致的单层显著图。

(2) 指数函数部分的结果属于 $[e^{-1/2}, 1]$ 区间，即权重系数的范围约为 $(0.6, 1]$ ，这意味着虽然多层融合结果将更偏向于权重较大的图像层，但加权后层与层之间的显著值差距不会过大。在空间坐标 (x, y) 处，由于无法确定哪个单层显著图得到的是正确的检测结果，当各层权重存在明显差异时，应选择尽量向三个显著值的均值靠拢。这样，小范围内变化的权重系数可以确保多层融合结果不会“严重”偏向于权重较大的图像层。可以想见，如果权重函数的构造导致当 d_i 较小时产生一个很大的权重，那么其它图像层提供的显著性信息几乎可以忽略不计。

(3) 虽然式(4.8)在计算 d_i 时并未考虑各层显著区域的空间对应关系，且层内所有像素都共享相同的权重，但由于式(4.9)是采用加权叠加的方式计算，所以客观上保留了各层显著区域的位置信息。在空间坐标 (x, y) 处，如果三个单层显著图都同时判定为前景像素或背景像素，由于权重系数之和必定大于 1.8，将使融合结果中前景像素的显著值进一步提升，而背景像素的显著值几乎保持不变^①。物体与背景间显著性差异的进一步加大，将有助于两者的分离，以在协同显著性检测阶段分割出物体区域并从中抽取出所需的特征表示。

图 4.5 示例了多层融合算法的结果，对于原始图像 4.5(a)而言，三个单层显著图（图 4.5(b) – 4.5(d)）之间各有优劣： $\widehat{\mathcal{L}}_{\text{HCN}}^1$ 对物体轮廓的检测最为精确，缺点是身体部分的显著值偏低； $\widehat{\mathcal{L}}_{\text{HCN}}^3$ 身体部分的显著值要明显优于其它两层，但头部的轮廓不如前两者清晰，同时头部上方背景区域的显著值偏高。图 4.5(e) 是多层融合图结果，可以看到，融合结果在保证前景物体轮廓清晰的同时，很好地提升了整个物体区域的显著值。需要说明的是，由于式(4.9)得到的融合结果并不在 $[0, 1]$ 区间内，最终输出 $\widehat{\mathcal{L}}_{\text{HCN}}$ 时使用了

^① 以特殊情况为例，假设三层的 d_i 分别为 $\{0.2, 0.3, 0.5\}$ ，那么三个权重系数就是 $\{0.9048, 0.8607, 0.7788\}$ ，权重系数之和为 2.5443。如果在空间坐标 (x, y) 处，三层都判定属于背景像素（值为 0），则融合结果中该像素的显著值依然为 0；若三层都判定属于前景物体像素（值为 1），则融合结果中该像素的显著值为 2.5443。与直接计算算术平均得到的显著值 0 或 1 相比，物体与背景之间的显著性差异被进一步加大。实际上，在多层融合算法设计时，也尝试了包括简单的直接叠加单层显著图或计算线性平均在内的多种加权方式，也尝试构造了不同的权重系数函数。最终通过三层 MaxF 指标均值评估不同融合机制的检测性能，确定如式(4.9)的融合方式。

饱和运算将显著值大于 1 的部分截断为 1。没有采用归一化操作主要是为了确保物体区域的显著值不被抑制，但带来的副作用是导致部分背景区域的显著值有所提升，例如头部上方的区域。针对这一问题，下一节将引入对多层次融合图的改进算法，以抑制掉这部分背景区域。

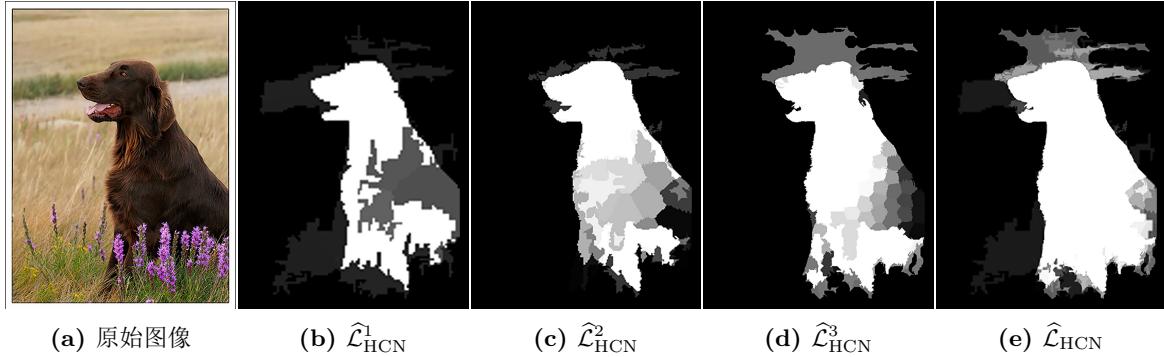


图 4.5 多层融合算法。(a) 原始图像; (b)–(d) 三个单层显著图; (e) 多层融合图

4.4.4 基于稀缺性与一致性的多层次融合图改进

为了解决多层次融合图中部分背景区域显著值偏高的问题，这里参考第 4.4.2 节的方法对多层次融合图进行改进。考虑到前景物体区域的显著值已足够突出，因此仅需要对背景进行抑制处理，这一过程采用形态学重构和 Hadamard 幂运算两步完成（即算法 4.1 中第 #1 和第 #2 行伪代码）。由于涉及到三个单层显著图，因此将针对单层融合图改进的式 (4.5) 和式 (4.6) 扩展为如下形式

$$\widehat{C} = \widehat{\mathcal{L}}_{\text{HCN}}^1 \circ \widehat{\mathcal{L}}_{\text{HCN}}^2 \circ \widehat{\mathcal{L}}_{\text{HCN}}^3, \quad (4.10)$$

$$\widehat{J} = (\widehat{W} \circ \widehat{C})^{\circ 3} + (\widehat{\mathcal{L}}_{\text{HCN}})^{\circ 3}, \quad (4.11)$$

其中， \widehat{W} 是利用原始图像的颜色名图像计算得到的权重矩阵，计算过程与式 (4.3) 相同；符号 $\circ 3$ 表示三次 Hadamard 幂运算符，定义见第 4.4.2 节。

式 (4.10) 中的 \widehat{C} 是对三个单层显著图执行强一致性融合的结果，其中显著值为 1 的部分，同样是共同的显著物体像素，以此作为标记图像 (marker)，将 \widehat{J} 作为掩模图像 (mask) 进行形态学重构，就可以去除掉显著值较高的背景区域。最后，将形态学重构和三次 Hadamard 幂运算两步操作合二为一，得到对多层次融合图 $\widehat{\mathcal{L}}_{\text{HCN}}$ 的改进结果，即单图显著图 S_s

$$S_s = (\text{RECONSTRUCT}(\widehat{C}, \widehat{J}))^{\circ 3}. \quad (4.12)$$

虽然单图显著图 S_s 只是 HCN 方法整个流程中的中间结果，但其也是针对单张输入图像检测得到的显著图。也就是说，HCN 方法本身也是一种可用于单张图像显著物体检测任务的显著性方法。而本章提出的基于 CNS 方法和 RBD 方法^[58] 的融合技术，可以看作是对现有显著性方法融合的一个特例。HCN 方法中所使用到的分层结构设计、

在单层上对两种显著性方法进行融合并改进、再进行多层融合并改进的处理思想，可以很容易推广到现有其它显著性方法中。通过综合两种（或多种）显著性方法的优点，得到更为理想的显著图输出。

图 4.6 示例了对多层融合图 4.5(e) 改进的过程，图 4.6(d) 是改进后得到的单图显著图，这也是 HCN 方法针对原始图像 4.5(a) 检测得到的显著图。与改进前的多层融合图相比，物体区域依然保持较高的显著值，同时背景区域被很好地抑制，包括物体左侧（原显著值约为 0.01）和头部上方在内的区域已全部被抑制为 0，显著图整体上接近于一个二值图像。下一节中，单图显著图的这种二值特性会被加以利用，通过简单阈值分割得到显著物体，并从中抽取出特征表示，完成图像之间的协同显著性检测。

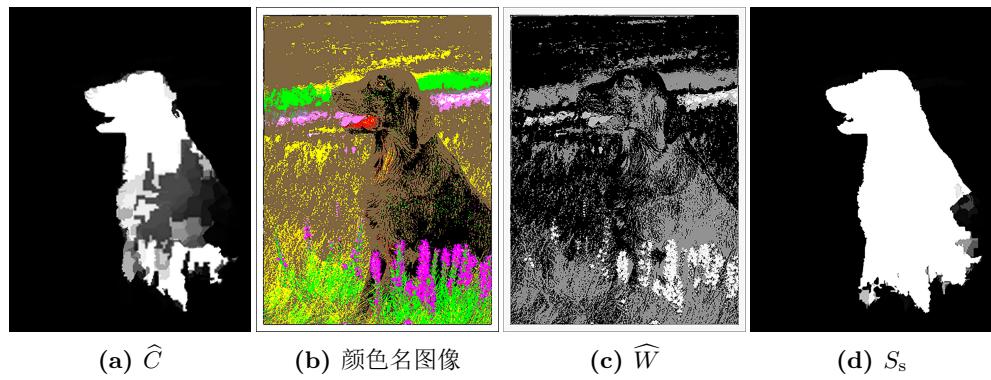


图 4.6 多层融合图改进。(a) 式(4.10)的输出结果；(b) 图 4.5(a)的颜色名图像；(c) 利用图(b)得到的权重矩阵；(d) 图 4.5(a) 的显著图

4.4.5 基于颜色名一致性的协同显著物体检测

针对单层/多层次融合与改进算法得到的单图显著图，协同显著物体检测的任务就是从中寻找到共同显著的前景物体，以输出协同显著图。既然单图显著图中已经包含了显著的前景物体区域，那么协同显著物体检测所要解决的就是如何构造出可以表征这些前景物体的视觉特征，并以此来度量前景物体之间的特征相似性，最后以相似程度作为协同显著性的判定依据。不难看出，这个过程主要包含三个核心问题：1) 如何从单图显著图中抽取出显著物体区域？2) 以何种视觉特征作为显著物体区域的特征表示？3) 如何定义特征表示之间的相似程度？本节将针对上述问题，介绍 HCN 方法的解决手段，并据此实现图像间的协同显著物体检测。

4.4.5.1 基于颜色名的显著物体区域特征表示

在自底向上的协同显著性检测方法中，一种被广泛采用的从多张图像中寻找共同显著物体区域的手段是^[76,81]

$$\text{Co-saliency} = \text{Saliency} \times \text{Repeatedness}, \quad (4.13)$$

即多张图像的协同显著性 (Co-saliency) 是单张图像显著性 (Saliency) 在这些图像间的重复 (Repeatedness)。从视觉角度出发，人类在识别出具体的物体类别之前，已可以快速地从多张图像中捕捉到重复的特征模式，这源于视觉系统对底层特征刺激的快速响应。而颜色作为底层特征表示中重要的一员，其有效性已在第 2 章和第 3 章的工作中被验证，因此，HCN 方法在处理协同显著问题时依然选择利用颜色信息。不同于传统的 RGB 或 LAB 颜色信息，第 3 章和本章多次利用了颜色名，作为一种表征人类在语言学上对场景颜色捕捉和理解能力的特征，HCN 方法将利用颜色名来解决协同显著中的特征度量问题。之所以选择使用颜色名，主要基于以下两个方面的考虑：

一方面，正如在第 2 章中所讨论的，如果以 RGB 颜色空间来度量，一幅图像可能包含几万乃至几十万种颜色。但具体到观察时，人们通常不能辨别出这些颜色间的细微差异，而只会以现有认知中熟悉的颜色名术语描述场景中的主要颜色。以框架图 4.1 中两幅输入图像 #1 和 #2 为例 (即图 4.7(a) 和 4.7(g))，前者包含 8,667 种颜色，后者包含 9,261 种颜色。假设现在要求描述出图像中所包含的颜色，通常只会得到诸如“图 #1 中含有绿色、红色、咖啡色、黑色、白色、黄色，图 #2 中含有绿色、蓝色、白色、黄色”这样的描述，而并不会去谈论具体的 RGB 颜色值。当然，伴随着人类认知和语言能力的发展，生活中实际使用到的颜色名会更加丰富，如草绿色、柠檬色、天蓝色、深红色等，但基本的颜色名术语已足以描述出场景中大部分的颜色信息。

另一方面，对场景的快速理解主要依赖于人类整体上的宏观把握能力，在此过程中，场景里一些不重要的细节信息通常会被人为抛弃。假设现在要求从宏观上描述图像内容时，通常会得到诸如“图 #1 中包含一个绿色的饮料盒和一个红色的饮料罐，图 #2 中包含一个绿色的饮料盒和三个人”这样的描述。显然，站在颜色的角度去看上述信息，“绿色”的饮料盒就是共同的显著物体。现在将这两幅原始图像转换为如图 4.7(b) 和 4.7(h) 所示的颜色名图像，可以发现这恰好与用颜色名术语表征的区域相一致。虽然在绿色的饮料盒内实际上还有一些其它颜色存在，但即便去掉这些颜色细节，也不会影响到人们对整个场景的宏观把握与理解。

基于此，HCN 方法采用颜色名信息完成对显著物体区域的特征表示。该过程包括以下两个步骤，分别解决本节最开始提出的前两个问题：

- **步骤 I—显著物体分割**

之前所有的工作都在朝着“有助于图形–背景分离”的方向所努力，无论是融合图还是改进后的显著图，都试图在抑制背景的同时、凸显物体区域，并让物体区域获得平滑且一致的显著值，由此导致的结果是得到的显著图更趋向于二值图像。因此，借助于简单的阈值分割，就可以很容易将显著物体区域从背景中抽取出来。HCN 方法中，将显著性检测中被广泛采用的显著图均值的两倍^[32] 作为分割阈值，计算公式见式(1.16)。图 4.7(c) 和 4.7(i) 是由 HCN 方法检测得到的单图显著图，对应的分割结果如图 4.7(d) 和 4.7(j) 所示。可以看到，基于一个简单的自适应阈值，显著物体区域就可以被很好地分割出来，这也印证了 HCN 方法的显著图确实能为显著物体分割带来便利。

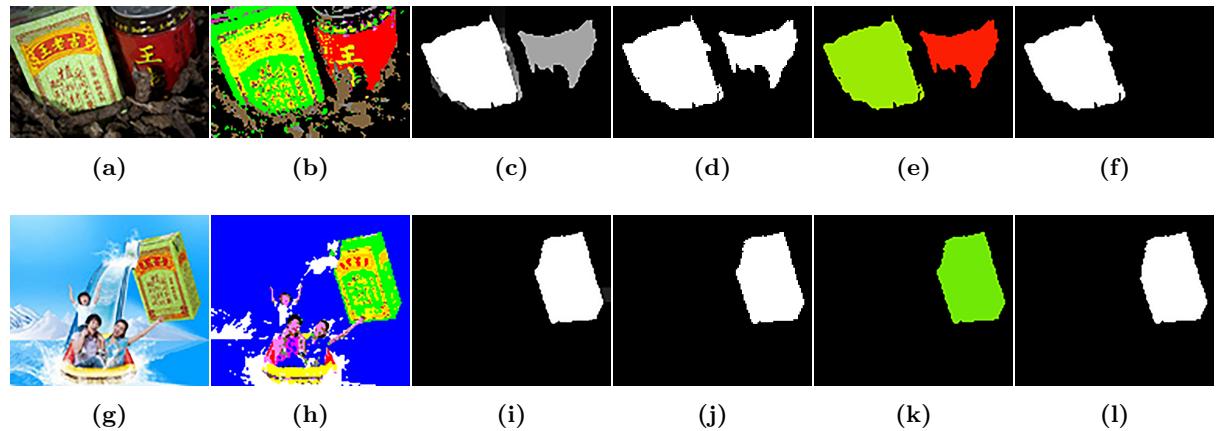


图 4.7 基于颜色名的协同显著性。每行中：第 1 列是原始图像，第 2 列是原始图像的颜色名图像，第 3 列是 HCN 方法的单图显著图，第 4 列是对显著图使用自适应阈值 T_a ^[32] 分割的结果，第 5 列是使用 $A(r)$ 填充的显著物体区域，第 6 列是协同显著图

• 步骤 II – 物体区域特征表示

在分割出显著物体后，就可以相应地从原图中抠出物体区域子图，再将其转换为颜色名图像，并利用所有颜色名的加权平均作为显著物体区域的特征表示。具体的，对于一个显著物体区域 r ，可被如下定义的特征所表示

$$A(r) = \sum_{j=1}^N f_j c_j, \quad (4.14)$$

其中， f_j 是第 j 个颜色名的概率，即该颜色名像素数与子图像素数的比值， c_j 是第 j 个颜色名的 RGB 颜色值， $N = 11$ 是颜色名总数，具体见表 3.1。

式 (4.14) 中的 $A(r)$ 是一个三维向量，本身是一个 RGB 颜色值。从其计算方式来看，概率 f_j 实际上充当的是权重的角色，像素数越多的颜色名被赋予的权重越大，使得 $A(r)$ 更趋近于显著物体区域内部占主导地位的颜色名。当把 $A(r)$ 作为对显著物体区域的特征表示时，其反映的是从语言学的角度对区域颜色的描述，并且这种描述是一种宏观上对整体颜色的把握。图 4.7(e) 和 4.7(k) 是使用 $A(r)$ 对各区域填充的结果，对于饮料盒而言，虽然计算得到的 $A(r)$ 并不是严格意义上的绿色，而是带有“偏绿色”的特征，但这对把握区域整体颜色并没有什么影响。因此，利用这种宏观上的颜色描述，就可以度量不同显著物体区域之间的差异。

4.4.5.2 基于颜色名一致性的非协同显著物体去除

在得到各显著区域的颜色特征表示后，HCN 方法解决协同显著问题的基本想法是利用区域之间的特征差异对比，将那些差异大的区域去除，剩下的就是协同显著物体。为了度量特征之间的差异性，首先对阈值分割得到的每个区域增加一个上/下标号进行标记，其中下标表示区域所在的图像号，上标表示区域在该图像中的 ID 号（流水号）。例如 r_1^i 表示第 1 幅图像中的第 i 个区域， r_2^j 表示第 2 幅图像中的第 j 个区域，由此两

个区域 r_1^i 和 r_2^j 的颜色特征差异就可以定义为

$$D(r_1^i, r_2^j) = \|A(r_1^i) - A(r_2^j)\|_2^2, \quad (4.15)$$

其中, 符号 $\|\cdot\|_2$ 表示 ℓ_2 范数, $A(r_1^i)$ 和 $A(r_2^j)$ 分别是两个区域的颜色特征表示。为方便起见, 将 $D(r_1^i, r_2^j)$ 简记为 D_{ij} , 下标 i, j 的先后顺序表示图像号的先后次序。

D_{ij} 实际上反映了区域 r_1^i 和 r_2^j 在语言学层面上的颜色名一致性, 显然, D_{ij} 越小, 表明两个区域在宏观上的颜色名越相似。因此, 对于本节最开始抛出的第三个问题, 就可以利用 D_{ij} 予以度量。将图像间的显著物体区域两两组合构成区域对 (r_1^i, r_2^j) , 并计算每个区域对的 D_{ij} , 再计算出所有区域对差异的均值 \bar{D} 。对于非协同显著物体区域的判定方法, 是将每个 D_{ij} 与 \bar{D} 依次进行比较: 如果 $D_{ij} > \bar{D}$, 则判定区域 r_1^i 和 r_2^j 为非协同显著区域; 否则, 认为 r_1^i 和 r_2^j 是协同显著区域。举例来说, 图 4.1 中, 显著图 #1 有 r_1^1 和 r_1^2 两个显著物体区域, 图 #2 中有一个显著物体区域 r_2^1 , 两两组合构成了两个区域对 $\{(r_1^1, r_2^1), (r_1^2, r_2^1)\}$ 。计算各区域对的差异得到 $D_{11} = 0.0274$, $D_{21} = 0.9303$, 它们的均值 $\bar{D} = 0.4789$ 。依据判定准则, 由于 $D_{21} > \bar{D}$, 因此区域对 (r_1^2, r_2^1) 中的两个区域属于非协同显著区域。需要说明的是, 由于一个区域可能被多个区域对所包含 (如区域 r_2^1), 因此只要有一次被判定为协同显著物体区域, 则该区域就予以保留。最后, 将非协同显著物体区域从单图显著图中去除, 即在单图显著图内将这些区域所包含像素的显著值全部置为 0, 就得到了最终的协同显著图, 如图 4.7(f) 和 4.7(l) 所示。

4.5 实验与讨论

为了实际评估本章提出的 HCN 方法的有效性, 在 Image Pair 数据集^[34] 上与 6 个协同显著性方法进行性能比较。除可用于检测协同显著物体外, 由于 HCN 方法本身也是一种针对单张图像的显著物体检测方法, 因此与第 3 章相同, 在 ASD^[9,32]、ECSSD^[60,61] 和 ImgSal^[45,46] 三个数据集上与 23 个显著性方法进行了性能比较。

4.5.1 数据集简介

出于研究协同显著物体检测的目的, 李宏亮等人发布了 Image Pair 数据集^[34]。该数据集共有 105 个图像对 (即 210 张图像), 其中包含不同的物体种类, 如花朵、人脸、卡通人物、交通工具和动物等。除原始图像外, 原作者还提供了精确的、像素级的人工标注结果, 供协同显著物体检测算法性能比较所使用, 部分示例图像和标注结果如图 1.3 所示^①。整个 Image Pair 数据集中, 一共包含 242 个人工标注的显著物体区域, 其中 191 张图像仅包含一个显著区域。图 4.8(a) 为所有人工标注区域的平均标注图, 可以看到, 该数据集同样带有明显的中心偏置现象, 其平均标注图中心点的显著值约为 0.8471 (区间 [0, 1]), 要明显高于之前使用的 ASD^[9,32]、ECSSD^[60,61] 和 ImgSal^[45,46] 数据集。在 242 个人工标注的显著物体区域中, 有 45 个区域与图像边界相连接, 平均标

^① Image Pair 数据集^[34] 下载网址: <http://ivipc.uestc.edu.cn/hlli/projects/cosaliency.html>。

注图如图 4.8(b) 所示，接近于 ECSSD 的贴边区域平均标注图。此外，整个数据集图像的平均分辨率为 131×105 像素（宽度 \times 高度），基本上都属于小尺寸图像。如果将人工标注显著物体区域的面积大小进行归一化处理，概率密度分布如图 4.8(c) 所示。平均而言，Image Pair 数据集中，显著物体区域的面积大小约占整个图像面积大小的 23.87%（绿色虚线所示），这一比例接近于 ASD 和 ECSSD。概率密度曲线峰值处所对应的 x 轴坐标约为 0.2003（红色虚线所示），这意味着在所有人工标注的显著物体区域中，占图像面积大小 20.03% 的区域数量最多，这一比例同样与 ASD 和 ECSSD 两个数据集类似。

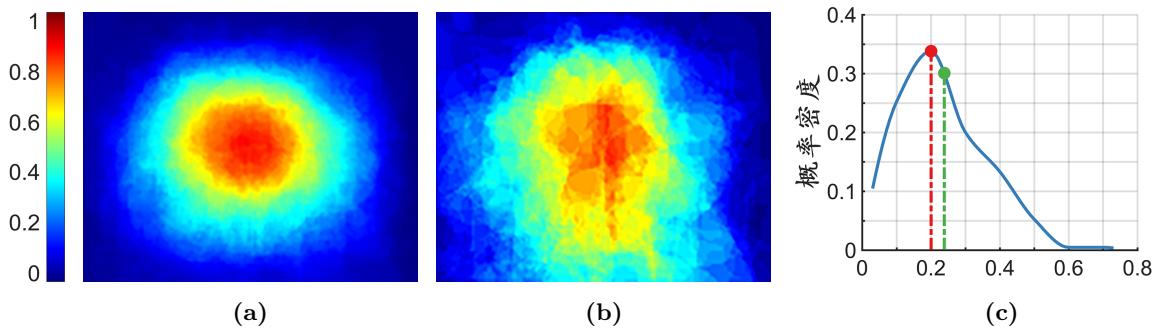


图 4.8 Image Pair 数据集特性统计。(a) 数据集的平均标注图 (归一化到 $[0, 1]$)；(b) 与图像边界连接的显著区域的平均标注图 (归一化到 $[0, 1]$)；(c) 显著区域大小的概率密度曲线： x 轴表示显著区域占图像面积大小的比例

4.5.2 参数分析

HCN 方法包含 6 个参数，其中 5 个参数来自于第 3 章的 CNS 方法，即采样间隔 δ ，结构元素半径 ω_c 和 ω_r ，以及亮度变换参数 ϑ_r 和 ϑ_g 。考虑到分层结构中各图像层的大小不同，且 ω_c 和 ω_r 会对 HCN 方法的结果产生实质性影响，因此对这两个参数在每个图像层上分别调参。对于剩下的 3 个参数 δ 、 ϑ_r 和 ϑ_g ，由于它们对结果的影响较小，不再分别进行调参操作，而是三个图像层共享参数值。上述 5 个参数预设的取值范围，与 CNS 方法相同。此外，HCN 方法需要一个额外的权重系数 w_f ，用于控制单层融合。该参数的取值范围被经验地设置为 $[0.1 : 0.1 : 0.9]$ ，为了简化参数分析， w_f 同样在三个图像层上共享参数值。

对 HCN 方法的调参过程与第 2 和第 3 章相同，即采用 MaxF 作为评价指标，以 MaxF 曲线峰值处的 x 坐标作为参数的最优取值。图 4.9(a) – 4.9(f) 是对应 6 个参数的参数曲线，其中 x 轴表示预设的参数范围、 y 轴表示 MaxF 的值，三个图像层以及三层的 MaxF 均值曲线分别用蓝色、红色、绿色、黑色表示。由于参数 ω_c 和 ω_r 在每个图像层上分别调参，各峰值点的颜色与对应层的曲线颜色相同；剩下的 4 个参数，仅以黑色标记 MaxF 均值曲线的峰值点。所有参数曲线的具体指标值见表 4.1，表中对参数 ω_c 和 ω_r 在各图像层的 MaxF 最大值分别使用蓝色、红色、绿色标记；其它 4 个参数仅以红色标记 MaxF 的最大均值。

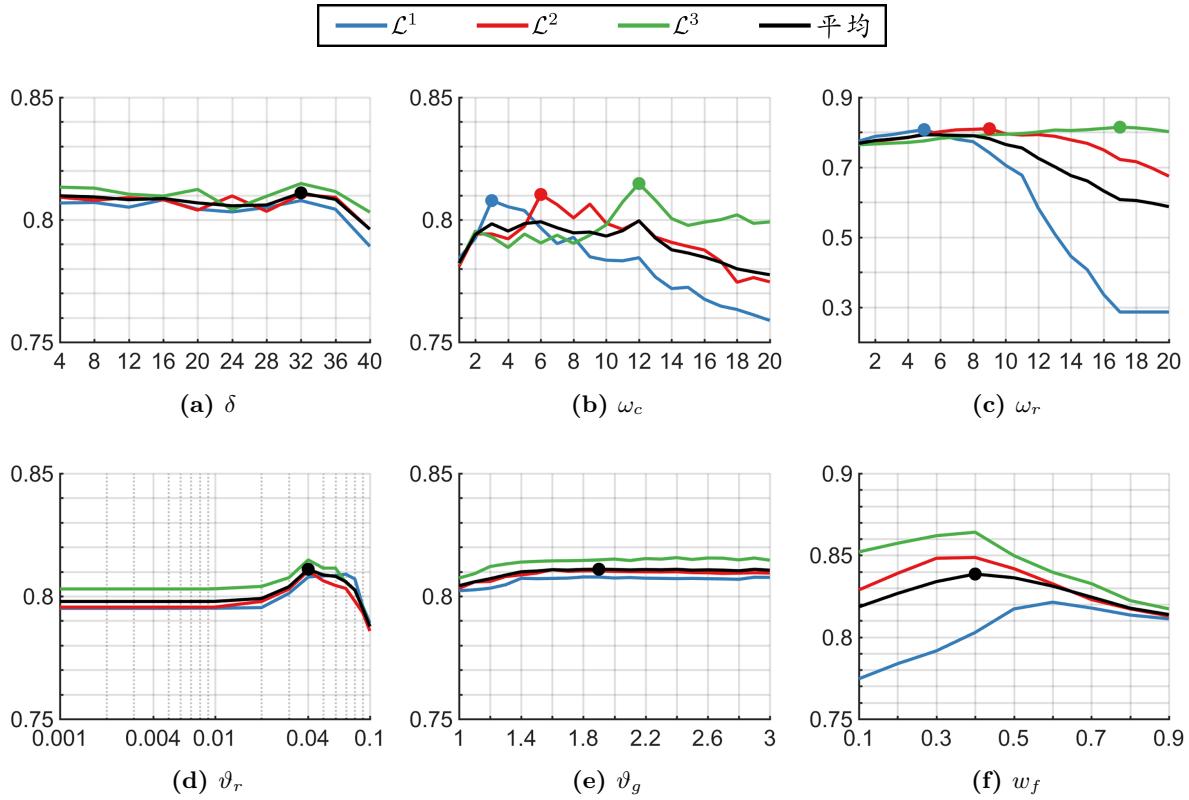


图 4.9 HCN 方法参数分析: x 轴表示参数取值范围, y 轴表示 MaxF 的值

对参数曲线的分析如下:

(1) 多层共享的参数中, HCN 方法对于 δ 、 ϑ_r 和 ϑ_g 这三个参数不敏感, 尤以 ϑ_g 的曲线变化最为平缓^①。

(2) 同样作为多层共享的参数, 权重系数 w_f 对方法性能有一定影响。在上升至峰值点 $w_f = 0.4$ 后, 方法性能随着 w_f 的进一步增大而逐渐下降, 整个变化幅度约为 5%。最优取值 0.4 意味着在单层融合时, CNS 方法和 RBD 方法的占比分别为 40% 和 60%, 说明两个方法都已充分参与到融合过程中。如果权重系数 w_f 的取值过小或过大, 则融合结果会严重偏向其中某个方法, 这种情况下得到的方法性能反而不优。

(3) 对于分层调参的两个参数 ω_c 和 ω_r , 参数曲线表明图像层的大小决定了参数的最优取值。显然, 由于这两个参数控制着形态学处理结构元素的大小, 因此在较大的图像层上会得到较大的最优取值, 而尺寸较小的图像层会对参数的变化更为敏感。从参数曲线中可以看到, 在到达峰值点后, \mathcal{L}^1 层性能的下降趋势更快。

(4) 就三个图像层间的比较而言, 无论是曲线均值还是曲线峰值, \mathcal{L}^3 层的指标更具有性能优势, 这表明 HCN 方法在较大的图像层上获得了更好的检测结果。但是, 由于更大尺寸的图像层意味着更高的计算复杂度, 导致在性能提升的同时、相应带来了执

^①表 4.1(e) 中的 MaxF 指标仅保留小数点后 4 位数字, 因此存在 3 个相同的大值 0.8110。如果保留小数点后 8 位数字, 则当 ϑ_g 分别取 {1.9, 2.4, 2.9} 时, 对应的 MaxF 值分别为 {0.81098816, 0.81095868, 0.81095547}, 据此参数 ϑ_g 的最优取值被确定为 1.9。

行效率的降低。虽然本章并未系统地分析图像层大小对 HCN 方法性能的影响，但正如第 4.4.1.1 节中所提到的，在尝试了 $\{25, 50, 800\}$ 等多种图像层大小后，最后确定将分层结构中三个图像层的大小固定为 $\{100, 200, 400\}$ 个像素宽度，这也是平衡检测性能和算法效率的一种折衷考虑。

综上，最终确定的 HCN 方法最优参数的取值见表 4.2。

表 4.1 HCN 方法参数分析

(a) δ		(b) ω_c			(c) ω_r							
δ	MaxF			平均	ω_c	MaxF			ω_r	MaxF		
	\mathcal{L}^1	\mathcal{L}^2	\mathcal{L}^3			\mathcal{L}^1	\mathcal{L}^2	\mathcal{L}^3		\mathcal{L}^1	\mathcal{L}^2	\mathcal{L}^3
4	.8068	.8092	.8133	.8098	1	.7841	.7809	.7819	1	.7749	.7662	.7645
8	.8071	.8080	.8129	.8093	2	.7924	.7942	.7954	2	.7886	.7744	.7669
12	.8051	.8090	.8104	.8082	3	.8078	.7942	.7930	3	.7933	.7796	.7690
16	.8082	.8082	.8097	.8087	4	.8053	.7922	.7887	4	.8010	.7850	.7711
20	.8043	.8039	.8124	.8069	5	.8038	.7973	.7942	5	.8078	.7954	.7755
24	.8032	.8097	.8042	.8057	6	.7967	.8103	.7905	6	.7931	.8017	.7828
28	.8050	.8035	.8096	.8060	7	.7903	.8062	.7937	7	.7803	.8073	.7867
32	.8078	.8103	.8148	.8110	8	.7928	.8008	.7904	8	.7733	.8086	.7899
36	.8043	.8091	.8115	.8083	9	.7849	.8064	.7937	9	.7416	.8103	.7934
40	.7892	.7961	.8031	.7961	10	.7835	.7986	.7979	10	.7056	.7955	.7943
					11	.7832	.7961	.8072	11	.6771	.7923	.7965
					12	.7845	.7995	.8148	12	.5829	.7935	.8009
					13	.7766	.7930	.8080	13	.5098	.7888	.8066
					14	.7719	.7908	.8005	14	.4453	.7784	.8053
					15	.7724	.7891	.7977	15	.4070	.7684	.8076
					16	.7675	.7876	.7990	16	.3363	.7497	.8112
					17	.7647	.7831	.8000	17	.2861	.7224	.8148
					18	.7633	.7745	.8020	18	.2861	.7160	.8127
					19	.7611	.7764	.7985	19	.2861	.6970	.8082
					20	.7589	.7746	.7991	20	.2861	.6745	.8020

(d) ϑ_r

ϑ_r	MaxF			平均
	\mathcal{L}^1	\mathcal{L}^2	\mathcal{L}^3	
0.001	.7951	.7956	.8030	.7979
0.002	.7951	.7956	.8030	.7979
0.003	.7951	.7956	.8030	.7979
0.004	.7951	.7956	.8030	.7979
0.005	.7951	.7956	.8030	.7979
0.006	.7951	.7956	.8030	.7979
0.007	.7951	.7956	.8030	.7979
0.008	.7951	.7956	.8030	.7979
0.009	.7951	.7956	.8030	.7979
0.01	.7951	.7956	.8031	.7979
0.02	.7954	.7979	.8040	.7991
0.03	.8011	.8029	.8076	.8039
0.04	.8078	.8103	.8148	.8110
0.05	.8083	.8063	.8114	.8087
0.06	.8084	.8044	.8114	.8081
0.07	.8090	.8032	.8056	.8059
0.08	.8070	.7980	.8025	.8025
0.09	.7960	.7933	.7954	.7949
0.1	.7892	.7858	.7879	.7877

(e) ϑ_g

ϑ_g	MaxF			平均
	\mathcal{L}^1	\mathcal{L}^2	\mathcal{L}^3	
1.0	.8023	.8032	.8074	.8043
1.1	.8026	.8058	.8092	.8059
1.2	.8033	.8060	.8121	.8071
1.3	.8047	.8080	.8131	.8086
1.4	.8073	.8087	.8139	.8100
1.5	.8072	.8094	.8142	.8103
1.6	.8073	.8108	.8144	.8108
1.7	.8074	.8102	.8145	.8107
1.8	.8078	.8103	.8146	.8109
1.9	.8078	.8103	.8148	.8110
2.0	.8074	.8101	.8151	.8109
2.1	.8076	.8100	.8145	.8107
2.2	.8073	.8099	.8153	.8109
2.3	.8073	.8099	.8151	.8108
2.4	.8072	.8100	.8157	.8110
2.5	.8073	.8096	.8150	.8106
2.6	.8071	.8096	.8156	.8108
2.7	.8071	.8093	.8155	.8106
2.8	.8069	.8094	.8148	.8104
2.9	.8077	.8096	.8156	.8110
3.0	.8077	.8095	.8147	.8106

(f) w_f

w_f	MaxF			平均
	\mathcal{L}^1	\mathcal{L}^2	\mathcal{L}^3	
0.1	.7746	.8289	.8521	.8185
0.2	.7838	.8390	.8574	.8267
0.3	.7916	.8482	.8620	.8339
0.4	.8029	.8487	.8641	.8386
0.5	.8172	.8418	.8498	.8363
0.6	.8213	.8326	.8396	.8312
0.7	.8178	.8229	.8327	.8245
0.8	.8135	.8173	.8223	.8177
0.9	.8112	.8129	.8173	.8138

表 4.2 HCN 方法最优参数

(a) 层间共享参数		(b) 层内参数		
参数名	最优参数值	最优参数值		
		\mathcal{L}^1	\mathcal{L}^2	\mathcal{L}^3
δ	32			
ϑ_r	0.04			
ϑ_g	1.9			
w_f	0.4			
ω_c		3	6	12
ω_r		5	9	17

4.5.3 融合算法的性能评估

HCN 方法中包含两个融合模块，一是第 4.4.1 节介绍的单层融合模块，二是第 4.4.3 节介绍的多层融合模块。之前已详细讨论了各模块的设计目的和理论意义，本节将对融合前后的性能变化做一个定量评估，以实际验证融合算法的有效性。

- 单层融合算法的性能评估

通过对比参与融合的 CNS 方法和 RBD 方法^[58]，评估单层融合与改进算法是否能够提升检测性能。表 4.3 中给出了不同方法在三个图像层上的 MaxF 指标以及三层的均值，需要说明的是，由于各图像层与人工标注图像的大小不同，MaxF 指标是通过将人工标注图像缩放到对应图像层大小后计算得到。

表 4.3 单层融合算法的性能比较

	$i = 1$	$i = 2$	$i = 3$	平均
$\mathcal{L}_{\text{CNS}}^i$.8078	.8103	.8148	.8110
$\mathcal{L}_{\text{RBD}}^i$.7597	.8288	.8526	.8137
$\mathcal{L}_{\text{HCN}}^i$.8029	.8487	.8641	.8386
$\tilde{\mathcal{L}}_{\text{HCN}}^i$.8148	.8510	.8657	.8438

(1) 对于两个参与融合的方法，CNS 方法在三个图像层上获得了较为接近的性能，同时在 \mathcal{L}^1 层上也要超过 RBD 方法。对比 CNS 方法，RBD 方法在后两层上的性能更优，表明其更适合于在较大的图像层上进行显著性检测。三层平均来看，两个方法的整体上较为接近。

(2) 对于单层融合结果即单层融合图 $\mathcal{L}_{\text{HCN}}^i$ ，图像层越大，方法性能越好。 $\mathcal{L}_{\text{HCN}}^i$ 除了在 \mathcal{L}^1 层上略低于 CNS 方法，其它都要优于 CNS 或 RBD 方法。三层平均来看， $\mathcal{L}_{\text{HCN}}^i$ 比 $\mathcal{L}_{\text{CNS}}^i$ 性能高出了 2.76%，比 $\mathcal{L}_{\text{RBD}}^i$ 高出了 2.49%，这表明在现有显著性方法的基础上执行的融合操作，有助于改进检测性能。

(3) 对于单层融合图的改进结果 $\tilde{\mathcal{L}}_{\text{HCN}}^i$ ，其在所有图像层上的性能都超过了改进前

的 $\mathcal{L}_{\text{HCN}}^i$, 表明了改进算法的有效性。虽然从三层平均来看, 性能只提升了 0.52%, 但改进算法的目的主要是为了凸显前景物体区域, 以便于后续的多层融合。

- 多层融合算法的性能评估

表 4.4 给出了多层融合的评估结果。需要说明的是, 表中 $\widehat{\mathcal{L}}_{\text{HCN}}^i$ ($i \in \{1, 2, 3\}$) 是对单层显著图 $\widetilde{\mathcal{L}}_{\text{HCN}}^i$ 进行图像缩放和边框补全的结果 (详见 4.4.3 节), 因此 $\widehat{\mathcal{L}}_{\text{HCN}}^i$ 与人工标注图像具有相同的尺寸, 可直接用于计算 AvgF、MaxF 和 AdpF 指标。

(1) 随着图像层的增大, 单层显著图在三个指标性能上都有所提升, 表明 HCN 方法的单层融合机制更适用于较大的图像层。三层平均来看, $\widehat{\mathcal{L}}_{\text{HCN}}^3$ 整体上要比 $\widehat{\mathcal{L}}_{\text{HCN}}^1$ 高出了 5.49%, 比 $\widehat{\mathcal{L}}_{\text{HCN}}^2$ 高出了 1.6%。

(2) 对于多层融合结果即单图显著图 S_s , 在各项指标上都要超过各单层显著图。虽然从三层平均来看, 这种性能上的提升有限 (仅比三层中性能最好的 $\widehat{\mathcal{L}}_{\text{HCN}}^3$ 高出了 0.16%), 但对比表 4.3 就可以看出 HCN 方法整体上的有效性。以最为关键的 MaxF 指标为例: S_s 比 CNS 方法的三层均值 (0.8110) 高出了 5.11%, 比 RBD 方法的三层均值 (0.8137) 高出了 4.84%; 即便是对比两个方法性能最高的 \mathcal{L}^3 层 ($\mathcal{L}_{\text{CNS}}^3$ 为 0.8148、 $\mathcal{L}_{\text{RBD}}^3$ 为 0.8526), S_s 也要分别高出 4.73% 和 0.95%。这说明作为同样适用于单张图像显著物体检测的 HCN 方法, 其利用的对现有显著性方法融合的技术, 可以有效提升检测性能。

表 4.4 多层融合算法的性能比较

	AvgF	MaxF	AdpF	平均
$\widehat{\mathcal{L}}_{\text{HCN}}^1$.7995	.8148	.8027	.8056
$\widehat{\mathcal{L}}_{\text{HCN}}^2$.8391	.8510	.8435	.8445
$\widehat{\mathcal{L}}_{\text{HCN}}^3$.8568	.8657	.8591	.8605
S_s	.8587	.8663	.8611	.8621

4.5.4 与其它显著性/协同显著性方法的性能比较

由于 HCN 方法可以同时输出显著图和协同显著图, 本章的性能比较分为两部分进行, 分别使用下标 “s” 和 “co” 区分比较的是显著性方法还是协同显著性方法。

- 对于 HCN_s : 分别与 AC^[108]、BMS^[75]、CA^[70]、COV^[110]、FES^[111]、FT^[32]、GC^[121]、GU^[121]、HC^[29]、HFT^[46]、MSS^[112]、PCA^[71]、RBD^[58]、RC^[29]、SEG^[113]、SeR^[114]、SIM^[115]、SR^[37]、SUN^[117]、SWD^[119]、TLLT^[59], 以及第 2 章提出的 RPC 方法和第 3 章提出的 CNS 方法, 共计 23 个显著性方法进行比较。其中前 21 个方法名的英文缩写, 与文献 [28] 或原作者论文保持一致。
- 对于 HCN_{co} : 分别与 CoIRS^[77]、CBCS^[76]、CSHS^[79]、IPCS^[34]、IPTDIM^[19]、SACS^[78] 共计 6 个协同显著性方法进行比较。这些方法名的英文缩写, 与文献 [19] 或原作者论文保持一致。

4.5.4.1 与显著性方法的性能比较

23 个参与性能比较的显著性方法的显著图来源如下：

- 对于第 2 章提出的 RPC 方法，利用第 2.5.2 节讨论的最优参数得到显著图；
- 对于第 3 章提出的 CNS 方法，利用第 4.5.2 节讨论的最优参数得到显著图；
- 对于 RBD 方法，使用原作者发布的 MATLAB 代码生成显著图^①。
- 其它 20 个方法显著图的获取过程与第 3 章相同（详见第 3.5.3 节）。

图 4.10 给出了三个数据集上基于固定分割阈值 T_f 得到的 $P-R$ 曲线图和 F_β 曲线图、以及基于自适应分割阈值 T_a ^[32] 得到的 $P-R-F_\beta$ 柱状图，性能分析如下：

(1) 图 4.10(a) – 4.10(c) 是不同方法的 $P-R$ 曲线图，图中 x 轴表示召回率 R ， y 轴表示精确率 P 。在每个数据集上，当固定分割阈值 $T_f = 0$ 时，HCN 方法与其它方法都汇聚到同一点，此时的召回率 $R = 1$ ，精确率 P 表示各数据集中显著物体平均在图像中的占比（见第 2.5.1 节和第 3.5.1 节）。具体为：ASD 数据集 $P = 0.1985$ ，ECSSD 数据集 $P = 0.2351$ ，ImgSal 数据集 $P = 0.0870$ 。

(2) 经过后处理算法后，HCN 方法的结果显著图更接近于二值图像，导致 $P-R$ 曲线被压缩在一个相当窄的范围内。在各数据集上，除汇聚点外，召回率 R 的范围分别是：ASD 数据集 [0.9205, 0.9493]，ECSSD 数据集 [0.6866, 0.7827]，ImgSal 数据集 [0.6639, 0.7458]，使得 HCN_s 的 $P-R$ 曲线并不能够完整地覆盖任何一个方法。在 ImgSal 数据集上，HCN 方法 $P-R$ 曲线被 CA、PCA、HFT 以及第 3 章提出的 CNS 方法完整覆盖，表明在该数据集上 HCN 不具有性能优势。由于随着 R 值的增大， $P-R$ 曲线逐渐下降，而较高的 P 值往往伴随着较低的 R 值，因此需要进一步通过 F_β 曲线评估各方法的性能。

(3) 图 4.10(d) – 4.10(f) 是不同方法的 F_β 曲线图，其中 x 轴表示固定分割阈值 T_f ， y 轴表示 F_β 值。在每个数据集上，当固定分割阈值 $T_f = 0$ 时，HCN 方法与其它方法都汇聚到同一点，具体为：ASD 数据集 $F_\beta = 0.2406$ ，ECSSD 数据集 $F_\beta = 0.2794$ ，ImgSal 数据集 $F_\beta = 0.1087$ 。除汇聚点外，HCN 方法的 F_β 曲线相较其它方法要更为平缓，各数据集上 F_β 值的范围分别是：ASD 数据集 [0.8652, 0.9062]，ECSSD 数据集 [0.6654, 0.7049]，ImgSal 数据集 [0.4292, 0.5040]， F_β 曲线较小的变化将使得对 HCN 方法结果显著图的分割得到更为相近的分割结果。此外， F_β 曲线的峰值点代表了方法的最优性能，HCN 方法在 ASD 和 ECSSD 两个数据集上都排名第一，但在 ImgSal 数据集上却不具有性能优势。

(4) 图 4.10(g) – 4.10(i) 是不同方法的 $P-R-F_\beta$ 柱状图，图中以 F_β 值的升序排列。在 ASD 和 ECSSD 两个数据集上，HCN 方法都排名第一；但在 ImgSal 数据集上，第 3 章提出的 CNS 方法排名第一，HCN 方法仅排名第三。

^① 网址：<https://www.microsoft.com/en-us/research/people/yichenw/>。

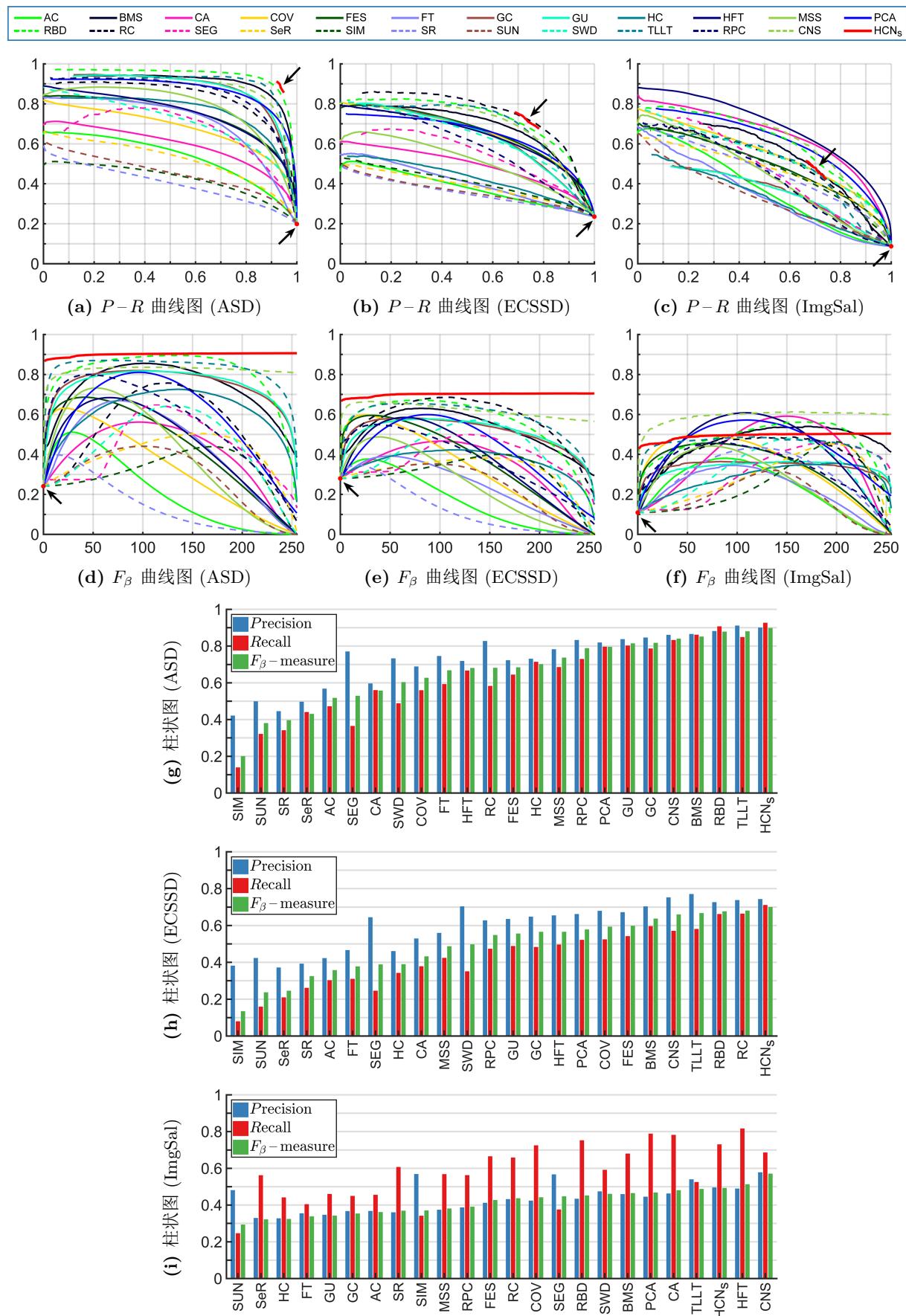


图 4.10 与 23 个显著性方法的性能比较

表 4.5 与 23 个显著性方法的性能比较

序号	方法	ASD [9,32]			ECSSD [60,61]			ImgSal [45,46]			平均		
		AvgF	MaxF	AdpF	AvgF	MaxF	AdpF	AvgF	MaxF	AdpF	AvgF	MaxF	AdpF
1	AC [108]	.2139	.5107	.5174	.1688	.3766	.3575	.2298	.3807	.3611	.2042	.4227	.4120
2	BMS [75]	.7285	.8555	.8515	.5214	.6302	.6370	.4605	.5401	.4646	.5701	.6753	.6510
3	CA [70]	.4043	.5615	.5569	.3403	.4661	.4314	.3913	.5910	.4801	.3786	.5395	.4895
4	COV [110]	.3413	.6305	.6264	.3347	.5973	.5931	.3485	.4960	.4419	.3415	.5746	.5538
5	FES [111]	.4484	.6859	.6840	.3762	.5951	.5976	.3371	.4557	.4268	.3872	.5789	.5695
6	FT [32]	.4342	.6681	.6677	.2419	.3915	.3775	.2234	.3451	.3380	.2998	.4682	.4611
7	GC [121]	.7474	.8193	.8169	.5118	.5814	.5652	.3381	.3642	.3531	.5324	.5883	.5784
8	GU [121]	.7454	.8164	.8141	.5103	.5774	.5558	.3339	.3646	.3419	.5299	.5862	.5706
9	HC [29]	.6113	.7255	.7009	.3642	.4224	.3894	.2849	.3561	.3238	.4202	.5013	.4714
10	HFT [46]	.4526	.6839	.6806	.3739	.5849	.5652	.4254	.6079	.5129	.4173	.6255	.5862
11	MSS [112]	.4116	.7321	.7369	.2543	.4873	.4864	.2656	.4415	.3807	.3105	.5536	.5347
12	PCA [71]	.5884	.8101	.7953	.4252	.5987	.5778	.4415	.5718	.4679	.4850	.6602	.6137
13	RBD [58]	.8070	.8951	.8779	.5615	.6716	.6762	.4659	.5629	.4517	.6115	.7099	.6686
14	RC [29]	.5192	.7570	.6809	.5766	.6860	.6801	.4048	.4871	.4365	.5002	.6434	.5992
15	SEG [113]	.4305	.6485	.5288	.3840	.4990	.3883	.3096	.4569	.4470	.3747	.5348	.4547
16	SeR [114]	.3975	.5037	.4300	.3179	.3818	.2452	.2855	.4513	.3216	.3336	.4456	.3323
17	SIM [115]	.3162	.4384	.2002	.3080	.3998	.1342	.2497	.4626	.3698	.2913	.4336	.2347
18	SR [37]	.1435	.3964	.3964	.1275	.3469	.3246	.3006	.4324	.3687	.1905	.3919	.3632
19	SUN [117]	.2916	.4402	.3803	.2442	.3522	.2365	.1764	.3198	.2937	.2374	.3708	.3035
20	SWD [119]	.4399	.6434	.6033	.4074	.5700	.4971	.3016	.4787	.4605	.3830	.5640	.5203
21	TLLT [59]	.8270	.8699	.8799	.5832	.6543	.6671	.4512	.4878	.4874	.6205	.6707	.6781
22	RPC	.5762	.8002	.7880	.3757	.5499	.5479	.3400	.4598	.3907	.4306	.6033	.5755
23	CNS	.8204	.8361	.8398	.6191	.6645	.6593	.5902	.6127	.5702	.6765	.7044	.6898
24	HCNs	.8977	.9062	.8981	.6983	.7049	.6999	.4907	.5040	.4925	.6956	.7050	.6968
平均		.5247	.6931	.6647	.4011	.5329	.4954	.3519	.4679	.4160	.4259	.5647	.5254

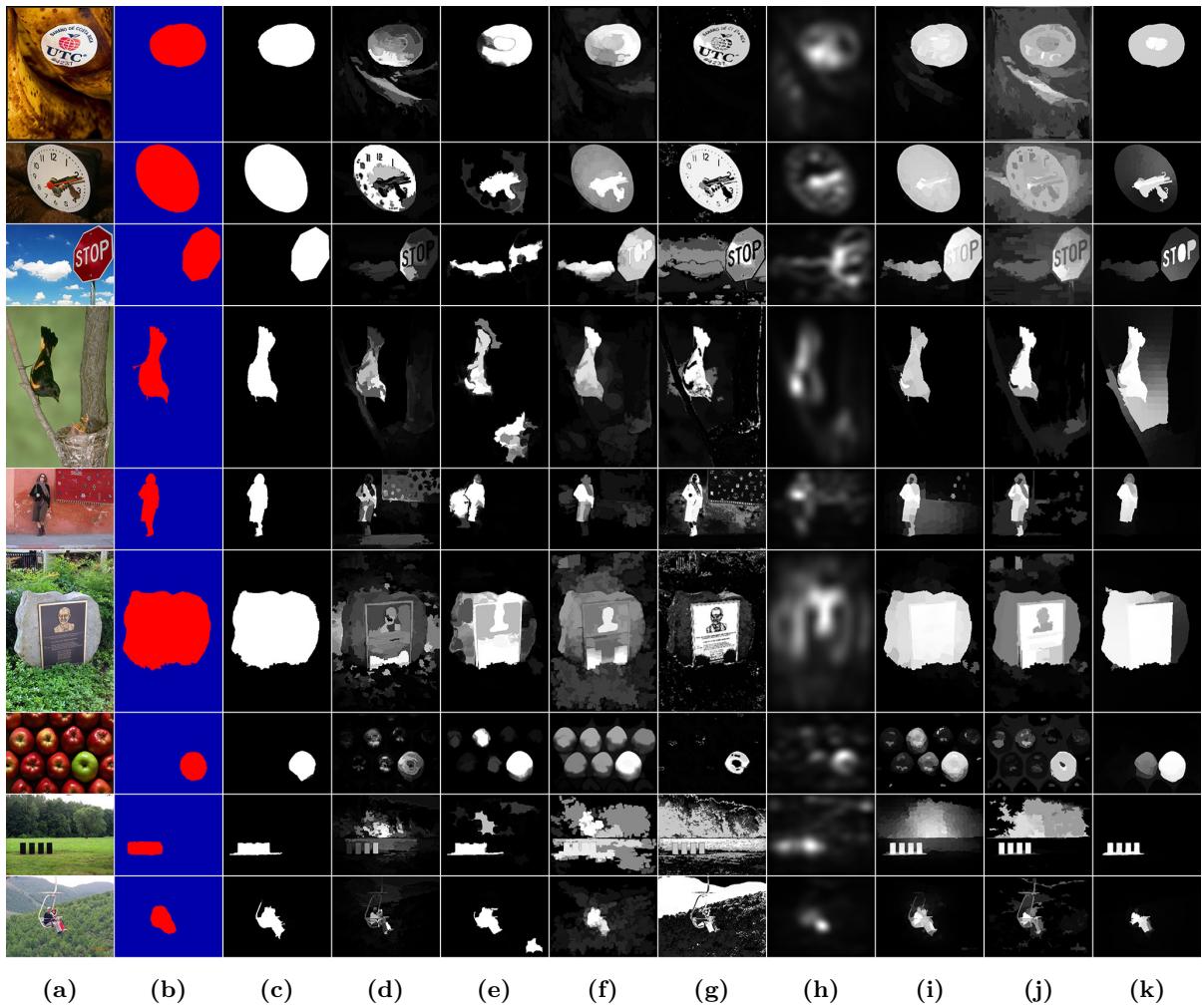


图 4.11 与 8 个显著性方法的视觉比较。(a) 和 (b) 分别是输入图像和人工标注结果, 其中前 3 行、中间 3 行、后 3 行分别来自于 ASD^[9,32]、ECSSD^[60,61]、ImgSal^[45,46] 数据集; (c)–(k) 是分别使用不同方法得到的显著图: (c) 本章提出的 HCN 方法; (d) 第 2 章提出的 RPC 方法; (e) 第 3 章提出的 CNS 方法; (f) BMS^[75]; (g) GC^[121]; (h) HFT^[46]; (i) RBD^[58]; (j) RC^[29]; (k) TLLT^[59]

对性能指标的定量评估见表 4.5, 表中包括三个数据集上的 AvgF、MaxF、AdpF 三项指标及均值, 各项指标的前三名分别使用红色、绿色、蓝色标记:

(1) 在 ASD 和 ECSSD 两个数据集上, HCN 方法的 AvgF、MaxF、AdpF 三项指标都排名第一, 表明其具有的性能优势。

(2) 在 ImgSal 数据集上, 第 3 章提出的 CNS 方法排名第一, 表明 CNS 方法在检测较小尺度的显著物体时具有优势。HCN 方法的 AvgF 指标排名第二, 低于 CNS 方法 9.95%; AdpF 指标排名第三, 低于 CNS 方法 7.77%。

(3) 三个数据集平均而言, HCN 方法的 AvgF 和 AdpF 指标排名第一, 而 RBD 方法的 MaxF 指标排名第一, 高于 HCN 方法 0.49%。三项指标平均来看, HCN 方法排名第一, 第 3 章提出的 CNS 方法排名第二, 三项指标分别低于 HCN 方法 1.91%、0.06%、

0.7%，两者性能接近，综合排名第三的是 RBD 方法。

最后是不同显著性方法检测结果在视觉上的比较。这里从 23 个横向对比的显著性方法中选取 8 个方法，图 4.11 给出了 HCN 方法与这 8 个方法对 9 幅输入图像计算得到的显著图。其中，图 4.11(c) 是 HCN 方法的检测结果，图 4.11(d) 是第 2 章提出的 RPC 方法的结果，图 4.11(e) 是第 3 章提出的 CNS 方法的结果。相对而言，RBD 方法（图 4.11(i)）在其它参与比较的方法中获得了更好的检测结果。但与本章的 HCN 方法相比，背景没有得到很好抑制，同时显著物体不够突出。从 HCN 方法的显著图中可以看到，显著物体和背景得到了很好的分离，检测结果更接近于二值的人工标注图。

4.5.4.2 与协同显著性方法的性能比较

6 个参与比较的协同显著性方法的显著图来源如下：

- 对于 CBCS^[76]、IPCS^[34]、IPTDIM^[19]，直接使用原作者提供的协同显著图①；
- 对于 CoIRS^[77]、CSHS^[79]、SACS^[78]，通过运行原作者发布的 MATLAB 代码生成协同显著图②。

图 4.12 分别给出了基于固定分割阈值 T_f 得到的 $P-R$ 曲线图和 F_β 曲线图，以及基于自适应分割阈值 T_a ^[32] 得到的 $P-R-F_\beta$ 柱状图，性能分析如下：

(1) 对于 $P-R$ 曲线：当阈值 $T_f = 0$ 时，所有方法都汇聚到 $P = 0.2387, R = 1$ 这一点。除该点外，HCN 方法的 $P-R$ 曲线几乎被压缩至一个点，具体为 $P \in [0.9014, 0.9105], R \in [0.7967, 0.8058]$ 。同样，这是由于 HCN 方法得到的显著图更接近于二值图像所致。

(2) 对于 F_β 曲线：当阈值 $T_f = 0$ 时，所有方法都汇聚到 $F_\beta = 0.2848$ 这一点。除该点外，HCN 方法的 F_β 曲线几乎为一条水平线，范围为 $F_\beta \in [0.8625, 0.8665]$ 。在大部分阈值区间，HCN 方法都覆盖了其它方法曲线，但曲线峰值要略低于 IPTDIM 方法。

(3) 对于 $P-R-F_\beta$ 柱状图：HCN 方法在精确率 P 、召回率 R 和 F_β 度量三项指标上都获得了第一。

对性能指标的定量评估见表 4.6，表中包括与 F_β 相关的 AvgF、MaxF、AdpF 指标以及上述三个指标的均值，各指标前三名分别用红色、绿色、蓝色标记：

(1) 对于关键的 MaxF 指标，HCN 方法排名第二，但仅低于排名第一的 IPTDIM 方法 0.06%，而在 AvgF 和 AdpF 两项指标上，HCN 方法具有明显性能优势，分别高于 IPTDIM 方法 24.59% 和 25.55%。

(2) 得益于近似于水平线的 F_β 曲线，HCN 方法在 AvgF 指标上占据绝对优势，比第二名的 CSHS 方法高出了 17.26%。

① CBCS^[76]: https://github.com/HzFu/Cosaliency_tip2013; IPCS^[34]: <http://ivipc.uestc.edu.cn/hlli/projects/cosaliency.html>; IPTDIM^[19]: <https://zdw-nwpu.github.io/dingwenz.github.com/>。

② CoIRS^[77]: <https://sites.google.com/site/fallcolor/projects/saliency>; CSHS^[79]: <https://github.com/ivpshu/Co-Saliency-Detection-Based-on-Hierarchical-Segmentation>; SACS^[78]: https://github.com/HzFu/SACS_TIP2014。

(3) 在 AdpF 指标上, HCN 方法同样排名第一, 比排名第二的 CSHS 方法和排名第三的 SACS 方法分别高出了 4.68% 和 5.11%。

(4) 三项指标平均而言, HCN 方法排名第一, 排名第二和第三的分别是 CSHS 方法和 SACS 方法, HCN 方法分别高出这两个方法 7.67% 和 9.09%。

最后是不同协同显著性方法检测结果在视觉上的比较。图 4.13 给出了 HCN 方法与这 6 个协同显著性方法对 5 对输入图像计算得到的协同显著图, 不同图像对之间已用绿色加粗横线隔开。图 4.13(c) 是 HCN 方法的检测结果, 从中可以很明显地看到显著图几乎是一个二值结果, 且与人工标注结果 4.13(b) 更为接近。相对而言, 其它协同显著性方法中, IPTDIM 和 SACS 方法的检测性能较好, 主要体现在对物体区域轮廓的检测较为精确。然而, 与 HCN 方法相比, 这两个方法并没有凸显出整个物体区域, 同时对背景的抑制程度也不够。显然, HCN 方法得到的这种更趋近于二值结果的显著图, 将更便于从背景中分割出所需的显著物体区域。

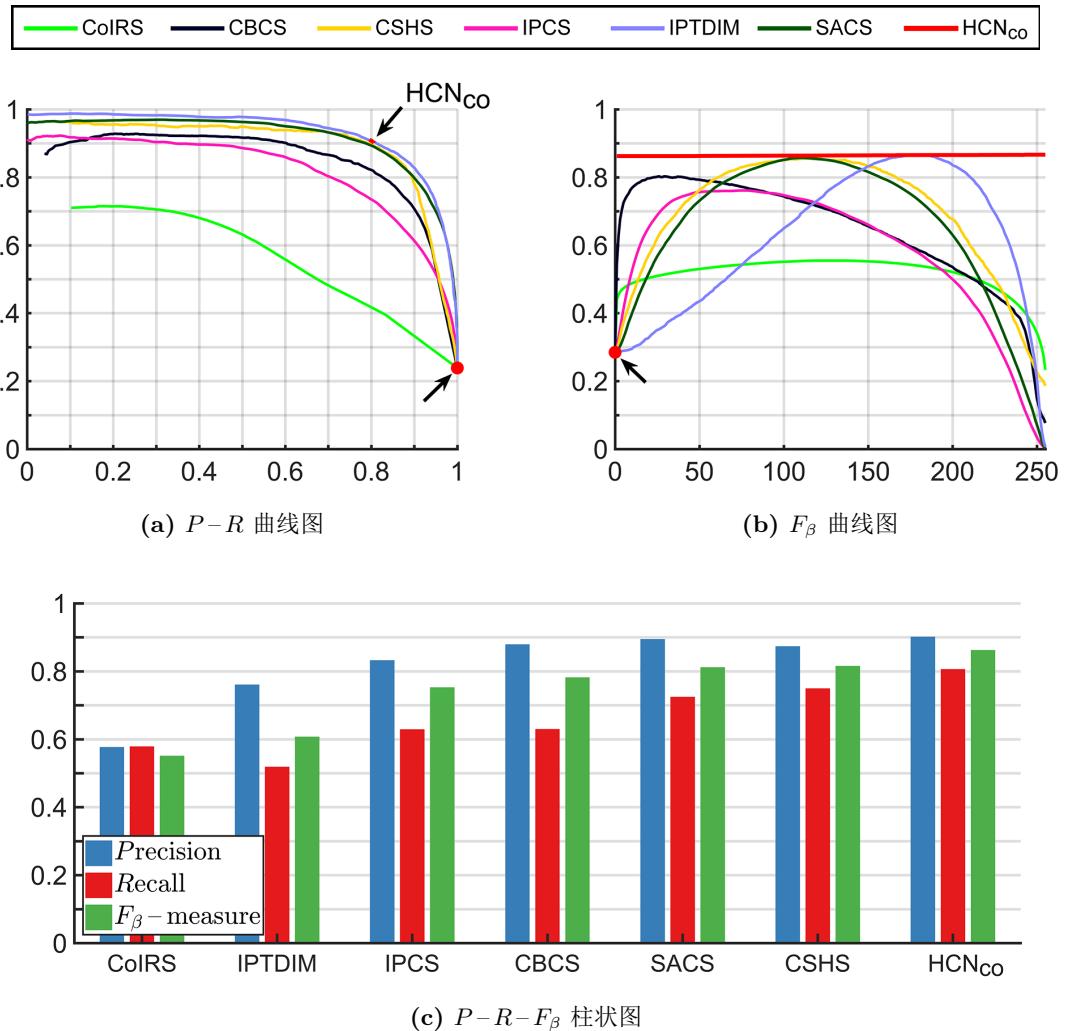


图 4.12 与 6 个协同显著性方法的性能比较。(a) $P-R$ 曲线: x 轴表示召回率 R , y 轴表示精确率 P ; (b) F_β 曲线: x 轴表示固定分割阈值 T_f , y 轴表示 F_β ; (c) 使用自适应分割阈值 T_a ^[32] 计算得到的 $P-R-F_\beta$ 柱状图: 以 F_β 值升序排列

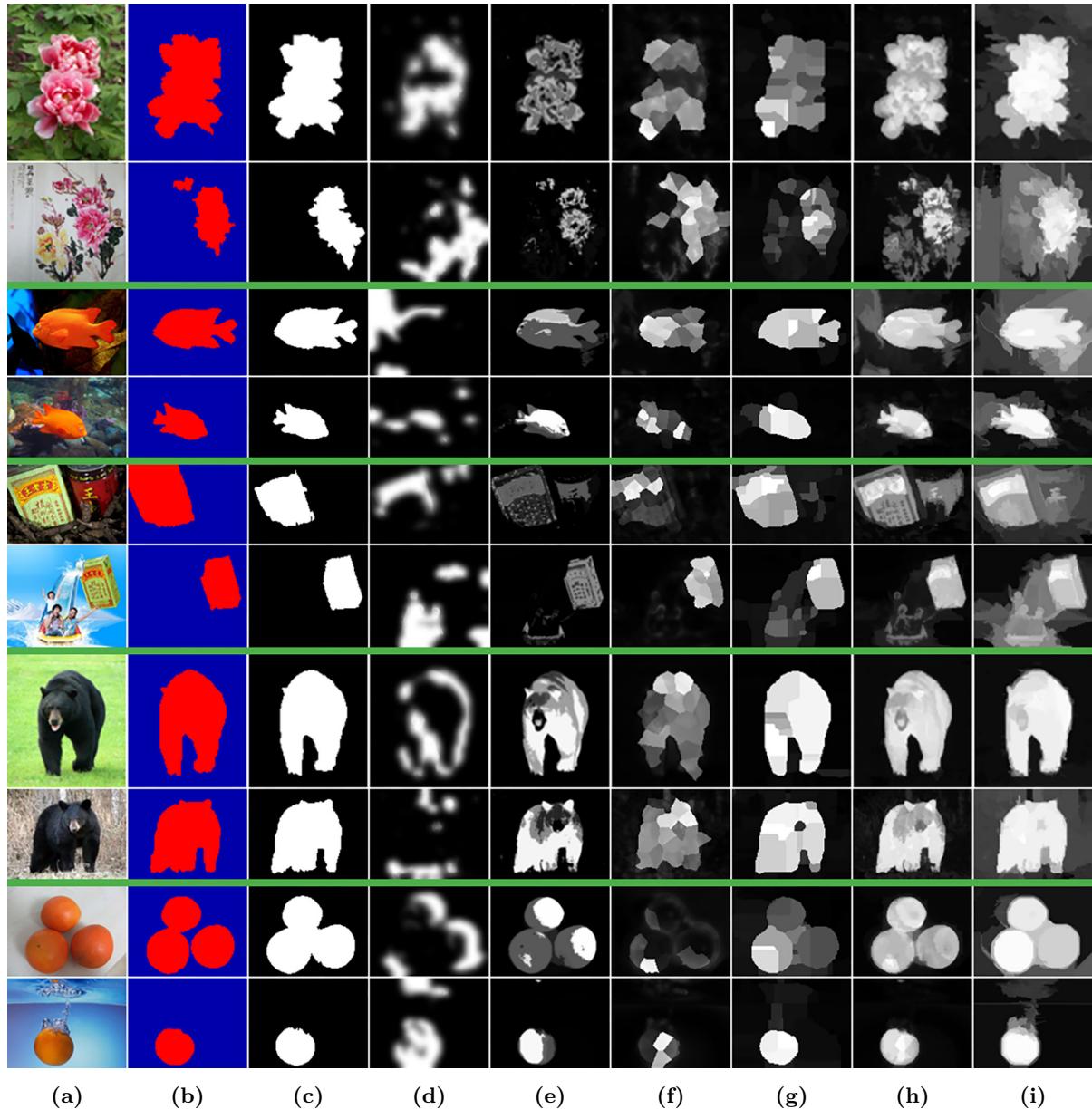


图 4.13 与 6 个协同显著性方法的视觉比较。(a) 和 (b) 分别是输入图像和人工标注结果^[34]; (c)–(i) 分别是使用不同方法得到的协同显著图, 其中: (c) 本章提出的 HCN 方法; (d) CoIRS^[77]; (e) CBCS^[76]; (f) IPCS^[34]; (g) CSHS^[79]; (h) SACS^[78]; (i) IPTDIM^[19]

表 4.6 与 6 个协同显著性方法的性能比较

序号	方法	AvgF	MaxF	AdpF	平均
1	CoIRS ^[77]	.5150	.5548	.5512	.5403
2	CBCS ^[76]	.6433	.8028	.7816	.7425
3	CSHS ^[79]	.6894	.8559	.8157	.7870
4	IPCS ^[34]	.5855	.7612	.7526	.6998
5	IPTDIM ^[19]	.6161	.8671	.6070	.6968
6	SACS ^[78]	.6499	.8571	.8114	.7728
7	HCN _{co}	.8620	.8665	.8625	.8637
平均		.6516	.7951	.7403	.7290

4.5.5 实验结果讨论

虽然通过定量的性能评估和对比分析，表明本章提出的 HCN 方法在显著物体检测和协同显著物体检测上的有效性，但是该方法仍然存在一定缺陷。与上两章相同，这里同样利用第 1.2.4.2 节介绍的 MAE 指标，挑选出五对 MAE 值较大且具有代表性的失败示例，在图 4.14 中予以展示，不同图像对之间已使用绿色加粗的竖线隔开。由于 HCN 方法主要基于第 3 章的 CNS 方法和 RBD 方法^[58] 融合而来，除了示例图像和人工标注结果外，也给出了这两个方法在三个图像层上的显著图，即 $\mathcal{L}_{\text{CNS}}^i$ 和 $\mathcal{L}_{\text{RBD}}^i$ ($i \in \{1, 2, 3\}$)，最后一行是 HCN 方法的检测结果。需要说明的是，CNS 和 RBD 方法在各图像层上产生的显著图大小并不相同，为了示例的美观，这里统一缩放至原图大小显示。

由于 HCN 方法是在单图显著图的基础上，利用不同显著物体之间的颜色名一致性来度量协同显著性，因此单图显著图的准确程度将直接决定协同显著物体检测结果的好坏。对于单图显著图而言，由于是基于 CNS 和 RBD 两个方法融合生成，因此这两个方法又对单图显著物体检测结果有实质性的影响。总体上，导致 HCN 方法检测失效主要有以下几个原因：

(1) 由图像对中的某一张图像的显著性检测失败而导致。此类情况出现在图 4.14(a)、图 4.14(c) 和图 4.14(d) 中：

- 对于图 4.14(a)：在 HCN 方法的左图结果中，尽管辛普森脚下存在一片非标注的区域，但仍然完整地检出了图像中的显著物体。但对于右图，CNS 和 RBD 两个方法在面对显著物体占据大面积图像范围时同时失效，最终导致仅检测出了显著的胳膊区域。
- 对于图 4.14(c)：得益于 CNS 方法良好的性能，HCN 方法在右图上获得了较好的检测结果。但对于左图，过大的显著物体导致 HCN 方法最终丢失了车后部的一大片区域。
- 对于图 4.14(d)：同样得益于 CNS 方法良好的性能，HCN 方法在左图上获得

了较好的检测结果。但对于右图，过大的显著物体、加上又是边界对象，导致 HCN 方法最终仅检测出了金钱豹的上肢部分，而身体部分则完全丢失。

(2) 由 CNS 和 RBD 两个方法对图像对中的两张图像同时检测失败导致。如图 4.14(b) 所示，两个方法都只检测出了摩托车身显著的红色区域，导致 HCN 方法最终的协同显著图中也仅包含这部分区域，而作为摩托车整体的一部分，轮胎区域的丢失使得 HCN 方法检测失败。

(3) 由 HCN 方法本身对协同显著性的处理机制导致。如图 4.14(e) 所示，对于右图，HCN 方法基本检测正确，多出的非标注区域主要是由于该部分区域与猩猩同为黑色。但对于左图，CNS 和 RBD 两个方法都将显著区域定位在穿红衣的人身上，尽管在部分图像层上检测出了猩猩区域，但仅被判为次显著区域，导致单图显著图中猩猩区域丢失。事实上，如果单看左图，通常会认为穿红衣的人更为显著，只有当左右两图放在一起时，才会关注共同的猩猩区域。由于 HCN 方法对协同显著性的处理机制是在单图显著图的基础上利用颜色名剔除非共同显著的区域，因此当单图显著图中都只有一个显著物体区域存在时，这种机制并不会剔除其中任何一个显著物体、或降低任何一个显著物体区域的显著值，从而导致了方法的失效。

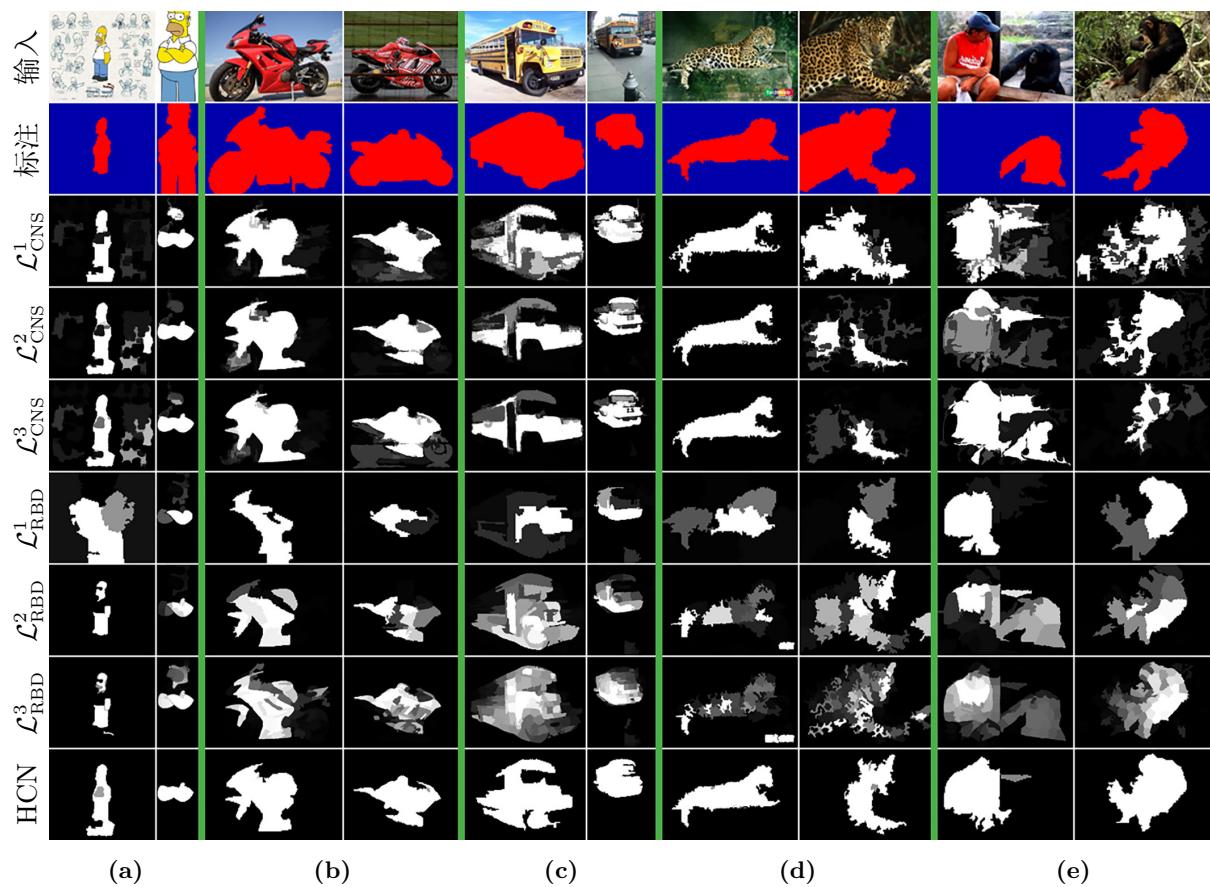


图 4.14 检测失败示例

综上不难看出，要想 HCN 方法获得理想的协同显著物体检测结果，在目前的处理框架和机制下，必须依赖于单图显著性的检测准确。但当显著物体占据图像大面积区域或其自身是一个重度边界对象时，参与融合的 CNS 方法或者 RBD 方法都出现了检测失效的问题。事实上，纯粹依靠自底向上的底层处理技术很难解决上述问题，但如果回归到显著物体检测任务本身、即首先要检测的是包含“物体”的区域，就可以引入似物性采样或目标检测模型，来帮助显著性方法得到更为精确的单图显著图。此外，对于像图 4.14(e) 这样的情况，如果改进后的显著性方法能在左图的单图显著图中给出猩猩区域，那么本章提出的基于颜色名一致性的协同显著物体检测机制将同样有效。

4.6 本章小结

本章综合采用颜色名线索、环绕性线索以及边界连通性线索，提出了用于检测一对图像间协同显著物体的 HCN 方法。首先对原始图像构造三个图像层，单层融合算法在每个图像层上基于显著一致性获得单层融合图，并基于颜色名稀缺性和显著一致性改进得到单层显著图；然后多层融合算法基于显著一致性对三个单层显著图进行融合，并同样基于颜色名稀缺性和显著一致性改进得到单图显著图；最后协同显著检测方法利用自适应的阈值分割得到各单图显著图中的显著区域，并基于显著区域的颜色名一致性去除非协同显著区域，得到协同显著图。HCN 方法除可用于检测协同显著物体外，也是一种可用于检测单张图像的显著物体检测方法，实验结果表明采用的单层/多层次融合算法以及颜色名线索对于协同显著物体检测的有效性。

由于本章提出的 HCN 方法是在单图显著图基础上，采用颜色名一致性来获取图像间的协同显著物体区域，因此单图显著物体检测方法从根本上决定了 HCN 方法检测性能的可靠性。针对第 4.5.5 节讨论的导致 HCN 方法失效的几个问题，下一步计划以下几个方面开展相关研究：

(1) 针对较大尺寸或与图像边界重度连接的显著物体，拟采用似物性采样或目标检测方法，引导单图显著性方法回到对物体区域的检测上来。同时除在本章采用的环绕性和颜色名线索外，考虑采用其它区域级的图像特征（如纹理、空间分布、HSV 直方图等）并加以融合，以适用于更为复杂的图像场景。

(2) 在 HCN 方法目前的框架下，协同显著性模块需要对单图显著图中的显著物体区域进行分割，然而基于自适应阈值分割得到的物体区域并不精确。下一步拟采用边缘检测方法，在单图显著图基础上得到物体区域更为精确的分割。

(3) HCN 方法对协同显著物体区域的判定依据是不同显著区域在整体上的颜色一致性，下一步拟在整体颜色一致性的基础上采用物体内部超像素级的颜色对比和空间信息，以适用于具有更为复杂颜色模式的显著物体区域。

(4) 本章仅在一个图像对（即两幅图像）上验证了 HCN 方法的有效性，下一步计划改进单图显著性检测技术和协同显著性度量方式，把 HCN 方法扩展到对两幅以上图像的协同显著物体检测。

5 融合区域稳定性与显著性的单帧彩色图像小目标检测

5.1 引言

在第2章–第4章中，分别介绍了两种显著物体检测方法和一种协同显著物体检测方法，并在几个公开的显著性数据集上进行了性能评估。本章将把显著性检测技术实际应用到具体场景中，研究如何在单帧彩色图像中检测小目标的问题。小目标检测在诸如预警系统、遥感、航空航天、视觉侦测与跟踪等领域有着非常广泛的应用^[153,154]。不同于传统的面向自然场景的目标检测问题，由于较远的成像距离，小目标区域通常缺乏高层的表观特征或语义模式。在一些应用场景中，小目标会淹没在海杂波或云杂波中，其它诸如传感器噪声、目标尺寸变化以及人为干扰等因素，都使得小目标检测是一个相当具有挑战性的课题。

在过去几十年里，研究人员提出了大量的小目标检测方法，从目标/背景的角度，主要划分为三类：目标增强^[155]、背景抑制^[156]、目标–背景分离^[153,157,158]。由于小目标区域也是一个前景图形，因此基于目标–背景分离的小目标检测方法通常会采用不同的特征对比技术进行建模，试图在突出小目标信号的同时抑制背景杂波。在自底向上的显著性检测方法中，基于全局稀缺性或局部中心环绕差异性的特征对比机制对场景中的前景图形进行检测，是一种常见的技术手段。但是，传统的显著性方法更趋向于检测场景中最大或较大的显著区域，当应用于小目标检测时可能存在如下问题：一是小目标较小的尺度使得其特征在场景中可能并不稀缺，从而导致基于特征全局稀缺性的手段失效；二是场景中不同颜色、强度、纹理等特征的区域交界处，具有较大的特征差异，因此基于局部差异性的显著性方法会检测出大量的虚警区域，使得小目标淹没其中，并很难被分离出来。从第5.5.3.2节的实验可以看到，传统的显著性检测方法，包括第2章至第4章提出的方法在内，并不适合解决小目标检测问题。

此外，目前的小目标检测方法主要研究红外小目标检测^[159–161]，利用的是红外场景下小目标的高热特性，但本章主要聚焦于可见光静态场景下的小目标检测问题。从字面上看，该问题包含两层含义，一是面向的应用是可见光波段场景，二是针对的是单帧图像而非序列化的视频。这主要出于如下考虑：一方面，与序列化检测相比，单帧检测方法更适合于具有快速变化背景或目标变化较大的应用场景^[162]；另一方面，虽然可见光谱波段在整个电磁波谱中只占很窄的一段，但可见光波段是所有波段中人们最为熟悉的^[148]，并且经常被拿来与红外波段相结合，应用到一些多光谱视觉任务中^[163]。因此，研究可见光波段下的小目标检测问题，可以将该问题扩展到更为广泛的应用场合，相关的技术手段也可为红外小目标检测技术的发展带来一定启示。

本章将单帧彩色图像小目标检测问题，转换为同时检测区域的局部稳定性和局部

显著性问题，下一节将介绍采用这种方式的动机。本章工作的创新点为：提出了一种基于稳定性区域提取候选小目标，并基于颜色局部差异去除虚警的单帧彩色图像小目标检测方法，该方法可突出小的显著区域并得到目标区域精确的特征描述。同时，标注并公开了一个针对彩色图像小目标检测的基准数据库，包括三个基准数据集共 1,093 张图像，提供了像素级的人工标注图以及小目标的 4 个区域属性，并发布了相应的 MATLAB 评估代码，用于在该基准数据库上的性能评估^①。

为方便起见，本章提出的融合区域稳定性 (Regional Stability) 和区域显著性 (Regional Saliency) 的小目标检测方法，简记为“RSS”方法^②。本章内容安排如下：第 5.2 节介绍研究动机与出发点；第 5.3 节介绍 RSS 方法框架；第 5.4 节详细介绍 RSS 方法，包括基于区域稳定性的候选小目标提取算法和基于区域显著性的虚警目标去除算法；第 5.5 节是实验方案和数据集介绍、参数分析、以及实验结果和讨论；最后是本章小结和未来工作展望。

5.2 研究动机与出发点

与红外波段相比，可见光波段下的小目标通常具有较低的强度值，但如果将分别在这两个波段下获取到的小目标放在一起比较，可以发现它们之间存在一些共性特征：1) 目标本身是连通的前景图形；2) 类斑点的形状；3) 较小的熵；4) 目标内部近似均匀的强度；5) 与其局部邻域之间的差异。上述五个特征隐含了两条极其重要的线索，即小目标首先是具有环绕性的前景区域 (针对特征 1)，同时这个区域应是局部稳定的 (针对特征 2–4)、也是局部显著的 (针对特征 5)。基于此，本章提出的 RSS 方法就将单帧彩色图像小目标检测，转换为从图像中分割出同时具有“局部稳定性”和“局部显著性”的前景区域。随之而来的两个问题是：1) 如何从图像中分割出局部稳定性区域？2) 如何定义图像中的局部显著性？

两个经典的工作与上述的两个问题紧密相关，首先是关于稳定性的。J. Matas 等人^[164,165] 提出了一种仿射不变局部区域特征提取方法，称为最大稳定极值区域 (Maximally Stable Extremal Region，简记为 MSER)，该方法利用地形学中分水岭的思想计算图像中的局部稳定区域，用于解决宽基线图像的匹配问题。简单来说，对于一幅灰度图，MSER 是其中的一个局部连通区域，在有限的连续阈值分割下该区域得到的二值面积没有较大的变化，也就是说是相对稳定的。然而，由于分割阈值通常采用间隔采样，无法确保这些阈值是否是最优分割阈值；同时，对于某个目标，可能在多个阈值区间内都满足稳定性约束条件，即得到了一个目标的多个稳定性表示，导致后续还需要人为地将冗余的表示剔除。更为重要的是，由于待检测目标区域较小的特点，通常会对 MSER 设置较大的面积变化阈值，而仅基于 MSER 方法很难有效解决由背景杂波导致

^① 下载网址：<https://github.com/jinglou/p2016-rss-small-target/tree/master/Benchmark>。

^② RSS 方法的 MATLAB 源代码、基于 MSER 的小目标检测代码、基准数据库及人工标注结果、性能评估代码及指标结果、视频演示以及幻灯片等资源，已发布在论文主页：<http://www.loujing.com/rss-small-target/>。

的检测虚警问题。

另一个工作与显著性有关，也就是在之前几章中提到过的 FT 方法^[32]。FT 方法使用 LAB 颜色的全局稀缺性来定义显著性，适用于检测场景中较大的显著物体。然而，在后续章节中可以看到，对于彩色图像中的小目标而言，其颜色或亮度特征在整个场景中并不一定具有全局的稀缺性，使用 FT 方法反而会抑制小目标在其局部邻域内的凸显程度。更为重要的是，显著性检测得到的结果是一个显著图，直接将显著图用于小目标检测将不可避免地带来分割阈值选取的问题。尤其是场景中的小目标尺寸存在变化时，很难找到合适的全局阈值或局部阈值来实现自适应的小目标分割。

受上述工作的启发，本章提出的 RSS 方法将小目标检测首先转换为一个稳定性前景图形分割问题，并基于五种区域描述子和四个稳定性度量准则将得到的稳定性区域作为候选小目标。为了解决检测虚警问题，本章采用局部颜色差异计算像素级的显著性，并基于候选小目标的平均显著值剔除非显著的局部区域。除 RSS 方法外，也对几个现有的稳定性和显著性方法采用本章提出的融合机制进行融合，实验结果表明融合机制比仅采用单一检测技术，对提升检测性能的有效性。

5.3 方法框架

RSS 方法框架如图 5.1 所示，其中包括一个用于生成候选小目标区域的稳定性提取器 (Stability Extractor)、一个用于计算像素级显著性的显著性检测器 (Saliency Detector)、以及一个融合模块。这种模块化的设计，不仅与上一节提到的五个小目标内在特征紧密关联，同时也表征了人类对于小目标场景的视觉观测机制。

- 模块 I – 稳定性提取器

基于小目标区域内几乎没有颜色变化的事实，稳定性提取器首先将输入的单帧彩色图像转换为灰度图，并利用一组间隔阈值对灰度图进行序列化的二值分割，得到一组布尔图。由于从布尔图中提取的前景区域可能是潜在的小目标，因此对这些区域分别构造五种区域描述子，并加入到候选小目标集合。同时，考虑到小目标内部具有的纹理平滑性和强度一致性，采用四个稳定性度量准则对每个候选区域进行稳定性度量，并将不满足稳定性约束的区域剔除，得到一幅“稳定图” (Stability Map)。本章对稳定图使用符号 M_T 表示，下标字母 T 代表稳定性 (Stability)。

- 模块 II – 显著性检测器

显著性检测器首先将输入的单帧彩色图像从 RGB 颜色空间转换到 LAB 空间，然后对三个颜色通道分别采用高斯平滑，获得像素级的显著性信息。本模块的作用并不是为了直接得到小目标检测结果，而是自动判断场景中的显著像素并赋给像素对应的显著值，来引导融合模块对候选小目标做进一步的筛选。显著性检测器会生成一个与输入图像具有相同分辨率大小的“显著图” (Saliency Map)，本章对显著图使用符号 M_A 表示，下标字母 A 代表显著性 (Saliency)。

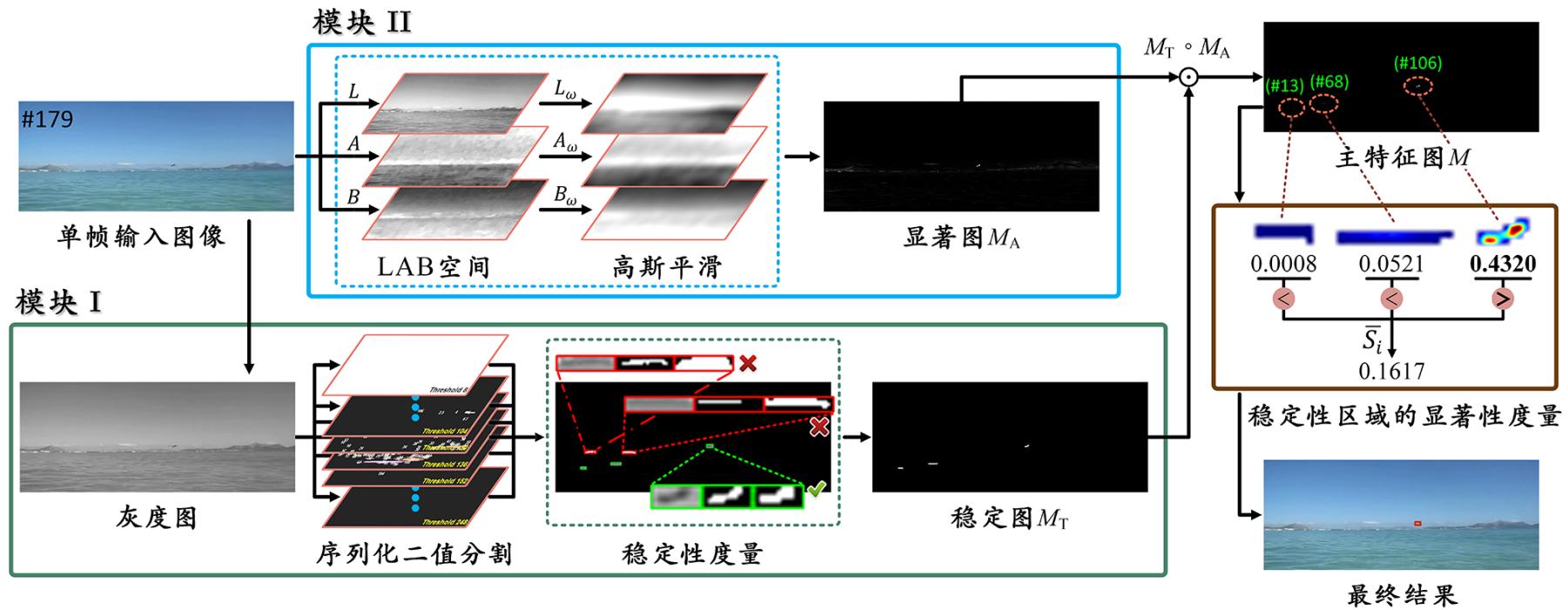


图 5.1 RSS 方法框架

- 融合模块

融合模块将上述两个模块得到的稳定图 M_T 和显著图 M_A , 采用 Hadamard 积的方式 ($M_T \circ M_A$) 计算出主特征图 M (Master Feature Map), 主特征图是对图像稳定性和显著性的一种编码表示。在主特征图中, 利用各候选小目标区域内像素的显著值均值得到目标的平均显著值。对于各候选小目标, 仅当其平均显著值超过所有小目标的平均显著值均值时, 该候选小目标才予以保留, 否则将被作为虚警目标从候选小目标集合中剔除, 最终候选小目标集合中保留的区域就是 RSS 方法的检测结果。如图 5.1 所示, 主特征图 M 中共有 3 个候选小目标, 编号分别为 #13、#68 和 #106, 对应的平均显著值分别为 0.0008、0.0521 和 0.4320, 三者的均值 \bar{S}_i 为 0.1617。由于只有 #106 候选小目标的平均显著值大于 \bar{S}_i , 因此只有该目标被保留下, 成为最终的检测结果。可以看到, 在整个 RSS 方法框架中, 稳定性提取器和显著性检测器是其中的关键。接下来, 将分别介绍这两个模块的设计, 并通过仅使用其中之一产生的检测缺陷, 来阐明采用融合机制的必要性和有效性。

5.4 融合区域稳定性与显著性的小目标检测方法

5.4.1 基于区域稳定性的候选小目标提取

在第 5.2 节, 之所以提出彩色图像中的小目标具有特征 2–4、进而抛出小目标具有稳定性这一观点, 主要源于如下观察: 在原始图像 5.2(a) 中, 以紧密包含小目标区域的窗口截取出目标子图, 灰度化并反相后得到图 5.2(b), 如果使用一组序列化的阈值对该子图做二值分割, 就得到相应的一组二值表示。本章将这种在不同阈值分割下得到的二值表示称为目标的“表象”, 图 5.2(c)–5.2(n) 示例了阈值区间 [104, 148]、采样间隔 4 的部分表象^①。可以看到, 虽然随着分割阈值的不断增大, 目标的表象也在不断变化, 但在有限的一段阈值区间内(最极端的情况是两个间隔为 1 的相邻阈值), 表象将在一定程度上保持不变, 也就是说, 在该段阈值区间内小目标是稳定的。

上面的观察除了给出稳定性这一线索外, 另一个重要信息就是利用一个最优阈值, 就可以有效地将小目标从子图窗口中分割出来, 但如何确定这个最优阈值又带来了新的问题。重新审视小目标的五个内蕴特征, 可以看到在图 5.2(b) 中, 小目标内部像素的强度值呈现均匀的分布, 且具有高度的一致性。又由于小目标自身面积较小, 子图窗口内剩余区域作为目标的邻域背景, 面积也相对较小, 因此背景像素的强度也可认为具有均匀的分布和高度的一致性。此外, 从视觉观察的角度, 人之所以能在整个场景中发现小目标, 主要源于小目标与其邻域背景间存在的强度差异。既然在整个子图窗口内部, 有两类具有明显强度差异的模式存在, 那么通过最大化两类的类间方差计算分割阈值的 Otsu 方法^[166] 应是最优的选择。

^① 原始图像 5.2(a) 中, 小目标区域子图窗口的宽度为 19 个像素、高度为 7 个像素。出于清晰显示的目的, 这里统一对子图以及 12 个二值分割结果做放大 10 倍处理。

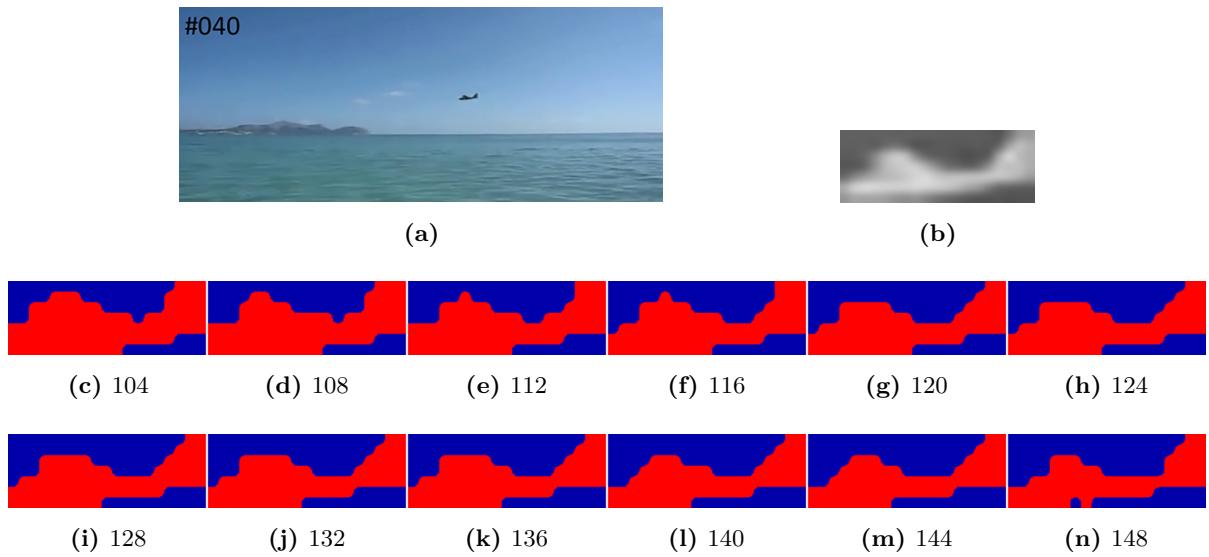


图 5.2 小目标区域稳定性示例。(a) 数据集 #2 的第 #40 帧图像; (b) 小目标区域的反相灰度子图; (c)–(n) 对图(b)序列化分割得到的 12 个二值结果, 各子图标题为分割阈值

因此, 本章的 RSS 方法将小目标检测问题, 转换为如何在单帧图像中寻找包含小目标的子图窗口问题, 而问题的关键也变为如何能够自动切分场景, 并得到每个小目标的最优子图。但这一问题仍然面临很大的挑战, 这是因为:

(1) 假设希望通过阈值分割来切分场景, 通常需要对不同区域使用不同的阈值, 然而这种自适应的阈值实际上很难自动确定。

(2) 子图窗口本质上是一个中心环绕的图像子块, 在图像中有大量不同尺寸的此类模式子块存在, 但其中大部分包含的都是虚警目标。在实际应用中, 特别是面向一些将准确率作为首要考虑的应用场合, 很难找到某种自适应约束来帮助检测算法自动地剔除掉这些虚警目标。

(3) 对某个具体的小目标而言, 以其为中心可以得到众多不同窗口大小的中心环绕子块, 导致通过这些子图计算得到的 Otsu 阈值不尽相同; 反之, 由于无法确定哪个 Otsu 阈值是最优阈值, 导致无法通过该阈值对应的表象来确定最优子图。看上去, 这进入了一个相悖的境地。考虑到小目标在场景中可能有较大的尺寸变化, 如何针对不同尺寸的目标区域确定出最优子图, 将成为决定检测系统性能优劣的关键。

基于上述讨论, RSS 方法最终把寻找包含小目标区域的最优子图窗口问题, 进一步转换为寻找图像中的稳定性区域, 并将稳定性区域的最小外接矩形作为最优子图窗口。这是因为: 既然小目标必须满足稳定性约束, 那么从图像提取出的稳定性区域中必定包含了小目标; 同时, 既然稳定性区域的最小外接矩形是包含该区域的最优子图窗口, 那么利用子图计算得到的 Otsu 阈值, 就是小目标区域的最优分割阈值。因此, 本章首先提出了一种基于稳定性度量准则的两步提取小目标区域的算法, 简记为 RS_T 算法, 这里下标字母 T 表示稳定性。但需要特别指出的是, 虽然小目标是稳定性区域, 但提取出的稳定性区域却未必都是小目标 (其中可能有虚警目标存在), 因此 RS_T 算法得

到的实际上是候选小目标。在第 5.4.2 节中，将讨论仅基于稳定性检测带来的虚警问题，并由此引入显著性。

5.4.1.1 区域稳定性度量准则

为了度量区域的稳定性，RS_T 算法采用五种区域描述子来标记一个二值前景区域（即二值图像中像素值为 1 的连通区域）。这些区域描述子是在小目标有限的信息量下，对目标区域属性以及稳定性的一种有效表示，包括区域面积 $|r|$ 、区域质心位置 c 、区域最小外接矩形 b 、区域填充率 f 、区域宽高比 a ，具体定义见表 5.1。

表 5.1 五种区域描述子的符号表示与定义

- **二值图像 I** : 是一个二值映射，即 $I : Q \subset \mathbb{Z}^2 \rightarrow \{0, 1\}$ 。
- **区域 $r \subset Q$** : 是一个闭合的连通区域，即 $\forall (x, y) \in r : I(x, y) = 1$ ，且区域 r 的欧拉数等于 1。 r 的面积（即该区域内包含的像素数）用符号 $|r|$ 表示，计算公式为 $|r| = \sum_r I(x, y)$ 。
- **质心 c** : 是区域 r 的几何中心，使用 1×2 的向量 $[c_x, c_y]$ 表示。其中， $c_x = (\sum_r xI(x, y)) / |r|$, $c_y = (\sum_r yI(x, y)) / |r|$ 。
- **最小外接矩形 b** : 是包含区域 r 的最小外接矩形，使用 1×4 的向量 $[b_x, b_y, b_w, b_h]$ 表示。其中， $[b_x, b_y]$ 表示矩形左上角的 (x, y) 坐标， b_w 和 b_h 分别表示矩形的宽度和高度。
- **填充率 f** : 用于表示区域 r 在其最小外接矩形 b 内的填充程度，计算公式为 $f = |r| / (b_w \times b_h)$ 。
- **宽高比 a** : 是区域 r 的最小外接矩形的宽度与高度的比值，计算公式为 $a = b_w / b_h$ 。

对于两个已经采用上述五种区域描述子标记好的二值区域 u 和 v ，进一步引入四个属性对比度量来定义两个区域之间的相似程度，包括面积变化 D_r 、质心距离平方 D_c 、填充率差异 D_f 、以及宽高比差异 D_a ，具体定义如下：

(1) 面积变化 D_r

$$D_r(u, v) = \frac{| |u| - |v| |}{\max(|u|, |v|)}, \quad (5.1)$$

其中， $|u|$ 和 $|v|$ 分别表示区域 u 和 v 的面积，外层符号 $| \cdot |$ 为绝对值运算符， $\max(\cdot)$ 函数返回两个区域的最大面积。

(2) 质心距离平方 D_c

$$D_c(u, v) = \|c_u - c_v\|_2^2, \quad (5.2)$$

其中， c_u 和 c_v 分别表示区域 u 和 v 的质心，符号 $\|\cdot\|_2$ 表示 ℓ_2 范数。

(3) 填充率差异 D_f

$$D_f(u, v) = \frac{\max(f_u, f_v)}{\min(f_u, f_v)}, \quad (5.3)$$

其中, f_u 和 f_v 分别表示区域 u 和 v 的填充率, 函数 $\max(\cdot)$ 和 $\min(\cdot)$ 分别返回最大和最小填充率。

(4) 宽高比差异 D_a

$$D_a(u, v) = \frac{\max(a_u, a_v)}{\min(a_u, a_v)}, \quad (5.4)$$

其中, a_u 和 a_v 分别表示区域 u 和 v 的宽高比, 函数 $\max(\cdot)$ 和 $\min(\cdot)$ 分别返回最大和最小宽高比。

对于上述四个相似性度量, 结果返回值越小, 表明区域 u 和 v 越相似。假设 u 和 v 是同一个目标的两个表象, 那么这种相似性就在一定程度上表征了该目标区域的稳定性。因此, RS_T 算法中的稳定性, 实际上度量的是同一个目标的一组表象之间的相似性。但这紧接着又产生了两个新的问题: 1) 如何得到同一个目标的一组表象, 或者换句话说, 当已有一组表象时如何确定它们隶属于同一个目标? 2) 何种相似程度才能够表明该目标是稳定的?

5.4.1.2 基于区域稳定性的提取算法

为了解决上述两个问题, RS_T 算法采用两步处理。简单概括来说, 首先通过步骤 1 将图像划分为一系列的表象, 并通过聚簇算法将隶属于同一个目标的表象归为一簇。然后在步骤 2 中对同一簇内的表象进行相似性度量, 并利用稳定性约束条件最终得到稳定性区域。

- 步骤 1 – 表象聚簇

聚簇算法首先将单帧彩色图像转为 $[0, 255]$ 区间的灰度图 G , 并使用一组阈值对 G 进行序列化的二值分割, 反相后得到一组布尔图。随后, 针对每个布尔图中任意一个前景连通区域, 使用表 5.1 中的五种区域描述子标记该区域的区域属性, 同时采用流水编码方式对每个区域标记一个 ID 号。图 5.3 示例了对图 5.2(a) 序列化分割得到的部分布尔图, 各子图右下角为分割阈值, 区域上的编号为 ID 号。

(1) 为了减少参与稳定性度量的二值区域个数, 对于面积小于 4 个像素或超过图像面积 20% 的区域(即太小或太大的区域), 被直接剔除。

(2) RS_T 算法仅检测具有环绕性的小目标区域, 即要求前景连通区域必须具有完整的闭合轮廓, 对于边界对象区域, 由于其轮廓的不闭合也被同时剔除。

(3) 对 G 的序列化分割并非是在 $[0, 255]$ 区间的连续分割, 而是使用了以 δ 为步长的间隔采样, 即实际的分割阈值为 $[0 : \delta : 255]$ 。显然, δ 的取值越大, 序列化分割得到的布尔图越少, 算法的执行效率越高。设置 δ 为采样间隔, 一是为了提高算法效率, 二是可以确保被提取出的稳定性区域至少在一个 δ 区间内保持稳定。但由于在 RS_T 算法的步骤 2 中, 对面积稳定性的度量是在 Otsu 阈值左右两侧各 $\delta/2$ 步长处进行, 因此实际采用的分割阈值为 $[\delta/2 : \delta : 255 - \delta/2]$ 。有关 δ 的参数分析, 详见第 5.5.2.2 节。

(4) 对各区域标记的 ID 号, 是在所有布尔图中连续编号, 编号以各区域内像素的

最小 x 坐标依次递增，图 5.3 中示例了这种编号过程。假设 $\delta = 16$ ，在阈值为 56 时得到第一个二值区域，编号为 #1，接着在阈值为 72 时得到第二个二值区域，编号为 #2，并以此类推。对于图 5.2(a) 中的小目标区域而言，在这些布尔图中的 ID 号分别为 #1、#2、#11、#32、#48 和 #76。

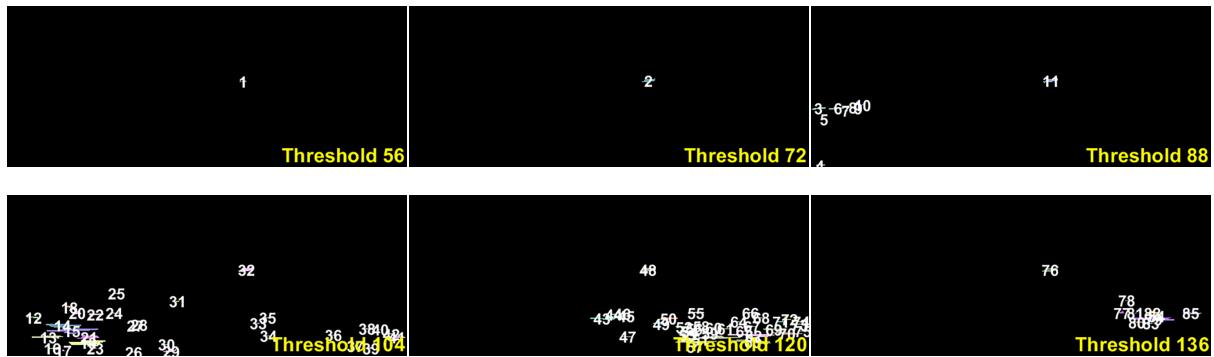


图 5.3 序列化采样分割得到的部分布尔图

采用序列化采样分割生成的布尔图中包含了场景的拓扑结构信息，本文中这并非首次出现。在第 3 章介绍的 BMS 方法^[58] 以及提出的 CNS 方法中，就利用了这一技术来实现场景的图形–背景分离。由于 RS_T 算法本身也不检测边界对象区域，因此标记好的二值区域中就包含了本应属于前景图形的小目标区域。同时，虽然对于场景中不同位置处的小目标，要直接确定自适应的分割阈值会比较困难，但无论如何这个最优阈值肯定是在 $[0, 255]$ 区间内，因此布尔图中必定有接近于小目标最优秀表象的区域存在，换句话说，最优阈值应在某个采样阈值的附近。接下来，如果能找到隶属于同一个目标的一组表象，就可以利用表象之间的相似性度量得到稳定的采样阈值区间，进而在其附近得到最优分割阈值。

在序列化采样分割时，二值区域不断进行着分离或合并，MSER 方法^[164,165] 采用组件树结构组织区域之间的这种关系，并通过对树的搜索求取稳定区域。不同于 MSER 方法， RS_T 算法通过直接度量 D_c 来得到目标与表象之间的隶属关系。这里就引入了第一个区域稳定性约束，即空间位置稳定性。事实上，对于同一个小目标而言，虽然其在不同采样阈值下的表象有所不同，但在有限的阈值区间内，它们的质心必定出现在相近的空间位置上。因此，把空间位置上相近的二值区域归为一个集合，那么集合内的二值区域就是隶属于同一个目标的一组表象。

上述过程通过一个聚簇算法实现，伪代码见算法 5.1。假设序列化采样分割共得到了 m 个二值区域，记为 $\{r_1, r_2, \dots, r_m\}$ ，这些区域的集合记为 R 。在初始化阶段，利用第一个区域 r_1 构成第一个簇 s_1 ，并把 s_1 放入簇集合 S 中。接着，从 R 中取出第二个区域 r_2 ，依次与簇集合 S 中每一个现有簇 s_{exist} 中的二值区域计算 D_c 。如果 D_c 不大于一个给定阈值 Δ_c ，则将 r_2 并入簇 s_{exist} ；否则，将 r_2 构成一个新簇 s_{new} ，并将 s_{new} 加入到簇集合 S 中。以此类推，不断取出集合 R 中的二值区域执行上述聚簇操作，直

至 R 为空时停止。算法中，位置稳定性约束 Δ_c 是利用二值区域的最小外接矩形自动求得，计算公式为

$$\Delta_c = \frac{(\min(b_{r_i,w}, b_{r_j,w})^2 + (\min(b_{r_i,h}, b_{r_j,h}))^2}{4}, \quad (5.5)$$

其中，下标 w 和 h 分别表示区域 r_i 和 r_j 的最小外接矩形的宽度和高度。对于 r_i 和 r_j ，如果 $D_c(r_i, r_j) \leq \Delta_c$ ，则认为 r_i 和 r_j 应属于同一个簇。

算法 5.1 表象聚簇

输入: 二值区域集合 R

输出: 簇集合 S

```

1:  $s_1 = \{r_1\}$ ,  $S = \{s_1\}$                                 ▷ 初始化
2: for each  $r_j \in R$  do                            ▷ 对于  $R$  中任一区域  $r_j$ 
3:   if  $\forall s_{\text{exist}} \in S, \exists r_i \in s_{\text{exist}} : D_c(r_i, r_j) \leq \Delta_c$  then
4:      $s_{\text{exist}} = s_{\text{exist}} \cup \{r_j\}$                 ▷ 将  $r_j$  加入现有簇  $s_{\text{exist}}$ 
5:   else
6:      $s_{\text{new}} = \{r_j\}$                                 ▷ 将  $r_j$  加入新簇  $s_{\text{new}}$ 
7:      $S = S \cup \{s_{\text{new}}\}$                           ▷ 将  $s_{\text{new}}$  加入簇集合  $S$ 
8:   end if
9: end for

```



图 5.4 目标与表象间的隶属关系

聚簇算法生成的簇集合 S 中，每一个簇 s_k 里都包含了一组空间位置上相近的二值区域，依据之前的讨论，它们是隶属于同一个目标的一组表象。而记录目标与表象之间的这种隶属关系，只需要简单地将这一组表象的 ID 号都标记为同一个编号即可。图 5.4 示意了执行聚簇算法后得到的目标与表象间隶属关系，对比聚簇前序列化采样分割得到的布尔图 5.3，可以看到小目标区域在不同布尔图中的 ID 号都已变成了 #85，表明这些区域是隶属于 #85 簇的一组表象。需要指出的是，由于在编码实现时每个簇的簇

号直接来自于区域的 ID 号，簇集合 \mathbf{S} 中的簇号实际上并不连续，但这并不带来什么问题。在下一步稳定性区域提取阶段，只需要对簇集合遍历即可。

- 步骤 2—稳定性区域提取

步骤 1 的表象聚簇算法已得到目标与表象之间的隶属关系，接下来引入填充率稳定性约束、宽高比稳定性约束、以及面积稳定性约束，通过稳定性区域提取算法，确定一簇表象所隶属的目标是否是稳定的，伪代码见算法 5.2。

算法 5.2 稳定性区域提取

输入: 灰度图 G , 簇集合 \mathbf{S}

输出: 稳定性区域集合 R_T

```

1:  $R_T = \emptyset$                                 ▷ 初始化
2: for each  $s_k \in \mathbf{S}$  do                  ▷ 对于集合  $\mathbf{S}$  中的每一个簇  $s_k$ 
3:    $r = \arg \max_{r \in \{r_i, r_j, r_m, r_n\}} |r|$ 
   其中  $\begin{cases} (r_i, r_j) = \arg \min_{r_i, r_j \in s_k} D_f(r_i, r_j) \\ (r_m, r_n) = \arg \min_{r_m, r_n \in s_k} D_a(r_m, r_n) \end{cases}$ 
4:    $G_r = \text{EXTRACT}(G, r)$                 ▷ 利用  $r$  从  $G$  中抽取子图  $G_r$ 
5:    $\vartheta = \text{OTSU}(G_r)$                    ▷ 计算 Otsu 阈值
6:    $r^\vartheta = \text{SEGMENT}(G_r, \vartheta)$     ▷ 利用  $\vartheta$  得到候选稳定性区域
7:    $r^{\vartheta_-} = \text{SEGMENT}(G_r, \vartheta_-)$   ▷  $\vartheta_- = \vartheta - \delta/2$ 
8:    $r^{\vartheta_+} = \text{SEGMENT}(G_r, \vartheta_+)$   ▷  $\vartheta_+ = \vartheta + \delta/2$ 
9:   if  $D_r(r^{\vartheta_-}, r^{\vartheta_+}) \leq \Phi_r$  then
10:     $R_T = R_T \cup \{r^\vartheta\}$                  ▷ 将区域  $r^\vartheta$  加入集合  $R_T$ 
11:   end if
12: end for

```

对于填充率和宽高比两个稳定性，为了避免手工设置硬阈值条件，算法 5.2 通过在簇中寻找一对最稳定的表象进行处理。具体而言，对于簇集合 \mathbf{S} 中的每一个簇 s_k ，从隶属于 s_k 的所有表象中找到填充率差异 D_f 最小的一对表象 (r_i, r_j) ，以及宽高比差异 D_a 最小的一对表象 (r_m, r_n) 。虽然这里并没有像位置稳定性那样设置一个硬约束阈值，但显然最稳定的一对表象应在上述的两组表象之中，在某些情况下这两组表象甚至可能是同一对表象。随后，考虑到最小外接矩形应完整地覆盖整个目标区域，因此将 $\{r_i, r_j, r_m, r_n\}$ 中面积最大的表象赋予 r ，并利用 r 从灰度图 G 中分割出子图 G_r 。此时得到的 G_r 虽非目标区域的最优子图，但其中已包含了完整的目标区域，且有两类明显强度差异的模式存在（即目标与背景）。因此对 G_r 计算得到的 Otsu 阈值^[166] 可被认为是最稳定的最优阈值，而基于最优阈值 ϑ 对子图 G_r 分割得到的二值区域 r^ϑ ，被认为

是该目标的最优表象。需要注意的是，虽然 r^ϑ 是簇 s_k 所属目标的最优表象，但由于对填充率和宽高比都只是最大相似性的度量，因此上述计算过程仅表征了簇 s_k 这组表象的相对稳定性，换句话说，得到的 r^ϑ 只是一个候选的稳定性区域。为了最终确认 r^ϑ 是否稳定，这里引入最后一个区域稳定性约束，即面积稳定性。

面积稳定性通过在最优阈值 ϑ 左右两边各 $\delta/2$ 处对子图 G_r 进行分割，可以首先确保最优表象 r^ϑ 至少在一个 δ 的阈值区间内保持稳定。对于分割得到的两个二值区域 r^{ϑ_-} 和 r^{ϑ_+} ，判断两者的面积变化 D_a 是否不大于给定阈值 Φ_r 。如果是，则认为 r^ϑ 是稳定性区域，加入稳定性区域集合 R_T ；否则认为是非稳定性区域，予以剔除。约束阈值 Φ_r 是一个与面积有关的函数，定义为

$$\Phi_r = \Delta_r \phi_r, \quad (5.6)$$

其中， Δ_r 用来约束面积变化的最大比例， ϕ_r 是与小目标临界大小相关的绝对面积值，计算公式为

$$\phi_r = \begin{cases} \max(|r^{\vartheta_-}|, |r^{\vartheta_+}|), & \text{if } \min(|r^{\vartheta_-}|, |r^{\vartheta_+}|) \geq t_s; \\ t_s, & \text{otherwise.} \end{cases} \quad (5.7)$$

上式中，当两个区域 r^{ϑ_-} 和 r^{ϑ_+} 的最小面积小于小目标的临界大小时， ϕ_r 被设置为临界大小 t_s ，否则取其中面积较大的值并利用 Δ_r 来约束 r^{ϑ_-} 和 r^{ϑ_+} 之间的面积变化。参数 Δ_r 和 t_s 的具体分析，详见第 5.5.2.2 节。



图 5.5 RS_T 算法检测结果

算法 5.2 最终产生的集合 R_T 中，包含了从场景中提取出的所有稳定性区域。需要再次强调的是，小目标是稳定性区域，但反之并不成立，即 R_T 只是候选小目标集合。如果将场景中候选小目标区域的像素值置为 1，其它部分的像素值置为 0，就得到了稳定性提取模块的输出结果，即稳定图 M_T 。图 5.5 示例了 RS_T 算法的部分检测结果，从上至下分别对应了实验中使用的三个数据集，图像左上角为帧号。从稳定图结果来看，RS_T 算法都正确检测出了场景中的目标区域。此外，第 1 行的各帧图像中，目标尺寸

存在较大的变化，这表明 RS_T 算法不仅可用于小目标检测，同时也适用于解决中等尺寸的无表观目标检测问题^①。

5.4.2 仅基于区域稳定性检测小目标的缺点

虽然图 5.5 表明了 RS_T 算法在小目标检测时的有效性，同时也显示出该算法对目标大小变化的适应能力，但仅基于区域稳定性的检测方式还是有其固有的缺陷。由于稳定性区域并不一定是小目标， RS_T 算法产生的集合 R_T 中有虚警目标存在，图 5.6 给出了这样的示例。在稳定图 5.6(c) 中，共检测到三个稳定性区域，从左至右的簇号分别为 #28、#96 和 #93。如果以它们的最小外接矩形从灰度图 5.6(b) 中分割出对应的子图（见图 5.6(c) 左下角），可以看到它们完全符合之前提出的稳定性假设：即本质上都是具有中心环绕差异的局部子块，目标内部具有近似均匀的强度，且与背景之间存在一定的强度差异。然而，后两者显然不是希望得到的小目标检测结果，也就是针对该帧输入，基于区域稳定性的 RS_T 算法产生了两个虚警目标。

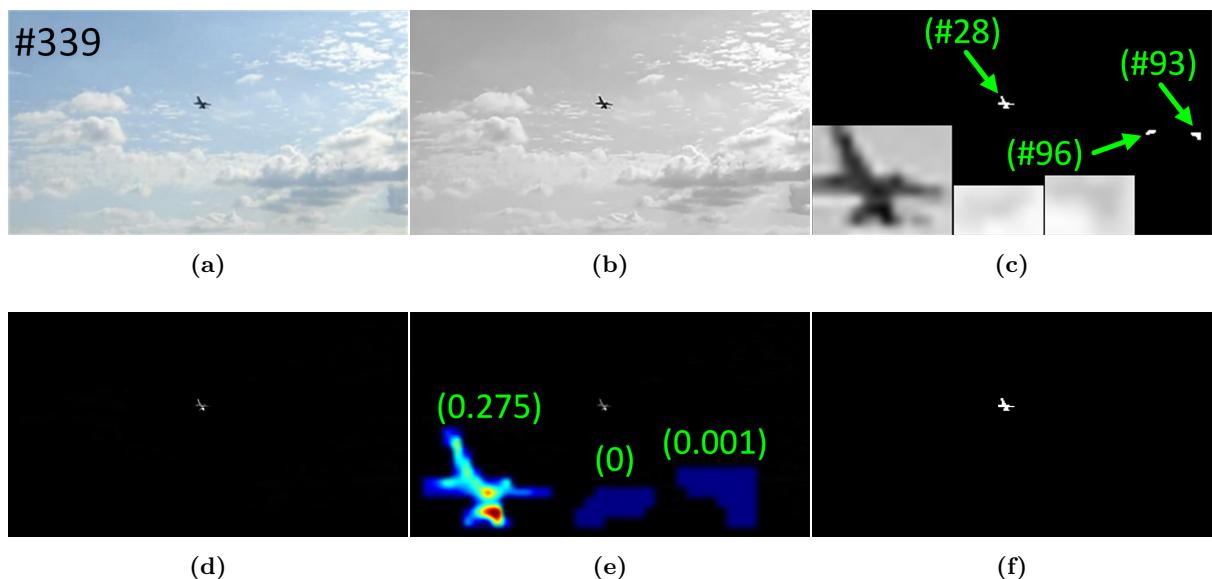


图 5.6 RS_T 算法缺点。(a) 数据集 #1 的第 #339 帧图像；(b) 图 (a) 的灰度图；(c) RS_T 算法得到的稳定图 M_T ；(d) 第 5.4.3 节提出的 RS_A 算法得到的显著图 M_A ；(e) 主特征图 M ；(f) RSS 方法检测结果

虚警目标的存在，将大大降低检测系统的性能。这是因为即便结果中已包含了待检测的小目标（即召回率 $R = 100\%$ ），精确率 P 仍将处于一个较低的水平，从而整体上拉低了 $F1$ 值（指标定义见第 1.2.4.2 节）。事实上，观察图 5.6(c) 左下角的三张子图，不难看出相较于 #28 目标，#96 和 #93 目标与背景之间的强度差异较小。虽然这些目标都至少在一个 δ 阈值区间内满足稳定性约束，但从人眼视觉的角度，#28 目标显然

^① 这里的无表观目标，是指同样具有小目标内蕴特征的前景图形区域，即区域内部具有近似均匀的强度、且与局部邻域之间存在一定的强度差异。

比 #93 和 #96 目标更为显著。因此，引入显著性检测方法，利用显著值来剔除掉那些不显著的稳定性区域，就可以有效减少虚警目标的数量，提高检测的精确率。

图 5.6(d) 是由下一节介绍的显著性方法生成的显著图，从该图中分割出的 3 个稳定性区域如图 5.6(e) 左下角所示，显著值分别为 0.275、0 和 0.001。由于后两者的显著值几乎为 0，基于显著性将它们从稳定性区域集合 R_T 中剔除后，就得到了图 5.6(f) 所示的检测结果。可以看到，区域显著性的引入，有效地解决了仅基于区域稳定性检测产生的虚警问题。但一个值得注意的现象是，假设仅基于显著图 5.6(d) 检测小目标的话，看起来也能得到理想的检测结果，这是否意味着仅基于显著性就可以完成检测任务，而不需要稳定性的参与？在下一节中，除提出显著性检测方法外，也将讨论仅基于显著性方法检测小目标的缺点，以表明稳定性与显著性两者融合的必要性与有效性。

5.4.3 基于区域显著性的虚警目标去除

与上一节稳定性 RS_T 算法相呼应，本章提出的显著性检测算法被简记为 RS_A ，这里下标字母 A 表示显著性。在正式介绍 RS_A 算法之前，首先来看一下基于频率调谐 (Frequency-tuned, 简记为 FT) 的方法^[32] 是如何实现显著物体检测的。FT 方法认为显著性检测器应该能够：1) 凸显最大的显著物体；2) 均匀地高亮整个显著物体区域；3) 检测出的显著物体有定义良好的轮廓；4) 忽略由纹理、噪声和块效应导致的高频；5) 输出与原图相同分辨率大小的显著图。因此，FT 方法将显著图 S 中任意坐标 (x, y) 处的显著值，定义为与图像平均特征向量之间的差异，即

$$S(x, y) = \|I_\mu - I_\omega(x, y)\|_2^2, \quad (5.8)$$

其中，符号 $\|\cdot\|_2$ 表示 ℓ_2 范数， I_μ 是整幅图像的平均特征向量， I_ω 是对原始图像使用 5×5 的二维高斯核平滑后的结果。如果在 LAB 颜色空间中度量特征差异的话，就可以将 I_ω 的均值作为平均特征向量 I_μ ， $S(x, y)$ 就是 I_μ 与三个通道 L_ω 、 A_ω 和 B_ω 在 (x, y) 坐标处的差异，即

$$S(x, y) = (L_\mu - L_\omega(x, y))^2 + (A_\mu - A_\omega(x, y))^2 + (B_\mu - B_\omega(x, y))^2, \quad (5.9)$$

其中， L_μ 、 A_μ 和 B_μ 分别是 L_ω 、 A_ω 和 B_ω 的均值。

5.4.3.1 小目标的局部中心环绕差异

式 (5.9) 直接体现了 FT 方法^[32] 检测显著物体的基本思想。以 LAB 颜色空间中的亮度通道 L 为例， L_μ 是平均亮度信息，如果把 L_μ 看作是场景中全局共性的亮度特征，那么 $L_\omega(x, y)$ 与 L_μ 的差值就是坐标 (x, y) 处的亮度特征与全局平均亮度的差异程度。进一步从 LAB 颜色特征整体上来看， $S(x, y)$ 就是坐标 (x, y) 处的颜色与全局平均颜色的差异程度。显然， $S(x, y)$ 越大，说明 (x, y) 处的颜色与全局平均颜色的差异越大；而与场景中全局共性的颜色差异大，说明 (x, y) 处的颜色是非共性的稀缺颜色。因此，FT 方法实际上是利用了全局颜色的稀缺性实现场景中的显著物体检测。接下来的问题

是, FT 方法是否同样适用于小目标检测呢? 或者说, 小目标区域的颜色特征是否同样具有上面所说的这种全局稀缺性?

这个问题不能简单地以“是”或“否”来回答, 因为这取决于显著性方法所针对的检测场景。在背景单一, 且小目标与背景之间有较大颜色差异的情况下, FT 方法也同样有效。但大部分情况下, 小目标区域往往不具有这种全局的颜色稀缺特性。例如, 图 5.7(a) 是来自于数据集 #2 的第 #200 帧图像, 海天背景下的红色箭头处有一个飞机小目标, 该目标所具有的灰色实际上并不属于场景中的稀缺颜色。包括海天交界处的山脉以及海面上的部分区域在内, 场景中有许多灰色区域存在。从 FT 方法检测得到的显著图 5.7(b) 也可以看到, 红色箭头处并没有检测出显著区域; 相反, 绿色和黄色椭圆处表明场景中更为稀缺的白色被 FT 方法赋予了更高的显著值。因此, 针对该帧图像, 基于全局稀缺性的 FT 方法的检测是失效的。图 5.7(c) 是提出的基于局部差异性的 RS_A 算法得到的显著图, 可以看到在红色箭头处成功检测出了小目标区域。当然, 基于局部差异性的方法对颜色特征差异明显的交界处同样会赋予较高的显著值(见绿色椭圆位置), 在介绍完 RS_A 算法的具体实现后, 将在第 5.4.4 节继续讨论这个问题。

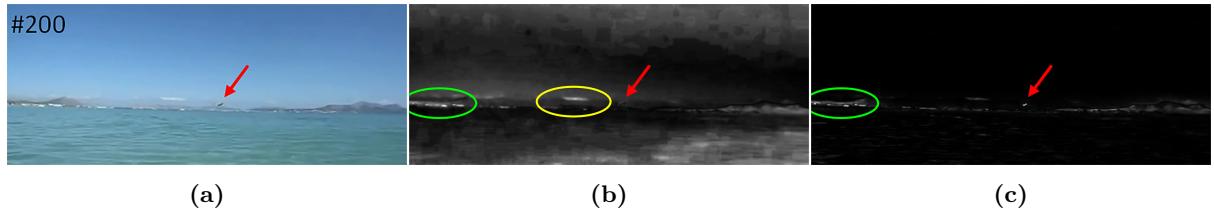


图 5.7 小目标的局部差异性。(a) 数据集 #2 的第 #200 帧图像; (b) FT 方法^[32] 得到的显著图; (c) RS_A 算法得到的显著图

5.4.3.2 基于局部差异性的显著性检测算法

虽然图 5.6(c) 左下角示例的是灰度子图, 但已反映了这里提出的 RS_A 算法在进行显著性检测时的基本思想, 即利用特征的局部差异性。由于包含小目标的局部区域是一个具有中心环绕模式的子块, RS_A 算法将显著性定义为中心目标与环绕背景之间的特征差异, 即显著图 M_A 中坐标 (x, y) 处的显著值为

$$M_A(x, y) = \|I(x, y) - I_\omega(x, y)\|_2^2, \quad (5.10)$$

其中, 符号 $\|\cdot\|_2$ 表示 ℓ_2 范数, I 是原始图像, I_ω 是对 I 高斯平滑后的结果。如果同样在 LAB 颜色空间中度量这种局部差异的话, 那么式 (5.10) 中的 $M_A(x, y)$ 可进一步定义为

$$M_A(x, y) = (L(x, y) - L_\omega(x, y))^2 + (A(x, y) - A_\omega(x, y))^2 + (B(x, y) - B_\omega(x, y))^2, \quad (5.11)$$

其中, L 、 A 和 B 是将图像 I 转换到 LAB 颜色空间后的三个通道, L_ω 、 A_ω 和 B_ω 是分别对这三个通道高斯平滑后的结果。在 RS_A 算法中, 高斯平滑使用的是一个大小由参

数 σ_s 控制的方形低通滤波器 (σ_s 的参数分析见第 5.5.2.2 节), 具体为 $3 \times \min(W, H)/\sigma_s$, 其中 W 和 H 是图像 I 的宽度和高度。

对比式(5.10)和式(5.8), 可以看到其中仅 ℓ_2 范数的第一项不同, 但两者实际上有着本质的区别。一方面, 高斯平滑的作用是舍弃了图像中的高频信号, 而与原始图像 I 的差值则恰恰凸显了作为高频信号的小目标区域; 另一方面, 对于高斯平滑结果中坐标 (x, y) 处的值 $I_\omega(x, y)$, 可以看作是 I 中以 (x, y) 为中心的一个小的局部邻域内的特征平均, 两者的差值体现了 $I(x, y)$ 与其局部邻域的平均特征的差异程度。显然, 局部特征的差异越大, 所得到的显著值越高。

图 5.8(c)–5.8(n) 示例了 12 个不同的显著性方法对图 5.8(a) 检测得到的显著图, 其中包括第 2 章提出的 RPC 方法、第 3 章提出的 CNS 方法、以及第 4 章提出的 HCN 方法, 图 5.8(o) 是 RS_A 算法得到的显著图。可以看到, RS_A 算法的一个显著优势就是在凸显小目标区域的同时, 很好地抑制了背景区域。需要说明的是, 虽然大部分的显著性方法并不是专门针对小目标检测任务而设计, 但其中部分方法也同样适合于检测小的显著区域。例如, 李健等人认为 SR 方法^[37] 对于中心环绕对比强烈的小的显著区域有较好的检测效果, 由此基于频域的尺度空间分析提出了 HFT 方法^[46], 该方法可同时针对不同尺寸大小的显著区域进行检测。因此, 在第 5.5.3.2 节中, 将对上述这些显著性方法在三个不同应用场景的检测性能进行全面评估。

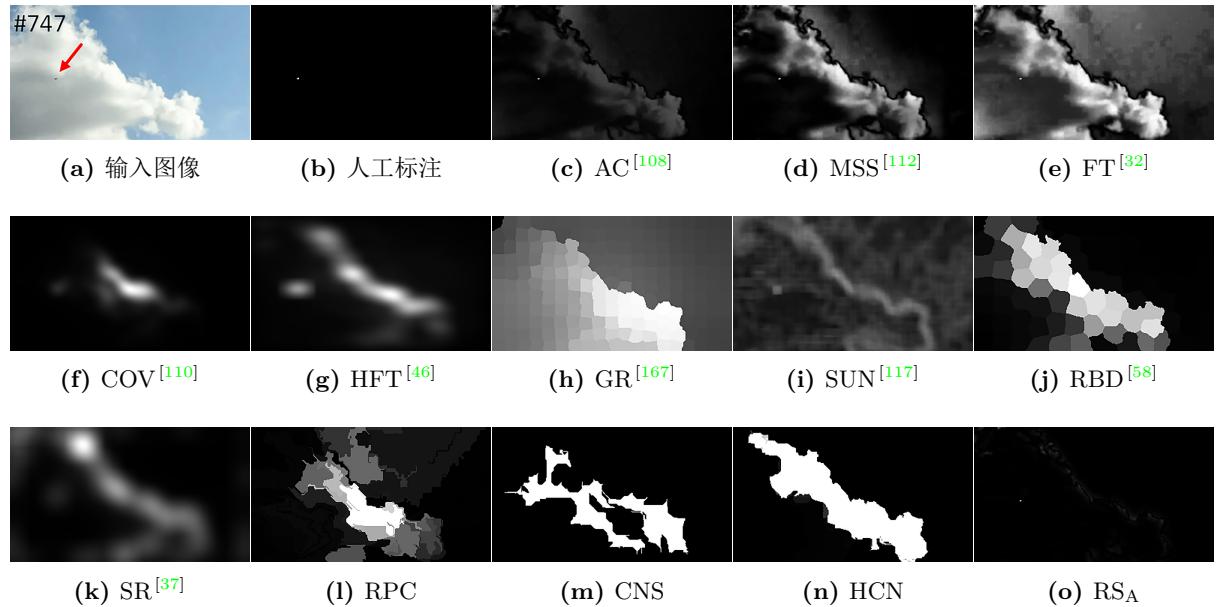


图 5.8 与 12 个显著性方法的视觉比较。(a) 数据集 #1 的第 #747 帧图像; (b) 图 (a) 的人工标注结果; (c)–(n) 不同显著性方法得到的显著图; (o) RS_A 算法得到的显著图

5.4.4 仅基于区域显著性检测小目标的缺点

虽然图 5.8 表明了 RS_A 算法针对小目标的检测优势, 但仅基于区域显著性的检测方式还有其一定的缺陷。同样以数据集 #2 的第 #200 帧图像为例, 图 5.9(c) 是 RS_A 算

法得到的显著图，可以看到海天交界的条带上获得了较高的显著值，这种情况的产生是由交界处较大的局部颜色差异导致。在某些情况下，差异对比强烈的局部区域甚至会超过小目标的显著值（见绿色椭圆处），这使得从显著图中分割出小目标有相当大的难度。更为关键的是，如果仅基于显著图实现小目标的分割操作，那么不可避免地又要带来阈值选取的问题。而在区域稳定性这一节中已讨论过，针对位于场景不同位置且具有一定尺寸变化的小目标，要寻找到有效的自适应阈值实际上非常困难。特别是当显著图中出现大量小的高显著性区域时，根本无从得知哪些才是真正的小目标。

为了避免对显著图的分割操作，这里将显著图 M_A 与区域稳定性产生的稳定图 M_T 相融合。如第 5.3 节融合模块功能所述，利用显著图计算每个稳定性区域的平均显著值，然后基于所有区域的显著值均值剔除掉非显著的稳定性区域，最终得到的就是理想的小目标。图 5.9(d) 是上一节的 RS_T 算法得到的稳定图，其中的两个稳定性区域的显著值分别为 0.05 和 0.44（见图 5.9(e））。依据上述规则，大于所有稳定性区域显著值均值（即 0.245）的簇 #97 予以保留，为最终的小目标检测结果，如图 5.9(f) 所示。

结合第 5.4.2 节，仅基于区域稳定性 RS_T 算法或者区域显著性 RS_A 算法检测小目标，都有着各自的缺陷。对于 RS_T 算法，由于稳定性区域并不一定都是小目标区域，由此导致的虚警问题将影响检测结果的精确率；而对于 RS_A 算法，局部差异对比强烈的区域可能产生比小目标区域更高的显著值，同时也面临着对显著图直接分割困难的问题。本章提出的 RSS 方法将稳定图和显著图通过 Hadamard 积、即 $M_T \circ M_A$ 的方式融合，其实是一种非常简单的融合技术，却是解决上述问题的一种有效手段。在下一节中，将实际评估这种融合手段的有效性，同时推广到对现有稳定性和显著性方法的融合，并加以验证。

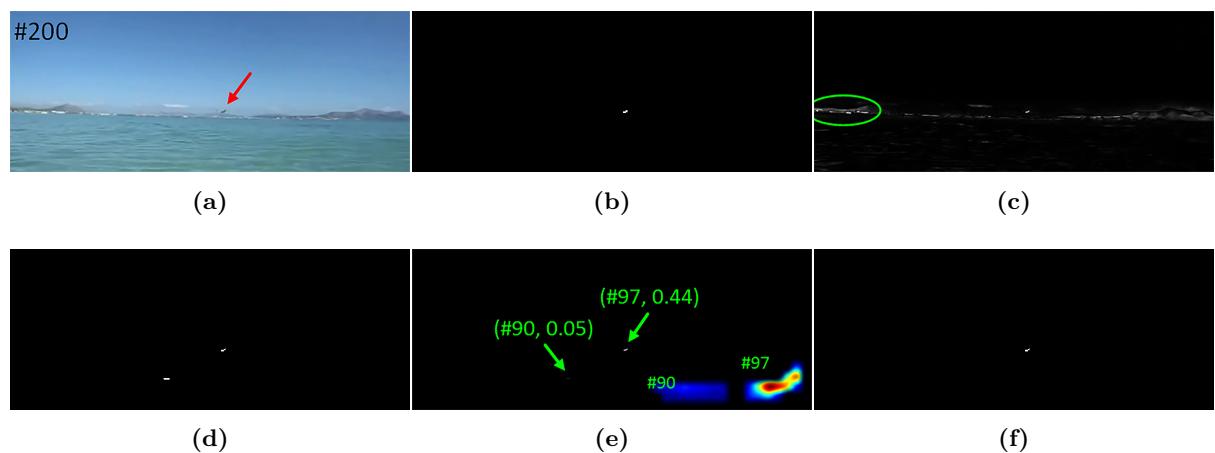


图 5.9 RS_A 算法缺点。(a) 数据集 #2 的第 #200 帧图像；(b) 图 (a) 的人工标注结果；(c) RS_A 算法得到的显著图 M_A ；(d) 第 5.4.1 节提出的 RS_T 算法得到的稳定图 M_T ；(e) 主特征图 M ；(f) RSS 方法检测结果

5.5 实验与讨论

为了验证本章提出的 RSS 方法针对单帧彩色图像小目标检测的有效性，在三个数据集上实际评估并分析了方法性能，这三个数据集涵盖了典型的小目标应用场景。同时，既然 RSS 方法是基于稳定性和显著性的融合，实验评估从以下三个方面分别进行：基于稳定性的 RS_T 算法、基于显著性的 RS_A 算法、融合稳定性与显著性的 RSS 方法。

- (1) 对于 RS_T 算法：与基于 MSER^[164,165] 的检测方法比较。这里首先通过调用 VLFeat 开源库^[168] 中的 VL_MSER 函数得到最大稳定极值区域即 MSER，然后利用这些稳定区域检测出场景中的小目标^①。MSER 方法的参数分析见第 5.5.2.1 节，与 RS_T 算法的比较见第 5.5.3.1 节。
- (2) 对于 RS_A 算法：分别与 AC^[108]、MSS^[112]、FT^[32]、COV^[110]、HFT^[46]、GR^[167]、SUN^[117]、RBD^[58]、SR^[37]、第 2 章提出的 RPC 方法、第 3 章提出的 CNS 方法、以及第 4 章提出的 HCN 方法，共计 12 个显著性方法进行比较。其中前 9 个方法名的英文缩写，与文献 [28] 保持一致，它们在三个数据集上的显著图，是通过运行 A. Borji 等人在显著性基准数据库^[28] 中发布的 C++ 或 MATLAB 代码得到。上述显著性方法与 RS_A 算法的比较见第 5.5.3.2 节。
- (3) 对于 RSS 方法：将上述稳定性方法和显著性方法采用与 RSS 相同的方式两两融合，来实现小目标检测，与 RSS 方法的比较见第 5.5.3.3 节。

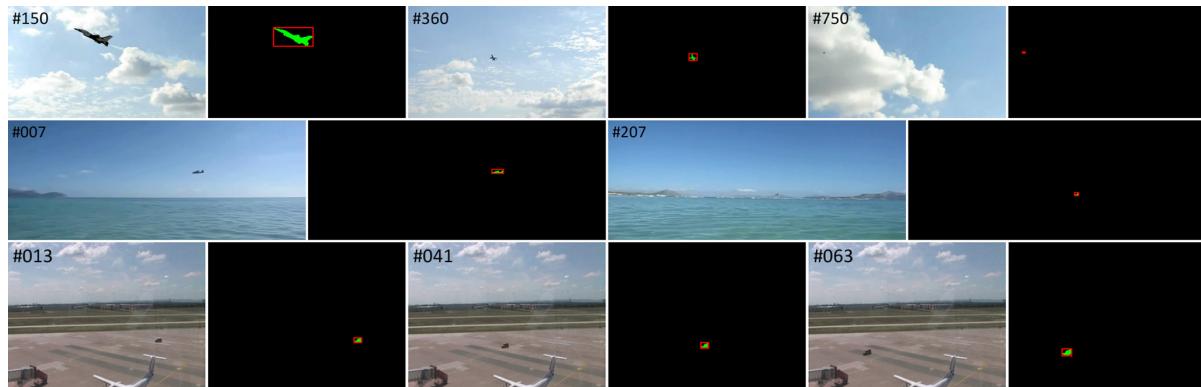


图 5.10 数据集部分图像与人工标注

5.5.1 数据库简介

为了综合评估 RSS 方法针对不同应用场景的有效性，本章标注并发布了一个新的针对彩色图像小目标检测的基准数据库。如图 5.10 所示，该数据库由三个数据集组成，涵盖了典型的应用场景，包括天空背景（第 1 行 – 数据集 #1）、海天背景（第 2 行 – 数据集 #2）和地面背景（第 3 行 – 数据集 #3），各数据集的详细介绍见表 5.2。在每帧图像的

^① 基于 MSER 检测小目标的 MATLAB 代码下载：<https://github.com/jinglou/p2016-rss-small-target/tree/master/MSER>。

人工标注中，提供了像素级的标注图以及帧内每个小目标的四个区域属性，包括目标面积、轮廓点坐标、最小外接矩形以及目标内各像素点坐标。

有两点需要特别指出的是：1) 数据集 #1 的第 #753 – #805 帧中只包含场景背景，而没有实际的小目标存在。保留这部分图像的意义，是希望能够测试小目标检测方法在只有背景的情况下是否会产生虚警。2) 数据集 #1 和 #2 中的各帧图像是由两个视频序列逐帧抽取得到，虽然论文主页中 RSS 的结果是以视频方式演示，但这些演示视频都是由单帧检测结果逐帧合并而成，RSS 方法本身并不解决时序状态下的小目标检测问题。

表 5.2 数据集简介

序号	场景	图像数	图像大小	备注 (单帧图像中包含的小目标个数)
#1	天空	805	320×180	#001 – #752: 单个目标 #753 – #805: 无目标
#2	海天	208	512×205	单个目标
#3 [†]	地面	80	288×216	单个目标

[†] 数据集 #3 的原始图像见：<http://people.ee.ethz.ch/~dragonr/943/>^[169]，原始图像大小为 1440×1080 ，实验中等比例缩小为原图的 1/5。

5.5.2 参数分析

RSS 方法共包含 4 个参数，分别是稳定性算法 RS_T 中的阈值采样间隔 δ 、小目标临界面积大小 t_s 、面积变化最大比例 Δ_r ，以及显著性算法 RS_A 中用于控制高斯低通滤波器大小的参数 σ_s 。最优参数的选取过程与第 2 – 4 章相同，也是在预设参数取值范围内，通过分析参数曲线方式进行。但与之前三章使用 F_β 曲线峰值确定最优参数不同，由于本章所解决的小目标检测问题认为精确率 P 和召回率 R 同等重要，因此不再像计算 F_β 那样对 P 赋予较高的权重，而是使用 $F1$ 指标，即对 P 和 R 赋予相同的权重。有关 F_β 指标和 $F1$ 指标的区别，详见第 1.2.4 节。

对于精确率 P 和召回率 R ，是通过小目标的检出率 (Detection Rate) 计算得到。判断单个小目标是否被正确检出的依据，是要求检出的目标与人工标注之间的像素重合率不低于 50%。对于单帧图像，在统计出帧内正确检出的小目标个数后，利用下式计算该帧的精确率 P 和召回率 R

$$P = \frac{\text{正确检出的小目标个数}}{\text{检测出的小目标个数}}, \quad (5.12)$$

$$R = \frac{\text{正确检出的小目标个数}}{\text{人工标注的小目标个数}}, \quad (5.13)$$

再利用式 (1.6) 计算该帧的 $F1$ 指标。对于单个数据集上的 P 、 R 和 $F1$ 指标，是该数据集中所有帧在这三项指标上的指标均值。

特例情况是数据集 #1 中的第 #753 – #805 帧，由于这些帧内没有实际的小目标存在，故如果检测方法同样未检测到任何的目标区域，则 P 和 R 都计为 1，相应的 $F1 = 1$ ；否则两者都计为 0，相应的 $F1 = 0$ 。

5.5.2.1 MSER 方法参数分析

在调用 VLFeat 开源库^[168] 中的 VL_MSER 函数实现 MSER 方法时需要提供 3 个参数，分别是采样阈值间隔 δ 、区域最小差异 d_m ^①、区域最大变化 v_m （绝对稳定值），默认参数值为 $\delta = 5$ 、 $d_m = 0.2$ 、 $v_m = 0.75$ 。之所以对 MSER 方法进行参数分析，一是针对本章数据集，使用默认参数得到的检测结果较差；二是由于 RSS 方法基于稳定性区域的平均显著值检测小目标，一旦稳定性算法产生漏检（即召回率不为 100%），就无法再通过显著性方法找回。因此，考虑到稳定性算法的重要性，同时也为了公平的比较，这里不再使用 VL_MSER 函数的默认参数，而是将 3 个参数的取值范围分别预设为： $\delta \in [4 : 4 : 64]$ 、 $d_m \in [0.1 : 0.1 : 0.9]$ 、 $v_m \in [0.1 : 0.1 : 0.9]$ 。

为了与 RS_T 算法保持一致的处理流程，在具体实现 MSER 方法时：1) 小目标区域的面积大小被限定在 $[4, 0.2 \times W \times H]$ 区间内，其中 W 和 H 分别表示输入图像的宽度和高度，即最小面积为 4 个像素，最大为图像面积的 20%；2) MSER 方法同样也不检测边界目标区域，同时当稳定性区域之间存在包含关系时剔除其中面积较小的区域；3) 由于 RS_T 算法仅对灰度图 G 的反相图进行检测，因此 VL_MSER 函数也仅检测亮背景下的暗区域，即设置参数 $BrightOnDark = 0$ 且 $DarkOnBright = 1$ 。

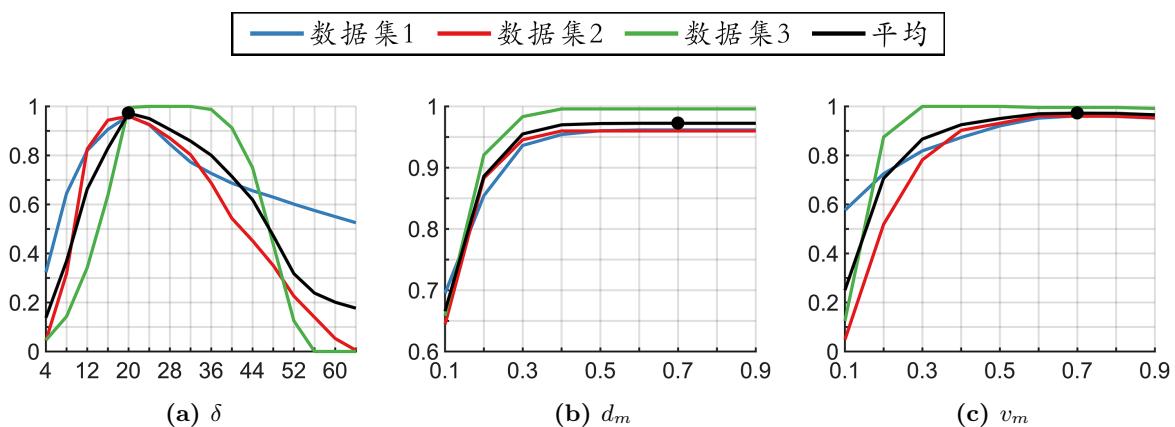


图 5.11 MSER 方法参数分析： x 轴表示参数取值范围， y 轴表示 $F1$ 的值

对 MSER 方法的调参过程与之前类似，区别仅在于这里是以 $F1$ 曲线峰值处的 x 坐标作为参数的最优取值。图 5.11(a) – 5.11(c) 是对应 3 个参数的参数曲线，其中 x 轴表示预设的参数范围、 y 轴表示 $F1$ 的值，三个数据集以及三者均值的 $F1$ 曲线分别用蓝色、红色、绿色、黑色表示，均值曲线的峰值点用黑色圆点标记。所有参数曲线的具体指标值见表 5.3，表中以红色标记 $F1$ 均值的最大值。参数分析如下：

^① 当两个区域存在包含关系且相应的面积变化小于该阈值时，仅保留其中最稳定的一个。

(1) 参数 δ 的曲线在上升到峰值点后开始逐渐下降, 这意味着随着 δ 的不断增大, 小目标已不能在更大的阈值区间内保持稳定。

(2) 参数 d_m 和 v_m 的曲线都是在上升到峰值点后趋于稳定, 表明峰值点后更大的面积变化比例已不会影响到对稳定性的度量。

(3) 三个数据集的参数 d_m 曲线都在上升至峰值点 $d_m = 0.6$ 后不再发生变化, 使得 $d_m \in \{0.6, 0.7, 0.8, 0.9\}$ 时获得了相同的 $F1$ 值。这里与参数 v_m 的最优取值保持一致, 对 d_m 取值 0.7。最终确定的 MSER 方法的参数最优取值见表 5.4。

表 5.3 MSER 方法参数分析 (三个数据集分别用 #1、#2、#3 表示)

(a) δ				(b) d_m				(c) v_m						
δ	F1			d_m	F1			v_m	F1					
	#1	#2	#3		#1	#2	#3		#1	#2	#3			
4	.3227	.0446	.0460	.1378	0.1	.6949	.6442	.6583	.6658	0.1	.5764	.0481	.1250	.2498
8	.6454	.3182	.1430	.3689	0.2	.8543	.8830	.9208	.8861	0.2	.7255	.5192	.8750	.7066
12	.8203	.8264	.3422	.6630	0.3	.9363	.9455	.9833	.9551	0.3	.8186	.7821	1.	.8669
16	.9061	.9439	.6371	.8290	0.4	.9541	.9599	.9958	.9700	0.4	.8733	.9022	1.	.9252
20	.9616	.9599	.9958	.9724	0.5	.9603	.9599	.9958	.9720	0.5	.9205	.9311	1.	.9505
24	.9251	.9263	1.	.9504	0.6	.9616	.9599	.9958	.9724	0.6	.9522	.9615	.9958	.9698
28	.8460	.8702	1.	.9054	<u>0.7</u>	.9616	.9599	.9958	.9724	<u>0.7</u>	.9616	.9599	.9958	.9724
32	.7727	.8029	1.	.8585	0.8	.9616	.9599	.9958	.9724	0.8	.9592	.9599	.9958	.9717
36	.7267	.6875	.9875	.8006	0.9	.9616	.9599	.9958	.9724	0.9	.9522	.9535	.9917	.9658
40	.6870	.5433	.9125	.7142										
44	.6559	.4519	.7500	.6193										
48	.6298	.3510	.4375	.4728										
52	.6012	.2260	.1250	.3174										
56	.5752	.1394	0	.2382										
60	.5503	.0529	0	.2011										
64	.5255	.0048	0	.1768										

表 5.4 MSER 方法最优参数

参数名	最优参数值
δ	20
d_m	0.7
v_m	0.7

5.5.2.2 RSS 方法参数分析

RSS 方法的 4 个参数中, δ 的预设取值范围与 MSER 方法相同, 即 $\delta \in [4 : 4 : 64]$; 其它 3 个参数的预设取值范围分别是 $t_s = k^2$, $k \in [2 : 16]$ 、 $\Delta_r \in [0.1 : 0.02 : 0.4]$ 、 $\sigma_s \in [4 : 2 : 32]$ 。对 RSS 方法的调参过程与 MSER 相同, 同样以 $F1$ 曲线峰值处的 x 坐标作为参数的最优取值。图 5.12(a) – 5.12(d) 是 RSS 方法 4 个参数的参数曲线, 其中 x 轴表示预设的参数范围、 y 轴表示 $F1$ 的值, 三个数据集以及三者均值的 $F1$ 曲线分别用蓝色、红色、绿色、黑色表示, 均值曲线的峰值点用黑色圆点标记。所有参数曲线的具体指标值见表 5.6, 表中以红色标记 $F1$ 均值的最大值。依据参数曲线, 最终确定的 RSS 方法中 4 个参数的最优取值见表 5.5, 相关分析如下:

(1) 阈值采样间隔 δ 的曲线同样在上升到峰值点后开始逐渐下降, 就三个数据集平均而言, 当超过峰值点 $\delta = 16$ 后, 小目标已不能在更大的阈值区间内保持稳定。

(2) 在数据集 #1 和 #2 上, 小目标临界面积大小 t_s 、面积变化最大比例 Δ_r 、以及滤波器大小 σ_s 这 3 个参数的变化趋势较为一致。其中, 参数 σ_s 的变化对于 RSS 方法在这两个数据集上的性能没有任何影响。

(3) 数据集 #3 的参数曲线变化趋势与前两个数据集有明显不同, 表明数据集 #3 受参数变化的影响最大, 这主要是三个数据集不同的背景复杂度导致。数据集 #1 是天空背景、复杂度最低, 且无论是蓝色天空还是白色云朵, 都与小目标有较大的反差。数据集 #2 是海天背景、复杂度次之, 虽然图像序列最后有海天交界处的山脉和海面干扰, 但小目标大部分情况处于天空背景中, 反差同样较为明显。相较于前两个数据集, 数据集 #3 的背景最为复杂: 由于 RSS 方法的性能主要依赖于稳定性 RS_T 算法, 数据集 #3 背景中诸多满足稳定性约束的区域带来了更多的虚警目标; 同时, 数据集 #3 对参数 σ_s 的变化最为敏感 (见表 5.6(d)), 由于 RS_A 算法得到的显著图更为杂乱, 导致利用显著性剔除虚警目标的手段失效。

(4) 国际光学工程学会 (Society of Photo-Optical Instrumentation Engineers, SPIE) 对小目标的定义是不超过 80 个像素 ($\approx 9 \times 9$), 这对于典型的 256×256 大小的图像而言, 小目标面积不会超过图像面积的 0.15%^[157,170]。RSS 方法中的小目标临界面积大小 t_s , 三个数据集的 $F1$ 均值曲线表明最优取值为 100。从表 5.6(b) 中可以看到, 当 $t_s = 81$ 时 $F1$ 均值指标排名第二, 低于 $t_s = 100$ 时约 0.43%。但考虑到待检测目标在数据集上特别是数据集 #1 中有较大的尺寸变化, t_s 的最优参数仍然取值为 100。

表 5.5 RSS 方法最优参数

参数名	最优参数值
δ	16
t_s	100
Δ_r	0.2
σ_s	18

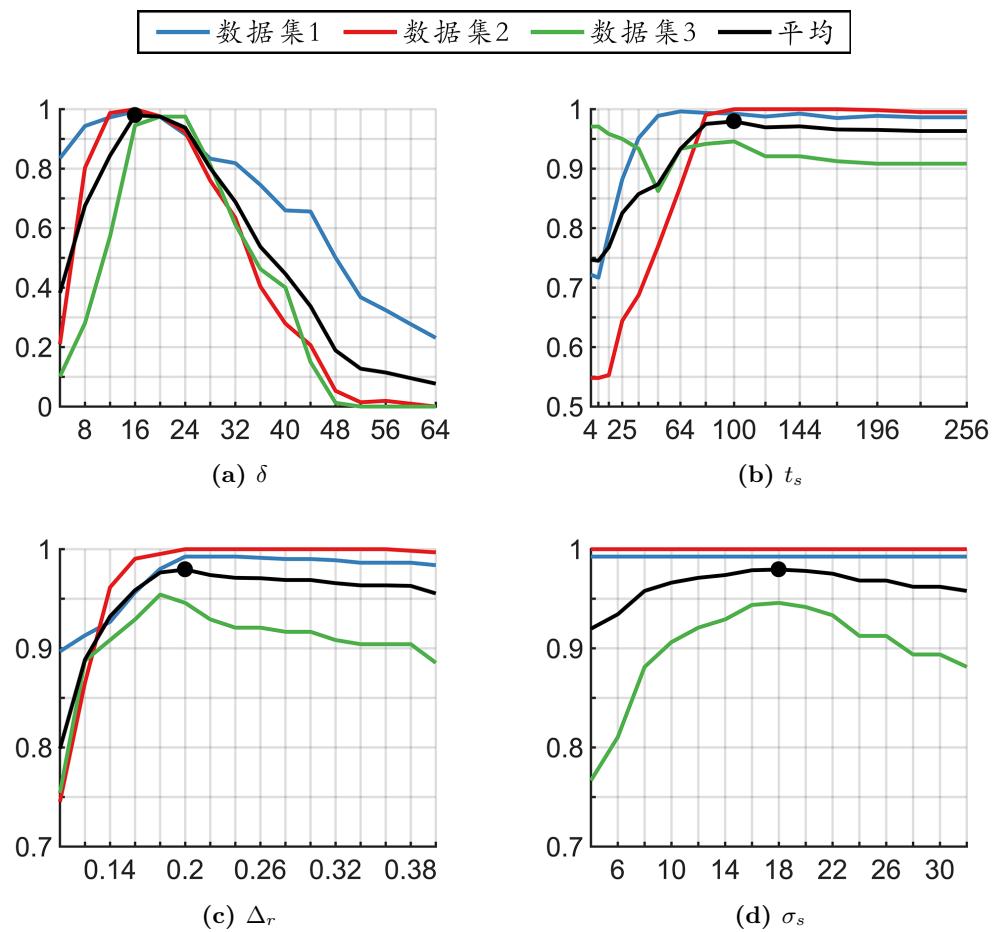


图 5.12 RSS 方法参数分析: x 轴表示参数取值范围, y 轴表示 $F1$ 的值

表 5.6 RSS 方法参数分析 (三个数据集分别用 #1、#2、#3 表示)

(a) δ				(b) t_s				(c) Δ_r				(d) σ_s							
δ	F1			t_s	F1			Δ_r	F1			σ_s	F1						
	#1	#2	#3		#1	#2	#3		#1	#2	#3		#1	#2	#3				
4	.8351	.2101	.1015	.3822	4	.7217	.5481	.9708	.7469	0.1	.8969	.7452	.7542	.7988	4	.9925	1.	.7667	.9197
8	.9433	.8030	.2805	.6756	9	.7168	.5481	.9708	.7452	0.12	.9130	.8654	.8875	.8886	6	.9925	1.	.8104	.9343
12	.9727	.9872	.5725	.8441	16	.7925	.5529	.9583	.7679	0.14	.9267	.9615	.9083	.9322	8	.9925	1.	.8812	.9579
16	.9925	1.	.9458	.9795	25	.8820	.6442	.9500	.8254	0.16	.9565	.9904	.9292	.9587	10	.9925	1.	.9063	.9663
20	.9739	.9760	.9750	.9750	36	.9516	.6875	.9333	.8575	0.18	.9801	.9952	.9542	.9765	12	.9925	1.	.9208	.9711
24	.9155	.9231	.9750	.9379	49	.9888	.7692	.8625	.8735	<u>0.2</u>	.9925	1.	.9458	.9795	14	.9925	1.	.9292	.9739
28	.8335	.7596	.8125	.8019	64	.9963	.8702	.9333	.9333	0.22	.9925	1.	.9292	.9739	16	.9925	1.	.9437	.9788
32	.8186	.6346	.6125	.6886	81	.9938	.9904	.9417	.9753	0.24	.9925	1.	.9208	.9711	<u>18</u>	.9925	1.	.9458	.9795
36	.7453	.4038	.4625	.5372	100	.9925	1.	.9458	.9795	0.26	.9913	1.	.9208	.9707	20	.9925	1.	.9417	.9781
40	.6596	.2788	.4000	.4462	121	.9876	1.	.9208	.9695	0.28	.9901	1.	.9167	.9689	22	.9925	1.	.9333	.9753
44	.6559	.2067	.1500	.3375	144	.9925	1.	.9208	.9711	0.3	.9901	1.	.9167	.9689	24	.9925	1.	.9125	.9683
48	.5006	.0529	.0125	.1887	169	.9851	1.	.9125	.9659	0.32	.9888	1.	.9083	.9657	26	.9925	1.	.9125	.9683
52	.3677	.0144	.0	.1274	196	.9888	.9984	.9083	.9652	0.34	.9863	1.	.9042	.9635	28	.9925	1.	.8938	.9621
56	.3242	.0192	.0	.1145	225	.9863	.9952	.9083	.9633	0.36	.9863	1.	.9042	.9635	30	.9925	1.	.8938	.9621
60	.2770	.0096	.0	.0955	256	.9863	.9952	.9083	.9633	0.38	.9863	.9984	.9042	.9630	32	.9925	1.	.8813	.9579
64	.2311	.0	.0	.0770						0.4	.9839	.9968	.8854	.9554					

5.5.3 与其它方法的性能比较

本节将把本章提出的 RS_T 算法、 RS_A 算法、RSS 方法，分别在三个数据集上与其它稳定性方法、显著性方法、融合方法进行定量的性能比较。

5.5.3.1 与基于稳定性方法的性能比较

表 5.7 给出了 RS_T 算法与 MSER 方法^[164,165] 在三个数据集上的指标结果。可以看到， RS_T 算法在数据集 #2 和 #3 上获得了 100% 的召回率 R ，而在数据集 #1 上召回率没有达到 100% 的原因是产生了一帧漏检，在第 5.5.4 节的结果讨论中将给出这个失败的示例。MSER 方法在数据集 #1 和 #2 上的召回率都要低于 RS_T 算法，这意味着该方法在部分帧上产生了漏检。

然而，与 MSER 方法相比， RS_T 算法在三个数据集上得到的精确率 P 都相对较低，特别是在数据集 #3 上，从而直接导致了较低的 $F1$ 值。精确率指标较低，意味着 RS_T 算法产生了更多的虚警。正如在第 5.4.2 中所讨论的，仅基于区域稳定性检测小目标的缺点就是虚警目标的存在，这也是在 RSS 方法中引入显著性 RS_A 算法的动机。事实上，相较于精确率，召回率要更为重要，这是因为后续总可以找到某种处理手段将虚警目标剔除，例如本章提出的显著性方法就是其中的一种。但一旦稳定性方法丢失了检测目标，则后续很难再找回，即便将稳定性与显著性结合可以解决精确率的问题，召回率却并不会得到改善。

表 5.7 稳定性方法的性能比较

方法	数据集 #1			数据集 #2			数据集 #3			平均		
	P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$
MSER ^[164,165]	.9525	.9801	.9616	.9543	.9712	.9599	.9938	1.	.9958	.9669	.9838	.9724
RS_T	.8985	.9925	.9239	.9060	1.	.9346	.5094	1.	.6446	.7713	.9975	.8344

5.5.3.2 与基于显著性方法的性能比较

在进行显著性方法的性能比较时，与稳定性方法有所不同。由于本章发布的基准数据库中的人工标注图同样是像素值 $\in \{0, 1\}$ 的二值图像，因此仍然采用之前三章使用的显著性通用评价方法，即通过固定阈值 T_f 和自适应阈值 T_a ^[32] 来评价不同显著性方法的精确率 P 、召回率 R 、以及 F_β 指标，指标定义同第 1.2.4.3 节。

图 5.13(a) – 5.13(f) 给出了基于固定分割阈值 T_f 得到的 $P-R$ 曲线图和 F_β 曲线图，表 5.8 给出了对应的 AvgF 和 MaxF 指标值，各指标的前三名分别用红色、绿色和蓝色标记。对于 AvgF 指标， RS_A 算法仅在数据集 #1 上要低于 AC 方法 1.89%、排名第二，在其它两个数据集上都排名第一。对于关键的 MaxF 指标， RS_A 算法在三个数据集上都排名第一。三个数据集平均而言， RS_A 算法的 AvgF 指标排名第一，高出排名第二的

AC 方法 10.89%；同时 MaxF 指标也排名第一，高出排名第二的 AC 方法 21.97%。此外，与第 2–4 章提出的方法相比， RS_A 算法具有明显的性能优势，表明 RS_A 算法在小目标检测问题上更具有适用性。

但从表 5.8 给出的基于自适应分割阈值 T_a 得到的 AdpF 值来看， RS_A 算法处于明显劣势。但是，针对小目标检测问题，采用显著图均值的两倍作为自适应分割阈值并统计 AdpF 指标的评估方式，实际上并不具有意义。以数据集 #1 为例，可以看到排名前三的 GR、RBD 和 SUN 方法的 AdpF 指标要明显优于其它显著性方法，但从图 5.8 的显著图来看，这三个方法得到的检测结果并不理想，远不如 AC、MSS、HFT 以及本章提出的 RS_A 算法。事实上，如果显著性检测方法不能很好地抑制背景区域的显著值，就会使得显著图的整体均值较大，而利用该均值的两倍作为阈值进行显著图分割，反而可以有效地剔除背景虚警，得到更高的 F_β 指标；相反，如果显著性检测方法在凸显小目标的同时很好地抑制了背景显著值，则对应的显著图均值会非常小，较小的分割阈值反正会产生更多的虚警区域，从而使得 F_β 指标相对较低。因此，利用自适应分割阈值 T_a 得到的 AdpF 指标，实际上并不能有效表征显著性方法对于小目标检测任务的性能。

基于上述理由，在图 5.13(g) 至 5.13(i) 中，不再使用基于自适应阈值 T_a 得到的 $P-R-F_\beta$ 柱状图，而转为统计基于固定阈值 T_f 得到的 P 、 R 和 F_β 的指标均值，即 AvgP–AvgR–AvgF 柱状图，图中以显著性方法的 AvgF 值做升序排列。各显著性方法的 AvgP、AvgR、AvgF 具体指标值见表 5.9。可以看到， RS_A 算法的 AvgP 指标在所有数据集上都排名第一，就三个数据集平均而言， RS_A 算法的 AvgP 指标比排名第二的 AC 方法高出了 29.4%。

虽然 RS_A 算法获得了最好的 AvgP 指标，但表 5.9 也反映出该算法的 AvgR 指标处于相对较低的水平，该指标在任何一个数据集上都未能进入前三名，表明单独使用 RS_A 算法检测小目标时存在着漏检问题。但值得注意的是，如果同时比较表 5.7 和表 5.9，就会发现基于稳定性的 RS_T 算法获得了更高的召回率 R 、而基于显著性的 RS_A 算法则获得了更高的精确率 P ^①。直觉上，这就可以将两个算法进行融合，来弥补单独使用其中某个算法的不足。正如在第 5.4.2 节和第 5.4.4 节中所讨论的，仅基于 RS_T 算法检测小目标时面临着虚警问题，仅基于 RS_A 算法检测小目标又面临着阈值分割的问题，从而可能导致产生虚警或漏检。正是由于仅基于稳定性或显著性来检测小目标存在着各自的缺点，本章才考虑采用融合方式加以解决。本章提出的由稳定性算法检测出候选小目标区域，再由显著性算法依据各候选目标的显著值剔除虚警目标的 RSS 方法，就是一种非常简单有效的融合方法。

^①为了与基于自适应阈值 T_a 得到的 $P-R-F_\beta$ 柱状图有所区别，本节使用 AvgP–AvgR–AvgF 来表示基于固定阈值 T_f 得到的柱状图。虽然 AvgP 或 AvgR 反映的是在不同固定阈值下的指标均值，但这些指标可以被看作是显著性方法的平均性能，因此表 5.9 中的 AvgP 和 AvgR 指标，可以与表 5.7 中的 P 和 R 指标被相应等同看待。

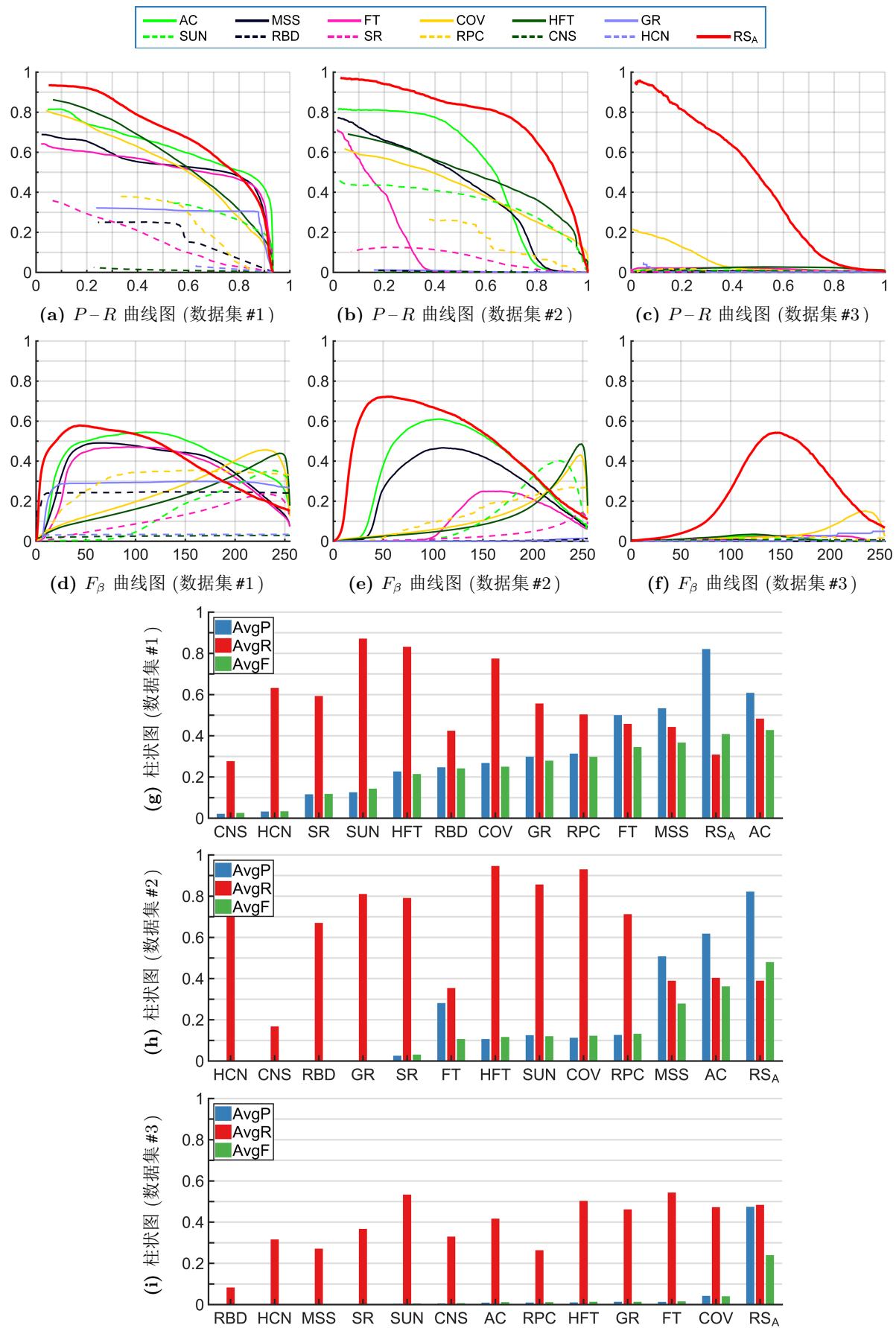


图 5.13 与 12 个显著性方法的性能比较

表 5.8 显著性方法的 F_β 相关指标比较

序号	方法	数据集 #1			数据集 #2			数据集 #3			平均		
		AvgF	MaxF	AdpF	AvgF	MaxF	AdpF	AvgF	MaxF	AdpF	AvgF	MaxF	AdpF
1	AC ^[108]	.4269	.5446	.0313	.3618	.6100	.0087	.0112	.0282	.0078	.2666	.3943	.0160
2	MSS ^[112]	.3668	.4907	.0285	.2779	.4665	.0026	.0033	.0090	.0060	.2160	.3221	.0124
3	FT ^[32]	.3450	.4711	.0249	.1059	.2498	.0029	.0147	.0297	.0096	.1552	.2502	.0125
4	COV ^[110]	.2493	.4560	.0390	.1220	.4302	.0071	.0399	.1513	.0066	.1371	.3459	.0175
5	HFT ^[46]	.2138	.4390	.0481	.1161	.4857	.0072	.0128	.0344	.0120	.1142	.3197	.0224
6	GR ^[167]	.2790	.2989	.3054	.0030	.0148	.0062	.0133	.0491	.0009	.0984	.1209	.1042
7	SUN ^[117]	.1428	.3526	.1516	.1198	.4026	.0131	.0052	.0138	.0137	.0892	.2563	.0595
8	RBD ^[58]	.2411	.2466	.1673	.0029	.0134	.0020	.0003	.0023	.0001	.0814	.0874	.0565
9	SR ^[37]	.1171	.2353	.0315	.0308	.1413	.0042	.0033	.0102	.0086	.0504	.1289	.0148
10	RPC	.2972	.3540	.0299	.1314	.2730	.0030	.0113	.0278	.0053	.1466	.2183	.0128
11	CNS	.0250	.0289	.0201	.0006	.0006	.0006	.0064	.0072	.0065	.0107	.0122	.0091
12	HCN	.0332	.0340	.0414	.0005	.0005	.0000	.0004	.0013	.0000	.0113	.0119	.0138
13	RS _A	.4080	.5779	.0525	.4795	.7217	.0044	.2391	.5423	.0091	.3755	.6140	.0220
平均		.2419	.3484	.0747	.1348	.2931	.0048	.0278	.0697	.0066	.1348	.2371	.0287

表 5.9 显著性方法的精确率、召回率和 F_β 指标的平均值比较

序号	方法	数据集 #1			数据集 #2			数据集 #3			平均		
		AvgP	AvgR	AvgF	AvgP	AvgR	AvgF	AvgP	AvgR	AvgF	AvgP	AvgR	AvgF
1	AC ^[108]	.6076	.4826	.4269	.6172	.4029	.3618	.0089	.4163	.0112	.4112	.4339	.2666
2	MSS ^[112]	.5331	.4423	.3668	.5076	.3893	.2779	.0025	.2706	.0033	.3478	.3674	.2160
3	FT ^[32]	.4995	.4570	.3450	.2804	.3538	.1059	.0123	.5431	.0147	.2641	.4513	.1552
4	COV ^[110]	.2677	.7743	.2493	.1121	.9295	.1220	.0418	.4717	.0399	.1406	.7252	.1371
5	HFT ^[46]	.2260	.8309	.2138	.1061	.9460	.1161	.0100	.5027	.0128	.1141	.7599	.1142
6	GR ^[167]	.2982	.5560	.2790	.0023	.8099	.0030	.0128	.4610	.0133	.1045	.6090	.0984
7	SUN ^[117]	.1254	.8713	.1428	.1244	.8562	.1198	.0040	.5330	.0052	.0846	.7535	.0892
8	RBD ^[58]	.2463	.4237	.2411	.0022	.6694	.0029	.0002	.0820	.0003	.0829	.3917	.0814
9	SR ^[37]	.1152	.5922	.1171	.0251	.7907	.0308	.0026	.3667	.0033	.0476	.5832	.0504
10	RPC	.3128	.5029	.2972	.1258	.7118	.1314	.0091	.2631	.0113	.1492	.4926	.1466
11	CNS	.0208	.2759	.0250	.0004	.1673	.0006	.0049	.3294	.0064	.0087	.2575	.0107
12	HCN	.0319	.6313	.0332	.0004	.7749	.0005	.0003	.3152	.0004	.0109	.5738	.0113
13	RS _A	.8201	.3079	.4080	.8220	.3888	.4795	.4735	.4835	.2391	.7052	.3934	.3755
平均		.3158	.5499	.2419	.2097	.6300	.1348	.0449	.3876	.0278	.1901	.5225	.1348

5.5.3.3 与融合方法的性能比较

最后是关于融合方法的性能比较。为了评估 RSS 方法中融合机制的有效性，这里将包括 RS_T 算法在内的 2 个稳定性方法、以及包括 RS_A 算法在内的 13 个显著性方法两两组合，并采用与 RSS 方法一样的方式加以融合。表 5.10 给出了除 RSS 方法外，新得到的 25 个融合方法，为方便起见，对这些方法直接以 Hadamard 积的方式命名。同时，表中也给出了这些方法在各数据集上的精确率 P 、召回率 R 和 $F1$ 指标、以及指标均值。需要说明的是，由于部分方法具有相同的指标结果，出于清晰展示的目的，这里仅对关键指标结果的并列前三名分别以红色、绿色和蓝色标记，而对非关键的相同指标结果不再以相应颜色标记。

对融合方法的性能分析如下：

- (1) 部分融合方法在 P 、 R 和 $F1$ 三项指标上的结果相同，这与所采用的在稳定性区域基础上利用显著性剔除虚警目标的融合机制有关。
- (2) 包括 RSS 方法 (即 $RS_T \circ RS_A$) 在内，与稳定性 RS_T 算法融合的方法在数据集 #1 和 #2 上获得了更好的检测结果。可以看到，RSS 方法的三项指标结果都排名第一，并列排名第一的也都是与 RS_T 融合的方法，这是因为 RS_T 算法在这两个数据集上具有最高的召回率。
- (3) 在数据集 #3 上，RSS 方法获得了并列第一的召回率 R 。得益于 MSER 方法在该数据集上具有更高的精确率 P ，与 MSER 融合的相关方法在 P 上并列排名第一，高于 RSS 方法 8.12%，而较低的 P 也导致了 RSS 方法获得了较低的 $F1$ 值，低于与 MSER 融合的相关方法 5.42%。
- (4) 三个数据集平均而言，在召回率 R 上 RSS 方法排名第一，但与 MSER 方法融合的相关方法获得了最高的精确率 P 和 $F1$ 指标。在指标 P 上，RSS 方法低于这些方法 1.34%；在指标 $F1$ 上，RSS 方法排名第三，低于这些方法 0.43%。
- (5) 虽然受 RS_T 算法在数据集 #3 上较低的精确率拖累，RSS 方法在整体上的 $F1$ 指标仅排名第三，但这里需要更多关注的还是融合机制本身。综合表 5.7、表 5.9 和表 5.10 可以看到，无论是稳定性方法还是显著性方法，当与另一种方法相融合时，检测性能都能得到或多或少的提升。以表 5.7 中的 MSER 方法为例，当与 RS_A 算法融合后 (表 5.10 中序号 #13)，三个数据集的 P 均值上升了 1.69%， $F1$ 均值上升了 1.14%。同样，表 5.10 中序号 #26 的 RSS 方法，相比融合前的 RS_T 算法 (见表 5.7)，在三个数据集上的 P 和 $F1$ 指标均值，分别上升了 19.91% 和 14.51%；同时相较融合前的 RS_A 算法 (见表 5.9)，三项指标均值分别上升了 26.52%、60.41% 和 60.4%，这就从实验层面验证了融合机制的有效性。本章通过将稳定性方法和显著性方法集成到一个统一的框架之中，弥补了仅基于其中一种方法检测小目标的不足。同时，本章提出的基于 Hadamard 积的融合方式，只是一种最简单直接的融合方式，这也为采用其它融合技术解决小目标检测问题留下了更多的扩展可能。

表 5.10 融合方法的性能比较

序号	方法	数据集 #1			数据集 #2			数据集 #3			平均		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
1	MSER \circ AC	.9801	.9801	.9801	.9712	.9712	.9712	1.	1.	1.	.9838	.9838	.9838
2	MSER \circ COV	.9801	.9801	.9801	.9712	.9712	.9712	1.	1.	1.	.9838	.9838	.9838
3	MSER \circ FT	.9801	.9801	.9801	.9712	.9712	.9712	1.	1.	1.	.9838	.9838	.9838
4	MSER \circ GR	.9801	.9801	.9801	.9519	.9519	.9519	1.	1.	1.	.9773	.9773	.9773
5	MSER \circ HFT	.9801	.9801	.9801	.9712	.9712	.9712	.9875	.9875	.9875	.9796	.9796	.9796
6	MSER \circ MSS	.9801	.9801	.9801	.9712	.9712	.9712	1.	1.	1.	.9838	.9838	.9838
7	MSER \circ RBD	.9478	.9478	.9478	.9567	.9567	.9567	1.	1.	1.	.9682	.9682	.9682
8	MSER \circ SR	.9801	.9801	.9801	.9663	.9663	.9663	.9875	.9875	.9875	.9780	.9780	.9780
9	MSER \circ SUN	.9783	.9789	.9785	.9712	.9712	.9712	.9875	.9875	.9875	.9790	.9792	.9790
10	MSER \circ RPC	.9801	.9801	.9801	.9567	.9615	.9583	1.	1.	1.	.9790	.9806	.9795
11	MSER \circ CNS	.9503	.9503	.9503	.9591	.9712	.9631	1.	1.	1.	.9698	.9738	.9712
12	MSER \circ HCN	.9596	.9764	.9652	.9543	.9712	.9599	.9938	1.	.9958	.9692	.9825	.9737
13	MSER \circ RSA	.9801	.9801	.9801	.9712	.9712	.9712	1.	1.	1.	.9838	.9838	.9838
14	RS _T \circ AC	.9925	.9925	.9925	1.	1.	1.	.7500	.9125	.8042	.9142	.9683	.9322
15	RS _T \circ COV	.9919	.9925	.9921	1.	1.	1.	.6312	.6875	.6500	.8744	.8933	.8807
16	RS _T \circ FT	.9925	.9925	.9925	.9856	.9904	.9872	.8354	1.	.8896	.9378	.9943	.9564
17	RS _T \circ GR	.9801	.9814	.9805	.9904	.9952	.9920	.5229	.5375	.5271	.8311	.8380	.8332
18	RS _T \circ HFT	.9919	.9925	.9921	1.	1.	1.	.5125	.5875	.5375	.8348	.8600	.8432
19	RS _T \circ MSS	.9925	.9925	.9925	1.	1.	1.	.7063	.8125	.7417	.8996	.9350	.9114
20	RS _T \circ RBD	.9733	.9776	.9747	1.	1.	1.	.4313	.4500	.4375	.8015	.8092	.8041
21	RS _T \circ SR	.9882	.9925	.9896	.9880	.9904	.9888	.4562	.5125	.4750	.8108	.8318	.8178
22	RS _T \circ SUN	.9874	.9925	.9890	1.	1.	1.	.4333	.5500	.4687	.8069	.8475	.8193
23	RS _T \circ RPC	.9851	.9851	.9851	.9952	.9952	.9952	.5000	.5375	.5125	.8268	.8393	.8309
24	RS _T \circ CNS	.9573	.9839	.9656	.9164	1.	.9418	.5625	.6750	.5979	.8121	.8863	.8351
25	RS _T \circ HCN	.9282	.9901	.9455	.9116	1.	.9386	.5010	.9750	.6321	.7803	.9884	.8387
26	RSS	.9925	.9925	.9925	1.	1.	1.	.9188	1.	.9458	.9704	.9975	.9795
平均		.9773	.9820	.9788	.9742	.9826	.9768	.7968	.8538	.8145	.9161	.9395	.9234

5.5.4 实验结果讨论

虽然 RSS 方法弥补了仅基于稳定性或显著性方法检测小目标的不足，实验结果也验证了融合机制的有效性，但还是存在一定的缺陷，图 5.14 展示了典型的失败案例。

首先是图 5.14 中第一行所示的小目标恰好是边界对象的情况。RSS 方法在数据集 #1 上之所有没有获得 100% 的指标结果，是由于稳定性 RS_T 算法在第 #752 帧上产生了漏检，而这也是 RS_T 算法在三个数据集上的唯一一帧漏检。从人工标注图 5.14(b) 可以看到，该小目标区域正好与图像的左边界连接，由于 RS_T 算法基于环绕性假设，会在序列化分割后直接剔除掉非闭合的边界对象，从而导致了该目标的丢失。但是，由于 RSS 方法并非用于解决时序状态下的小目标检测问题，因而单帧图像上的漏检并不会影响到对之后帧的检测，这也是单帧检测方法的优势所在^[162]。

另一个是如图 5.14 中第二行所示的检测虚警问题。从稳定图 5.14(c) 可以看到， RS_T 算法共检测到 4 个稳定性区域，虽然在 RS_A 算法得到的显著图 5.14(d) 中，小目标处于明显的显著地位，但由于这些区域的平均显著值较小，导致基于显著值均值的筛选方式尽管已经剔除了其中两个非显著的稳定性区域，但最终的检测结果中仍然留有 1 个虚警目标（见图 5.14(e) 红色箭头处）。事实上，如果将筛选规则改为仅保留显著值最大的那个稳定性区域，那么 RSS 方法在数据集 #3 上的精确率 P 同样可以达到 100%，而并不会产生虚警问题。但显然，这一规则的改动仅适用于那些仅包含单个小目标的输入图像，如果图像中有多个小目标同时存在的话，这种规则的改动将不可避免地带来漏检的问题。虽然本章使用的三个数据集，在单帧图像内都只至多包含一个小目标，但考虑到 RSS 方法对多小目标检测问题的兼容性，这里仍然保留之前的筛选规则，即以所有稳定性区域的显著值均值作为筛选阈值。要从根本上解决检测虚警问题，可以从两个方面着手考虑：一是引入更多的视觉线索，使稳定性算法产生更少的候选小目标区域；二是改进显著性与稳定性的融合方式，或是找到更具适应性的显著性筛选规则，使稳定性区域中的虚警目标能被有效剔除。

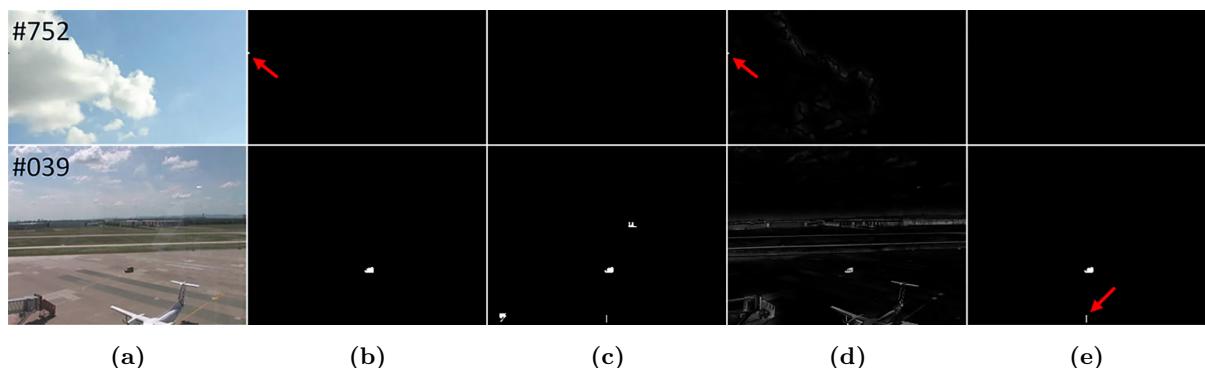


图 5.14 检测失败示例。(a) 原始图像; (b) 人工标注结果; (c) RS_T 算法得到的稳定图 M_T ; (d) RS_A 算法得到的显著图 M_A ; (e) RSS 方法检测结果

5.6 本章小结

本章针对单帧彩色图像小目标检测问题，提出了融合区域稳定性和区域显著性的 RSS 方法。考虑到小目标自身所包含的五个内蕴特征，首先将小目标检测看作是稳定性区域分割问题，在序列化分割得到二值区域后，区域稳定性算法采用五种区域描述子和四个稳定性度量准则对这些区域进行表象聚簇，在获得候选小目标的同时得到目标区域精确的特征描述；其次，将小目标检测看作是局部显著性检测问题，区域显著性算法采用局部的 LAB 颜色差异计算像素级的显著性，可在突出小的显著区域的同时，抑制均匀的背景模式；最后，融合算法对稳定性算法得到的稳定图以及显著性算法得到的显著图进行融合，利用候选小目标的平均显著值剔除虚警目标，得到小目标检测结果。实验结果表明，稳定性算法和显著性算法可以有效互补各自的不足，以及融合机制对提升检测性能的有效性。此外，标注并发布了一个用于彩色图像小目标检测的基准数据库，并提供了相应的人工标注结果和性能评估代码。

本章提出的 RSS 方法除可用于单帧彩色图像小目标检测外，也可用于解决单帧红外图像小目标检测问题，但这需要经过少量的代码修改。一是对于稳定性算法，由于 RS_T 仅检测亮背景下的暗区域，在序列化分割之前对灰度图 G 进行了反相操作。如果将 RS_T 算法用于红外小目标检测时，由于检测的是暗背景下的亮区域，故无需再执行反相操作。同样，如果使用本章发布的改进后的 MSER 代码检测红外小目标，应与第 5.5.2.1 节中的设置相反，即设置参数 $BrightOnDark = 1$ 且 $DarkOnBright = 0$ 。二是对于显著性算法，由于红外图像本身是一幅强度图，可以直接被当作显著图使用，也可以同样利用原图与高斯平滑的结果计算出显著图。而对于稳定性算法与显著性算法的融合，可同样使用 RSS 方法中所采用的融合机制。

对于后续的研究工作，计划从以下几个方面开展：

- (1) 针对红外多小目标跟踪应用场景，拟将本章工作直接扩展到红外图像，或将本章的稳定性算法与现有的红外小目标检测技术结合，用于解决作为轨迹关联前序步骤的单帧红外图像多小目标检测问题。
- (2) 进一步提高检测方法特别是稳定性算法的执行速度，以满足实时的应用需求，并在更为杂乱的场景中验证 RSS 方法的鲁棒性。
- (3) 虽然小目标自身可供抽取的特征描述较少，但还是可利用更为复杂的特征线索将小目标与背景分离。下一步计划引入更多自顶向下的小目标先验信息，如形状^[170]、熵等，以有效解决 RSS 方法中存在的虚警问题。

6 结束语

视觉显著性计算在过去的二十年里得到了蓬勃的发展，作为视觉显著性计算的一个研究分支，显著物体检测旨在从场景中检测出显著的物体并得到这些物体的精确分割，目前已被广泛地应用于诸多计算机视觉任务中。本文针对静态图像场景，基于自底向上、底层数据驱动的图像处理技术，围绕显著物体检测的计算方法及其应用开展了相关研究，主要研究成果包括一种基于区域主色的显著物体检测方法、一种基于颜色名空间的显著物体检测方法、一种基于分层结构与颜色名的协同显著物体检测方法，以及显著物体检测技术在单帧彩色图像小目标检测中的应用，即一种融合区域稳定性与显著性的小目标检测方法。

6.1 论文工作总结

本文首先介绍了显著性检测课题的研究背景和主要任务，并结合本文研究内容，对显著物体检测和协同显著物体检测两个研究方向进行了介绍，同时回顾了相关研究工作，介绍了评估数据集的建立过程并给出了性能指标定义与评估分析方法。本文的研究工作和主要成果如下：

(1) 采用全局颜色差异以及区域空间关系，提出了基于区域主色的 RPC 方法。RPC 方法首先对原始图像采用最小方差量化将颜色数减少到 32 色，并度量颜色差异计算全局显著性，实验结果表明最小方差量化可在更少的颜色数下获得更好的检测性能，同时降低了算法复杂度；其次，考虑到区域更容易引起视觉注意且区域内部高度的颜色一致性，在对量化图像进行超像素分割后，对每个局部区域仅保留区域主色并利用区域主色间的差异计算局部显著性，实验结果表明区域主色可有效降低局部显著性计算时的颜色比较或查表次数；最后，计算区域间距离以及区域与图像中心间距离，并基于这两种空间关系对局部显著性加权，实验结果表明中心加权模式对提升 RPC 方法检测性能的有效性。

(2) 由于 RPC 方法利用全局颜色稀缺性检测显著物体，因此当显著物体的颜色属于图像中的高频数量颜色时，会导致 RPC 方法的失效。考虑到颜色差异度量对显著性检测的有效性以及物体自身具有完整闭合的外部轮廓，采用环绕性线索和全局颜色名线索，提出了基于颜色名空间的 CNS 方法。CNS 方法首先将原始图像转换到含有 11 个颜色名通道的颜色名空间，基于环绕性线索对每个颜色名通道进行图形-背景分离得到注意图，并对注意图线性融合和后处理获得显著图；其次，将原始图像转换为颜色名图像，从中计算全局颜色名统计特征和对比特征，并利用这两种特征对注意图加权获得加权显著图；最后，对显著图和加权显著图采用高亮截断的融合操作和后处理操作，得到显著物体检测结果。实验结果表明，CNS 方法可有效抑制非显著的图像背景，并在

突出显著物体的同时使其具有均匀且平滑的显著值。

(3) CNS 方法存在的问题在于当显著物体是边界对象时将导致检测失效，同时单一的结构元素大小不能很好地适应不同尺寸的显著物体。因此，综合利用颜色名线索、环绕性线索以及边界连通性线索，提出了用于检测一对图像协同显著物体的 HCN 方法。HCN 方法首先对原始图像构造三个图像层，采用显著一致性单层融合算法计算单层融合图，并基于全局颜色名稀缺性改进得到单层显著图；其次，对三个单层显著图采用多层融合算法进行融合，并基于颜色名稀缺性和显著一致性改进得到单图显著图；最后，对各单图显著图进行自适应阈值分割得到单图显著区域，并基于单图显著区域的颜色名相似性去除非协同显著区域，获得协同显著图。实验结果表明，单层融合算法与多层融合算法可有效提升检测性能，且颜色名线索也同样适用于协同显著物体检测。HCN 方法除可用于检测一对图像间的协同显著物体外，也是一种可用于单张图像的显著物体检测方法。

(4) 作为一项辅助技术，显著物体检测真正的价值在于面向的具体应用。因此，采用区域稳定性度量和局部颜色差异性度量，提出了用于单帧彩色图像小目标检测的 RSS 方法。RSS 方法首先将小目标检测看作是稳定性区域提取问题，区域稳定性算法通过序列化阈值分割获得二值区域后，基于五种区域描述子和四个稳定性度量准则对二值区域进行表象聚簇，得到隶属于每个候选小目标的一组表象，然后对每组表象进行稳定性度量，在获得候选小目标的同时得到目标区域精确的特征描述；其次，RSS 方法将小目标检测看作是局部显著性检测问题，区域显著性算法基于局部颜色差异计算像素级的显著性，在突出小的显著区域的同时抑制均匀的背景模式；最后，融合算法对稳定图和显著图进行融合，并基于候选小目标的平均显著值剔除虚警目标得到检测结果。实验结果表明，区域稳定性算法和区域显著性算法可有效弥补各自的不足，以及融合机制对提升检测性能的有效性。此外，标注并发布了一个含有三种典型应用场景的小目标检测基准数据库，并提供了相应的性能评估代码。

6.2 未来工作展望

为了验证提出的四种方法的有效性，本文分别在三个显著物体检测数据集、一个协同显著物体检测数据集以及三个小目标检测数据集上进行了实验评估。虽然整体上取得了较好的检测结果，与其它方法相比也展现出一定的性能优势，但从理论和应用层面来看，提出的这些方法仍然存在一定的不足，有待进一步的探索与完善。针对本文方法存在的问题，未来考虑从以下几个方面开展研究：

(1) 无论是 RGB/LAB 颜色空间还是颜色名空间，对颜色特征的使用在本文中占据了相当大的比重。作为容易引起视觉注意的一种底层特征，虽然颜色已被众多工作验证了其在显著性检测中的有效性，但有时并不鲁棒。从之前章节失败示例中可以看到，当显著物体与图像背景具有较高的颜色相似性时，仅利用颜色特征会导致检测方法的失效。而人类在观察此类图像时，通常会从物体轮廓、形状等多个角度同时进行，以有

效地将物体区域与图像背景区分。因此，在颜色特征的基础上采用更多其它视觉特征并有效结合，如纹理、大小、位置、空间频率等，是未来值得研究的一个方向。

(2) 本文第4章开展了协同显著性相关研究，并将提出的 HCN 方法用于一对图像的协同显著物体检测问题。未来除了需要进一步验证并将其扩展到对一组图像的检测任务外，从多张图像中发现显著性线索也有助于解决单张图像的显著物体检测问题。目前，有部分学者认为，对于单张图像的显著物体检测不应局限于在单张图像内发掘特征，而应从其它相似图像或视频序列中找到显著物体的共同模式。通过外部线索的挖掘与引入，不仅可为解决单张图像的显著物体检测问题提供帮助，也可被协同显著性检测、视频显著性检测等诸多课题所借鉴。随着传感器设备的日益增多，未来在多模态数据中进行显著物体检测是另一个值得研究的方向。

(3) 尽管视觉显著性的内在生物机理目前尚不明确，研究人员从多个学科角度开展研究的最终目的是为了揭示人类为何具有出色的对场景的整体把握和快速理解能力，在这一点上，和源于神经科学的深度学习异曲同工。与依赖于手工提取特征并加以融合的传统检测手段不同，基于深度学习的显著物体检测方法不仅能够获取到局部上下文信息，还具有对全局上下文建模的优势。随着深度学习技术的飞速发展，其优异的性能表现决定了基于该技术的显著物体检测方法将逐渐成为研究的主流。除了直接设计端到端的显著物体检测框架外，利用深度网络学习到的深度特征也可以帮助人类理解选择性注意的内在机理，下一步将重点在此方向上开展相关的研究。

(4) 无论如何，作为一项通常用于预处理或后处理的辅助手段，对显著物体检测相关技术的研究归根到底要落地到具体的应用中。本文第5章在单帧彩色图像小目标检测中的应用，只是显著物体检测技术众多应用中的冰山一角。随着移动互联网、智能手机、社交媒体的迅速普及，以及计算机视觉、机器人技术的不断发展，显著物体检测技术未来还有巨大的应用发展空间。具体的，下一步将跳出静态的图像场景，开展动态视频场景中显著物体检测与分割的相关研究，同时计划面向跨场景的视频监控领域，研究显著物体检测技术辅助解决多摄像机下鲁棒的行人检测与跟踪问题。

致 谢

又到了二月兰盛放的季节，校园里弥漫着沁人心脾的花香。三号路、一号门，从本科到博士，反反复复不知走过了多少回，而这段旅程，也终将画上一个完整的句号。想象中，这或许应是博士论文中最好写的一节，但提起笔来，却发现并没那么简单。人生，无非山海，出发时激情如海，到达时淡然如山。这一路走来，太多的人需要感谢，太多的事值得留念，太多的话却不知从何说起。夜深人静，独自在实验室中掩卷而思，回顾当初义无反顾的选择、走过的这些年和洒下的那些汗水，过往艰难痛苦与困惑彷徨的记忆都变得模糊不清，脑海中只剩下感恩与感怀。在博士生涯即将结束之际，谨向那些曾经关心支持和无私帮助过我的老师、同学、同事和朋友，还有我最亲爱的家人，致以最由衷的感谢和最诚挚的祝福！

首先感谢我的博士生导师任明武教授。与恩师相识于我攻读硕士学位期间，因《数字图像分析》这门课与他结缘，他虽非我的硕士生导师，却在我硕士期间自始至终给予了悉心的指导。此生的第一篇学术论文以及硕士论文，正是在他的帮助下才得以完成。恩师渊博的知识、严谨的治学态度、诲人不倦的崇高师德、深厚的编程造诣、敏锐的行业洞察力、平易近人的工作作风、以及积极乐观的生活心态，都时时刻刻感染和影响着我，使我终身受益。恩师虽然科研与教学工作繁忙，但每每遇到技术上的难题求助于他，他总能第一时间倾囊相授。他不仅为我创造了良好的学习和科研氛围，也指导我应该如何发现问题、独立思考和解决问题；他不仅传授了我大量的专业知识和最新的行业动向，也指引我应该如何扩宽个人学术视野和拥有为人处世的博大胸怀与境界。虽常常感慨此生必然不能达到他的高度，却不断激励和鞭策着我朝着他的高度努力奋斗，能拜入恩师的门下攻读博士学位，实乃人生一大幸事！

衷心感谢三位曾经的师兄、如今奋战在各自工作岗位上的高校老师，他们是南京理工大学计算机科学与工程学院的王欢副教授、东南大学自动化学院的杨万扣副研究员、以及南京航空航天大学计算机科学与技术学院的孙涵副教授。感谢他们对我课题研究和论文撰写过程中所给与的指导与帮助，也要感谢他们对我日常生活和工作的关心与照顾，他们的无私、平易和对师弟的提携，是我终身学习的榜样。

深深感谢南京理工大学的诸位老师，他们是计算机科学与工程学院的张二华副教授、赵春霞教授、金忠教授、刘传才教授、肖亮教授、陆建峰教授、於东军教授、唐振民教授、孔慧教授、刘华军副教授、李蔚清副教授、宫辰教授、夏青元老师、赵建老师、薛振华老师、朱凯老师和刘行景老师，以及电子工程与光电技术学院的柏连发教授，感谢他们在我攻读博士学位期间提供的大力支持和无私帮助。

特别感谢南京大学计算机科学与技术系的高阳教授、南京航空航天大学计算机科学与技术学院的刘宁钟教授、南京邮电大学自动化学院的荆晓远教授、国防科技大学智

能科学学院的胡德文教授、以及各位匿名审稿专家，他们牺牲个人时间对我攻读博士学位期间撰写的论文和本文进行了细致的评阅，并提出了大量的改进意见和修改建议，使这些论文的质量得到了极大的提升，与他们的讨论使我获益匪浅。

感谢南开大学计算机与控制工程学院的程明明教授对文中部分数据集提供了诸多使用上的帮助和建议，以及与北京交通大学计算机与信息技术学院的李尊博士和南京航空航天大学电子信息工程学院的戴一冕博士进行的讨论。还要感谢江苏理工学院的郭芮老师、James S. Krugh 老师、章淮平副教授，合肥工业大学学报(自然科学版)的张镅副主编，以及 Luby Hough、Jo Probut、Joseph S. Senate、Razel Gerona-Avanzado、Rebecca Hytowitz 等编辑对部分手稿的排版与校对。

感谢南京理工大学计算机科学与工程学院的众多兄弟姐妹们，他们包括但不限于：冯博远、张海洋、朱伟、陆华齐、李想、王拴、王安东、陈龙涛、徐峰磊、胡炳文、陈力能、黄波、王超、杨阳、杨荣根、朱俊、王建、朱雪娇、李斐斐、祁磊、梅鸿翔、任薇、钱青、邓安良、金务文、李强、袁峻、张鸿志、赵起超、曹明星、杜泽炎、沈洁、杜芳、朱旻华、崔晨、汪锡平、洪诗颖、乔笑、姚铖、谢宇、陈小辉、陆伟、韩叶飞、郭箭、王群、张笃振、张键、王海琳、吴小伟、杨薇薇、舒振球、顾迎节、顾兴健。感谢读博路上有他们的陪伴，常常怀念与他们一起研讨课题、感悟人生、畅谈情怀的幸福时光，与他们的朝夕相处，是人生路上一段多么美好的回忆。

感谢常州机电职业技术学院的沈琳校长、刘贤锋教授、陶国正教授、顾卫杰副教授和陈培中老师，常州市委党校的周庆元副教授，常州大学的王洪元教授、朱正伟教授，江苏理工学院的龙一平教授、徐鸿翔副教授、张建斌教授、宋国龙高级实验师、蒋益锋高级工程师、郑恩兴高级实验师、江波副教授、刘冉冉副教授、邹丽阳副研究员、高修坦、刘栋梁、钱成、刘振平、臧爱琴、钱秀芳、卢丹、陆剑、王云白、顾添笑、林进、唐子俊等老师，以及光大环保技术装备(常州)有限公司的梁德亮部长，感谢他们多年以来对我个人的成长与生活所给与的关心与帮助，在我迷茫和彷徨时，让我没有彻底质疑自己的未来，而是选择凝眸远方、继续前行。

感谢部分企业横向课题的参研人员，包括大庆油田有限责任公司的姜群发先生、杨宝春先生、丁原先生和牟玉琦先生，以及无锡市速波精密机械有限公司的张栋总经理、穆志遥先生和邓志先生。正是因为有了他们的大力支持和全力协助，才使得这些课题能够顺利地实施与推进。

感谢南京理工大学机械工程学院程杰博士提供的学位论文 L^AT_EX 模板，南京大学计算机科学与技术系胡海星博士提供的 GB/T 7714-2005 参考文献 B_IB^AT_EX 样式，以及 Carles F. Julia 提供的用于处理 B_IB 文件的 Python 脚本。还要感谢来自 L^AT_EX 社区里众多素未谋面的热心网友，对论文中诸多格式排版问题提供了耐心细致的解答与建议。正是因为有了他们的帮助，才使我能够轻松完成博士期间的论文以及本文的写作。

本文相关的研究工作，受到国家自然科学基金面上项目(项目号 60875010)，国家自然科学基金重点项目(项目号 61231014)，国家自然科学基金青年科学基金项目(项

目号 61403202、61703209), 中国博士后科学基金面上资助项目(项目号 2014M561654), 以及部分 863 计划军口项目的资助, 特此感谢。

尤其感谢父母和岳父母对我的包容, 常常遗憾我努力前行的程度赶不上他们逐渐老去的速度。我不在身边的日子里, 他们牺牲了许多其他父母应享有的幸福时光, 年迈之时反而给予了我无私的照顾和帮助, 用他们的全力支持解除我的后顾之忧, 使我可以心无旁骛、安心学业。特别感谢我的妻子岳燕芬女士, 离家求学多年, 她义无反顾、默默撑起了本应属于我的家庭重担, 独力承担了对孩子的教育和对家庭、对老人的照顾, 每每想到她的艰辛与付出, 总让我倍感愧疚。还要感谢我那两个可爱的幼子, 他们稚气憨态的身影和天真烂漫的笑颜, 是我在求学路上不断前进的动力。

最后, 请允许我再一次向这些年来所有关心、支持和帮助过我的人致以最诚挚的感谢和最真诚的祝福!

楼 竞

2018 年 3 月于鼎新楼

参考文献

- [1] CORBETTA M. Frontoparietal Cortical Networks for Directing Attention and the Eye to Visual Locations: Identical, Independent, or Overlapping Neural Systems? [C] // Proceedings of the National Academy of Sciences : Vol 95. 1998 : 831–838.
- [2] ZHANG L, LIN W. Selective Visual Attention: Computational Models and Applications [M]. 1st. Solaris South Tower, Singapore : John Wiley & Sons Singapore Pte. Ltd., 2013.
- [3] TEUBER H-L. Physiological Psychology [J]. Annual Review of Psychology, 1955, 6(1) : 267–296.
- [4] TREISMAN A M, GELADE G. A Feature-Integration Theory of Attention [J]. Cognitive Psychology, 1980, 12(1) : 97–136.
- [5] DESIMONE R, DUNCAN J. Neural Mechanisms of Selective Visual Attention [J]. Annual Review of Neuroscience, 1995, 18(1) : 193–222.
- [6] KOCH C, ULLMAN S. Shifts in Selective Visual Attention: Towards the Underlying Neural Circuitry [J]. Human Neurobiology, 1985, 4 : 219–227.
- [7] 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.
- [8] BORJI A, CHENG M-M, HOU Q, et al. Salient Object Detection: A Survey [J]. arXiv preprint, 2017 : 1–20.
- [9] LIU T, SUN J, ZHENG N-N, et al. Learning to Detect A Salient Object [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2007 : 1–8.
- [10] SHEN H, LI S, ZHU C, et al. Moving Object Detection in Aerial Video Based on Spatiotemporal Saliency [J]. Chinese Journal of Aeronautics, 2013, 26(5) : 1211–1217.
- [11] WANG Z, DU L, ZHANG P, et al. Visual Attention-Based Target Detection and Discrimination for High-Resolution SAR Images in Complex Scenes [J]. IEEE Transactions on Geoscience and Remote Sensing, 2018, 56(4) : 1855–1872.
- [12] RUTISHAUSER U, WALther D, KOCH C, et al. Is Bottom-Up Attention Useful for Object Recognition? [C] // Proceedings of the IEEE Conference on Computer

- Vision and Pattern Recognition : Vol 2. 2004: 37–44.
- [13] REN Z, GAO S, CHIA L-T, et al. Region-Based Saliency Detection and Its Application in Object Recognition [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2014, 24(5) : 769–779.
- [14] FRINTROP S, GARCÍA G M, CREMERS A B. A Cognitive Approach for Object Discovery [C] // Proceedings of the International Conference on Pattern Recognition. 2014 : 2329–2334.
- [15] ZHOU C, LIU C. An Efficient Segmentation Method Using Saliency Object Detection [J]. Multimedia Tools and Applications, 2015, 74(15) : 5623–5634.
- [16] QIN C, ZHANG G, ZHOU Y, et al. Integration of the Saliency-based Seed Extraction and Random Walks for Image Segmentation [J]. Neurocomputing, 2014, 129 : 378–391.
- [17] MARGOLIN R, ZELNIK-MANOR L, TAL A. Saliency for Image Manipulation [J]. Visual Computer, 2013, 29(5) : 381–392.
- [18] CHENG M-M, MITRA N J, HUANG X, et al. SalientShape: Group Saliency in Image Collections [J]. The Visual Computer, 2014, 30(4) : 443–453.
- [19] ZHANG W, BORJI A, WANG Z, et al. The Application of Visual Saliency Models in Objective Image Quality Assessment: A Statistical Evaluation [J]. IEEE Transactions on Neural Networks and Learning Systems, 2016, 27(6) : 1266–1278.
- [20] BORJI A, ITTI L. Defending Yarbus: Eye Movements Reveal Observers' Task [J]. Journal of Vision, 2014, 14(3) : 29: 1–22.
- [21] BORJI A, ITTI L. Scene Classification with a Sparse Set of Salient Regions [C] // Proceedings of the IEEE International Conference on Robotics and Automation. 2011 : 1902–1908.
- [22] LI S, XU M, REN Y, et al. Closed-Form Optimization on Saliency-Guided Image Compression for HEVC-MSP [J]. IEEE Transactions on Multimedia, 2018, 20(1) : 155–170.
- [23] 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, 2010, 19(1) : 185–198.
- [24] LEE Y J, GHOSH J, GRAUMAN K. Discovering Important People and Objects for Egocentric Video Summarization [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2012 : 1346–1353.
- [25] MADEMLIS I, TEFAS A, PITAS I. Summarization of Human Activity Videos

- Using a Salient Dictionary [C] // Proceedings of the IEEE International Conference on Image Processing. 2017: 625–629.
- [26] BORJI A, FRINTROP S, SIHITE D N, et al. Adaptive Object Tracking by Learning Background Context [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2012: 23–30.
- [27] MA C, MIAO Z, ZHANG X-P, et al. A Saliency Prior Context Model for Real-Time Object Tracking [J]. IEEE Transactions on Multimedia, 2017, 19(11): 2415–2424.
- [28] 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.
- [29] CHENG M-M, ZHANG G-X, MITRA N J, et al. Global Contrast based Salient Region Detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2011: 409–416.
- [30] LI Y, HOU X, KOCH C, et al. The Secrets of Salient Object Segmentation [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 4321–4328.
- [31] BORJI A. What is a Salient Object? A Dataset and a Baseline Model for Salient Object Detection [J]. IEEE Transactions on Image Processing, 2015, 24(2): 742–756.
- [32] ACHANTA R, HEMAMI S, ESTRADA F, et al. Frequency-tuned Salient Region Detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2009: 1597–1604.
- [33] ZHANG D, FU H, HAN J, et al. A Review of Co-saliency Detection Technique: Fundamentals, Applications, and Challenges [J/OL]. arXiv preprint, 2017: 1–18. <https://arxiv.org/abs/1604.07090>.
- [34] LI H, NGAN K N. A Co-Saliency Model of Image Pairs [J]. IEEE Transactions on Image Processing, 2011, 20(12): 3365–3375.
- [35] ROTHER C, KOLMOGOROV V, MINKA T, et al. Cosegmentation of Image Pairs by Histogram Matching—Incorporating a Global Constraint into MRFs [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2006: 993–1000.
- [36] BATRA D, KOWDLE A, PARIKH D, et al. iCoseg: Interactive Co-segmentation with Intelligent Scribble Guidance [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2010: 3169–3176.
- [37] HOU X, ZHANG L. Saliency Detection: A Spectral Residual Approach [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recogni-

- tion. 2007: 1–8.
- [38] MILANESE R, GIL S, PUN T. Attentive Mechanisms for Dynamic and Static Scene Analysis [J]. Optical Engineering, 1995, 34(8): 2428–2434.
- [39] BALUJA S, POMERLEAU D A. Expectation-Based Selective Attention for Visual Monitoring and Control of a Robot Vehicle [J]. Robotics and Autonomous Systems, 1997, 22(3-4): 329–344.
- [40] SUGANO Y, MATSUSHITA Y, SATO Y. Appearance-Based Gaze Estimation Using Visual Saliency [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(2): 329–341.
- [41] SUGANO Y, BULLING A. Self-Calibrating Head-Mounted Eye Trackers Using Egocentric Visual Saliency [C] // Proceedings of the Annual ACM Symposium on User Interface Software and Technology. 2015: 363–372.
- [42] KELLNHOFER P, DIDYK P, MYSZKOWSKI K, et al. GazeStereo3D: Seamless Disparity Manipulations [J]. ACM Transactions on Graphics, 2016, 35(4): 68: 1–13.
- [43] MAHADEVAN V, VASCONCELOS N. Saliency-based Discriminant Tracking [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2009: 1007–1013.
- [44] MATHE S, SMINCHISESCU C. Dynamic Eye Movement Datasets and Learnt Saliency Models for Visual Action Recognition [C] // Proceedings of the European Conference on Computer Vision. 2012: 842–856.
- [45] LI J, LEVINE M D, AN X, et al. Saliency Detection Based on Frequency and Spatial Domain Analysis [C] // Proceedings of the British Machine Vision Conference. 2011: 86: 1–11.
- [46] LI J, LEVINE M D, AN X, et al. Visual Saliency Based on Scale-Space Analysis in the Frequency Domain [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(4): 996–1010.
- [47] PAPAGEORGIOU C, POGGIO T. A Trainable System for Object Detection [J]. International Journal of Computer Vision, 2000, 38(1): 15–33.
- [48] VIOLA P, JONES M J. Robust Real-Time Face Detection [J]. International Journal of Computer Vision, 2004, 57(2): 137–154.
- [49] ALEXE B, DESELAERS T, FERRARI V. What is an Object? [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2010: 73–80.

- [50] CHENG M-M, ZHANG Z, LIN W-Y, et al. BING: Binarized Normed Gradients for Objectness Estimation at 300fps [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 3286–3293.
- [51] ENDRES I, HOIEM D. Category-Independent Object Proposals with Diverse Ranking [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 36(2): 222–234.
- [52] HOSANG J, BENENSON R, DOLLÁR P, et al. What Makes for Effective Detection Proposals? [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(4): 814–830.
- [53] EVERINGHAM M, Van Gool L, WILLIAMS C, et al. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results [EB/OL]. 2007. <http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html>.
- [54] SIVA P, RUSSELL C, XIANG T, et al. Looking Beyond the Image: Unsupervised Learning for Object Saliency and Detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2013: 3238–3245.
- [55] MA Y-F, ZHANG H-J. Contrast-based Image Attention Analysis by Using Fuzzy Growing [C] // Proceedings of the ACM International Conference on Multimedia. 2003: 374–381.
- [56] PERAZZI F, KRÄHENBÜHL P, PRITCH Y, et al. Saliency Filters: Contrast Based Filtering for Salient Region Detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2012: 733–740.
- [57] JIANG H, WANG J, YUAN Z, et al. Salient Object Detection: A Discriminative Regional Feature Integration Approach [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2013: 2083–2090.
- [58] ZHU W, LIANG S, WEI Y, et al. Saliency Optimization from Robust Background Detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 2814–2821.
- [59] GONG C, TAO D, LIU W, et al. Saliency Propagation from Simple to Difficult [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 2531–2539.
- [60] SHI J, YAN Q, XU L, et al. Hierarchical Image Saliency Detection on Extended CSSD [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(4): 717–729.
- [61] YAN Q, XU L, SHI J, et al. Hierarchical Saliency Detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2013: 1155–

- 1162.
- [62] 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.
 - [63] HE S, LAU R W H, 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.
 - [64] ZHAO R, OUYANG W, LI H, et al. Saliency Detection by Multi-Context Deep Learning [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015 : 1265–1274.
 - [65] HOU Q, CHENG M-M, HU X, et al. Deeply Supervised Salient Object Detection with Short Connections [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017 : 5300–5309.
 - [66] XIE S, TU Z. Holistically-Nested Edge Detection [C] // Proceedings of the IEEE International Conference on Computer Vision. 2015 : 1395–1403.
 - [67] ZHANG J, DAI Y, PORIKLI F. Deep Salient Object Detection by Integrating Multi-level Cues [C] // Proceedings of the IEEE Winter Conference on Applications of Computer Vision. 2017 : 1–10.
 - [68] LUO Z, MISHRA A, ACHKAR A, et al. Non-Local Deep Features for Salient Object Detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017 : 6593–6601.
 - [69] ZHANG P, WANG D, LU H, et al. Amulet: Aggregating Multi-level Convolutional Features for Salient Object Detection [C] // Proceedings of the IEEE International Conference on Computer Vision. 2017 : 202–211.
 - [70] GOFERMAN S, ZELNIK-MANOR L, TAL A. Context-Aware Saliency Detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2010 : 2376–2383.
 - [71] MARGOLIN R, TAL A, ZELNIK-MANOR L. What Makes a Patch Distinct? [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2013 : 1139–1146.
 - [72] TATLER B W. The Central Fixation Bias in Scene Viewing: Selecting an Optimal Viewing Position Independently of Motor Biases and Image Feature Distributions [J]. Journal of Vision, 2007, 7(14) : 4: 1–17.
 - [73] JUDD T, EHINGER K, DURAND F, et al. Learning to Predict Where Humans Look [C] // Proceedings of the IEEE International Conference on Computer Vision. 2009 : 2106–2113.

- [74] LI X, LU H, ZHANG L, et al. Saliency Detection via Dense and Sparse Reconstruction [C] // Proceedings of the IEEE International Conference on Computer Vision. 2013: 2976–2983.
- [75] ZHANG J, SCLAROFF S. Saliency Detection: A Boolean Map Approach [C] // Proceedings of the IEEE International Conference on Computer Vision. 2013: 153–160.
- [76] FU H, CAO X, TU Z. Cluster-Based Co-Saliency Detection [J]. IEEE Transactions on Image Processing, 2013, 22(10): 3766–3778.
- [77] CHEN Y-L, HSU C-T. Implicit Rank-Sparsity Decomposition: Applications to Saliency/Co-Saliency Detection [C] // Proceedings of the International Conference on Pattern Recognition. 2014: 2305–2310.
- [78] CAO X, TAO Z, ZHANG B, et al. Self-Adaptively Weighted Co-Saliency Detection via Rank Constraint [J]. IEEE Transactions on Image Processing, 2014, 23(9): 4175–4186.
- [79] LIU Z, ZOU W, LI L, et al. Co-Saliency Detection Based on Hierarchical Segmentation [J]. IEEE Signal Processing Letters, 2014, 21(1): 88–92.
- [80] JACOBS D E, GOLDMAN D B, SHECHTMAN E. Cosaliency: Where People Look When Comparing Images [C] // Proceedings of the Annual ACM Symposium on User Interface Software and Technology. 2010: 219–228.
- [81] CHANG K-Y, LIU T-L, LAI S-H. From Co-saliency to Co-segmentation: An Efficient and Fully Unsupervised Energy Minimization Model [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2011: 2129–2136.
- [82] ZHANG D, MENG D, LI C, et al. A Self-paced Multiple-instance Learning Framework for Co-saliency Detection [C] // Proceedings of the IEEE International Conference on Computer Vision. 2015: 594–602.
- [83] ZHANG D, HAN J, LI C, et al. Co-saliency Detection via Looking Deep and Wide [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 2994–3002.
- [84] WANG Z, LI B. A Two-stage Approach to Saliency Detection in Images [C] // Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing. 2008: 965–968.
- [85] 周志华. 机器学习 [M]. 第 1 版. 北京: 清华大学出版社, 2016.
- [86] 马儒宁, 涂小坡, 丁军娣, et al. 视觉显著性凸显目标的评价 [J]. 自动化学报, 2012, 38(5): 870–876.

- [87] Van Rijsbergen C. Information Retrieval [M]. 2nd. London : Butterworths, 1979.
- [88] 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.
- [89] LIU T, YUAN Z, SUN J, et al. Learning to Detect a Salient Object [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33(2) : 353–367.
- [90] SPACKMAN K A. Signal Detection Theory: Valuable Tools for Evaluating Inductive Learning [C] // Proceedings of the International Workshop on Machine Learning. 1989 : 160–163.
- [91] HAREL J, KOCH C, PERONA P. Graph-Based Visual Saliency [C] // Proceedings of the Annual Conference on Neural Information Processing Systems. 2006 : 545–552.
- [92] ZHAI Y, SHAH M. Visual Attention Detection in Video Sequences Using Spatiotemporal Cues [C] // Proceedings of the ACM International Conference on Multimedia. 2006 : 815–824.
- [93] FELZENSZWALB P F, HUTTENLOCHER D P. Efficient Graph-Based Image Segmentation [J]. International Journal of Computer Vision, 2004, 59(2) : 167–181.
- [94] WAN X, KUO C-C J. Color Distribution Analysis and Quantization for Image Retrieval [C] // Proceedings of the SPIE Storage and Retrieval for Still Image and Video Databases IV : Vol 2670. 1996 : 8–16.
- [95] MAX J. Quantizing for Minimum Distortion [J]. IRE Transactions on Information Theory, 1960, 6(1) : 7–12.
- [96] LLOYD S P. Least Squares Quantization in PCM [J]. IEEE Transactions on Information Theory, 1982, 28(2) : 129–137.
- [97] GAREY M R, JOHNSON D S, WITSENHAUSEN H S. The Complexity of the Generalized Lloyd-Max Problem [J]. IEEE Transactions on Information Theory, 1982, 28(2) : 255–256.
- [98] HECKBERT P. Color Image Quantization for Frame Buffer Display [J]. Computer Graphics, 1982, 16(3) : 297–307.
- [99] WAN S J, PRUSINKIEWICZ P, WONG S K M. Variance-Based Color Image Quantization for Frame Buffer Display [J]. Color Research and Application, 1990, 15(1) : 52–58.
- [100] WU X. Color Quantization by Dynamic Programming and Principal Analysis [J].

- ACM Transactions on Graphics, 1992, 11(4) : 348–372.
- [101] XIANG Z. Color Image Quantization by Minimizing the Maximum Intercluster Distance [J]. ACM Transactions on Graphics, 1997, 16(3) : 260–276.
- [102] LOU J, REN M, WANG H. Regional Principal Color Based Saliency Detection [J]. PLoS ONE, 2014, 9(11) : e112475: 1–13.
- [103] EINHÄUSER W, KÖNIG P. Does Luminance-Contrast Contribute to a Saliency Map for Overt Visual Attention? [J]. European Journal of Neuroscience, 2003, 17(5) : 1089–1097.
- [104] KLEIN D A, FRINTROP S. Center-surround Divergence of Feature Statistics for Salient Object Detection [C] // Proceedings of the IEEE International Conference on Computer Vision. 2011 : 2214–2219.
- [105] TORRALBA A, OLIVA A, CASTELHANO M S, et al. Contextual Guidance of Eye Movements and Attention in Real-World Scenes: The Role of Global Features in Object Search [J]. Psychological Review, 2006, 113(4) : 766–786.
- [106] WANG P, WANG J, ZENG G, et al. Salient Object Detection for Searched Web Images via Global Saliency [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2012 : 3194–3201.
- [107] SHEN X, WU Y. A Unified Approach to Salient Object Detection via Low Rank Matrix Recovery [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2012 : 853–860.
- [108] ACHANTA R, ESTRADA F, WILS P, et al. Salient Region Detection and Segmentation [C] // Proceedings of the International Conference on Computer Vision Systems. 2008 : 66–75.
- [109] BRUCE N D, TSOTSOS J K. Saliency Based on Information Maximization [C] // Proceedings of the Annual Conference on Neural Information Processing Systems. 2005 : 155–162.
- [110] ERDEM E, ERDEM A. Visual Saliency Estimation by Nonlinearly Integrating Features Using Region Covariances [J]. Journal of Vision, 2013, 13(4) : 11: 1–20.
- [111] TAVAKOLI H R, RAHTU E, HEIKKILÄ J. Fast and Efficient Saliency Detection Using Sparse Sampling and Kernel Density Estimation [C] // Proceedings of the Scandinavian Conference on Image Analysis. 2011 : 666–675.
- [112] ACHANTA R, SÜSSTRUNK S. Saliency Detection Using Maximum Symmetric Surround [C] // Proceedings of the IEEE International Conference on Image Processing. 2010 : 2653–2656.

- [113] RAHTU E, KANNALA J, SALO M, et al. Segmenting Salient Objects from Images and Videos [C] // Proceedings of the European Conference on Computer Vision. 2010: 366–379.
- [114] SEO H J, MILANFAR P. Static and Space-Time Visual Saliency Detection by Self-Resemblance [J]. *Journal of Vision*, 2009, 9(12): 15: 1–27.
- [115] MURRAY N, VANRELL M, OTAZU X, et al. Saliency Estimation Using a Non-Parametric Low-Level Vision Model [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2011: 433–440.
- [116] HOU X, HAREL J, KOCH C. Image Signature: Highlighting Sparse Salient Regions [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, 34(1): 194–201.
- [117] ZHANG L, TONG M H, MARKS T K, et al. SUN: A Bayesian Framework for Saliency using Natural Statistics [J]. *Journal of Vision*, 2008, 8(7): 32: 1–20.
- [118] CHANG K-Y, LIU T-L, CHEN H-T, et al. Fusing Generic Objectness and Visual Saliency for Salient Object Detection [C] // Proceedings of the IEEE International Conference on Computer Vision. 2011: 914–921.
- [119] DUAN L, WU C, MIAO J, et al. Visual Saliency Detection by Spatially Weighted Dissimilarity [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2011: 473–480.
- [120] TORRALBA A, EFROS A A. Unbiased Look at Dataset Bias [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2011: 1521–1528.
- [121] CHENG M-M, WARRELL J, LIN W-Y, et al. Efficient Salient Region Detection with Soft Image Abstraction [C] // Proceedings of the IEEE International Conference on Computer Vision. 2013: 1529–1536.
- [122] 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*, 2015, 37(3): 569–582.
- [123] RUBIN E. Figure and Ground [G] // Readings in Perception. 1958: 194–203.
- [124] MAZZA V, TURATTO M, UMLITÀ C. Foreground-Background Segmentation and Attention: A Change Blindness Study [J]. *Psychological Research*, 2005, 69(3): 201–210.
- [125] VECERA S P, FLEVARIS A V, FILAPEK J C. Exogenous Spatial Attention Influences Figure-Ground Assignment [J]. *Psychological Science*, 2004, 15(1): 20–26.

- [126] DRIVER J, BAYLIS G C, RAFAL R D. Preserved Figure-Ground Segregation and Symmetry Perception in Visual Neglect [J]. *Nature*, 1992, 360(6399) : 73–75.
- [127] KIMCHI R, PETERSON M A. Figure-Ground Segmentation Can Occur Without Attention [J]. *Psychological Science*, 2008, 19(7) : 660–668.
- [128] PALMER S E. *Vision Science: From Photons to Phenomenology* [M]. Cambridge, Massachusetts : The MIT Press, 1999.
- [129] van de WEIJER J, SCHMID C, VERBEEK J. Learning Color Names from Real-World Images [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2007 : 1–8.
- [130] van de WEIJER J, SCHMID C, VERBEEK J, et al. Learning Color Names for Real-World Applications [J]. *IEEE Transactions on Image Processing*, 2009, 18(7) : 1512–1523.
- [131] VECERA S P, VOGEL E K, WOODMAN G F. Lower Region: A New Cue for Figure-Ground Assignment [J]. *Journal of Experimental Psychology: General*, 2002, 131(2) : 194–205.
- [132] KLYMENKO V, WEISSTEIN N. Spatial Frequency Differences Can Determine Figure-Ground Organization [J]. *Journal of Experimental Psychology: Human Perception and Performance*, 1986, 12(3) : 324–330.
- [133] HOFFMAN D D, SINGH M. Salience of Visual Parts [J]. *Cognition*, 1997, 63(1) : 29–78.
- [134] BAHNSEN P. Eine Untersuchung über Symmetrie und Asymmetrie bei visuellen Wahrnehmungen [J]. *Zeitschrift für Psychologie*, 1928, 108 : 129–154.
- [135] BERLIN B, KAY P. *Basic Color Terms: Their Universality and Evolution* [M]. Berkeley and Los Angeles, California : University of California Press, 1969.
- [136] HOFMANN T. Probabilistic Latent Semantic Indexing [C] // Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval. 1999 : 50–57.
- [137] QUELHAS P, MONAY F, ODOBEZ J-M, et al. Modeling Scenes with Local Descriptors and Latent Aspects [C] // Proceedings of the IEEE International Conference on Computer Vision. 2005 : 883–890.
- [138] VERBEEK J, TRIGGS B. Region Classification with Markov Field Aspect Models [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2007 : 1–8.
- [139] LIU Y, ZHANG D, LU G, et al. Region-based Image Retrieval with High-Level Se-

- mantic Color Names [C] // Proceedings of the International Multimedia Modelling Conference. 2005 : 180–187.
- [140] BENAVENTE R, VANRELL M, BALDRICH R. A Data Set for Fuzzy Colour Naming [J]. Color Research and Application, 2006, 31(1) : 48–56.
- [141] MENEGAZ G, Le Troter A, SEQUEIRA J, et al. A Discrete Model for Color Naming [J]. Eurasip Journal on Advances in Signal Processing, 2007 : 29125: 1–10.
- [142] SOILLE P. Morphological Image Analysis: Principles and Applications [M]. 2nd. New York : Springer-Verlag Berlin Heidelberg, 1999 : 173–174.
- [143] VINCENT L. Morphological Grayscale Reconstruction in Image Analysis: Applications and Efficient Algorithms [J]. IEEE Transactions on Image Processing, 1993, 2(2) : 176–201.
- [144] 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] // Proceedings of the IEEE International Conference on Computer Vision. 2001 : 416–423.
- [145] EVERINGHAM M, Van Gool L, WILLIAMS C, et al. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results [EB/OL]. 2012. <http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html>.
- [146] LOU J, WANG H, CHEN L, et al. Exploiting Color Name Space for Salient Object Detection [J/OL]. arXiv preprint, 2017: 1–13. <https://arxiv.org/abs/1703.08912>.
- [147] YU L, ZHANG L, van de WEIJER J, et al. Beyond Eleven Color Names for Image Understanding [J]. Machine Vision and Applications, 2018, 29: 361–373.
- [148] GONZALEZ R C, WOODS R E. Digital Image Processing [M]. 2nd. Upper Saddle River, New Jersey : Prentice Hall, 2002.
- [149] YANG C, ZHANG L, LU H, et al. Saliency Detection via Graph-Based Manifold Ranking [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2013 : 3166–3173.
- [150] ACHANTA R, SHAJI A, SMITH K, et al. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(11) : 2274–2281.
- [151] JOHNSON D B. Efficient Algorithms for Shortest Paths in Sparse Networks [J]. Journal of the ACM, 1977, 24(1) : 1–13.

- [152] CANNY J. A Computational Approach to Edge Detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986, 8(6) : 679–698.
- [153] KIM S, LEE J. Scale Invariant Small Target Detection by Optimizing Signal-to-Clutter Ratio in Heterogeneous Background for Infrared Search and Track [J]. Pattern Recognition, 2012, 45(1) : 393–406.
- [154] MALANOWSKI M, KULPA K. Detection of Moving Targets With Continuous-Wave Noise Radar: Theory and Measurements [J]. IEEE Transactions on Geoscience and Remote Sensing, 2012, 50(9) : 3502–3509.
- [155] ZHU B, XIN Y. Effective and Robust Infrared Small Target Detection With the Fusion of Polydirectional First Order Derivative Images Under Facet Model [J]. Infrared Physics & Technology, 2015, 69 : 136–144.
- [156] BAE T-W, ZHANG F, KWEON I-S. Edge Directional 2D LMS Filter for Infrared Small Target Detection [J]. Infrared Physics & Technology, 2012, 55(1) : 137–145.
- [157] CHEN C L P, LI H, WEI Y, et al. A Local Contrast Method for Small Infrared Target Detection [J]. IEEE Transactions on Geoscience and Remote Sensing, 2014, 52(1) : 574–581.
- [158] LI Y, LIANG S, BAI B, et al. Detecting and Tracking Dim Small Targets in Infrared Image Sequences Under Complex Backgrounds [J]. Multimedia Tools and Applications, 2014, 71(3) : 1179–1199.
- [159] LI W, PAN C, LIU L-X. Saliency-based Automatic Target Detection in Forward Looking Infrared Images [C] // Proceedings of the IEEE International Conference on Image Processing. 2009 : 957–960.
- [160] QI S, MA J, TAO C, et al. A Robust Directional Saliency-Based Method for Infrared Small-Target Detection Under Various Complex Backgrounds [J]. IEEE Geoscience and Remote Sensing Letters, 2013, 10(3) : 495–499.
- [161] LEE E, GU E, PARK K. Effective Small Target Enhancement and Detection in Infrared Images Using Saliency Map and Image Intensity [J]. Optical Review, 2015, 22(4) : 659–668.
- [162] GAO C, MENG D, YANG Y, et al. Infrared Patch-Image Model for Small Target Detection in a Single Image [J]. IEEE Transactions on Image Processing, 2013, 22(12) : 4996–5009.
- [163] CHEN H-Y, LEOU J-J. Multispectral and Multiresolution Image Fusion Using Particle Swarm Optimization [J]. Multimedia Tools and Applications, 2012, 60(3) : 495–518.
- [164] MATAS J, CHUM O, URBAN M, et al. Robust Wide Baseline Stereo from Max-

- imally Stable Extremal Regions [C] // Proceedings of the British Machine Vision Conference. 2002: 384–393.
- [165] MATAS J, CHUM O, URBAN M, et al. Robust Wide-Baseline Stereo from Maximally Stable Extremal Regions [J]. Image and Vision Computing, 2004, 22(10): 761–767.
- [166] OTSU N. A Threshold Selection Method from Gray-Level Histograms [J]. IEEE Transactions on Systems, Man, and Cybernetics, 1979, 9(1): 62–66.
- [167] YANG C, ZHANG L, LU H. Graph-Regularized Saliency Detection With Convex-Hull-Based Center Prior [J]. IEEE Signal Processing Letters, 2013, 20(7): 637–640.
- [168] VEDALDI A, FULKERSON B. VLFeat: An Open and Portable Library of Computer Vision Algorithms [EB/OL]. 2008. <http://www.vlfeat.org/>.
- [169] DRAGON R, OSTERMANN J, Van Gool L. Robust Realtime Motion-Split-And-Merge for Motion Segmentation [J]. Proceedings of the German Conference on Pattern Recognition, 2013: 425–434.
- [170] ZHANG W, CONG M, WANG L. Algorithms for Optical Weak Small Targets Detection and Tracking: Review [C] // Proceedings of the IEEE International Conference on Neural Networks and Signal Processing. 2003: 643–647.

附录

攻读博士学位期间发表及已完成的论文情况：

• 第一作者论文

- [1] **Jing Lou**, Mingwu Ren, Huan Wang. Regional Principal Color Based Saliency Detection. *PLoS ONE*, vol. 9, no. 11, pp. e112475: 1–13, 2014. (SCI: 000344863100105, 三区, 影响因子 3.234)
- [2] **Jing Lou**, Wei Zhu, Huan Wang, Mingwu Ren. Small Target Detection Combining Regional Stability and Saliency in a Color Image. *Multimedia Tools and Applications*, vol. 76, no. 13, pp. 14781–14798, 2017. (SCI: 000403039400013, 四区, 影响因子 1.530)
- [3] **Jing Lou**, Fenglei Xu, Qingyuan Xia, Wangkou Yang, Mingwu Ren. Hierarchical Co-salient Object Detection via Color Names. In *Proceedings of the Asian Conference on Pattern Recognition (ACPR)*, pp. 718–724, 2017. (Spotlight)
- [4] **Jing Lou**, Huan Wang, Longtao Chen, Qingyuan Xia, Wei Zhu, Mingwu Ren. Exploiting Color Name Space for Salient Object Detection. *arXiv preprint*, pp. 1–13, 2017. <https://arxiv.org/abs/1703.08912>

• 其他作者论文

- [1] Wei Zhu, **Jing Lou**, Longtao Chen, Qingyuan Xia, Mingwu Ren. Scene Text Detection via Extremal Region based Double Threshold Convolutional Network Classification. *PLoS ONE*, vol. 12, no. 8, pp. e0182227: 1–17, 2017. (SCI: 000408010000004, 三区, 影响因子 2.806)
- [2] Huan Wang, **Jing Lou**, Congcong Zhang, Yong Liu. Infrared Small Dim Target Detection Based on Weighted Nuclear Norm Minimization. In *Proceedings of the IEEE International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp. 1–6, 2017. (Poster, EI: 20183905873663)
- [3] 张从丛, 王欢, 楼竟. 基于加权核范数最小化的红外弱小目标检测. *华中科技大学学报 (自然科学版)*, 45(10): 31–37, 2017. (EI: 20174804465041)
- [4] Longtao Chen, **Jing Lou**, Wei Zhu, Qingyuan Xia, Mingwu Ren. Multi-appearance Segmentation and Extended 0-1 Program for Dense Small Object Tracking. *arXiv preprint*, pp. 1–14, 2017. <https://arxiv.org/abs/1712.05116>

- [5] Wei Zhu, **Jing Lou**, Qingyuan Xia, Mingwu Ren. Single Shot Text Detector with Rotational Prior Boxes. *Neural Processing Letters*, pp. 1–15, 2018. (Online)

攻读博士学位期间参加的科学研究情况：

- [1] 2011.03 – 2011.12 863 计划军口项目 “XX 弱小目标检测技术”
- [2] 2011.03 – 2012.01 国家自然科学基金面上项目 “基于目标区域演化分割的图像描述研究” (项目号 60875010, 经费 28 万元)
- [3] 2011.07 – 2016.12 863 计划军口项目 “XX 综合目标识别与仿真技术研究”
- [4] 2011.10 – 2013.01 大庆油田有限责任公司 “庆哈输油管道泄漏检测系统” (经费 68 万元)
- [5] 2013.10 – 2014.10 无锡市速波精密机械有限公司 “LCD 缺陷检测系统” (经费 15 万元)
- [6] 2014.07 – 2017.06 国家自然科学基金重点项目 “基于认知计算和信息挖掘的多波段夜视图像融合技术” (项目号 61231014, 经费 320 万元)