中图分类号: 学校代码: 10055
UDC: 密级: 公开

南開大學

博 士 学 位 论 文

面向复杂场景的交互式图像分割

Interactive Image Segmentation in Complex Scenes

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

南开大学研究生院
二〇二三年十二月

摘要

交互式图像分割指的是一种用户在图像上不断添加交互以获得目标对象精确掩膜的分割任务。该任务对于基于深度学习的计算机视觉领域所需的大规模数据标注具有重大意义，同时也作为图像编辑、目标标识等工作的基础。由于用户的任务需求不同，交互式图像分割需要面对各类复杂场景，其可能存在不同的图像类型、物体分布、目标结构等。在此情况下，该任务主要存在以下四个逐步递进的难点与挑战：（1）目标定位不准确；（2）局部区域精度低；（3）细小结构交互难；（4）医学图像分割差。面对这些难点，如何设计高性能的网络模型以及高效率的交互模式成为了面向复杂场景的交互式图像分割的关键问题。

对应上述挑战，本文由主到次提出了四个研究目标，从网络模型和交互模式两个维度提供了解决办法，并在多个数据集上取得了领先性能。具体内容如下：

1. 为了实现针对目标定位的全局物体分割，本文提出了基于初始交互点注意力的交互式分割。该工作突出了初始交互点的目标定位作用，并提出了初始交互点注意力网络。该模型利用初始交互点的指导信息解决了目标的定位问题，并使其他交互点能更好地实现修复目的。

2. 为了实现针对精确细节的局部区域分割，本文提出了深入聚焦视角的交互式分割。该工作提出了聚焦分割的流程框架。该框架从交互点的聚焦视角出发，在整体分割的基础上，对交互点周围的局部分割进行精细化修复，有效地提升了交互式分割方法对于细节的分割性能。

3. 为了实现针对细小结构的复杂拓扑分割，本文提出了修复细小结构的切割线交互式分割。该工作针对细小结构物体提出了新的切割线交互模式并设计了相应的网络模型。该模型利用细小区域的相似性，让用户自由地对局部或全局的细小结构分割进行修复，有效地减轻了用户的交互负担。

4. 为了实现针对低对比度的医学图像分割，本文提出了面向医学图像的多模式交互式分割。该工作设计了一个多模式的交互式医学图像分割框架。该框架不仅集合了多种初始及修复的交互模式，而且通过一个共享网络使这些交互相互协作，对低对比度医学图像中的目标进行高效分割。

关键词： 交互式图像分割；交互模式；神经网络；精确分割；用户行为

Abstract

Interactive image segmentation is a kind of segmentation task where the user continuously adds interactions to the image to obtain an accurate mask of the target object. This task is of great significance for the large-scale data annotation required in the field of computer vision based on deep learning, and also serves as the basis for image editing, target identification, etc. Due to the different task requirements of users, interactive image segmentation needs to face various complex scenes, which may contain different image types, object distributions, target structures, etc. Under this circumstance, the task mainly faces the following four progressive difficulties and challenges: (1) Low accuracy for target localization; (2) Poor details for local areas; (3) Difficulty for thin structures; (4) Poor segmentation for medical images. Faced with these difficulties, how to design high-performance network models and efficient interaction modes has become a key issue for interactive image segmentation in complex scenes.

In response to the above challenges, this thesis proposes four research objectives from primary to secondary, provides solutions from the dimensions of network models and interaction modes, and achieves leading performance on multiple datasets. The specific content is as follows:

1. To achieve global object segmentation with target localization, this thesis proposes interactive segmentation with first click attention. This work highlights the target localization role of the first click and proposes the first click attention network. This model utilizes the guidance information of the first click to solve the problem of target localization, and enables other clicks to better achieve the purpose of repair.

2. To achieve local region segmentation for precise details, this thesis proposes interactive segmentation with diving into a focus view. This work proposes a process framework for focused segmentation. This framework starts from the focus view of the click and performs detailed repairs for the local segmentation around the click based on the overall segmentation, effectively improving the segmentation performance of the interactive segmentation method for details.

3. To achieve complex topology segmentation for thin structures, this thesis proposes interactive segmentation with cutting lines to repair thin part segmentation. This work proposes a new interaction mode called cutting lines for thin objects and designs a corresponding network model. This model utilizes the similarity of thin parts to allow users to freely repair local or global segmentation of thin parts, effectively reducing the burden on users to make interactions.

4. In order to achieve low-contrast medical image segmentation, this thesis proposes a multi-mode interactive segmentation for medical images. This work designs a multi-mode interactive medical image segmentation framework. This framework not only integrates multiple initial and repair interaction modes, but also enables these interactions to cooperate with each other through a shared network to efficiently segment targets in low-contrast medical images.

Key Words: Interactive image segmentation; interaction mode; neural network; accurate segmentation; user behavior

目录

插图索引

表格索引

1 绪论

1.1 研究背景和意义

计算机视觉是信息技术领域一项重要的研究方向。它指的是让计算机模拟人类视觉，对可视媒体进行感知和理解的研究。随着科技的进步，移动设备、摄影设备、可穿戴设备等都具有了视觉感知器，从而能广泛地获取计算机图像数据。而计算机图像作为一种极为重要的可视媒体，在日常生活中无处不在，因此基于计算机图像的视觉任务成为了热门的研究方向。随着深度学习研究的逐渐成熟并得以广泛应用，如今的计算机视觉研究常常以卷积神经网络这一新兴技术作为基础。由于其对图像的理解分析能力大大增强，因此诞生了许多相关的人工智能应用，比如人脸识别 [1]、自动驾驶 [2]、机器人技术 [3] 等等。这些应用服务于人类社会的方方面面，推动了当今社会的发展。

图像分割是计算机视觉中的一项重要任务，它指的是利用计算机对图像中的用户感兴趣的目标进行分割，以得到分割后的掩膜结果。一般意义上的图像分割，主要是指自动类图像分割任务，即将图像输入模型之中，从而自动得到分割结果的任务。该任务本质上是一种图像理解任务，分割结果反映了模型对某一类目标的感知理解能力。根据用户的需求和目的不同，图像分割任务又分为许多的子类任务。最为常见的为图像的语义分割 [4]，即对图像中的各个像素进行分类。除此之外，还有分割出同类物体中不同个体的实例分割 [5]、对像素同时分割语义和实例的全景分割 [6]、检测出图像中显著目标的显著性分割 [7] 等等。这些任务广泛地应用在自动驾驶中的场景理解 [8]、手机摄影中的人像检测 [9] 等工作上。此外，一些特殊图像的分割任务也极其重要，比如医学图像分割任务 [10] 可以对疾病病灶、人体器官等进行分割，辅助医生进行诊断和治疗。

对于自动类的图像任务，由于各种算法和模型的性能参差不齐，输出的结果也有好坏之分。而且算法和模型存在性能瓶颈，用户总会得到一些不令人满意的结果。图像分割任务也类似，各种图像分割算法和模型总会存在分割错误或分割质量差的掩膜结果。而对于有些分割工作，用户需要获得较为精确的掩膜结果。比如，随着深度学习的普及，众多的图像分割任务需要训练一个性能良

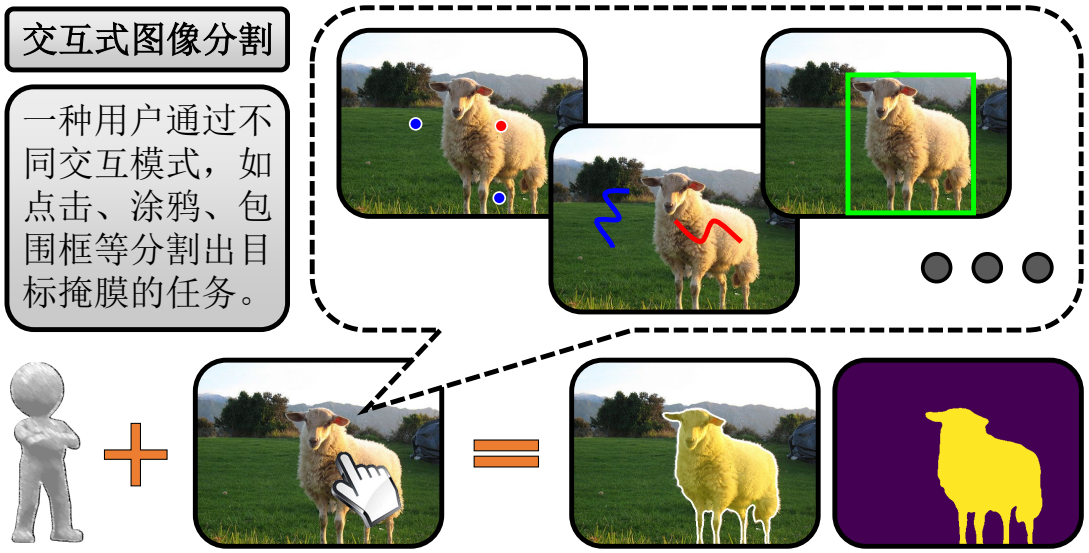


图 1.1　交互式图像分割任务介绍。

好的模型。而对于训练模型来说，大规模的图像数据和真值标注必不可少。这些数据的真值掩膜先前往往是通过纯人工标注的方式，不仅耗时耗力，还需要高昂的费用。如何高效地获得图像精确的掩膜标注，是计算机视觉研究的一个重点问题。考虑到计算机视觉领域，有许多引入人工因素的交互类图像任务。这些任务可以根据用户交互来进行图像检索 [11]、图像编辑 [12]、图像生成 [13] 等操作。如果输出结果不令人满意，用户则可以继续添加交互以逐渐生成更合乎需求的结果。如果把用户交互引入图像分割任务，则理论上可以获得更加精确的分割结果。交互式图像分割任务也因此成为了研究的热点问题。

如图 1.1所示，交互式图像分割 [14]，它是指一种用户通过不同交互模式分割出目标掩膜的任务。用户可以选择基于原始图像添加交互以获得初始的分割结果，也可以直接使用自动类模型得到粗糙的分割结果。基于该结果，用户可以判别预测错误的区域，并在物体内部、外部、边界等处添加交互。这种交互可以是点击、涂鸦、包围框等一系列基于绘制的交互模式。算法和模型会基于用户添加的交互，对错误预测的区域进行修复，以重新生成更精确的分割结果。如果得到的结果仍然无法让用户满意，用户可以反复迭代添加一系列交互，直到最终获得满足用户需求的结果。该任务与一般的分割任务最大的不同就是，通过引入用户的交互行为，分割结果的性能可以不断提高，以逼近最真实的目标掩膜。

交互式图像分割任务从上世纪以来一直广受关注。在早期，交互模式主要为能较大范围标识前背景的涂鸦交互 [15]、包围框交互 [16] 等。分割模型主要利用图像的底层特征，如颜色、纹理等，通过图割 [17]、随机游走 [18] 等传统算法来进行预测。如今的交互式分割研究则主要集中在基于深度学习的方法。由于神经网络能充分提取用户交互信息与图像的高层特征，点击的交互模式 [19] 大为流行。此外，其他丰富的交互模式，如边界点 [20]、极值点 [21] 等，也取得了良好效果。针对交互设计的各种模型也使得高性能的交互式分割成为了可能。

由于现实场景是复杂的，用户对于交互式图像分割的目标也不尽相同。模型面对的图像可能是自然图像，也可能是医学图像。除此之外，复杂场景中的物体各不相同，呈现不同的结构特征。用户可能需要分割猫、狗这样的生物目标，也可能需要分割自行车、电视这样的人造物体；可能面对西瓜、水壶这样的简单结构物体，也可能面对球拍、树枝这样的复杂结构物体。因此研究面向复杂场景的交互式图像分割是极其重要的。由于有着用户的参与，交互式图像分割的主要研究方向可以分为算法模型的研究和交互模式的研究。首先，针对流行的交互模式，主要集中在研究如何设置合理的算法与模型，更好地感知用户的交互意图，以提升分割的性能，这也是本文第3章、第4章重点研究的内容。其次，针对特殊的图像和物体，主要集中在研究如何设计更高效的交互模式来分割目标对象，以减轻用户的负担，这也是本文第5章、第6章重点研究的内容。

关于交互式图像分割的研究不断进步，对如今以大规模的数据标注 [22] 为基础的图像分割领域，有着重要意义。此外，许多图像相关任务，如图像编辑 [23]、图像修复 [24] 等，都需要相应的物体掩膜，交互式分割正是这类任务的基础。对于医学图像，通过交互精确地分割出目标掩膜，有助于医学上的定量分析 [25]、疾病标识 [26] 等工作。总之，研究面向复杂场景的交互式图像分割任务具有重要的实际价值与社会意义，值得广大研究者在此方向上进行探索。

1.2 研究现状和难点

交互式图像分割需要面对复杂场景中的各种目标，由于图像以及物体之间的差异性，该任务存在着各式各样的挑战。图 1.2中展示了复杂场景下的交互式图像分割面对的循序渐进的四大重要难点和挑战，分别是目标定位不准确、局部区域精度低、细小结构交互难和医学图像分割差。对于这四个研究难点，本章节将结合该任务的研究现状做简要介绍。

目标定位不准确。 交互式图像分割的首要目标是分割出物体的主体，因此首先需要定位目标物体。而且如果目标在混杂场景中或者场景中存在多个同类物体，如图 1.2中展示的多个同类动物的场景，此时物体的目标定位就更为重要。在这类场景中，准确的定位有助于模型区分其他物体或同类对象，从而获得更准确的主体分割。一些工作会使用特殊的初始交互来定位物体。比如，在一些方法中，包围框 [27–29] 作为了初始交互，后续则使用多边形编辑进行分割调整。还有的方法使用包围框协同内部点 [30] 或只用极值点 [21] 来进行目标定位。然而在交互模式方面，广为流行的是基于前背景点击的交互式分割方法。该模式下，用户首先添加一个交互点来进行初始分割，随后不断添加其他交互点来修复掩膜。对于该交互模式下的工作，有些方法研究交互点的训练策略 [19, 31, 32]，有些方法侧重研究模型的输入编码 [33]，有些方法专注于设计有效的网络结构 [34, 35], 还有些方法针对交互产生的歧义性进行研究 [36, 37]。这些交互式分割方法都能让分割性能和效率进一步提升，但因为它们将所有交互点同等对待，在一定程度上缺乏了对目标的着重定位能力。如果利用初始交互点进行目标定位，后续的交互点就能更好地去着重修复错误区域。

局部区域精度低。 当分割好目标对象的主体部分后，更需要关注的就是对象的细节部分。对于一些物体，如图 1.2中展示的菠萝，由于其细节较多，因此目标的分割掩膜就需要更高的精度。对于基于前背景点击的交互式分割方法，无论是真实用户，还是基于模拟算法的机器人用户，都会倾向在交互式分割过程中，对比目标区域与当前分割结果，选择点击最大的错误区域。这些错误区域有时候是物体的主体部分，有时候是物体的细节部分。但即使交互点位于物体细节部分，由于分割网络是全局图像感知的，而且存在分辨率降采样和交互点过密等因素，物体的细节分割质量有限。现在的一些方法关注每个交互点对网络输入 [38]、模型参数 [39, 40] 等整体的调节作用，还有的方法着重于将交互点的影响从局部扩散到全局 [41]，一定程度上缺乏了对局部细节分割的关注。随着研究的发展，一些方法也开始逐步关注分割细节，比如通过一对前背景交互点来修复部分细节 [42]，使用边缘信息提高整体分割质量 [43] 等。但这些方法都没有从单个交互点的局部视角出发，就该交互点的针对区域进行着重修复，这在一定程度上与用户添加该交互点的目的不符。如果能对局部的精确细节进行分割，得到的掩膜结果则可以更好地作为模型训练的图像标注或用于其他图像任务。

细小结构交互难。 在交互式图像分割任务中，由于物体的复杂性，用户可能会遇到一些有着细小结构的物体，如图 1.2中展示的昆虫。这些物体可能是狭长的，比如电线杆、标枪等；也可能有着复杂的拓扑结构，比如蜘蛛网、树枝等。前面描述的一些基于前背景点击的交互式图像分割方法能对大部分目标物体的主体和细节进行良好的分割，但对于这类物体存在一定弊端，主要表现在用户的交互负担大。由于这些目标可能存在狭长结构或孔洞结构，前景交互点往往需要仔细瞄准才能准确地点击在目标前景区域上，而对于那些孔洞中的背景像素，也难以进行点击，这大大提升了用户的交互时间和体力负担。而且前背景交互点过于密集，也容易造成分割结果错误或质量下降。此外，其他的一些交互模式，如包围框等，则不能很好地标识细小结构，还可能存在包围区域过大而导致分割出错误目标等问题。针对细小结构物体的交互式分割，已经存在一些初步的探索。传统的一些基于前背景涂鸦的方法 [44, 45]，以及基于深度学习的极值点交互 [46]、覆盖细小部分的涂鸦交互 [47] 能一定程度提升对细小结构物体的分割效率，但仍然存在较大的交互负担。如何设计一个针对细小结构物体能进行高效分割的交互模式和对应的模型方法，成为了一个迫切且重要的研究课题。

医学图像分割差。 不同于自然图像，医学图像是计算机图像中的一类特殊图像。它们许多是通过医学设备生成的，比如超声设备、核磁共振设备等等。如图 1.2中展示的超声图像所示，这类医学图像有一个重要特征，即低对比度特征，表现为颜色单一、差异不明显等。对于医学图像的交互式分割往往集中在分割器官、病灶等，然而这类医学目标的特征往往复杂且多样，它们可能存在前背景相似导致的边缘不清晰、自身特性导致的结构不规则等问题。如何对这些医学目标进行准确的交互式分割，成为了一个重要的研究问题。由于医学图像底层特征的相似性，基于传统算法的交互式医学图像分割方法 [48–50] 的分割结果往往不尽人意。深度学习的出现一定程度上缓解了这个问题，一些相关的方法也开始得以发展。实验证明，使用包围框 [51] 能对医学目标进行一定程度的定位。基于前背景点击和涂鸦的方法 [52–54] 能够修复医学目标的主体区域。而使用边界点 [55] 和边界涂鸦 [56] 则能很好地解决边界的精确预测问题。这些交互式分割方法往往都是针对一种交互模式设计对应的单一模型。如果能将这些交互模式集合到一个模型，用户同时可以使用多种交互模式，就能较好地解决低对比度医学图像造成的歧义性问题，大大提高医疗工作者的交互式分割效率。

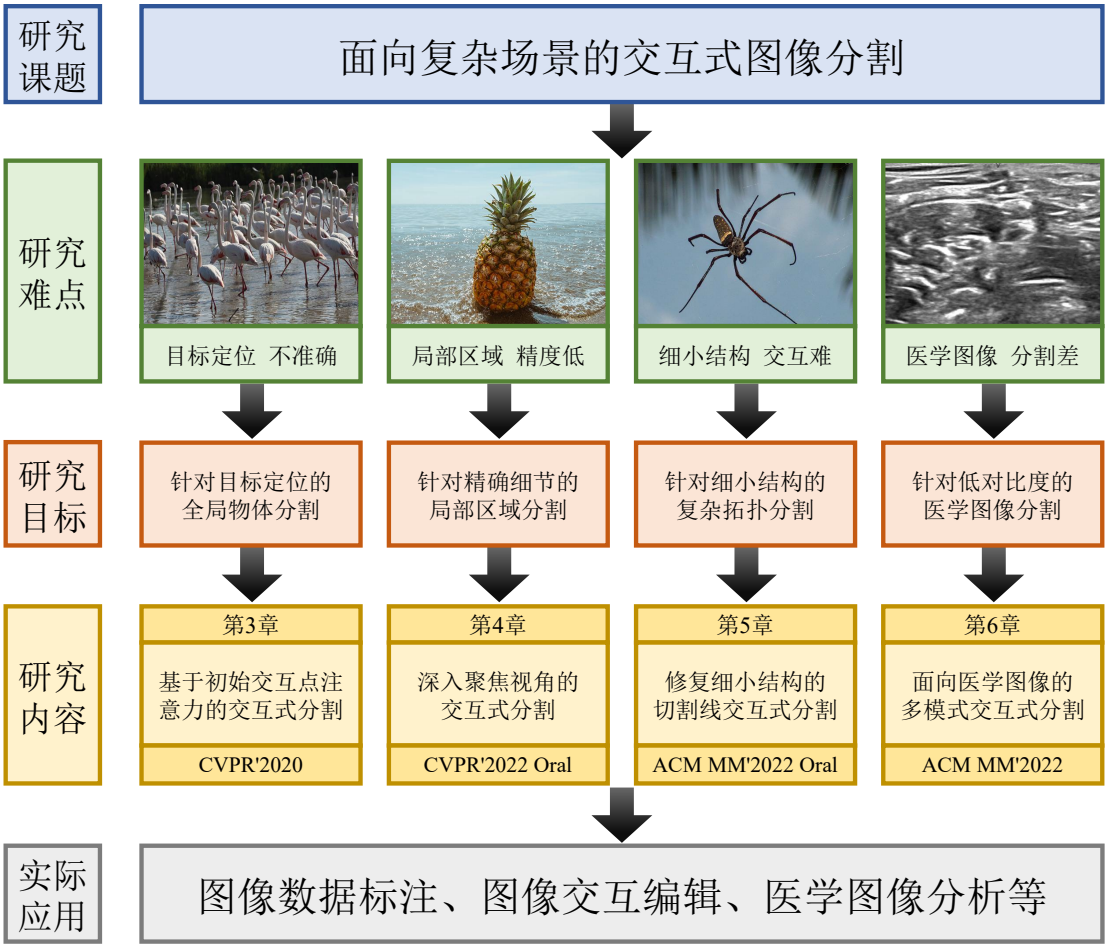


图 1.2 面向复杂场景的交互式图像分割的研究架构。

1.3 研究目标和贡献

图 1.2展示了本文的研究架构。本文的研究课题是面向复杂场景的交互式图像分割。面对复杂场景下目标定位不准确、局部区域精度低、细小结构交互难、医学图像分割差这四个逐步递进的难点，本文设立了四个由主到次的研究目标，即针对目标定位的全局物体分割、针对精确细节的局部区域分割、针对细小结构的复杂拓扑分割、针对低对比度的医学图像分割。基于这四个研究目标，本文提出了对应的研究内容，分别是基于初始交互点注意力的交互式分割、深入聚焦视角的交互式分割、修复细小结构的切割线交互式分割、面向医学图像的多模式交互式分割。这几个研究内容从多方面逐步地解决了复杂场景下的交互式图像分割存在的问题。具体而论，本文的研究内容和贡献如下：

（1）本文面对交互式图像分割中目标定位不准确这一难点，以针对目标定位的全局物体分割为目标，提出了基于初始交互点注意力的交互式分割。对于基于前背景点击的交互式分割任务，其可能要面对各种图像场景，解决分割对象在复杂场景中的目标定位是首要关注点。目标定位不准确会造成网络模型识别出错误对象或者将多个同类对象识别成一个对象等问题。因此，有效地定位目标能提高全局物体的分割效果。在该任务中，用户初次点击来分割目标对象的主体，然后在错误预测区域上迭代地提供更多的交互点，以不断修复分割结果。现有的方法将所有交互点同等对待，忽略了初始交互点和其他交互点之间的区别。本文展示了初始交互点对于目标定位的关键作用。为了更好地利用初始交互点，本文提出了初始交互点注意力网络，简称 FCA-Net。该网络仅仅在基础网络结构上添加了初始交互点注意力模块，使模型具有了聚焦不变能力、位置指导能力、容错能力等优点，充分证明了初始交互点的重要作用。同时本文还为了更好地进行目标整体分割，提出了基于交互点的损失函数和结构完整性策略来提升性能。可视的分割结果和充分的实验证明了初始交互点对于目标定位的重要性和提出的 FCA-Net 方法的优越性。

（2）本文面对交互式图像分割中局部区域精度低这一难点，以针对精确细节的局部区域分割为目标，提出了深入聚焦视角的交互式分割。对于基于前背景点击的交互式分割任务，在主体分割好后，需要关注分割对象的细节分割质量。交互式分割在像素级别的图像标注和编辑中是一项重要工具，而拥有精确细节的掩膜则更有利于这些工作。为了获得高准确率的二值分割掩膜，用户倾向于在边缘、孔洞等对象细节周围添加用来修复的交互点以精细化局部分割。由于模型的分辨率降采样问题，目标的细节难以被精确感知。除此之外，许多方法会从全局视角利用这些交互点作为指导，共同确定整体分割结果，而忽视了许多交互点的目的是用来修复先前分割中存在的局部错误。为了充分考虑交互点的细节分割目的，本文深入交互点的聚焦视角，并赋予它们在分割物体细节时决定性的作用。基于聚焦视角，本文设计了一个简单但有效的交互式分割流程框架，简称 FocusCut。该方法集成了全局分割和局部修复的功能。在获得目标的全局分割后，它以自适应的范围在原图上裁剪以交互点为中心的区块并输入网络中，来获得更为精细的局部分割，最后再反馈回全局分割中。在不增加网络参数的基础上，用户可以得到更为精细的掩膜。可视的分割结果和充分的实验证明了 FocusCut 方法使得交互式分割任务中的精细分割成为了可能。

（3）本文面对交互式图像分割中细小结构交互难这一难点，以针对细小结构的复杂拓扑分割为目标，提出了修复细小结构的切割线交互式分割。先前基于前背景点击的交互式分割方法已经可以处理众多的图像，但对于一些物体拥有的细小结构，由于其可能存在复杂拓扑性，用户的交互负担较大或难以分割出满意的结果。此外，自动类图像分割技术的输出结果中细小部分的分割时常也不太理想。在实际应用中，不可避免地需要对这些输出进行后处理。然而，无论是使用专业的图像编辑软件，还是使用基于点击、多边形、笔刷等交互模式的方法，修复这些输出都是费时费力的。为了高效分割出目标的细小结构，本文提出了一种有效的切割线交互模式。该模式只需要用户在错误标记的细小部分处划一条线，就像用刀切割一样。这种低负担、直观的操作不需要用户特意瞄准，并且对使用鼠标、触摸板和移动设备的用户非常友好。本文还基于切割线交互模式提出了一个细小结构分割修复框架，简称 KnifeCut。该方法设计了相应的网络模型，利用细小结构的局部与全局相似性，对细小部分的分割进行修复。此方法为用户提供了两个结果，其中一个只修复用户交互所针对的细小部分，而另一个则修复所有与该部分相似的细小结构。可视的分割结果和充分的实验证明了 KnifeCut 方法能有效地减轻用户修复细小结构分割的负担。

（4）本文面对交互式图像分割中医学图像分割差这一难点，以针对低对比度的医学图像分割为目标，提出了面向医学图像的多模式交互式分割。先前的方法已经能很好地处理自然图像，但对于医学图像，往往由于其存在的低对比度特性以及各类医学目标的不同形状和纹理特征而难以分割。对于现在数据紧缺的医学图像分析模型来说，大规模的像素级标注是十分缺乏的。为了能快速生成标注信息，迫切需要一种便捷高效的交互式医学图像分割方法。现有的交互模式往往只能处理医学图像中目标的部分歧义性。为了解决这一问题，本文提出了一种多模式的交互式医学图像分割框架，简称 MMIIS。在该框架下，用户可以选择不同的交互模式，并通过一个共享的网络模型来允许其相互协作以共同发挥作用。用户首先可以根据目标结构复杂程度使用各种初始交互模式，如包围框、包围多边形、包围涂鸦，来生成初始分割结果。然后在其基础上，用户可以综合利用区域和边界的交互，如前背景区域的点击或涂鸦、边界的点击或涂鸦，来修复不同歧义性导致的错误分割。本文在 X 光图像、超声图像等六种医学图像上评估了提出的框架。可视的分割结果和充分的实验证明了 MMIIS 方法能基于用户交互来准确预测低对比度医学图像下的目标。

1.4 本文的组织结构

本文的组织结构如下：第1章介绍了交互式图像分割的研究背景和意义、研究现状和难点，以及研究目标和贡献。第2章先介绍了相关的自动类图像分割任务和交互类图像任务，然后着重从交互模式的角度详细介绍了交互式图像分割这一任务。对应第1章提出的研究目标和研究内容，后续分别进行了阐述。第3章以针对目标定位的全局物体分割为目标，介绍了基于初始交互点注意力的交互式分割。第4章以针对精确细节的局部区域分割为目标，介绍了深入聚焦视角的交互式分割。第5章以针对细小结构的复杂拓扑分割为目标，介绍了修复细小结构的切割线交互式分割。第6章以针对低对比度的医学图像分割为目标，介绍了面向医学图像的多模式交互式分割。第7章对本文工作进行了总结，并对基于该工作的未来工作的可改进点和可探索点进行了展望。

2 相关工作综述

本章主要对本文的研究内容所涉及的相关工作进行了详细介绍。章节2.1介绍了与交互式图像分割任务相关的自动类图像分割任务。章节2.2介绍了与交互式图像分割任务相关的交互类图像任务。章节2.3对交互模式进行了三类划分并详细介绍了各类型的交互式图像分割方法。章节2.4对该章节进行了总结。

2.1 自动类图像分割任务

图像分割指的是分割出目标对象的掩膜。大多数图像分割任务指的是自动类图像分割，即算法或模型对输入图像进行自动处理以输出分割结果。根据任务属性不同，像素被赋予了不同的标签。下面将介绍一些常见的图像分割任务。

在早期，图像分割主要是利用图像的底层特征，如亮度、颜色、纹理、边缘等，来进行二值分割。它们包含了基于阈值的分割方法 [57]、基于边缘的分割方法 [58]、基于区域生长的分割方法 [59]、基于图的分割方法 [60] 等等。

随着深度学习的发展，图像的高层特征，比如语义特征等，得到了更好的提取。如今的语义分割任务 [61–63] 是常见的图像分割任务，它的目标是将图像上的属于同语义的像素打上相同的标签。这些标签不再是二值的，而是丰富多样的。比如对于自然图像，像素将被打上天空、地面、人、河流等一系列标签。该任务是机器人感知识别周围环境、汽车自动驾驶等工业界重要视觉应用的基础。该任务得到的结果是一张类别预测图，每个像素对应了预测的类别。如果要获得某个语义的分割图，比如天空的分割图，则将结果中的具有天空标签的像素和其他像素进行二值化处理，就能得到对应结果。

实例分割任务 [64–66] 是目标检测任务 [67] 的进一步扩展。目标检测会将图像中的一类物体的各个实例包围框检测出来，而实例分割会进一步将包围框中的物体分割出来。该任务与语义分割不同。对于某类物体，语义分割已经将所有该语义的目标像素获得，但该类目标实例之间并无区分。实例分割就是为了解决该问题而存在的，它会将同类语义的不同实例区别开。比如“汽车”这个语义，实例分割会将图像中的每一辆汽车对应的像素打上不同的标签。该任务对于目标计数、安防监控、自动驾驶等应用也具有重要意义。

基于语义分割和实例分割，研究人员引申出了许多种不同的任务。全景分割任务 [68] 是语义分割和实例分割的结合。它会给图像中的每个像素赋予语义标签和实例标签。医学图像分割也是语义分割的一种特例。它针对的是医学图像，目的是将医学图像上的目标分割出来。这种目标可以是生物 [69]、器官 [70]、病灶 [71] 等等。和实例分割类似，如果将每个医学目标实例区分开，就是医学图像实例分割任务 [72]。除了医学图像外，其他特殊的图像类型，也会有对应的图像分割子任务，如遥感图像分割任务 [73]、红外图像分割任务 [74] 等等。

显著性物体检测任务 [75–77]，其目的是，检测出图像中显著的物体。其真值标注图为二值的掩膜图，分别代表了其上的像素是否属于显著性物体。该任务生成的结果是每个像素的显著性置信度图，经过二值化处理后，会得到二值分割的显著性物体图，所以也可以把该任务称作显著性物体分割任务。除了一般图像的显著性物体检测任务外，针对不同的图像类型，该任务有许多子任务。RGB-D 显著性物体检测 [78, 79] 旨在分割出带有深度图的图像中的显著性物体。光场显著性物体检测 [80, 81] 是通过输入的光场数据来检测显著性物体。显著性实例物体分割 [82, 83] 是实例分割任务的子任务，会将图像中显著物体的每个实例打上不同的标签。与显著性物体检测相对应，伪装物体检测 [84, 85] 旨在检测出图像中伪装的物体。同样，伪装物体检测也存在相类似的子任务，如 RGB-D 伪装物体检测 [86]、伪装实例物体分割 [87] 等等。以上这些显著性物体或伪装物体的检测分割任务都可以通过阈值化得到二值分割的掩膜。

无论是上述的哪一种分割任务，都属于自动类图像分割任务。由于算法或模型的性能瓶颈，其得到的掩膜结果往往与真值标注存在一定差异。而交互式图像分割提供了不断向真值标注逼近的方法，即用户通过不断修正错误预测结果，以提高分割掩膜的准确度。交互式图像分割方法也可以先获得这些自动类图像分割任务的预测结果图，并通过在其上添加交互不断修复它们。

2.2 交互类图像任务

计算机视觉中许多图像任务需要用户的参与和交互。用户的交互可以分为多种形式，比如用户提供文字 [88] 或标签分布图 [89] 来生成图像，用户通过不断地对话检索图像 [90]、生成图像 [91]，以及用户从样本中进行选择 [92] 等等。而交互式图像分割任务的交互是指用户在图像上进行绘制的交互操作，比如绘制点、涂鸦、包围框等等。这里将具体介绍绘制型的图像交互类任务。

交互式图像搜索是指通过用户绘制的图案，从一系列图像中搜索出与图案相关的图像。用户可以使用绘制的草图从互联网上检索与草图相似结构的图像 [11]，也可以通过绘制的骨架在一系列图像中寻找指定的目标图像 [93]。

交互式图像编辑指的是根据用户的交互对原始图像的色彩、形态等进行转化。对于单张图像，可以进行操纵变换来改变图像形态。有调整图像控制点的基于自由变形的操纵变换方法 [94]，调整算法生成的目标骨架的基于骨架的操纵变换方法 [12]，以及对目标刚性建模并调整来改变目标形状的基于物理的操纵变换方法 [95] 等等。交互式图像修复 [96] 可以通过交互方式选取图像部分区域，对这部分区域进行隐藏，使图像该部分与周围环境保持一致。交互式图像编辑还有许多是对多个图像进行操作，用户可以利用多个图像对其中一个图像进行编辑。图像插入或图像和谐化 [97] 指通过交互方式选择一定图像部分，将该部分置入另外一张图像但尽可能减少违和感。交互式图像风格化 [98] 是通过交互，将其他图像的局部风格迁移到一张图像的不同区域上。

交互式图像生成指的是根据用户的交互生成全新的图像。随着深度学习的发展，神经网络的生成能力使这一类任务得到了快速的发展。有的工作是在原图像上进行交互，使其产生较大改变。比如通过在两点之间进行拖拽 [99, 100]，控制眼睛的开合、物体的转向等，以产生新的图像。还有的工作是不依赖原始图像，完全通过简单的交互，生成全新的内容。比如可以通过交互绘制的草图 [101] 或者逐步添加的简单绘画 [13] 等来生成丰富的面部图像。除此之外，还可以生成各式各样的内容，比如生成服装图像 [102] 等来进行辅助设计。

2.3 交互式图像分割任务

交互式图像分割 [14] 是指通过用户交互对图像中的目标进行分割。本节根据交互类型不同，分为三个部分，分别是基于前背景标注的交互式分割、基于边界标注的交互式分割、基于混合标注的交互式分割。

2.3.1 基于前背景标注的交互式分割

基于前背景标注的交互式分割指的是用户的交互操作集中在目标对象的前景像素和背景像素上。此类交互模式可以是点击、涂鸦等用来标识一定区域的绘制方式。相关方法主要通过标识一定的前景像素和背景像素作为种子，利用两种像素的不同特征，使算法或模型能分割出指定的目标对象。

早期时候的交互式分割方法主要利用的是图像的底层特征，如颜色、纹理、边缘等等。有许多方法是基于图论，将每个像素当作图节点，来进行分割的算法，称作图割。Boykov 等人 [17, 103] 最先提出了经典的基于最小割最大流算法 [104] 的 GraphCut 方法。Blake 等人 [105] 将 GraphCut 的前景和背景分割的能量最小化模型形式化为概率高斯混合马尔可夫随机场，并开发了一种用于参数学习的伪似然算法。Vicente 等人 [44] 在 GraphCut 之前施加了额外的连接性来处理细小结构物体的交互式分割。Bai 等人 [15] 和 Price 等人 [106] 都使用基于测地距离的图论方法来进行图像分割与抠图。Kim 等人 [107] 提出了一种非参数学习技术，从区域中像素的似然性中递归估计区域似然性来作为高阶线索。De Miranda 等人 [108] 利用图的协同弧权重估计来进行图割。Gulshan 等人 [109] 采用了星形先验的 GraphCut 算法 [110], 该方法根据用户交互从图像的多个点向各方向星形膨胀来进行分割。Bai 等人 [111] 提出了允许用户犯错的交互式分割方法，允许用户的交互可以有一定错误，从而减少了用户的负担。Wang 等人 [112] 解决了当用户存在错误交互时的标签先验估计问题。

除此之外，还有基于传统算法的各类交互式分割方法。Adams 等人 [59] 提出了基于种子区域生长的方法。Vezhnevets 等人 [113] 使用了基于细胞自动机的理论进行图像的交互式分割。Grady 等人 [18] 提出了经典的随机游走算法，把图像构建成一个无向图模型来求解第一边值问题。Kim 等人 [18] 改进了该方法，提出了重启随机游走算法，在游走过程中有一定重启概率回到起点。Dong 等人 [45] 设计了一种新的带有标签的亚马尔可夫随机游走算法来针对细小结构的物体。Protiere 等人 [114] 通过涂鸦的加权距离得到分割，加权距离由一系列 Gabor 滤波器计算得到。Xiang 等人 [115] 把交互式分割当成一种样条线回归问题。Ding 等人 [116] 使用概率超图分割图像，该超图对图像像素之间的空间和外观关系进行建模。Ning 等人 [117] 提出了基于最大相似度的区域合并的方法。Zhang 等人 [118] 提出了一个贝叶斯网络模型用来进行自动类图像分割和交互式图像分割。Nguyen 等人 [119] 提出了一种基于连续域凸主动轮廓模型的鲁棒且精确的交互式分割方法。Noma 等人 [120] 提出了一种基于模型的图匹配方法，并用在交互式图像分割上。Panagiotakis 等人 [121] 在交互式分割中使用基于合成坐标的图聚类算法。Hu 等人 [122] 利用将用户的交互信息传播到了整个图像的核传播 [123] 中学习到的数据结构来进行交互式图像分割。Souratietal 等人 [124] 提出基于迭代地吸收用户反馈的受约束的频谱聚类的交互式分割方

法。Chen 等人 [125] 通过迭代融合像素似然信息和上下文信息来进行交互式分割。Jian 等人 [126] 使用自适应约束传播进行半监督核矩阵学习，将交互信息自适应传播到整个图像中并保持了原始数据的一致性。Zemene 等人 [127] 在交互式分割中利用了基于一组与优势集有关的二次优化问题的一些性质。Wang 等人提出了可以同时分割多个目标的方法 [128]，基于可能性扩散和感知学习的方法 [129]，以及将分割问题认作基于用户意图先验的概率估计问题的方法 [130]。

针对医学图像，Andrews 等人 [48] 使用了预计算的具有优先级的快速随机游走算法来分割医学目标。Wang 等人使用动态平衡在线随机森林 [49] 以及多视图中逐切片传播 [50] 来进行交互式医学图像分割。

随着深度学习的发展，图像的高层特征，如语义特征等，可以得到有效提取。越来越多的研究人员利用神经网络进行交互式分割任务的研究。由于神经网络可以深度提取信息，因此点击作为较轻松的交互模式逐渐取代了早期的涂鸦交互模式。Xu 等人 [19] 最早提出了基于深度学习的交互式图像分割方法，提供了交互点转化编码以及模拟交互点采样策略等。Liew 等人 [42] 提出了 RIS-Net，利用一对前背景交互点来对局部区域进行分割以优化分割结果。Li 等人 [36] 使用两个卷积神经网络来训练和挑选出合适的结果，以解决分割目标的潜在多样性问题。Song 等人 [131] 提出了 SeedNet，应用强化学习，根据用户交互点以产生更多潜在的交互点。Mahadevan 等人 [31] 提出了一种迭代训练的策略。Hu 等人 [34] 使用一种双分支神经网络结构，将图像和交互输入不同的两分支网络来进行交互式分割。Majumder 等人 [33] 提出了一种新的用户交互点转换编码策略，以生成内容感知的指导图，并输入神经网络来获得分割结果。Liew 等人 [37] 提出了 MultiSeg，将尺度多样性引入到模型中来帮助用户快速定位他们想要的目标。Jang 等人 [38] 提出了 BRS，通过反向传播修复策略来纠正初始结果中与用户交互不符的错误标记像素。Sofiiuk 等人 [39] 提出了 f-BRS，在图像特征层面进行反向传播来提高网络分割精度。Kontogianni 等人 [40] 采用用户的修正交互作为训练样本，并且立即更新模型的参数，来更好地拟合数据集。Hao 等人 [43] 利用网络预测的边缘来提升分割质量。Chen 等人 [41] 提出了 CDNet，引入一种非局部的方法来充分利用用户的交互点信息对全局进行分割。Chen 等人 [132] 还提出了 FocalClick 来预测和更新局部区域，实现了高效计算。Sofiiuk 等人 [32] 提出了交互式图像分割任务的新训练范式。Liu 等人 [133] 利用网络生成伪交互点来减少用户交互负担，并提升性能。Faizov 等人 [35] 使用 Transformer 模

型 [134] 来进行交互式分割。Gui 等人 [135] 使用点击的表征嵌入和空间注意力同时进行多物体的交互式分割。Li 等人 [136] 增强了用户交互点的空间感知能力。Zhou 等人 [137] 把交互式图像分割任务作为基于高斯过程的逐像素二值分类模型。Du 等人 [138] 提出了轻量的掩膜修复网络，从而使交互式分割过程不用每次重新输入主网络，以提高效率。Wei 等人 [139] 基于高层特征的相似性修复先前预测。Yang 等人 [140] 提出了遥感图像的交互式分割。Wei 等人 [141] 通过利用所有可用的语义线索来提升分割的准确率。

针对医学图像，Wang 等人 [52] 提出了 DeepIGeoS，通过测地距离变换将用户交互与神经网络相结合，提出了一种能提供更好的密集预测的分辨率保持网络。Liao 等人 [53] 在交互式医学图像分割中使用强化学习进行迭代修复。Wang 等人 [54] 提出了不确定引导下的交互式医学分割方法。Zhang 等人 [142] 提出了只基于一个交互点的医学图像分割方法。Liu 等人 [143] 提出 iSegFormer，使用 Transformer 模型 [134] 进行交互式医学图像分割。

2.3.2 基于边界标注的交互式分割

基于边界标注的交互式分割指的是用户的交互操作集中在标识目标的外轮廓。包围框交互属于一种泛边界，用来标识目标的大体范围。除此之外，还可以采用在目标真实轮廓边界上进行标记，可以是轮廓边界上的点击和涂鸦、极值点、套索等一系列交互模式。下面将分别对这些交互模式进行详细介绍。

包围框是一种常见的泛边界交互。它是一个矩形框，它的外部全是背景像素，所有目标对象的像素都在包围框内。Rother 等人 [16] 使用高斯混合模型改进了 GraphCut 方法，并提出了经典的基于包围框的 GrabCut 方法。Lempitsky 等人 [144] 在 GrabCut 方法基础上提供了边界框的先验信息，以提高分割的前景对象与包围框的紧密性。Cheng 等人 [145] 提出了 DenseCut，使用密集连接的条件随机场替代 GrabCut 方法中耗时的全局色彩模型来进行迭代优化。Rajchl 等人 [145] 将 GrabCut 中的高斯混合模型用深度神经网络替代来进行分割。Wu 等人 [146] 提出了 MILCut，将交互式分割问题定义为多实例学习任务，通过从包围框内的扫掠线上的像素来生成前景包并进行分割。Yu 等人 [147] 使用不贴合目标的包围框作为交互，并采用了基于马尔可夫随机场模型的分割方法。Xu 等人 [148] 提出了基于深度学习的方法，将包围框作为交互输入神经网络进行分割。Wang 等人 [51] 针对医学图像提出了 BIFSeg，在模型推理过程中进行了针对图像的微调。以下的一些方法利用包围框生成的并非图像分割结果，而是目标的

轮廓多边形，用户可以手动调整多边形顶点进行进一步修复。Castrejon 等人 [27] 提出了 Polygon-RNN，其利用循环神经网络对图像进行分割，得到由多个点组成的物体轮廓多边形。Acuna 等人 [28] 提出了改进后的 Polygon RNN++，使用强化学习对网络进行训练并使用图神经网络来增加输出结果的分辨率。Ling 等人 [29] 提出了 Curve-GCN，通过使用图卷积网络模型来同时预测多边形所有顶点，以减轻了多边形预测过程中的顶点顺序性问题。

轮廓边界交互是指用户的交互在目标的轮廓边界像素上。它的交互可以是点击，也可以是涂鸦，用来标识具体的边界。Le 等人 [149] 将用户提供的边界点输入神经网络模型来进行物体轮廓边界预测。Jain 等人 [20] 提出了点击雕刻算法，通过物体的边界点进行图像和视频的交互式分割。Aresta 等人 [55] 针对医学图像提出了 iW-Net，基于自动分割的结果，只使用 2 个边界点进行分割修复。

极值点是一种特殊的边界点，它是物体最上、最下、最左、最右的四个点，这四个点自然同时在物体的边界上。Papadopoulos 等人 [150] 提出了极值点的交互模式，认为极值点相比于包围框有着更多的信息，并用来增强基于包围框的 GrabCut 类方法。Maninis 等人 [21] 提出了基于四个极值点交互的使用神经网络模型进行分割的方法。Liew 等人 [46] 为了针对细小结构物体进行分割，合成了细小结构物体的数据集 ThinObject-5K，并基于极值点交互提出了 TOS-Net，额外增加了高分辨率流来获取更精细的分割。Wang 等人 [151] 以端到端的方式将强大的卷积神经网络模型与水平集优化相结合，利用极值点交互进行分割。针对医学图像，Khan 等人 [152] 使用极值点导出的置信度图进行分割。Girum 等人 [153] 使用弱监督的深度学习训练方法，并采用极值点交互来分割医学目标。

Mortensen 等人 [154, 155] 提出了智能剪刀，它是一种类似套索的工具。用户首先选择目标边缘上的一个点，然后随着边缘移动，就会得到自动贴紧目标边缘的线作为物体轮廓。Mishra 等人 [156] 则针对医学图像提出了强化的智能剪刀。

外轮廓线交互指的是用户提供包围目标轮廓的外轮廓线，以生成目标轮廓。用户可以直接绘制外轮廓线，比如 Pizenberg 等人 [157] 提出了包围轮廓线的交互模式，以适用于在触摸设备上进行交互式分割。针对医疗图像分割，主动轮廓模型 [158] 是一种常用的基于该交互的经典模型，通过优化轮廓线内外能量进行轮廓线收缩优化，以得到最终结果。用户还可以交互设置控制点来绘制线 [159] 以生成外轮廓。Zhou 等人 [160] 在此基础上还引入了多尺度曲线编辑。Karasev 等人 [161] 则提出了基于水平集偏微分方程控制的主动轮廓模型。

2.3.3 基于混合标注的交互式分割

基于混合标注的交互式分割指的是同时应用前背景标注和边界标注的方法，以及一些不属于这二者的交互方法，下面将详细介绍。

基于前背景标注和边界标注的协同方法有许多。Li 等人 [162] 提出了 Lazy Snapping 方法。其在前背景上绘制涂鸦，使用基于分水岭算法的初始分割进行图割来分割目标并生成轮廓多边形，可以在轮廓多边形上添加删除顶点或者修改顶点来调整分割结果。Spina 等人 [163] 提出了一种结合了边界跟踪和区域划界的被称为实时标记的混合交互范式。Benenson 等人 [22] 提出了一个结合了包围框和点击的交互式分割框架并标注了大规模图像数据集（Open Images）。Majumder 等人 [164] 利用初始交互点分割目标后，使用边界点替代前背景交互点作为修复点。Zhang 等人 [30] 将包围框和框内点击相结合，以提高目标物体定位和分割的准确性。在医学图像上，Jones 等人 [165] 使用了边界点击加区域涂鸦的方法。Zhou 等人 [166] 基于自动分割的结果，使用选择工具添加或擦除大块区域，使用笔刷修复精确边界，使用调节工具来膨胀和收缩边界。Zhou 等人 [167] 提出了可以使用包围框、区域涂鸦、极值点这三种交互的体记忆网络框架。Gong 等人 [56] 提出 PIMedSeg，使用前背景交互点加边缘涂鸦来进行分割。

除此之外还有一些特殊的交互模式。Liu 等人 [168] 使用了矩形框的交互来进行基于水平集的方法。该交互不同于包围框，只需要在初始分割或者修复时候覆盖主要区域就行。Agustsson 等人 [169] 在全图使用多组极值点分割出全图目标，再使用带方向的涂鸦来修改边界。Han 等人 [47] 则提出了细小剪刀，通过在细小区域上进行覆盖性涂鸦，利用合成的背景分割细小物体。

2.4 本章小结

本章主要针对该论文的相关工作进行了详细介绍。首先，本章介绍了自动类图像分割任务，该任务是交互式图像分割任务的基础。然后，本章介绍了交互类图像任务，主要描述了基于用户绘制操作的交互任务，其交互模式与本文的交互模式相似。最后，本章详细介绍了交互式图像分割任务，根据交互模式的不同，分别介绍了基于前背景标注、边界标注、混合标注的具体工作。这些工作都对该领域起到了促进作用。但面对复杂场景中的物体，交互式图像分割仍然存在目标定位不准确、局部区域精度低、细小结构交互难、医学图像分割差这些研究问题。本文对这些研究问题展开了进一步的探索，后续章节将一一进行详述。

3 基于初始交互点注意力的交互式分割

对于交互式图像分割，最重要的就是目标的整体分割。要想获得优异的整体分割结果，则需要对目标进行准确地定位。面对复杂场景中的目标定位不准确这一难点，以针对目标定位的全局物体分割为目标，本章提出了基于初始交互点注意力的交互式分割。该方法将初始交互点与其他交互点区别开，充分挖掘了初始交互点对于目标定位的作用，使得物体整体分割获得更好的质量。实验证明，本章提出的方法对于交互式分割任务中针对目标定位的全局物体分割具有显著作用。在本章中，首先，章节3.1对该工作的背景、动机、贡献等进行了介绍。其次，章节3.2详细描述了该工作提出的初始交互点注意力网络、交互点损失函数、结构完整性策略等。然后，章节3.3描述了实验设置，进行了消融实验，并结合其他方法进行了性能对比与分析。最后，章节3.4对该工作进行了总结。

3.1 本章引言

交互式图像分割的目的是用较少的用户输入来分割出感兴趣的目标物体。它对于许多任务都有实际作用，如图像编辑 [162] 和医疗图像分析 [52] 等。近年来，随着数据驱动的深度学习技术的普及，在某些领域，对于像素级别标注的需求急剧增加，如显著性物体检测 [77, 170]、语义分割 [61]、实例分割 [64]、伪装物体检测 [171] 等等。因此许多工作迫在眉睫地需要高效的交互式图像分割技术以减轻纯人工标注所带来的时间成本和经济成本。而对于交互式图像分割最重要的就是分割出定位准确的主体掩膜。因此，越来越多的研究者正在这一方向上进行广泛的探索，以便以最小的交互有效地分割出目标主体。

在交互式分割中，许多形式的交互模式都被实践过，如包围框 [16, 145]、点击 [19, 33, 36, 38, 42] 和涂鸦 [17, 111]。包围框的交互模式是一种应用广泛、便捷的方法。然而，在大多数情况下，用户通常需要对分割结果进行进一步的修正，而该交互难以满足此种需求。因此，更实用的方法是基于前背景的点击或涂鸦，通过迭代标记错误区域，进一步提高分割结果。点击与涂鸦相比，因为不需要拖动的过程，所以对用户的负担更小。图 3.1展示了前背景点击这一交互模式的典型工作流程，表述如下：用户首先在目标对象上点击一个前景点来获得初始

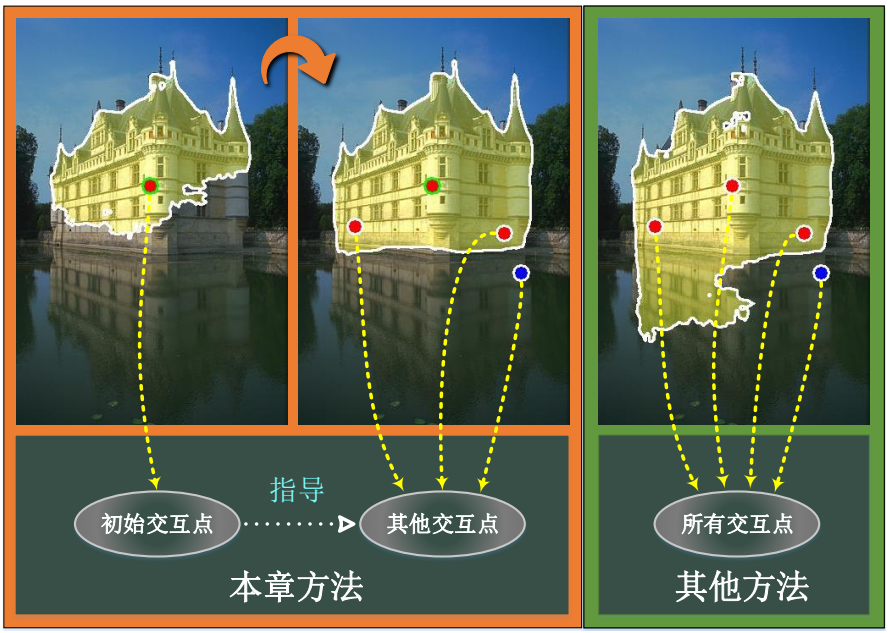


图 3.1　初始交互点在本章方法中的关键作用。该方法利用初始交互点作为定位锚点来指导其他交互点进行分割修复，而其他的交互式分割方法对所有交互点进行了不加区分的处理。

分割结果，根据该结果，用户进一步在图像上点击一个前景点或背景点对分割结果进行修复，不断循环这个过程，直到分割结果满足用户的需求。

许多传统的以及基于深度学习的方法已经在这个方向上探索了许久。对于大多数现有的工作，它们不加区别地使用所有的交互点来生成最终的预测结果。然而，本章观察到并非所有交互点都具有相同的分割效果。从图 3.1可以看出，只要点击一次，初始分割结果已经相当不错。相反，其他交互点的作用主要是在初始交互点的基础上实现更多细节的分割。因此，初始交互点更有利于获取对象的定位信息和整体信息，而其他交互点则侧重于细节修复。如表 3.1所示，本章使用其中一种交互式分割方法 [36] 来收集了 2000 例关于真实用户交互的统计数据。本章发现初始交互点（即第一个交互点）在交互式图像分割中起着重要的作用。首先，初始交互点的性能改善非常明显，并且初始交互点通常靠近目标对象的中心。结合之前描述的工作流程，通过直观的观察分析可以得到，初始交互点极其重要，可以作为目标对象的位置指示和全局信息指导。基于以上分析，本章推测特殊处理初始交互点将有利于交互式图像分割方法。

本章工作首先将这两种交互点区别对待，并提出了一个初始交互点注意力网络（FCA-Net），在该网络中构造了一个简单的辅助分支来进一步验证本章的

表 3.1 真实用户的交互点统计数据。性能提升：加入不同交互点后的平均交并比的性能提升。中心程度：描述交互点靠近物体中心的程度（只统计前景交互点）。中心程度越高代表交互点越靠近中心。具体计算细节在章节3.2.5中提及。

第 N 个交互点	1	2	3	4	5	6	7	8	9	10
性能提升	.751	.076	.045	.027	.020	.017	.015	.015	.009	.010
中心程度	.769	.312	.243	.207	.201	.211	.189	.188	.178	.186

猜想。在该网络中，本章使用初始交互点作为侧边输入来监督全局分割。利用初始交互点作为锚点来进行交互式分割，可以更好地引导目标对象的位置和主体信息。预测结果展示出集中在初始交互点周围的区域可以得到更好的分割结果。对于网络训练，本章提出了一种改进的损失函数，它考虑了用户提供的所有交互点，并将分割重心集中在交互点周围的这些区域。最后本章提出了一种新的后处理策略，可以有效地去除一些小的预测错误区域，并保持分割对象的结构完整性，这将更有利于目标整体的分割。本章在 GrabCut [16]、Berkeley [172]、PASCAL VOC [173]、DAVIS [174] 和 MSCOCO [175] 五个数据集上进行了全面的实验，取得了领先的性能。消融实验、对比实验的结果和分析证明了初始交互点的重要性以及本章提出的方法的独特性和有效性。

该工作的贡献可以总结如下：

1. 这是第一个展示初始交互点关键作用的工作。该工作还提出了初始交互点注意力网络（FCA-Net），它包含了一个简单而有效的模块来充分利用初始交互点提供的目标定位和全局分割的指导信息。
2. 提出了一种基于交互点的损失函数以及一种结构完整性策略，有助于掩膜结果具有更优异的分割性能和更整体的分割效果。
3. 五个数据集上的实验结果证明了初始交互点的重要性和提出的 FCA-Net、交互点损失函数和结构完整性策略的有效性。

3.2 交互框架与网络模型

本章节包括五个部分。首先，该章节介绍了提出的特殊处理初始交互点的 FCA-Net 网络，它的结构展示在图 3.2中。为了更好地说明初始交互点的有效性，该模型没有对交互式分割网络的结构做太多的改变。取而代之的是，一个简单的称为初始交互点注意力模块的附加模块被添加到基本分割网络中。因此，

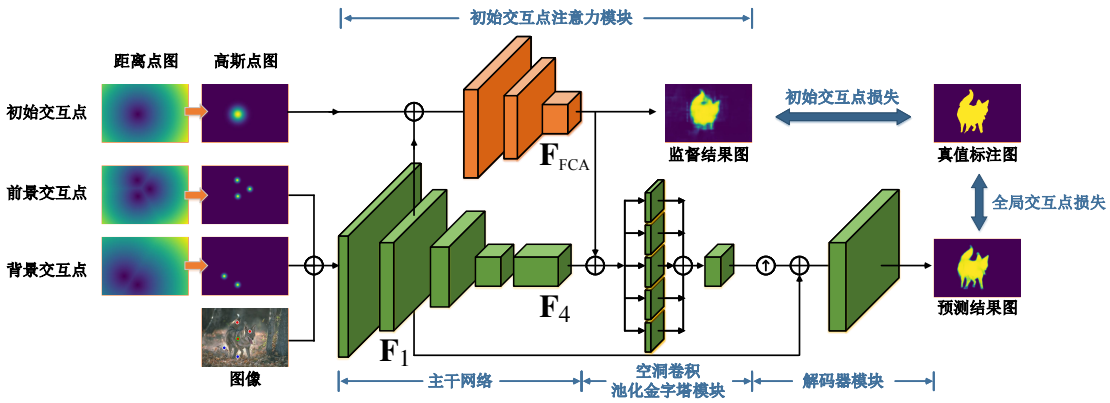


图 3.2　FCA-Net 方法的网络结构图。绿色部分显示了基础分割网络部分，包括主干网络、空洞卷积池化金字塔模块和解码器模块。橙色部分显示了该网络核心的初始交互点注意力模块。符号“⊕”和“↑”分别表示拼接和上采样操作。

FCA-Net 可以分为章节3.2.1中的基本分割网络和章节3.2.2中的初始交互点注意力模块。然后，章节3.2.3描述了提出的基于交互点的损失的计算过程，以帮助提出的交互式分割网络获得更好的性能。随后，章节3.2.4阐述了用于后处理的结构完整性策略。最后，章节3.2.5详述了交互点模拟策略的实现细节。

3.2.1　基础分割网络

同 [19, 33, 36, 38, 42] 一样, 该模型的基础网络为常见的 FCN [61] 架构网络，其结构与 DeepLab v3+ [62] 相似。如图 3.2所示，它包含三个模块：主干网络、空洞卷积池化金字塔模块和解码器模块。该模型采用 ResNet-101 [176] 主干网络作为特征提取器。该网络的后四层的特征定义为 $\{\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3, \mathbf{F}_4\}$。为了捕捉交互式分割中的多尺度物体，该模型同样在 ResNet-101 的最后一层采用了空洞卷积而不是采用步长为 2 的采样策略。因此，网络的输出步长为 16。主干网络的输入是 RGB 彩色图像与两个基于交互点的高斯点图的拼接。如图 3.2所示，高斯点图是通过欧式距离点图计算而来。该框架中的高斯点半径设置为 10。

如图 3.2所示，空洞卷积池化金字塔模块的输入是拼接的特征（$\mathbf{F}_4 \oplus \mathbf{F}_{\text{FCA}}$），其中 $\oplus$ 代表拼接操作，$\mathbf{F}_{\text{FCA}}$ 代表初始交互点注意力模块的输出。拼接后的特征被输入进四个尺寸分别为 1、6、12、18 的空洞卷积层，以及一个全局池化层。接着这 5 路输出的特征被拼接后输入到一个额外的卷积层。如图 3.2中所示的解码器模块，它将底层特征 $\mathbf{F}_1$ 和空洞卷积池化金字塔模块的输出特征作为输入并

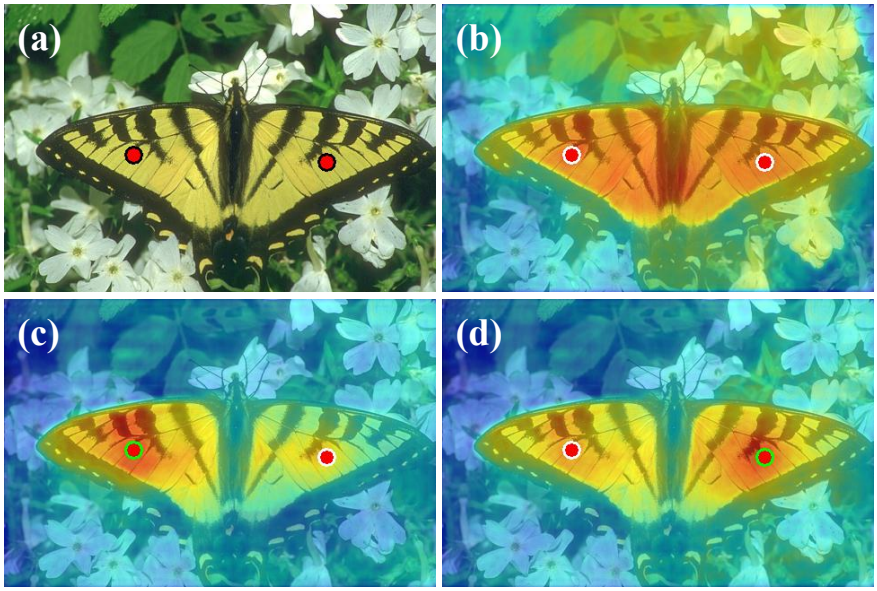


图 3.3 初始交互点注意力的可视化。(a) 是原始图像；(b) 是没有引入初始交互点注意力得到的预测可视图；(c) 和 (d) 是初始交互点注意力分别作用在左翅和右翅得到的预测可视图。

使用卷积层来获得最终的预测结果。为了对预测结果进行监督，该框架设计了一个基于交互点的损失函数去替换传统的二值交叉熵损失函数。这个被称作全局交互点损失，具体细节在章节3.2.3中有详细的描述。

3.2.2 初始交互点注意力模块

为了更好地利用初始交互点信息，该模型在基础分割网络之外设计了一个简单的模块，称作初始交互点注意力（First Click Attention，FCA）模块。它使用底层特征 $\mathbf{F}_1$ 和基于初始交互点的高斯点图 $\mathbf{M}_f$ 作为输入。这些拼接的特征（$\mathbf{F}_1 \oplus \mathbf{M}_f$）被输入进 6 个 3×3 的卷积层。在第 1 层和第 4 层该模块使用步长为 2 来降低分辨率。前 3 层的通道数是 256，后 3 层的通道数是 512。因此，输出的特征 $\mathbf{F}_{\text{FCA}}$ 拥有 512 个通道，它将在空洞卷积池化金字塔模块前被融合进基础分割网络。除此之外，本章使用一个初始交互点损失来监督 $\mathbf{F}_{\text{FCA}}$，它会重点关注初始交互点周围的像素，损失函数的相关细节在章节3.2.3中有详细的描述。

为了更好地说明初始交互点注意力的效果，在图 3.3中，本章分别使用初始交互点注意力模块（c-d）和不加该模块（b）对模型的预测图进行可视化。值得注意的是，在这三个测试样本（b-d）中，这些前景点的坐标完全一致。如图 3.3（b）所示，在没有初始交互点注意力模块的情况下，这两个前景交互点具有差不

多相同的重要性。通过引入初始交互点注意力模块（c-d），模型的注意力转移。在测试样本（c）和（d）中，用户标记前景交互点的顺序不同。可以看到无论它在哪里，第一次点击会引起更多的关注以作为分割的锚点，其余点击则更多起到辅助作用进行细节修复。与将所有交互点同等处理相比，初始交互点注意力模块的引入使模型更符合章节3.1中讨论的用户实际交互行为。

3.2.3 交互点损失

为了在下文中更好地进行说明，本章在这里定义了一些符号和操作。所有的像素被表示作 $\mathcal{G}$。本章使用 $\mathcal{G}_p$ 和 $\mathcal{G}_n$ 来表示根据真值标注图得到的前景像素点和背景像素点。$\mathcal{A}$ 表示所有交互点。$\mathcal{A}_p$ 和 $\mathcal{A}_n$ 分别表示所有的前景交互点和背景交互点。本章使用 $d(p_1,p_2)$ 来表示点 p_1 到点 p_2 的欧氏距离。$\phi(p,\mathcal{S})$ 函数用来表示点 p 到一个区域 $\mathcal{S}$ 的最短距离，它的定义如下：

$$\phi(p,\mathcal{S}) = \min_{\forall p_s \in \mathcal{S}} d(p,p_s). \tag{3.1}$$

在二值分割任务中，二值交叉熵通常被作为损失函数来监督卷积神经网络的输出结果。该损失函数有利于关注图像的全局分割质量，然而交互式分割任务则更希望看到用户提供的交互能够同样起到监督作用。最好在这些交互点处及其周围能得到更为准确的结果。因此，本章设计了一个基于用户交互点的损失函数来帮助提出的 FCA-Net 网络模型获得更好的性能。

此交互点损失可以被视作一种特殊的二值交叉熵损失。传统的二值交叉熵损失函数可以表示如下：

$$\ell(p) = -\left(y_p \log(x_p) + (1-y_p)\log(1-x_p)\right), \tag{3.2}$$

其中 x_p 代表点 p 在预测结果图中的概率值，而 y_p 代表点 p 在真值标注图中的标签，该标签为 0 或者 1。

本章预先定义了一个函数 ψ 来表示点 p 和一个交互点集 $\mathcal{S}$（如 $\mathcal{A}_p$ 和 $\mathcal{A}_n$）的距离，其计算如下：

$$\psi(p,\mathcal{S}) = 1 - \frac{\min\left(\phi(p,\mathcal{S}),\tau\right)}{\tau}, \tag{3.3}$$

其中 τ 是每个交互点的影响范围。

对于监督预测结果图的损失函数，本章提出了一个考虑了所有交互点的全局交互点损失 $\mathcal{L}_g$。它的计算如下：

$$\mathcal{L}_g = \frac{1}{N}\sum_{p\in\mathcal{G}}(\widehat{w}_p \cdot \ell(p)), \tag{3.4}$$

其中 N 是像素个数。公式 3.4中的权重 $\widehat{w}_p$ 可以表示如下：

$$\widehat{w}_p = \begin{cases} \alpha + \psi(p,\mathcal{A}_p)(\beta-\alpha), & y_p = 1 \\ \alpha + \psi(p,\mathcal{A}_n)(\beta-\alpha), & y_p = 0 \end{cases}, \tag{3.5}$$

其中 α 和 β 用来调整损失的范围。

对于监督初始交互点注意力模块输出的损失函数，本章提出了一个初始交互点损失 $\mathcal{L}_f$，它会集中关注初始交互点周围的区域。它的计算如下：

$$\mathcal{L}_f = \frac{1}{N}\sum_{p\in\mathcal{G}}(\widetilde{w}_p \cdot \ell(p)). \tag{3.6}$$

公式 3.6中的权重 $\widetilde{w}_p$ 可以表示如下：

$$\widetilde{w}_p = \alpha + \psi\left(p,\{a_f\}\right)(\beta-\alpha)y_p, \tag{3.7}$$

其中 a_f 代表 $\mathcal{A}_p$ 中的初始交互点。

在该实验中，τ 被设置成 100，α 被设置成 0.8，β 被设置成 2.0。

3.2.4 结构完整性策略

通过一些实验发现，神经网络的预测很可能包含一些错误分割的分散区域。在大多数情况下，在交互式分割任务中，用户更希望得到保持结构完整性的对象结果。结构完整性指的是物体的分割掩膜最好不存在许多零散的错误分割。因此，本章提出了一种基于交互点的后处理策略，即结构完整性策略。

通常情况下，用户会以 0.5 作为阈值，从神经网络的输出中得到最终的二值化预测。让 $\mathcal{P}$ 表示这些预测为前景的点，本章将根据交互点对这些预测区域进行后处理，得到新的 $\mathcal{P}'$，其计算如下：

$$\mathcal{P}' = \{p \in \mathcal{P} \mid \exists_{a\in\mathcal{A}_p}\sigma(p,a) = 1\}, \tag{3.8}$$

其中，当存在一条点 p_1 到点 p_2 的八连通路径时，$\sigma(p_1,p_2) = 1$。图 3.4展示了该策略的直观步骤的示意图。该结构完整性策略可以在大多数情况下起到一定效果。它所带来的性能提升可以在表 3.3中看到。

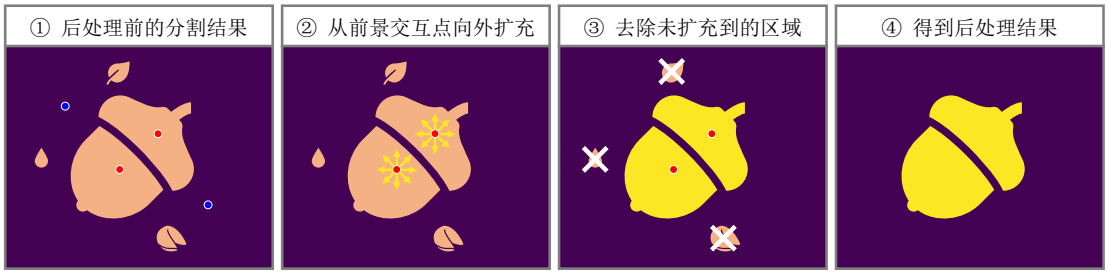


图 3.4 结构完整性策略的示意图。从每个前景交互点向外进行洪泛法填充，将未被填充到的零散区域去除，就得到了使用结构完整性策略后的分割结果。

3.2.5 交互点模拟

由于分割数据集不存在用户交互的标注，本章同其他论文类似，采用一些模拟策略来生成各类交互点，其中包括全局交互点和初始交互点。

全局交互点模拟。 对于大多数交互点，本章使用与 [19] 相似的策略。前背景交互点数量分别在 [1, 10] 和 [0, 10] 之间。对于前景交互点，它们来自前景，远离物体边界 P_1 像素以上并且它们之间相距 P_2 像素以上。本章定义 $\mathcal{A}^*$ 为先前设置的交互点集合，一个新的前景交互点来自一个候选点集 $\mathcal{C}_p$，表示如下：

$$\mathcal{C}_p = \{p \in \mathcal{G}_p \mid \phi(p, \mathcal{G}_n) > P_1,\ \phi(p, \mathcal{A}^*) > P_2\}. \tag{3.9}$$

对于背景交互点，它们来自背景，远离物体边界 $N_1 \sim N_2$ 像素并且它们之间相距 N_3 像素以上。一个新的背景交互点来自一个候选点集 $\mathcal{C}_n$，表示如下：

$$\mathcal{C}_n = \{p \in \mathcal{G}_n \mid N_1 < \phi(p, \mathcal{G}_p) < N_2,\ \phi(p, \mathcal{A}^*) > N_3\}. \tag{3.10}$$

在本章的实验中，P_1 来自集合 $\{5, 10, 15, 20\}$，P_2 来自集合 $\{7, 10, 20\}$，N_1 来自集合 $\{15, 40, 60\}$，N_2 来自集合 $\{80\}$，N_3 来自集合 $\{10, 15, 25\}$。

初始交互点模拟。 初始交互点总是来自目标物体内部，它通常靠近物体中心。因此本章使用 $\mathcal{E}(p)$（在表 3.1中被称作中心程度）来表示点 p 距离物体中心的程度，它的计算公式如下:

$$\mathcal{E}(p) = \frac{\phi(p, \mathcal{G}_n)}{\max_{\forall p_0 \in \mathcal{G}_p} \phi(p_0, \mathcal{G}_n)}. \tag{3.11}$$

$\mathcal{E}(p)$ 越接近 1 代表初始交互点越靠近物体中心。本章实验选择裁剪后的图像中 $\mathcal{E}(p)$ 为 1 的点作为初始交互点，并且将它的高斯半径设置成一般交互点的三倍。

3.3 实验结果与分析

本章节包括三个部分。首先，章节3.3.1介绍了该方法的实验设置，包括使用的数据集、评测指标、实现细节和模型推理。然后，章节3.3.2介绍了该方法的消融实验，包括最重要的初始交互点注意力模块和其他辅助部分的消融实验。最后，章节3.3.3介绍了该方法与其他方法的平均交互点数对比、NoC-IoU 曲线对比，还结合具体样例进行了一些本方法的局限性分析。

3.3.1 实验设置

数据集。 本章采用了以下广泛使用的数据集进行评测：

- **GrabCut [16]**：该数据集包含 50 幅图像，在大多数交互式图像分割方法中使用。大多数图像的前景和背景有明显的差异。
- **Berkeley [172]**：该数据集包含 96 幅图像上的 100 个对象。由于前景和背景的相似性，在这个数据集中有些图像很难进行分割。
- **PASCAL VOC [173]**：该数据集中的验证集将用于进行评测，该验证集包含 1449 个图像和 3427 个实例。本实验使用这些实例级对象掩膜进行性能评测。这些对象与用于训练的数据在语义上是一致的。
- **DAVIS [174]**：该数据集常被用于视频对象分割。它包含 50 个视频，并有着高质量的掩码标注。本实验采用和 [38] 同样的 10% 的帧用来评测。
- **MSCOCO [175]**：该数据集包含 80 个类别的对象。同 [19, 42] 一样，本实验根据与训练数据类别是否一致将此数据集分为 MSCOCO（seen）和 MSCOCO（unseen）两部分，并为每个类别抽取 10 张图像进行性能评测。

评测指标。 同 [19, 31, 33, 34, 36, 38, 42] 一样，本章方法使用数据集上平均的交并比（Intersection over Union，IoU）作为评测指标，还同样使用一个机器人用户来模拟评测中的交互点击。首先，初始交互点无疑是目标对象的一个前景点，网络模型将得到一个基于初始交互点的预测图。然后在最大错误区域的中心选取下一个模拟交互点。本章绘制了点击次数和平均交并比的曲线，以比较每种方法在固定交互下的性能。本章采用数据集上平均的交互点数（Number of Click，NoC）作为评估指标，它反映了在数据集的每个样本上获得特定 IoU 阈值的平均交互效率。对于每个数据集，IoU 阈值的选择是不同的每个样本的默认最大交互次数限制为 20 次。以上设置与之前的工作保持一致。

表 3.2 FCA-Net 方法的消融实验。该实验在 Berkeley 和 PASCAL VOC 两个数据集上进行，以 NoC 为评测指标。BS: 基础分割网络; BS2: 以 Res2Net [180] 为主干网络的基础分割网络; FCA: 初始交互点注意力模块; CL: 交互点损失; Iter: 迭代训练。

#	FCA-Net 消融配置	Berkeley	PASCAL VOC
1	BS	5.74	4.21
2	BS + FCA	5.22	3.66
3	BS + FCA + CL	4.94	3.33
4	BS + FCA + CL + Iter	4.23	2.98
5	BS2 + FCA + CL + Iter	3.92	2.79

实现细节。 本章实验使用增强的 PASCAL VOC 数据集（PASCAL VOC [173] 合并上 SBD [177] 再去除 PASCAL VOC 验证集后的数据集）的 10582 个训练图像对 FCA-Net 进行训练。实际上，最终得到的 25832 个实例级的样本和相应的真值标注图被用于训练。首先，该实验会使输入图像的短边固定为 512 像素，并按等比例调整图像大小。然后，再随机裁剪 512×512 像素的区域，且保证裁剪后的图像至少包含对象的一部分。该实验采用了相同的迭代训练策略 [31,33] 进行训练过程中的交互点模拟。该实验用在 ImageNet [178] 上预训练的 ResNet-101 为主干网络。网络模型的批大小被设置为 8。该实验设定主干网络的初始学习率为 7×10^{-3}，其他部分为 7×10^{-2}，并采用动量值为 0.9 的随机梯度下降方法进行参数优化。训练过程先采用多项式学习率衰减策略进行 30 个周期的训练，尾部额外添加采用恒定学习速率的 3 个周期。该方法中所有的实验都是采用 PyTorch [179] 深度学习框架实现的，并都在单个 NVIDIA Titan XP GPU 上运行。

模型推理。 本实验在 Intel i7-8700K 3.70GHz CPU 和单个 NVIDIA Titan XP GPU 上测试推理时间。在 512×512 的图像上进行每次点击大约需要 0.07 秒。由于交互式分割过程中需要用户的参与，这个速度足够满足实时交互的需要。

3.3.2 消融实验

如表 3.2所示，本章在 Berkeley 和 PASCAL VOC 两个数据集上进行消融实验。该实验以基础分割网络为基线模型（No.1），并逐步添加上本章提到的相关内容（No.2-5）。各消融配置下的 NoC 指标展示在表中，其中 Berkeley 数据集是以 90% 的 IoU 为阈值，PASCAL VOC 数据集是以 85% 的 IoU 为阈值。

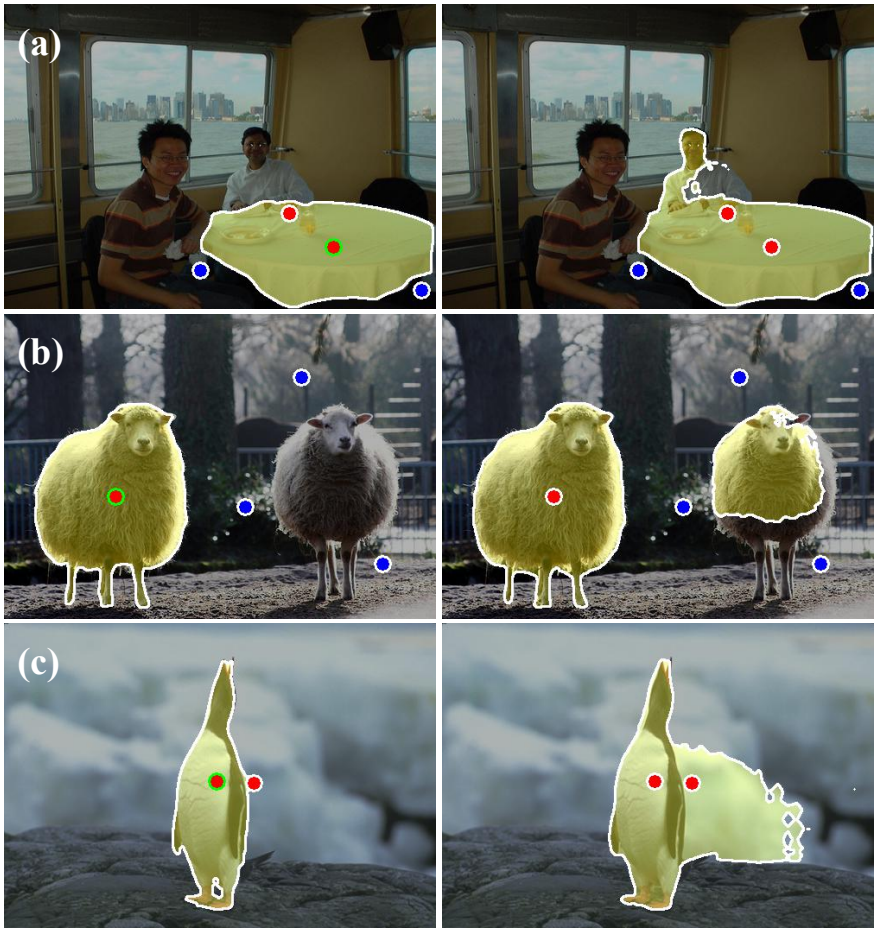


图 3.5　初始交互点注意力的优点展示。左右列分别显示了是否带有 FCA 模块的预测结果。

引入初始交互点注意力模块。　与基线模型比较，加入 FCA 模块后，两数据集上 NoC 降低了 0.52 和 0.55，性能得到了显著提高。这一改进符合本方法的期望，通过引入 FCA 模块，模型可以更有效地利用第一次点击的引导信息。初始交互点注意力真的能提高分割结果吗？图 3.5中的一些可视结果说明了添加初始交互点注意力模块的一些优点。（1）聚焦不变能力。在大多数方法中，所有的前背景交互点都是同等处理的。它们将所有交互点作为输入，以生成最终结果。初始交互点外的其他交互点往往被用来修复局部细节，并且可能靠近目标对象的边界。如果神经网络将这些交互点同等对待，往往会导致错误的分割。例如，在图 3.5（a）中，用户想要分割带有白色桌布的桌子，初始交互点靠近桌子的中心，另一个前景交互点用于修复桌子边缘附近的错误分割。如果没有初始交互点引导，网络模型对每个交互点平等对待，因此会错误地分割出图像中的人。在

初始交互点的帮助下，错误的分割将会大大减少。（2）位置指导能力。初始交互点指导了目标对象的位置。如果场景中有多个物体，那么利用初始交互点可以减少局部区域的错误分割。例如，在图 3.5（b）中，用户想要分割左边的羊。如果在右边的羊周围处有三个背景交互点，由于没有对全局位置信息的准确理解，神经网络可能会误认为在这些背景点包围的区域中有一个目标对象，这可能会导致对右边的羊的错误预测。有了初始交互点的帮助，预测结果就会集中在初始交互点的位置附近，网络则可以得到更准确的结果。（2）容错能力。在交互式分割过程中，不可避免会出现一些点击错误，特别是在目标边缘或背景与前景相似的区域。例如，在图 3.5（c）中，用户想要分割企鹅。右侧靠近对象边界的前景交互点意外地落入了背景区域。如右图所示，可以看到，如果不使用初始交互点注意力，这可能会导致严重的分割错误。而在初始交互点注意力的引导下，如左图所示，这些交互错误产生的影响将大大减小。

引入其他辅助部分。 在表 3.2中，比较 2 号和 3 号实验，可以看到，本章提出的交互点损失带来了性能提升。4 号实验表明，本方法采用的同样的迭代训练策略 [31, 33]，在一定程度上提高了最终的模型效果。由于本章所提出的 FCA-Net 只是一个简单的实现来探索初始交互点的关键作用，FCA-Net 模型并未对广泛使用的基础分割模型进行过多的修改。因此，在实践中，可以通过更换更有效的主干网络或更复杂的设计结构来获得更好的结果。例如，5 号实验使用 Res2Net [180] 代替 ResNet [176] 作为主干网络，进一步提升了模型性能。最后，本实验使用提出的结构完整性策略对结果进行后处理，并在表 3.3中展示它的性能结果。可以发现，该策略有时候可以进一步提升整个数据集上的性能指标。

3.3.3 性能分析

本章节将 FCA-Net 的实验结果与其他传统方法和深度学习方法进行了比较，包括了章节2.3.1中提及的 GraphCut (GC) [17]、GrowCut (GRC) [113]、Random Walk (RW) [18]、Geodesic Matting (GM) [15]、Euclidean Star Convexity (ESC) [109]、Geodesic Star Convexity (GSC) [109]、Deep Object Selection (DOS) [19]、Regional Image Segmentation (RIS) [42]、Latent Diversity (LD) [36]、Backpropagating Refinement Scheme (BRS) [38] 和 Content-aware Multi-level Guidance (CMG) [33]。实验结果展示在表 3.3和图 3.6中。部分数据来自 [19, 36, 38, 42] 的文中结果。最后，本章节还针对 FCA-Net 方法进行了一定的局限性分析。

表 3.3 FCA-Net 方法和其他方法的 NoC 指标对比。该表展示了数据集上每个样本到达指定 IoU 阈值（%）的平均交互点数（NoC），其阈值用（@XX）表示。SIS 表示使用了结构完整性策略进行后处理。FCA-Net* 表示该模型使用 Res2Net [180] 作为主干网络。

方法	GrabCut @90	Berkeley @90	PASCAL VOC @85	DAVIS @90	MSCOCO (seen)@85	MSCOCO (unseen)@85
GC [17] *ICCV01*	11.10	14.33	15.06	17.41	18.67	17.80
GRC [113] *POG05*	16.74	18.25	14.56	N/A	17.40	17.34
RW [18] *PAMI06*	12.30	14.02	11.37	18.31	13.91	11.53
GM [15] *IJCV09*	12.44	15.96	14.75	19.50	17.32	14.86
ESC [109] *CVPR10*	8.52	12.11	11.79	17.70	13.90	11.63
GSC [109] *CVPR10*	8.38	12.57	11.73	17.52	14.37	12.45
DOS [19] *CVPR16*	6.04	8.65	6.88	12.58	8.31	7.82
RIS [42] *ICCV17*	5.00	6.03	5.12	N/A	5.98	6.44
LD [36] *CVPR18*	4.79	N/A	N/A	9.57	N/A	N/A
BRS [38] *CVPR19*	3.60	5.08	N/A	8.24	N/A	N/A
CMG [33] *CVPR19*	3.58	5.60	3.62	N/A	5.40	6.10
FCA-Net	2.24	4.23	2.98	8.05	4.49	5.54
FCA-Net (SIS)	2.14	4.19	2.96	7.90	4.45	5.33
FCA-Net*	2.16	3.92	2.79	7.64	4.34	5.36
FCA-Net* (SIS)	2.08	3.92	2.69	7.57	4.08	5.01

平均交互点数对比。 表 3.3显示了五个数据集上六个子集的 NoC 指标。本章提出的 FCA-Net 在五个数据集中达到了最好水平。对于经典的 GrabCut 和 Berkeley 两个数据集，FCA-Net 提升了 1 个以上的 NoC。对于与训练数据相同语义的 PASCAL VOC 数据集，NoC 也减少了 0.64。DAVIS 数据集上的提升有限，NoC 仅比最好的方法减少了 0.19。对比 MSCOCO 数据集中的见过（seen）和未见过（unseen）子集部分，可以发现见过部分性能更好，这也符合神经网络的拟合行为，当目标与训练数据类别一致时，神经网络的性能更好。在采用结构完整性策略对结果进行后处理后，性能将进一步提高。该方法在网络结构上并没有做太多的改变，只是设置了一个简单的初始交互点注意力模块。然而，效果的提高是显著的，这也间接反映了初始交互点的独特性。表中还展示了使用更强大的 Res2Net [180] 主干网络，性能还能进一步提升。

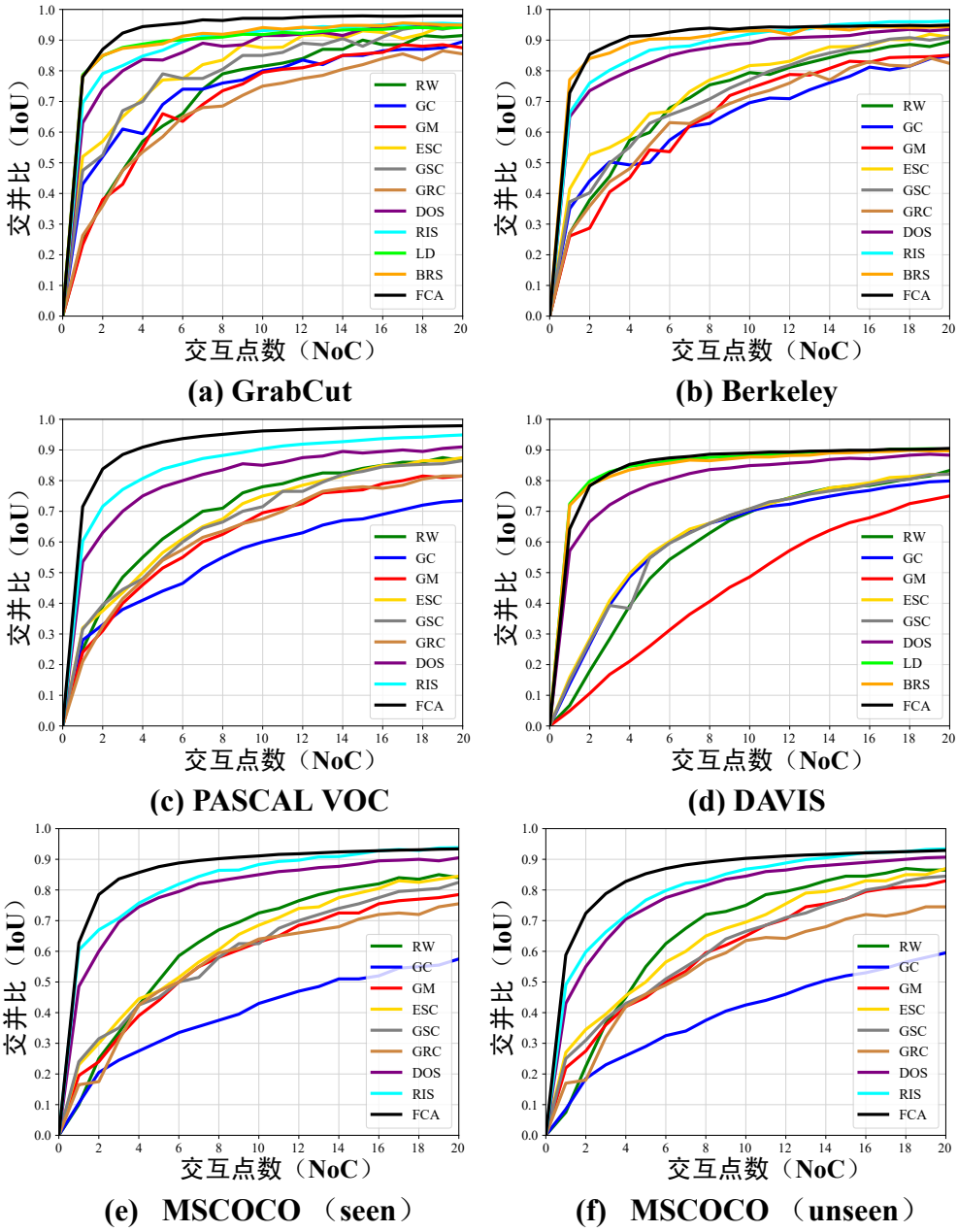


图 3.6　FCA-Net 方法和其他方法的 NoC-IoU 曲线图。图例中 FCA 表示 FCA-Net。

NoC-IoU 曲线对比。　图 3.6 展示了各个方法在不同交互点下的 IoU 指标。其中 FCA-Net 方法的曲线是根据没有使用结构完整性策略的结果绘制的。可以看出，在大多数情况下，本章提出的 FCA-Net 模型在初始交互点后的曲线优于其他方法。这也符合该方法的期望，以初始交互点作为主体引导，神经网络模型的预测会包含较少的错误区域，得到的结果将更加准确。

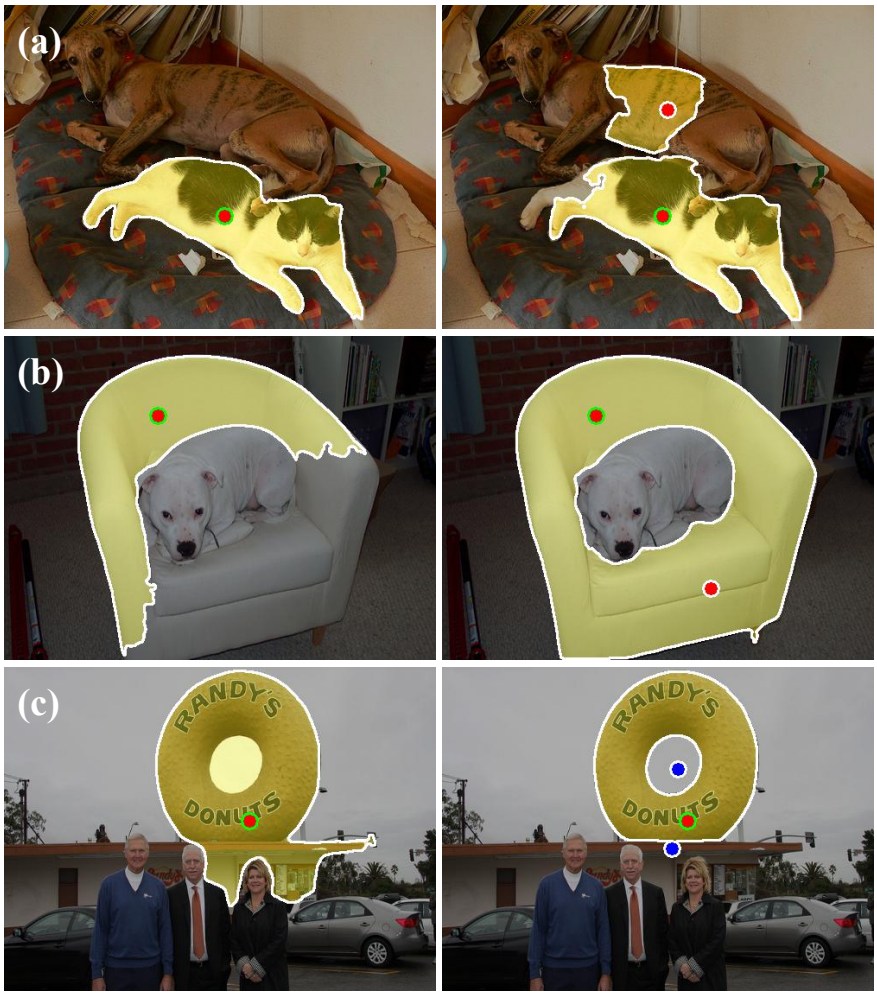


图 3.7　FCA-Net 方法可能存在的局限性示例。绿色圆环的点表示初始交互点。这里展示了三种情况：（a）分割多物体情况；（b）物体中心被遮挡情况；（c）物体中心不属于物体情况。

局限性分析。　图 3.7展示了 FCA-Net 在某些特殊情况下可能存在的局限性。如图 3.7（a）所示，由于初始交互点提供了很强的位置先验，本章的 FCA-Net 不擅长同时分割图像中的多个实例。不过，在实际应用中，通过初始交互点为每个实例对象添加标注，可以不受此限制约束。在图 3.7（b）中，沙发的中心被遮挡了。如果点击沙发中心，则会分割出沙发上的狗，从而导致错误分割。此时初始交互点不得不偏离中心，导致了过少的分割，而后需要添加额外的前景交互点来分割出目标。在图 3.7（c）中，目标是圆环状物体，物体中心是镂空的，并不属于物体。此时，同之前一样，用户只能在一侧添加初始交互点，模型对目标的错误估计导致了多余分割，需要添加额外的背景交互点来进行修复。

3.4 本章小结

针对交互式图像分割任务中，复杂场景下的目标定位不准确的问题，本章探讨并论证了初始交互点对于交互式图像分割中基于目标定位的全局分割的重要性。本章还提出了一个初始交互点注意力网络，命名为 FCA-Net。它在基本分割网络上增加了一个简单的初始交互点注意力模块，将更多的注意力转移到初始交互点上。此外，本章还提出了一种基于交互点的损失函数和一种结构完整性策略来提升性能。五个数据集上的性能表明了初始交互点的重要性和该方法的优越性。有了该方法，复杂场景下的基于目标定位的物体整体分割能更加准确。

4 深入聚焦视角的交互式分割

对于交互式图像分割，当完成目标的整体分割后，需要更加关注目标的细节分割质量。用户要想获得优异的细节分割结果，则需要对目标局部进行精细化分割。面对复杂场景中的局部区域精度低这一难点，以针对精确细节的局部区域分割为目标，本章提出了深入聚焦视角的交互式分割。该方法从初始交互点之外的其他交互点的聚焦视角出发，关注这些交互点周围的局部分割细节，使得物体的分割获得更精细的结果。实验证明，本章提出的方法对于交互式图像分割任务中针对精确细节的局部区域分割具有显著作用。在本章中，首先，章节4.1对该工作的背景、动机、贡献等进行了介绍。其次，章节4.2详细描述了该工作提出的 FocusCut 框架以及聚焦区块模拟、聚焦范围计算、渐进式聚焦策略等。然后，章节4.3描述了实验设置，进行了消融实验，并结合其他方法进行了模型性能和可视结果的对比与分析。最后，章节4.4对该工作进行了总结。

4.1 本章引言

交互式图像分割旨在以较小的交互成本获得目标对象的精确二值分割掩膜，现已发展成为提供像素级数据标注和图像编辑必不可少的工具。近年来，随着大屏幕设备的增加和人们审美水平的提高，图像标注和图像编辑对更精细化的分割掩膜的需求量不断增加。在高精度交互式图像分割中，对于前背景点击的交互模式，边缘、孔洞等目标细节的精细化修复通常需要更多的交互点和交互时间。当用户在预测错误的区域添加交互点时，他们往往倾向于关注细节区域来实现更精细的修复。但是，目前的方法把之前的交互点考虑到一起来确定目标的全局预测。在新一轮交互中，一个所有交互点共同预测的过程可能会弱化新输入的交互点对其周围细节的决定性影响，并且返回不合意的结果。因此，如何更好地理解用户添加交互点的局部精细化修复意图，是一个值得研究的重点。

在分割任务中，局部信息已经被许多工作所利用。HAZN [181] 能够自适应地调整整体或部分物体的视角范围来优化分割结果。GLNet [182] 聚合了局部和全局分支捕获的特征图。此外，对于语义分割，AWMF-CNN [183] 分别为不同放大率的局部区块赋予权重。CascadePSP [184] 使用一个级联网络，将原始图像

图 4.1 FocusCut 方法的示意图。鹰爪在全局视角下的细节分割质量有限，该方法通过额外的聚焦视角对其进行精细化分割。用户提供的红色和蓝色的交互点分别表示交互式分割中的前景交互点和背景交互点。黄色的掩膜表示网络模型的预测结果。

中的局部图像区块输入到精细化模块中进行细节修复。相似地，MagNet [185] 以一种渐进的方式来优化不同尺度下局部区块的分割结果。但是，对于语义分割任务，许多工作都采用了滑动窗口策略，这不可避免地造成大量的计算与时间成本。由于交互式分割的特殊性，局部视角可以通过交互来决定，因此可以避免这个缺点。在交互式图像分割中，RIS-Net [42] 已经初步证明了局部优化的重要性。它通过为每个前景交互点寻找最近的背景交互点来生成局部区块。局部特征通过主要分支的局部区域池化层来提取。该主要分支以图像和交互点图的拼接结果作为输入。也就是说，局部优化依然受到整幅图像和其他交互点的影响，这会在某种程度上弱化局部交互点的主导作用。此外，由于网络的下采样操作，局部特征会存在丢失现象。本章方法希望在此基础上更进一步，采用了一种更纯粹的局部视角来进行分割优化，即直接将以每个交互点为中心的局部区块送入网络并且完全地忽略整个图像和其他远距离交互点的影响。

如图 4.1所示，为了实现更加精细的分割修复，本章深入一个交互点的视角，称之为聚焦视角，来考虑其周围的信息。本章通过设计了一个名为 FocusCut 的简洁流程框架来验证聚焦视角的重要性。该框架中，交互式图像分割网络的原始功能已经被改变，它被赋予了一个新的功能，除了能够分割目标对象外，还可以进行局部细节修复。具体而言，图像经过全局视角下的整体分割后，该框架

会从原始图像中裁剪出一个以新添加的交互点为中心的局部区块作为聚焦视角，并使用同一个网络来进一步精细化目标的细节，最后再粘贴回全局的粗糙分割结果中。裁剪范围会根据全局视角中的预测变化进行动态调整。之后，裁剪范围会依据本章提出的渐进式聚焦策略逐步减小。图 4.1中展示了一个样例，图中的鹰爪在全局视角下的分割是粗糙的，经过了聚焦视角下的精细化分割，细节得到了修复。为了公平地与其他方法做对比，并且证明本章观点的有效性，几乎没有额外的参数和特定的模块添加到交互式分割任务常用的网络结构中。本章方法在 GrabCut [16]、Berkeley [172]、SBD [177] 和 DAVIS [174] 四个数据集上开展的所有实验都证明了 FocusCut 这一流程框架的有效性。

该工作的贡献可以总结如下：

1. 在交互式分割任务中引入了聚焦视角，通过考虑交互点周围的局部精细化分割来充分理解用户提供交互点的修复意图。
2. 基于提出的聚焦视角，本章提出了 FocusCut，一种简单但有效的流程框架来进行交互点周围的局部精细化过程。
3. FocusCut 在不增加额外参数的情况下取得了优异的性能，四个数据集上的性能指标和可视结果反映了该框架在精细分割上的有效性。

4.2 交互框架与网络模型

本章节包括五个部分。首先，章节4.2.1回顾了经典的交互式分割流程框架。然后，章节4.2.2介绍了本章提出的 FocusCut 流程框架。最后，章节4.2.3、章节4.2.4和章节4.2.5 分别介绍了聚焦视角的模拟算法、范围计算和渐进式策略。

4.2.1 经典的流程框架

随着深度学习的发展，近年来大多数交互式图像分割研究是通过引入卷积神经网络来开展的。因为交互式分割可以被视为一种特殊类型的分割任务，很多方法是基于语义分割中以 DeepLab v3+ [62] 为代表的 DeepLab 系列网络来设计的。这个网络结构包含一个主干网络、一个空洞卷积池化金字塔模块和一个解码器模块。对于主干网络，在交互式分割中大多采用 ResNet [176]。空洞卷积池化金字塔模块包含四个空洞卷积分支和一个全局平均池化分支。解码器模块通过融合主干网络的低层次特征来优化空洞卷积池化金字塔模块的输出以生成最终的预测结果。对于交互式分割任务，输入还应该包含交互点的信息。交互

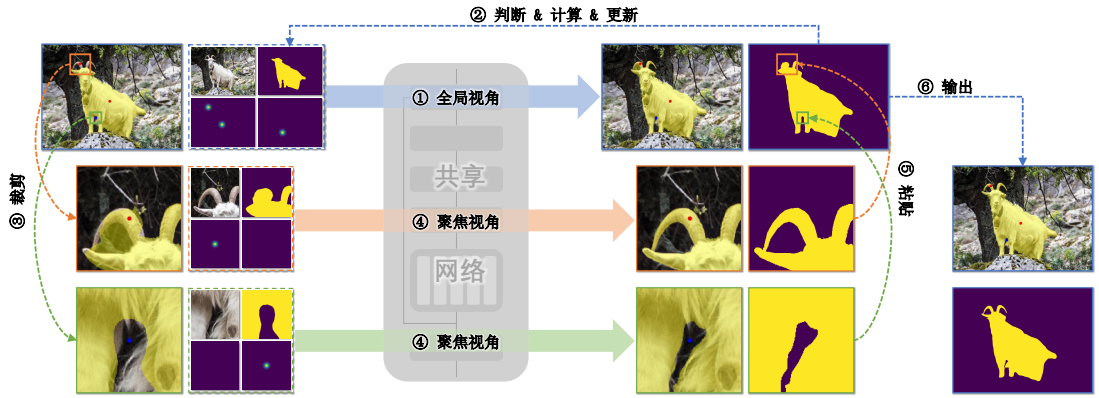


图 4.2　FocusCut 方法的流程框架图。该流程框架被划分为 6 个阶段：（1）将图像与交互生成的 6 通道图输入到共享网络中来生成全局视角的预测；（2）对于当前交互点，基于全局视角下的当前预测图和先前预测图的变化判断是否需要聚焦并且计算聚焦的范围，然后使用当前预测图更新先前预测图；（3）分别为每个不同聚焦范围的交互点从原始图像中裁剪对应区块；（4）将聚焦区块的 6 通道图输入到网络模型中生成聚焦视角下的局部区块预测；（5）将区块预测粘贴回全局预测中；（6）输出最终的预测结果图。

点位置将会被转换为两个交互点图，例如距离点图、圆盘点图或本工作使用的高斯点图，来分别表示前景交互点和背景交互点。交互式图像分割任务中的大多数工作修改了网络的输入部分，采用一个由 RGB 图像和两个交互点图拼接而成的 5 通道图作为输入。在具体实现时，可以添加额外的头部转化模块将一个 5 通道图编码为一个 3 通道图来满足标准的结构或者像本工作一样直接改变第一个卷积层的结构。预测结果图与真值标注图的二值交叉熵损失将用来监督网络模型的输入。最终，预测结果图会被二值化成目标对象的分割掩膜。

4.2.2　FocusCut 流程框架

在交互式分割的过程中，用户经常通过提供更多的前景交互点和背景交互点来修正错误的分割区域。随着交互点数目的增加，交互点会逐渐被用于修正更加局部的区域。特别是在较后的阶段，用户可能会使用许多交互点来修正一个小区域。由于感受野的尺寸和网络的下采样操作，模型很难同时分割出整个对象和细节区域。因此本章提出了深入聚焦视角的 FocusCut 流程框架。

图 4.2展示了 FocusCut 流程框架。该流程框架包含两个交互视图。一个是全局视角，对整个目标对象进行分割；另一个是聚焦视角，根据交互点附近的全局分割结果进行精细化分割。为了体现有效性，该方法尽可能不改变常用的网

络结构。该方法以输出步长为 16 的 DeepLab v3+ [62] 作为基础网络。不同的是，该流程框架将其视为共享网络，不仅学习整个目标对象的分割，还学习局部区域的精细化分割。为此，该框架需要统一这两个输入。由于聚焦视角中的精细化是基于粗糙掩膜生成的，该框架为输入添加一个额外通道来表示先前预测。该方法希望此网络除了目标对象的分割之外，还学习基于先前预测和交互点生成更准确的分割。为了实现这个目标，该框架将交替使用全局视角和聚焦视角的数据进行网络训练。对于全局视角，该框架同样会采用迭代训练策略 [31]。如果是迭代步骤，则将粗糙预测设置为先前的分割结果，否则，则设置为全 0 的空图。输入网络的 RGB 图像将包含整个对象，并且交互点将根据对象真值掩膜模拟生成，其中包含至少一个前景交互点来指示目标位置。在全局视角中，网络将此 6 通道图作为输入来生成整体预测。对于该框架的聚焦视角，则使用表示目标局部信息的区块样本来训练网络。本章将会在章节4.2.3中详细描述生成区块样本的过程。如图 4.2所示，这个阶段的输入图依然包含 6 个通道。但是，RGB 图像是从原始图像裁剪出来的局部区域，它们将不代表物体整体，而会更加注重细节。这些交互点图是从原始图像的交互点中裁剪区域得到局部交互点后生成的。与全局视角中的交互点图不同，裁剪区域的中心点必然存在一个前景交互点或背景交互点。该方法将通过算法处理裁剪区域的真值标注图以降低其精细度来生成粗糙分割。这些图将会被拼接到一起并且输入到网络中进行训练。

图 4.2详细地展示了模型推理过程。在这个阶段，用户会不断地点击，直到结果满足需求。由于初始交互点必然会分割整个对象，FocusCut 框架从第二个交互点开始引入聚焦视角。如图 4.2的顶部所示，添加当前交互点时，框架首先将采用全局视角的处理流程来获得全局预测结果。根据当前点击的位置以及当前预测 $\mathbf{P}$ 与先前预测 $\mathbf{P}'$ 在全局视角中的差异，判断交互点是否应该经过额外的聚焦视角处理流程。如果采用聚焦视角，则计算此点击的聚焦范围 r，计算方式会在章节4.2.4中介绍。然后，如图 4.2的底部所示，根据聚焦范围，从原图、交互点图和当前预测结果上裁剪下相应的局部区块，输入到聚焦视角的处理流程中来生成局部区块预测 $\hat{\mathbf{P}}$。值得注意的是，这里的图像区块是从原始图像中裁剪出来的。对于高分辨率图像，这有助于避免信息丢失并获得更清晰的 RGB 图像区块。最后，局部区块预测将被粘贴回原始预测中。如果区块之间存在重叠，则重叠部分采用它们的平均值作为预测结果。章节4.2.4还提供了一种渐进式聚焦策略，以迭代的方式不断关注更局部的区域来取得更好的分割效果。

算法 1 聚焦区块模拟

输入： 真值标注图 $\mathbf{G}$，常数 $\alpha_{\mathtt{min}}$，$\alpha_{\mathtt{max}}$，$\beta_{\mathtt{min}}$，$\beta_{\mathtt{max}}$；

1: 目标物体大小系数 $k = \sqrt{\sum_{i,j} \mathbf{G}_{i,j}}$，其中 $\mathbf{G}_{i,j} \in \{0,1\}$；
2: 在 $[\alpha_{\mathtt{min}}, \alpha_{\mathtt{max}}]$ 区间内随机选择 α；
3: 聚焦范围 $r = \alpha \cdot k$；
4: 根据 $\mathbf{G}$ 生成边界图 $\mathbf{B}$，其中 $\mathbf{B}_{i,j} \in \{0,1\}$；
5: 随机选择一个边界点 $\tilde{\mathbf{p}}(x,y)$，其满足 $\mathbf{B}_{x,y} = 1$；
6: 在 $[\beta_{\mathtt{min}}, \beta_{\mathtt{max}}]$ 区间内随机选择 β_x 和 β_y ；
7: $\mathbf{p} = (\tilde{\mathbf{p}}_x + \beta_x \cdot r, \tilde{\mathbf{p}}_y + \beta_y \cdot r)$；

输出： 区块中心点 $\mathbf{p}$，聚焦范围 r。

4.2.3 聚焦区块模拟

本章节将会介绍用户生成交互点周围聚焦区块的模拟算法，以用于模型的训练。在交互式分割的中后期，用户通常点击对象边界附近以使边界更准确。此外，对象的细节也通常在边界附近。所以在训练模型时，需要生成区块以模拟以上情况。首先在对象的边界上选择一个点，并且赋予该点一个随机的偏移值 β 作为该区块的中心点。β 选取自 $[\beta_{min}, \beta_{max}]$ 区间。聚焦范围 r 是一个与对象尺寸相关的随机数字。对象尺寸由从真值标注图 $\mathbf{G}$ 计算出的 k 来反映。随机系数 α 选取自 $[\alpha_{min}, \alpha_{max}]$ 区间。详细的计算过程在算法 1中描述。在本章实验中，α_{min}、α_{max}、β_{min} 和 β_{max} 的默认值分别设置为 0.2、0.8、-0.3 和 0.3。

基于区块中心 $\mathbf{p}$ 和聚焦范围 r，FocusCut 框架将在图像和真值标注图的 $(\mathbf{p}_x - r, \mathbf{p}_y - r)$ 到 $(\mathbf{p}_x + r, \mathbf{p}_y + r)$ 范围裁剪出一个正方形区块。基于区块的掩膜标注，该方法通过同 CascadePSP 方法 [184] 中一样的随机膨胀和腐蚀算法来生成一个粗糙的掩膜作为先前预测图。区块的中心点将会作为用户点击始终包含在用户交互点中。该方法还将在区块中随机选出 0 ~ 3 个前景交互点和背景交互点来模拟区块中心周围的交互点。这些区块交互点将被转换为交互点图，并和原始图像，粗糙分割结果合并成 6 通道图一起输入到网络模型中进行训练。

图 4.3展示了从一个包含椅子的图像和对应的真值标注图中模拟裁剪的区块。可以看到，该算法模拟了用户的交互位置，并且裁剪了不同的区块部分。在每个区块中至少包含一个交互点。这些粗糙掩膜的分割质量低，但是保留了相对完整的信息，使得神经网络模型可以更加关注于分割的精细化。

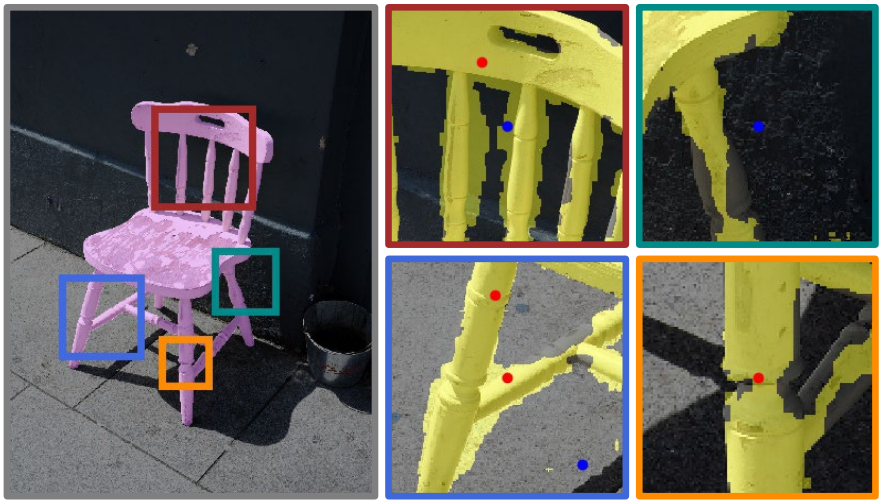


图 4.3　聚焦区块模拟算法的具体样例结果。右图区块中的边界颜色表示从左图对应颜色的位置裁剪放大而来。左图的粉红色掩膜表示真值标注图，右图的黄色掩膜表示模拟算法生成的粗糙掩膜图。这些模拟的前背景交互点也会显示在区块中。

4.2.4　聚焦范围计算

在聚焦视角的推理阶段，如何选择聚焦范围对精细化具有重要意义。实验发现，在全局视角中，面向局部修复的交互点尽管不足以精细化细节，但依然存在一定影响。因此，可以通过比较当前和先前预测来估计出当前交互点的影响范围。根据变化的预测区域大小和目标对象的尺寸，FocusCut 框架可以决定是否深入这个交互点的聚焦视角。如果使用聚焦视角，则可以通过变化区域计算聚焦范围。图 4.4展示了该算法的简要示意图，直观表述为：首先获得全局视角下的变化区域，然后利用交互点计算最小覆盖范围，最后按比例放宽范围。上述过程是基于机器人用户始终点击在预测错误的区域。在实际中，用户有时会点击在已经正确预测的区域，例如，他们会在已经预测的前景上添加前景交互点来精细化小部件或者在已经预测的背景上添加背景交互点来限制边界。对于这种情况，该方法将以交互点和先前预测的边界之间的距离作为聚焦范围来设置聚焦视角。因为该方法的裁剪基于一个正方形，所以在实际计算时算法采用切比雪夫距离。这里定义函数 η 来计算交互点 $\mathbf{a}$ 和 $\mathbf{b}$ 之间的切比雪夫距离：

$$\eta(\mathbf{a},\mathbf{b}) = \max(|\mathbf{a}_x - \mathbf{b}_x|, |\mathbf{a}_y - \mathbf{b}_y|). \tag{4.1}$$

算法 2描述了详细的计算过程，其中 λ 和 ω 的默认值分别为 0.2 和 1.75。

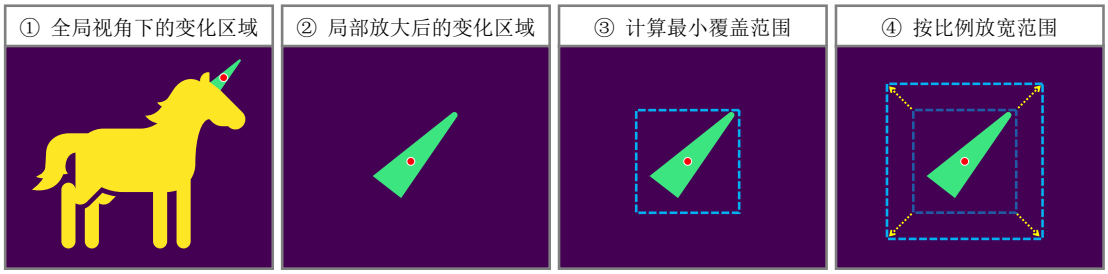


图 4.4 聚焦范围计算算法的示意图。首先获得全局视角下预测结果的变化区域，然后利用交互点计算最小覆盖范围，最后按比例放宽得到最终聚焦范围。

算法 2 聚焦范围计算

输入： 先前的全局预测 $\mathbf{P}'$，当前的全局预测 $\mathbf{P}$，
在全局视角中的区块中心 $\mathbf{p}$，判定系数 λ 和放宽系数 ω；

全局预测中的变化区域 $\Delta\mathbf{P} = |\mathbf{P} - \mathbf{P}'|$；
if $\Delta\mathbf{P}_\mathbf{p}$=1 **then**
　在 $\Delta\mathbf{P}$ 上的 $\mathbf{p}$ 周围使用洪泛法生成区域 $\mathbf{A}$；
　根据 $\sum\mathbf{A} < \lambda \cdot \sum\mathbf{P}$ 得到聚焦判定（True 或 False）；
　聚焦范围 $\tilde{r} = \max_{\forall\{\mathbf{a}|\mathbf{A}_\mathbf{a}=1\}} \eta(\mathbf{p},\mathbf{a})$；
else
　聚焦范围 $\tilde{r} = \min_{\forall\{\mathbf{a}|\mathbf{P}'_\mathbf{a}=1-\mathbf{P}'_\mathbf{p}\}} \eta(\mathbf{p},\mathbf{a})$；
　聚焦判定设为 True；
end if
通过放宽系数 ω 生成 r，$r = \omega \cdot \tilde{r}$；

输出： 聚焦判定，聚焦范围 r。

4.2.5 渐进式聚焦策略

对于本章提出的聚焦视角，聚焦范围越小，越能关注更多的局部细节信息。基于这一点，本章提出了渐进式聚焦策略（Progressive Focus Strategy，PFS），以逐渐地关注需要被修复的更加局部的区域。渐进式聚焦策略与传统的多尺度方式不同，其尺度是根据先前和当前区块预测的变化而动态变化的。在渐进式聚焦策略中，每次获得新的预测，其相应部分将被用作下一次的输入来获得下一轮的预测结果。图 4.5中展示了渐进式聚焦策略算法的示意图。通过多次使用局部预测变化图来反复计算聚焦范围，聚焦范围可以不断缩小以分割出更局部的精细结果。在算法 3中，本章详细展示了提出的渐进式聚焦策略所对应的计算过程。其中，T 的默认值设置为 3，$\hat{\omega}$ 设置为 1.1，ε 设置为 2。

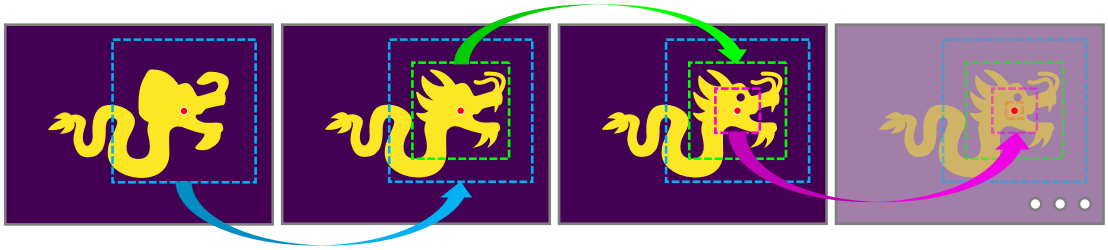


图 4.5 渐进式聚焦策略算法的示意图。通过多次使用局部预测变化图来反复计算聚焦范围，聚焦范围可以不断缩小以分割出更局部的精细结果。

算法 3 渐进式聚焦策略

输入： 之前的区块预测 $\hat{\mathbf{P}}'$，在聚焦视角中的区块中心 $\hat{\mathbf{p}}$，
聚焦轮次 T，放宽系数 $\hat{\omega}$，腐蚀轮次 ε；

for $t = 1, 2, \cdots, T$ 且 $\hat{r} \neq 0$ **do**
  生成新的区块预测 $\hat{\mathbf{P}} =$ 网络模型$(\hat{\mathbf{P}}')$；
  预测的变化区域 $\Delta\hat{\mathbf{P}} = |\hat{\mathbf{P}} - \hat{\mathbf{P}}'|$；
  将 $\Delta\hat{\mathbf{P}}$ 腐蚀 ε 像素以生成区域 $\hat{\mathbf{A}}$；
  if $\sum \hat{\mathbf{A}} > 0$ **then**
    $\tilde{r} = \max_{\forall\{\mathbf{a}|\hat{\mathbf{A}}_\mathbf{a}=1\}} \eta(\hat{\mathbf{p}}, \mathbf{a})$；
    通过放宽系数 $\hat{\omega}$ 生成 $\hat{r}$，$\hat{r} = \hat{\omega} \cdot \tilde{r}$；
    更新之前的预测结果 $\hat{\mathbf{P}}' \leftarrow \hat{\mathbf{P}}$；
    根据 $\hat{r}$ 裁剪新的区块；
  else
    $\hat{r} = 0$；
  end if
end for

输出： 最终区块预测结果 $\hat{\mathbf{P}}$。

标准的渐进式聚焦策略需要迭代地使用当前预测结果来获得下一轮的区块预测，所以在多个迭代过程中不能实现并行操作。这对于神经网络模型常用的图形处理器（GPU）硬件不够友好，从而可能会导致运行速度较慢。因此，本章也提出了一个快速版本来缓解这个问题，可以在牺牲一些性能的情况下提升速度。对于每一个轮次，快速版本会使用之前聚焦范围的 0.8 倍作为当前的聚焦范围。与此同时，裁剪区块的先前预测都来自原始的全局预测。在这个版本的方法中，三个轮次可以并行执行，因此用户可以使用 GPU 硬件来加速该计算过程。对于这两种版本的推理时间，章节4.3.1中给出了详细的分析。

4.3 实验结果与分析

本章节包括三个部分。首先，章节4.3.1介绍了该方法的实验设置，包括使用的数据集、评测指标、实现细节和模型推理。然后，章节4.3.2介绍了该方法的消融实验，包括最重要的聚焦视角以及渐进式聚焦策略的消融实验。最后，章节4.3.3介绍了该方法的性能分析，包括与其他方法的对比和分割结果分析。

4.3.1 实验设置

数据集。 本章工作在实验中采用了以下被广泛使用的数据集：

- **GrabCut [16]:** 该数据集包含了 50 张前景和背景存在明显差异的图像。
- **Berkeley [172]:** 该数据集包含了 96 张带有 100 个对象掩膜的图像，其中的一些样本对交互式图像分割任务具有挑战性。
- **SBD [177]:** 该数据集包含 8498 张训练集图像和 2857 张验证集图像。本章工作在其训练集上训练，在包含 6671 个物体掩膜的验证集上进行评测。
- **DAVIS [174]:** 该数据集包含 50 个视频。遵循先前的工作 [38, 39, 41]，相同的 345 个具有高质量掩膜的帧图像被用于评测。

评测指标。 遵循先前的工作 [19, 33, 36–42, 186]，本工作采用相同的机器人用户来模拟点击，即通过对比标注结果和预测结果，下个交互点将被置于最大错误区域的中心。本章采用数据集上平局的交互点数（Number of Click，NoC）作为评价指标，该指标记录了每个样本达到一个交并比（Intersection over Union，IoU）阈值所需的平均交互点数，表示为 NoC@XX。每个实例的默认最大点击次数限制为 20。数据集上使用最大点击次数仍然无法达到目标 IoU 的失败数（Number of Failure，NoF）也会得到统计。本章使用第 N 次点击时的 IoU 指标来表示分割质量，还绘制了 NoC-IoU 曲线来表示交互后期阶段的收敛趋势。因为本工作对细节分割更有作用，本章还引入了两个细节评价指标。首先是边界交并比（Boundary IoU，BIoU） [187]，该指标只计算靠近对象边界的交并比，以反映边界分割质量。其次是评价预测边界和标注边界相似性的平均对称表面距离（Average Symmetric Surface Distance，ASSD），该指标也常被用于交互式医学图像分割 [52]。对于这两个指标，本章采用第 5 次点击时的指标值（表示为“BIoU&5”和“ASSD&5”）来评价模型的性能。IoU 和 BIoU 指标的数值越大，表明模型的性能越好，NoC 和 ASSD 指标则与之相反。

#	候选项	Berkeley				DAVIS			
		NoC@90 ↓	IoU&5 ↑	ASSD&5 ↓	BIoU&5 ↑	NoC@90 ↓	IoU&5 ↑	ASSD&5 ↓	BIoU&5 ↑
ResNet-50	全局视角	4.510	0.917	2.451	0.785	7.899	0.862	9.711	0.771
	+ 聚焦视角	3.560	0.923	2.365	0.793	6.649	0.870	9.424	0.785
	+ 渐进式聚焦策略	3.440	0.929	2.170	0.804	6.377	0.870	9.338	0.787
ResNet-101	全局视角	4.280	0.922	2.787	0.792	7.713	0.868	9.547	0.777
	+ 聚焦视角	3.350	0.930	2.272	0.805	6.475	0.876	9.038	0.793
	+ 渐进式聚焦策略	3.010	0.933	2.050	0.811	6.223	0.879	8.840	0.796

表 4.1　FocusCut 方法的消融实验。“NoC@90”和“IoU&5”用来评测整体分割，“ASSD&5”和“BIoU&5”用来评测细节分割。“↑”和“↓”用来表示数值越大或越小时模型性能越好。

实现细节。　在 ImageNet [178] 上预训练的 ResNet [176] 被用作特征提取器。训练时批大小为 8，训练周期为 40。训练过程使用初始学习率为 7×10^{-3}、γ 值为 0.9 的指数学习率衰减策略，使用动量值为 0.9、权重衰减值为 5×10^{-4} 的随机梯度下降算法优化参数，并使用二值交叉熵损失来监督输出。本工作使用随机翻转和边长为 384 像素的随机裁剪来进行数据增强。全局视角的标注模拟遵循 [186] 使用的策略。从初始交互点开始，Zoom-In 策略 [39] 被应用到推理阶段。所有实验都在单个 NVIDIA Titan XP GPU 上使用 PyTorch [179] 深度学习框架实现。

模型推理。　由于本章的方法由一个共享网络的两个分支组成，推理时间是便于计算的。本章用“1×”表示这个网络的基础速度。当引入聚焦视角时，由于交互点可以被并行计算，因此速度是“2×”。当引入渐进式聚焦策略时，如果采用默认的 T 值，则标准版本的速度为“4×”。而快速版本的所有轮次依然可以并行计算，故速度依然为“2×”。对于不同分辨率的图像，输入图像将始终等比例调整以得到固定长度的短边。在固定长度为 384 个像素的设置中，ResNet-50 和 ResNet-101 的“1×”速度分别为每次点击 0.0295 秒和 0.0346 秒。即便是标准版本的方法，其模型推理速度也足以满足现实世界中实际应用的需要。

4.3.2　消融实验

表 4.1展示了 FocusCut 流程框架的核心消融实验。由于 Berkeley 数据集与 GrabCut 数据集相似且规模相对较大，而 SBD 数据集的标注质量不如 DAVIS 数据集，因此本工作选择在 Berkeley 和 DAVIS 数据集上开展消融实验。这些消融实验使用了四个评测指标，其中指标“NoC@90”和“IoU&5”用来评价整体分

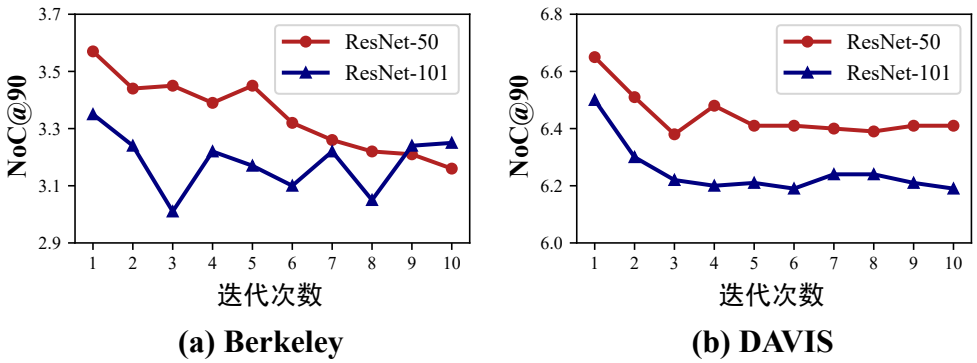


图 4.6　FocusCut 方法在不同迭代次数的渐进式聚焦策略下的性能曲线。

表 4.2　FocusCut 方法在有无迭代预测情况下的性能对比。评测指标为 NoC@90。

设置	Berkeley		DAVIS	
	ResNet-50	ResNet-101	ResNet-50	ResNet-101
无迭代预测	3.51	3.11	6.56	6.38
有迭代预测	3.44	3.01	6.38	6.22

割，指标“ASSD&5”和“BIoU&5”用来评价细节分割。对于渐进式聚焦策略，该实验还针对不同迭代次数和不同设置开展了额外的消融实验。

引入聚焦视角。　对于 FocusCut 方法中的引入聚焦视角这一核心贡献，无论是对整个对象的分割还是细节的分割，性能的提升都是显著的。作为核心的指标，NoC 在 Berkeley 和 DAVIS 两个数据集中大致减少了 1 次点击数。表 4.1在第 5 次点击时将这些消融候选项在其他三个指标上进行了对比。IoU 指标的提升表明聚焦视角带来了一个更完整的对象。BIoU 的增加和 ASSD 的减少表明此方法明显改善了细节分割质量，并提供了一个更精确的边界。无论是 ResNet-50 还是 ResNet-101 作为主干网络，无论是哪个评测指标，网络模型性能的提升都是明显的。因此，聚焦视角的引入对于交互式分割无疑是有意义的。

引入渐进式聚焦策略。　如表 4.1所示，渐进式聚焦策略的使用能进一步提高 FocusCut 方法的性能。在标准版本中，先前预测的通道会根据上一轮的输出进行迭代更新。表 4.2 展示了不使用迭代预测的实验结果。可以发现，在这种情况下，模型的性能会有一定程度的下降。图 4.6还展示了使用该策略时，不同迭

表 4.3 FocusCut 方法和其他方法的 NoC 指标对比。符号 † 和 § 分别表示使用 SBD 数据集和增强的 PASCAL VOC 数据集 [173, 177] 进行训练。* 表示 FocusCut 的快速版本。

方法		GrabCut		Berkeley	SBD		DAVIS	
		@85	@90	@90	@85	@90	@85	@90
§ DOS w/o GC [19] *CVPR16*		8.02	12.59	N/A	14.30	16.79	12.52	17.11
§ DOS w/ GC [19] *CVPR16*		5.08	6.08	N/A	9.22	12.80	9.03	12.58
§ RIS-Net [42] *ICCV17*		N/A	5.00	6.03	N/A	N/A	N/A	N/A
† Latent diversity [36] *CVPR18*		3.20	4.79	N/A	7.41	10.78	5.05	9.57
§ CM guidance [33] *CVPR19*		N/A	3.58	5.60	N/A	N/A	N/A	N/A
† BRS [38] *CVPR19*		2.60	3.60	5.08	6.59	9.78	5.58	8.24
§ MutiSeg [37] *ICCV19*		N/A	2.30	4.00	N/A	N/A	N/A	N/A
§ Continuous Adaptation [40] *ECCV20*		N/A	3.07	4.94	N/A	N/A	5.16	N/A
§ FCA-Net [186] *CVPR20*	ResNet-50	2.18	2.62	4.66	N/A	N/A	5.54	8.83
	ResNet-101	1.88	2.14	4.19	N/A	N/A	5.38	7.90
† f-BRS [39] *CVPR20*	ResNet-50	2.50	2.98	4.34	5.06	8.08	5.39	7.81
	ResNet-101	2.30	2.72	4.57	4.81	7.73	5.04	7.41
† CDNet [41] *ICCV21*	ResNet-50	2.22	2.64	3.69	4.37	7.87	5.17	6.66
	ResNet-101	2.42	2.76	3.65	4.73	7.66	5.33	6.97
† FocusCut* *Ours*	ResNet-50	1.58	1.78	3.48	3.76	5.86	5.18	6.59
	ResNet-101	1.48	1.68	3.22	3.54	5.55	4.98	6.32
† FocusCut *Ours*	ResNet-50	1.60	1.78	3.44	3.62	5.66	5.00	6.38
	ResNet-101	1.46	1.64	3.01	3.40	5.31	4.85	6.22

代次数下 NoC@90 指标的数值。可以看到，前几轮迭代带来的性能提升很明显，而后几轮则因区块尺寸太小而出现波动。由于迭代预测和逐步确定聚焦范围的操作需要根据先前的预测结果进行，因此无法并行实现。该策略的标准版本可能会牺牲一定的速度，所以本章也提供了一个快速版本。如表 4.3所示，其中聚焦范围的缩减系数为常数。这个快速版本可以在达到出色性能的同时，节省迭代所用的时间。用户可以根据需求和环境选择自己想要的版本。

4.3.3 性能分析

本章节将 FocusCut 方法的实验结果与其他方法在多方面进行了比较，具体包括平均交互点数对比、NoC-IoU 曲线对比、细节分割指标对比、性能瓶颈对比和可视结果对比。还对分割结果进行了展示和分析。

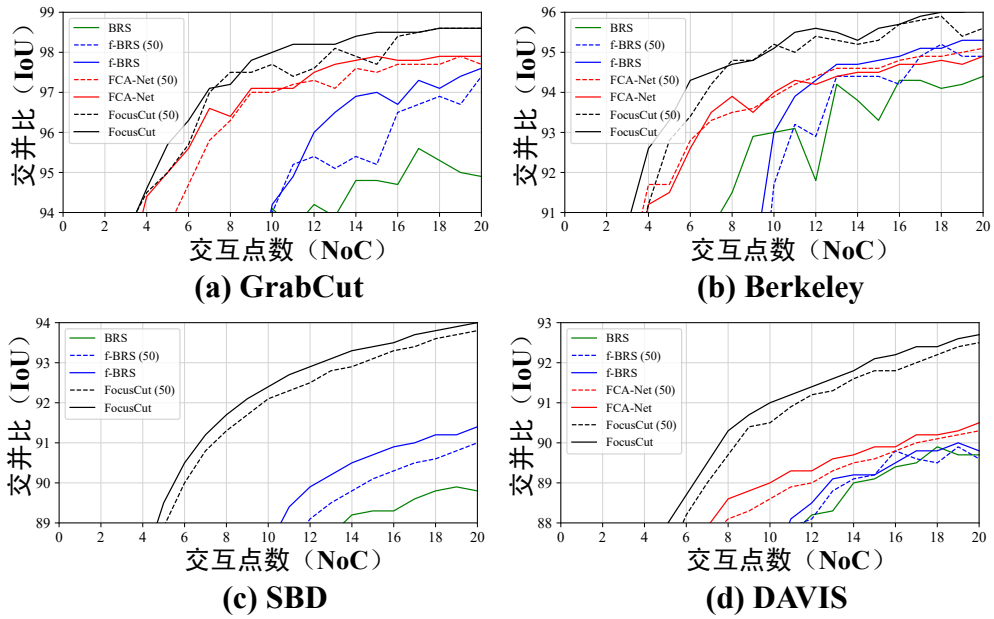


图 4.7　FocusCut 方法和其他方法的 NoC-IoU 曲线图。“(50)”表示该模型采用 ResNet-50 作为主干网络。SBD 数据集的子图上没有 FCA-Net 方法的曲线是由于 FCA-Net 方法在训练过程使用的是增强后的 PASCAL VOC 数据集 [173, 177]，与 SBD 数据集本身有重合。

平均交互点数对比。　基于最常用的 NoC 指标，FocusCut 方法和其他方法的对比结果如表 4.3 所示。本章的方法在 GrabCut、Berkeley、SBD 和 DAVIS 四个数据集上都进行了评估。表中提供了以 ResNet-50 和 ResNet-101 为主干网络的所有 NoC 性能指标。可以发现，FocusCut 方法在所有数据集中都取得了先进的性能。此外，FocusCut 快速版本的性能略差于标准版本，但与其他方法相比，仍然具有良好的表现。值得注意的是，FocusCut 在基准网络的基础上几乎没有添加任何参数或模块，这也反映了 FocusCut 方法的有效性。

NoC-IoU 曲线对比。　为了反映交互式分割过程的收敛趋势，本章裁剪并放大了局部 NoC-IoU 曲线，并将它们显示在图 4.7中。在该图中，本工作选择了一些最近的具有开源代码的方法进行比较。此外，由于 FCA-Net 方法使用了增强的 PASCAL VOC 数据集 [173, 177] 进行训练，故它不在 SBD 数据集的子图中。可以发现，在交互过程的较后期阶段，本章提出的 FocusCut 方法仍然有一定的上升趋势。第 20 次点击结果的 IoU 指标表明了本章提出的 FocusCut 方法具有更高的上限，这同时也反映了它可以更精细地分割出目标对象的细节部分。

表 4.4 FocusCut 方法和其他方法的细节分割指标对比。表中选取了第 20 次点击时的评测数值。后四个模型的主干网络为 ResNet-101。“↑”和“↓”表示数值越大或越小时性能越好。

方法	Berkeley		DAVIS	
	ASSD ↓	BIoU ↑	ASSD ↓	BIoU ↑
DOS [19]	4.150	0.594	7.402	0.741
LD [36]	2.218	0.773	7.186	0.776
BRS [38]	1.099	0.866	6.188	0.829
f-BRS [39]	1.218	0.866	6.318	0.825
FCA-Net [186]	1.147	0.861	6.051	0.834
FocusCut	0.928	0.892	4.427	0.874

表 4.5 FocusCut 方法在不同交互点设置情况下和其他方法的对比。该表选取了各方法以 ResNet-50 为主干网络，在 DAVIS 数据集上的结果。$\text{NoF}_N@90$ 表示 N 次点击后 IoU 未能达到 90% 的样本数。$\text{NoC}_{100}@90$ 指最大交互点数为 100 时的平均交互点数。

方法	$\text{NoF}_{20}@90$	$\text{NoF}_{100}@90$	$\text{NoC}_{100}@90$
BRS [38]	77	51	20.89
f-BRS [39]	78	50	20.70
FCA-Net [186]	87	54	22.56
CDNet [41]	65	48	18.59
FocusCut	57	43	17.42

细节分割指标对比。 在表 4.4中，本章展示了 FocusCut 方法与其他方法的细节分割指标的对比情况。评测指标采用 ASSD 和 BIoU 这两种反映边界质量和边界周围分割的指标。由于细节分割的指标需要数据集标注精度较高，所以该实验采用 Berkeley 和 DAVIS 两个数据集进行评测。由表中可以看出，对于 ASSD 指标，在两个数据集上，FocusCut 方法相对于其他方法是领先的。在 DAVIS 这一拥有高精度标注的数据集上尤为明显。对于 BIoU 指标，FocusCut 方法达到了最好的性能，反映了边界周围的细节得到了更好的分割。

性能瓶颈对比。 表 4.5中展示了在不同交互点设置的情况下，FocusCut 与其他方法的性能瓶颈对比。首先是 $\text{NoF}_{20}@90$ 指标的对比，FocusCut 的数值最小，代表了该方法在 DAVIS 数据集中，只有 57 个样本无法在 20 个点以内达到 90% 的

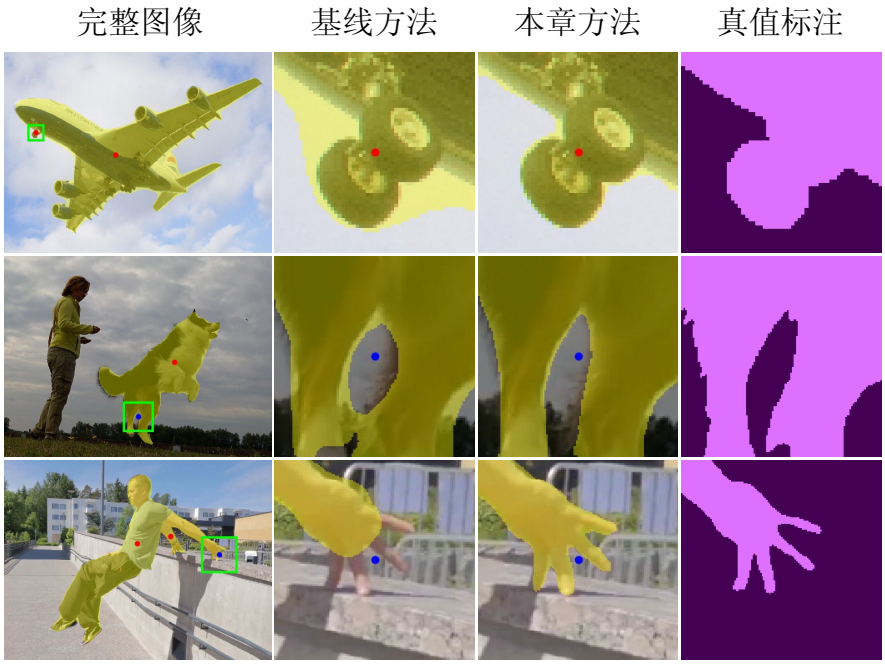


图 4.8 FocusCut 方法的分割结果及其同基线方法的对比。基线方法即在全局视角下的分割，而本章方法指的是使用了聚焦视角和渐进式聚焦策略的 FocusCut 标准版本。

IoU 指标，相对于其他方法是最少的。$NoF_{100}@90$ 指标同理，但它反映了模型的瓶颈，由于一些极难图像的存在，可能造成使用这些交互式分割方法难以达到预期目标。$NoC_{100}@90$ 指标反映了在 100 个交互点为阈值的情况下的平均交互点数，FocusCut 同样达到了最好。综上，FocusCut 方法的性能瓶颈相对较小。

分割结果分析。 图 4.8展示了 FocusCut 方法可以发挥主导作用的一些情况。例如，在飞机轮子等小部件的位置，FocusCut 方法只需用户在前景中提供一个交互点即可生成准确的预测。在一些有很多缝隙的地方，比如图中小狗的腿之间或者人的手指之间的区域，虽然提供了背景交互点，但是神经网络在全局视角下仍然无法得到满意结果，而本章提出的 FocusCut 方法可以很好地处理该情况。

可视结果对比。 在图 4.9中，本章还展示了在同等交互点数下，本章提出的 FocusCut 方法与其他方法的可视结果对比图。图上的交互点是由基于模拟算法的机器人用户生成的。可以看出，FocusCut 方法在性能上领先其他方法。此外，对于一些细节区域，比如腿部间隙、手臂间隙等局部区域，FocusCut 也能排除掉一些全局交互点的干扰，从而得到更好的分割质量。

图 4.9　FocusCut 方法和其他方法的分割结果对比。

4.4 本章小结

针对交互式图像分割任务中，复杂场景下的局部区域精度低的问题，本章引入了聚焦视角来理解用户新输入的交互点的意图以获得更精细的局部分割。基于此，本章提出了一个简单而有效的流程框架，命名为 FocusCut。该框架中，聚焦视角下以每个交互点为中心裁剪的区块通过一个与全局视角共享的网络进行精细分割，并粘贴回原图的粗糙分割以得到最终精细的分割结果。在渐进式聚焦策略下，区块分割可以进行迭代更新以获得更好的效果。在四个数据集上的大量实验中，本章提出的 FocusCut 方法取得了优异的性能，证明了该方法的优越性。有了该方法，复杂场景下的物体可以获得更为精细的分割结果。

5 修复细小结构的切割线交互式分割

对于交互式图像分割任务，用户有时候会遇到一些拥有细小结构的特殊物体。这类物体使用一般的交互式分割方法往往难以分割或者交互负担极大。要想减少对这类物体的交互负担，则需要设计一个高效的交互模式与方法。面对复杂场景中的细小结构交互难这一难点，以针对细小结构的复杂拓扑分割为目标，本章提出了修复细小结构的切割线交互式分割。该工作设计了一个新的交互模式，即切割线交互模式，并基于此设计了 KnifeCut 方法，使用户可以低负担、高效率地进行细小结构物体的分割与修复。实验证明，本章提出的方法对于交互式图像分割任务中针对细小结构的复杂拓扑分割具有显著作用。在本章中，首先，章节5.1对该工作的背景、动机、贡献等进行了介绍。其次，章节5.2展示了该工作提出的新的切割线交互模式及其模拟算法。之后，章节5.3详细描述了该工作基于切割线交互模式提出的 KnifeCut 方法的模型结构。然后，章节5.4描述了实验设置，进行了消融实验，结合其他方法进行了性能比较与分析，并且对 KnifeCut 方法进行了用户调研。最后，章节5.5对该工作进行了总结。

5.1 本章引言

在现实世界中，许多物体都具有细小的结构，例如栏杆、球网和树枝等。当前的图像分割方法在处理这些细小部分时往往具有局限性。在大多数情况下，这些方法的分割结果在主体部分处表现很好，但在细小部分处的结果往往不尽人意。因此，当需要高精度的分割结果时，不可避免地要引入人工参与的后处理修复方式。人们通常采用专业的图像处理工具（如 PhotoShop 等）或流行的交互式分割技术 [17, 39, 46] 来试图精细化细小结构的分割。然而，目前的交互模式，例如点击、多边形和笔刷等，都不能高效地解决细小部分的修复。如图 5.1的顶行所示，用多边形标记边界或用笔刷绘制整个腿部区域可能既耗时又费力。基于前背景点击的方法在一定程度上降低了标注过程的复杂性，但是将前景交互点、背景交互点分别点击在昆虫腿上及腿旁对于用户也是一种较重的负担。

为了有效修复细小结构分割，本章引入了一种快速、低负担的交互模式，称作切割线交互模式，即用户只需像用刀切割一样，绘制一条线穿过错误标记的

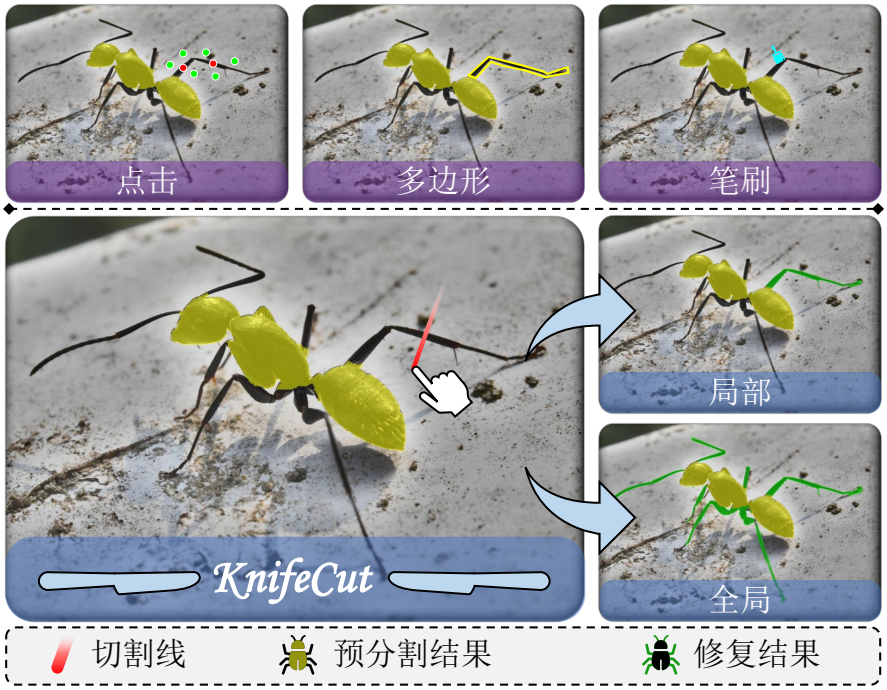


图 5.1 KnifeCut 方法的展示以及与其他交互的对比。顶部：处理细小部分时，点击、多边形和笔刷等交互模式的效率低下。底部：使用 KnifeCut 方法，用户可以在细小部分处像用小刀切割一样绘制一条切割线，进行精细化修复。KnifeCut 方法为用户提供了两个结果：一个是交互目标的局部细小部分修复，另一个是全局的所有相关的细小部分修复。

像素区域。如图 5.1所示，在错误标记的蚂蚁腿上进行切割的动作是直观并且快速的。图 5.2则展示了更多具体的样例。无论是在分割缺失的伞和球拍上，还是在分割过度的爪子和网上，本章提出的切割线交互模式都可以在瞬间完成，而对于先前那些需要仔细瞄准前背景和边界的交互而言，处理这些情况是难以想象的。此外，由于切割线上既有前景内容，也有背景内容，并且上面一定具有细小部分的像素，因此它提供了对比度先验来增加预测的准确率。

基于提出的切割线交互模式，本章进一步设计了一个名为 KnifeCut 的分割框架。如图 5.1所示，KnifeCut 旨在获得包含目标蚂蚁腿的局部精细化分割和包含所有蚂蚁腿的全局精细化分割，同时还要保持良好分割的主体部分不发生明显的变化。具体而言，KnifeCut 方法首先预测切割线针对的细小部分区域，然后利用其对应的特征相似性来激活所有相关的细小部分。利用切割线的先验信息和激活的相似度图，KnifeCut 方法得以估计局部和全局细小部分区域。它利用这些区域估计图进一步通过两个分支来进行局部和全局细小部分的分割精细

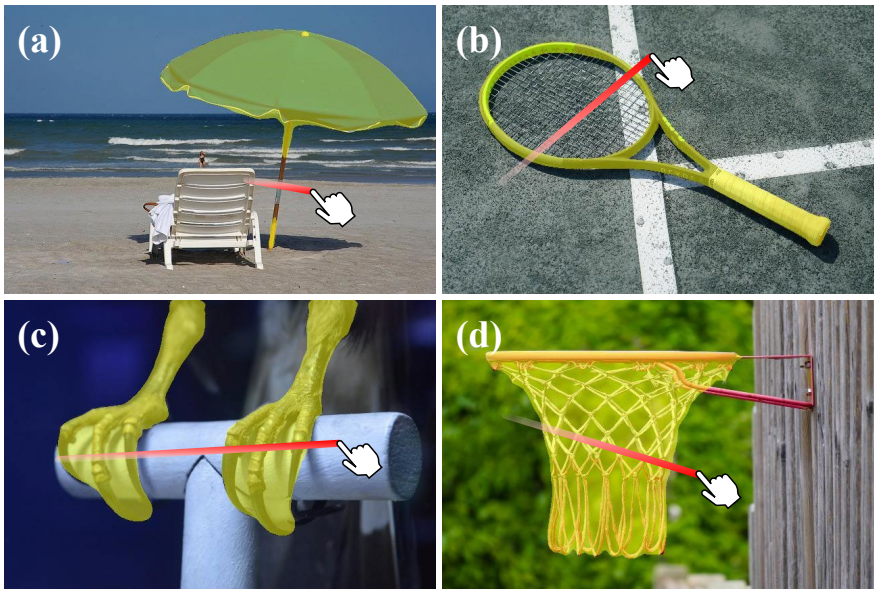


图 5.2 KnifeCut 方法适用的不同细小结构情况。（a）沙滩伞的把手缺少部分分割；（b）网球拍的网线未被分割；（c）爪子没有被准确地分割；（d）球网的间隙被错误分割。

化，同时避免对主体的影响。用户可以自由选择两个精细化结果中的任何一个，其中局部分支仅关注目标细小部分，全局分支会精细化所有同类型的细小部分。本方法在 ThinObject-5K [46]、HRSOD [188] 和 COIFT [189,190] 这三个拥有细小结构物体的数据集上开展了相关的实验，并且进行了真实的用户调研。实验结果证明了 KnifeCut 方法中的交互模式和网络模型的有效性。

该工作的贡献可以总结如下：

1. 提出了一种用于细小结构分割精细化的高效交互模式，该模式只需要像刀切一样绘制一条切割线穿过分割较差的细小结构区域。
2. 基于提出的切割线交互模式，本章设计了 KnifeCut 方法，它根据用户交互从局部和全局角度提供修复后的精细化分割结果。
3. KnifeCut 是第一种专门设计用于交互式细小结构分割精细化的方法。大量评测实验和用户调研结果进一步证明了它的便捷性和有效性。

5.2 切割线交互模式

本章节包括两个部分。章节5.2.1结合具体样例介绍了提出的切割线交互模式。章节5.2.2则详细描述了训练和测试中切割线的模拟算法。

5.2.1 交互模式介绍

为了更好地解释之后部分，本章在这里定义了一些符号。**G** 表示图像 **I** 的真值标注图。通过使用与 [46] 相同的策略，本章从 **G** 中提取细小部分的真值标注，表示为 $\mathbf{G}_{\text{thin}}$。而非细小部分的真值标注 $\mathbf{G}_{\text{non-thin}}$ 可以通过以下获得：

$$\mathbf{G}_{\text{non-thin}} = \mathbf{G} - \mathbf{G}_{\text{thin}} \tag{5.1}$$

本章将其他分割方法获得的预分割结果表示为 $\mathbf{P}'$。与 **G** 作对比，$\mathbf{P}'$ 通常在以交并比为衡量指标时已经取得了良好性能。然而，由于细小部分的像素太少，这些方法往往无法处理细小部分的分割，如图 5.2所示，$\mathbf{P}'$ 主要面临以下两种细小结构的情况：（1）分割缺失：细小结构的细节，甚至整个区域都丢失了，例如图 5.2（a-b）中的沙滩伞和网球拍。（2）分割过度：细小结构由于过于密集导致间隙被一起分割出，例如图 5.2（c-d）中的爪子和球网。

为了解决这些问题，本章试图提出一种专门为细小结构设计的高效修复工具。在 $\mathbf{G}_{\text{non-thin}}$ 标识的区域，$\mathbf{P}'$ 已经表现得足够好。因此该工具只需要关注细小区域的精细化而几乎不用改变主体分割。考虑到上述复杂情况，并受到人类行为的启发，本章提出了一种有效的切割线交互模式来处理这种细小结构。正如人们倾向于用刀切割细小物体一样，用户只需要在细小部分的错误标记区域绘制一条切割线就可以完成这一交互。图 5.2展示了切割线交互模式的实际样例。无论是分割缺失还是分割过度的预分割结果，该交互都可以在不需要太多思考和精力的情况下完成。除了方便和高效之外，切割线还包含其他交互模式无法提供的信息。因为切割线上既有前景内容，也有背景内容，所以它包含着前背景对比的先验信息。章节5.4.2中展示的进一步实验将证明本章提出的切割线交互模式在处理细小结构物体时相较于常见的点击交互模式的优越性。

5.2.2 交互模拟算法

即使面对同一个目标，不同用户往往会做出不同的交互行为。为了让网络模型能更好地适应真实情况，需要使用各种形态的切割线交互来进行训练。由于从真实用户收集这些交互的成本过高，因此本章设计了一种算法来模拟用户行为并自动生成切割线交互。此外，为了避免训练过程中的过拟合，模拟算法具有一定的随机性。但在测试过程中，由于需要保证每次测试的公平性，并尽量符合用户的操作习惯，本章假设用户会大致垂直于细小结构骨架方向来绘制切割线以穿过最大错误区域。图 5.3展示了该模拟算法。模拟仿真算法的细节如下：

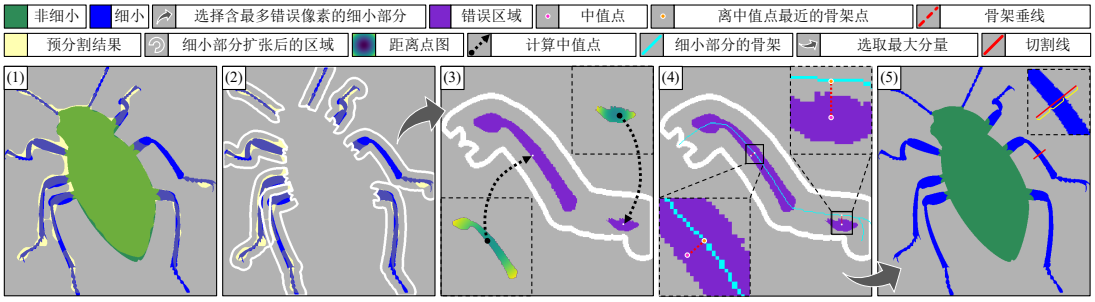


图 5.3　切割线交互模式的模拟算法可视步骤。（1）展示了预分割结果、非细小部分和细小部分。（2）细小部分的扩展区域由白色轮廓线表示。对每个扩展区域中的错误像素进行计数后，选择错误像素最多的细小部分。（3）计算距离图，其中每个点的数值为到其他点的距离和，然后在错误区域中找到距离和最小的中值点。（4）提取细小部分的骨架，然后使用离中值点最近的骨架点作为锚点。基于锚点，绘制骨架的垂直线。（5）采用错误区域的最大分量子区域绘制切割线。两端将在细小部分上被扩展相同的长度。

步骤 1：　在图像的所有细小部分中，用户倾向于首先精细化预分割结果中最差的部分，因此需要评估其当前分割质量。为了简化，本步骤假设 $\mathbf{G}_{\text{thin}}$ 上的每个连通分量可以被视为一个细小部分，用 $\mathbf{T}$ 表示。遵循 [46] 中的做法，本步骤首先设置了一个可以用如下公式表示的阈值 τ：

$$\tau = 10 \times \frac{\max(H_{\text{box}}, W_{\text{box}})}{300}, \tag{5.2}$$

其中，H_{box} 和 W_{box} 分别指的是物体的包围框的高度和宽度。本步骤使用 $d(\mathbf{p}_1, \mathbf{p}_2)$ 表示 $\mathbf{p}_1$ 和 $\mathbf{p}_2$ 两点之间的欧式距离。函数 $\phi(\mathbf{p}, \mathbf{T})$ 用来计算点 $\mathbf{p}$ 到 $\mathbf{T}$ 中前景的最短距离，该距离变换公式如下：

$$\phi(\mathbf{p}, \mathbf{T}) = \min_{\forall\{\mathbf{q}|\mathbf{T}_\mathbf{q}=1\}} d(\mathbf{p}, \mathbf{q}), \tag{5.3}$$

其中，$\mathbf{T}_\mathbf{q}$ 指的是 $\mathbf{T}$ 中位置 $\mathbf{q}$ 处的数值。因此，每个细小部分的扩展区域 $\mathbf{E}$（也被用作评价掩膜）可以用如下公式表示：

$$\mathbf{E}^i = \{\mathbf{p} : (\phi(\mathbf{p}, \mathbf{T}^i) \le \tau) \wedge ((\mathbf{G}_{\text{non-thin}})_\mathbf{p} = 0)\}, \tag{5.4}$$

其中，$\mathbf{T}^i$ 和 $\mathbf{E}^i$ 分别指的是第 i 个细小部分和对应的扩展区域。图 5.3（2）中采用了一个白色的轮廓线来标识扩展区域 $\mathbf{E}$。

步骤 2：　本步骤将计算每个细小部分 $\mathbf{T}^i$ 周围的错误预测像素的数目 $\mathbf{N}^i$：

$$\mathbf{N}^i = \sum(|\mathbf{P}' - \mathbf{G}| \times \mathbf{E}^i), \tag{5.5}$$

其中，$\times$ 指的是逐元素乘法。对于训练过程，本步骤将随机地挑选细小部分进行交互模拟，其中随机概率与 $\mathbf{N}^i$ 成正比。但是，在测试阶段，本步骤将为后续步骤选择错误像素最多的细小部分。图 5.3（2-3）展示了该过程。

步骤 3： 对于选定的细小部分，模拟算法希望切割线穿过错误区域。因此，本步骤首先计算错误区域，该区域在图 5.3中以紫色表示。对于训练阶段，本步骤将在错误区域中随机采样一个点 $\mathbf{m}$。但在测试阶段，本步骤将选择错误区域的最大分量，并将其表示为 $\mathbf{C}$。点 $\mathbf{m}$ 将位于 $\mathbf{C}$ 的中值点，该中值点与其他点的距离和最小。这个过程可以用如下公式表示：

$$\mathbf{m} = \underset{\forall\{\mathbf{p}|\mathbf{C_p}=1\}}{\arg\min} \sum_{\mathbf{C_q}=1} d(\mathbf{p},\mathbf{q}). \tag{5.6}$$

图 5.3（3）展示了距离图的两个示例，其中每个点的数值为到其他点的距离和。

步骤 4： 为了使切割线垂直于细小部分，本步骤首先提取出所选细小部分的骨架，并在上面选择一个锚点 $\mathbf{a}$。本步骤使用 [191] 提出的算法提取骨架，并将其记为 $\mathbf{K}$。无论在训练阶段还是测试阶段，都将从 $\mathbf{K}$ 中选择与点 $\mathbf{m}$ 的距离最短的点作为锚点 $\mathbf{a}$，这个过程可以用如下公式表示：

$$\mathbf{a} = \underset{\forall\{\mathbf{p}|\mathbf{K_p}=1\}}{\arg\min}\ d(\mathbf{p},\mathbf{m}). \tag{5.7}$$

确定锚点后，本步骤需要选择切割线的角度 θ。在训练阶段，为了避免训练过程中的过拟合，角度 θ 在 $[0,180°)$ 内随机选取，切割线将以顺时针方向进行旋转。在测试阶段，本步骤根据锚点 $\mathbf{a}$ 作一条骨架的垂直线，它将通过点 $\mathbf{m}$。图 5.3（4）展示了基于骨架的垂直线的两个具体样例。

步骤 5： 确定了锚点 $\mathbf{a}$ 和垂直于骨架的角度，本步骤可以相应绘制出切割线。本步骤首先在细小部分的边界上定位切割线的两端。为了迎合用户的操作习惯，本步骤给这条切割线的两端增加了一定的扩展长度 l。在训练阶段，扩展长度 l 在两端是随机的。在测试阶段，l 被设置成与细小部分上的切割线长度相同。图 5.3（5）显示了模拟算法得到的切割线的最终可视结果。

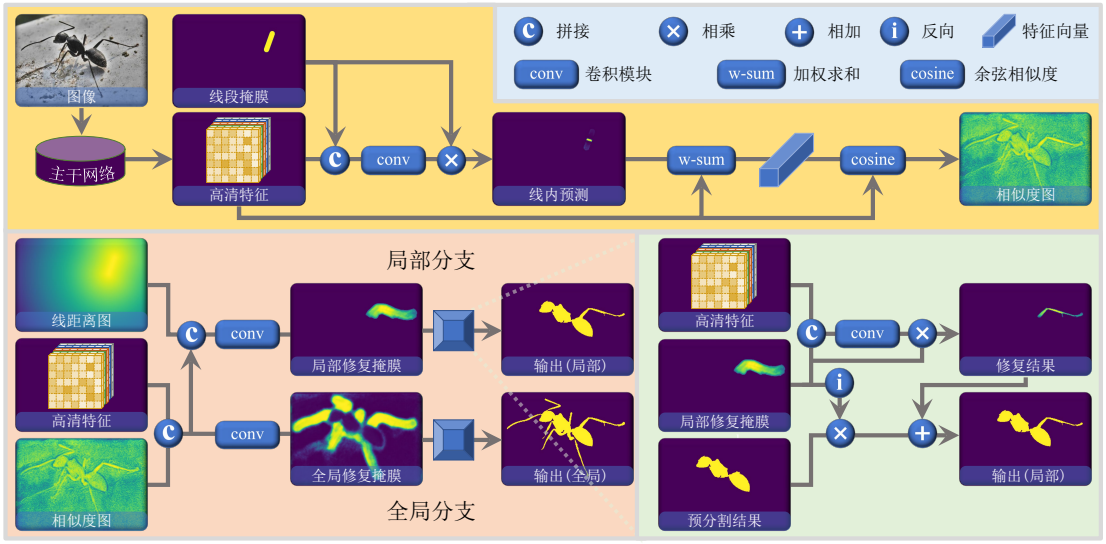


图 5.4　KnifeCut 方法的网络结构图。顶部是细小结构相似性模块，它通过切割线上的相似性特征向量激活所有相关的细小结构。底部是具有两个分支的分割精细化模块。它首先估计局部和全局的修复掩膜，然后在两个掩膜的帮助下获得精细化结果。结合精细化结果和预分割结果，网络输出局部和全局的最终分割结果。

5.3　交互框架与网络模型

图 5.4展示了 KnifeCut 方法的网络模型。围绕该模型，本章节分为四个部分。首先，章节5.3.1介绍了特征提取器，然后，章节5.3.2介绍了相似性模块。之后，章节5.3.3介绍了分割精细化模块。最后，章节5.3.4介绍了相应的损失函数。

5.3.1　特征提取器

随着基于深度学习的方法的普及，各模型通常采用经典的卷积神经网络来提取图像特征。由于大多数网络是通过从高分辨率到低分辨率的方式提取特征，因此在降采样过程中可能会丢失细小部分的信息，从而导致细小结构的分割性能不佳。因此，为了在整个框架中保持高分辨率特征，本章将 HRNet [192] 作为 KnifeCut 方法的特征提取器。本章将该网络输出的特征用作 KnifeCut 中使用的特征，并将其记为 **F**。值得注意的是，此特征尺寸为原始图像的四分之一。

此外，为了与其他交互式分割方法保持一致，本章还采用 ResNet [176] 作为特征提取器。但是，由于它是通过降采样过程来提取特征，前几层输出的高分辨率特征包含有限的高层信息。因此，为了保持与 HRNet 相同分辨率的特征，

本方法构建了与 DeepLab v3+ [62] 类似的结构。ResNet 输出的特征将被送入空洞卷积池化金字塔模块和解码器模块，同时其分辨率将恢复为原始的四分之一。由 HRNet 和 ResNet 提取特征带来具体效果差异详见章节5.4.3。

5.3.2 相似性模块

基于观察到物体的许多细小部分（例如蚂蚁的腿）通常具有相似的特征，KnifeCut 方法设计了一个简单但有效的模块来利用相似信息。首先把线段掩膜 **L** 和特征 **F** 拼接在一起，并将它们输入到七个拥有批归一化层和 ReLU 激活函数的 3×3 卷积块。然后将输出结果乘以 **L**，该方法便可以得到切割线上的细小部分的预测，并将其记为 $\mathbf{P}_{\text{line}}$。基于特征 **F** 和预测结果 $\mathbf{P}_{\text{line}}$，可以使用如下公式计算出细小部分的特征向量 $\boldsymbol{s}$：

$$\boldsymbol{s}=\frac{\tilde{\sum}(\mathbf{F}\times\mathbf{P}_{\text{line}})}{\sum\mathbf{P}_{\text{line}}}, \tag{5.8}$$

其中，$\tilde{\sum}$ 指的是逐通道相加的操作，$\boldsymbol{s}$ 的维度与 **F** 的网络层数相同。

然后，该方法通过余弦相似度获得相似度图 **S**：

$$\mathbf{S}_{\mathbf{p}}=\frac{\boldsymbol{s}\cdot\mathbf{F}_{\mathbf{p}}}{\|\boldsymbol{s}\|\|\mathbf{F}_{\mathbf{p}}\|}, \tag{5.9}$$

其中，$\|\boldsymbol{s}\|$ 和 $\|\mathbf{F}_{\mathbf{p}}\|$ 分别指的是 $\boldsymbol{s}$ 和 $\mathbf{F}_{\mathbf{p}}$ 的二范数。$\mathbf{F}_{\mathbf{p}}$ 指的是像素位置 **p** 处的特征向量，此向量和 $\boldsymbol{s}$ 具有相同的维度。可视化的相似度图详见图 5.4和图 5.5。

5.3.3 分割精细化模块

考虑到用户的不同意图，该方法为分割精细化模块配备了两个分支。一个分支是根据用户提供的交互，只对目标细小部分进行精细化。而另一个分支则利用细小部分共同的相似性，一次性精细化所有相似的细小部分。该方法首先计算得到线距离图 **D**，其中：

$$\mathbf{D}_{\mathbf{p}}=\phi(\mathbf{p},\mathbf{L}). \tag{5.10}$$

为了利用交互信息，分割精细化模块的局部分支以特征 **F**、相似度图 **S** 和线距离图 **D** 作为输入，而全局分支的输入则不包括线距离图 **D**。

拼接而成的结果图被送入到和先前相似结构的卷积块中。然后该方法可以分别获得局部和全局分支的修复掩膜 **M**。之后该方法将 **M** 与 **F** 拼接起来，并输入到同样的卷积块中，从而获得两个分支的预测结果 **R**。由于该方法的目标是根据其他方法获得的预分割结果 $\mathbf{P}'$ 来精细化细小部分，因此该方法保留了 $\mathbf{P}'$ 的

主体部分的分割结果，仅对细小结构部分的分割进行了修复。局部和全局两个分支的最终预测结果 **P** 可以用如下公式表示：

$$\mathbf{P}_{\text{branch}} = \mathbf{M}_{\text{branch}} \times \mathbf{R}_{\text{branch}} + (1 - \mathbf{M}_{\text{branch}}) \times \mathbf{P}', \tag{5.11}$$

其中，`branch` 指的是局部（local）或者全局（global）。

5.3.4 损失函数

对于二值分割任务，通常采用带权重图 **W** 的二值交叉熵作为损失函数，其具体计算过程可以表示为：

$$\ell(\mathbf{P}, \mathbf{G}, \mathbf{W}) = -\frac{1}{N} \sum \mathbf{W} \times (\mathbf{G} \times \log(\mathbf{P}) + (1 - \mathbf{G}) \times \log(1 - \mathbf{P})), \tag{5.12}$$

其中，**P** 指的是预测概率图，**G** 指的是由 0 和 1 标签组成的真值标注图，N 是整个图像的像素的总个数。权重图可以用于使网络专注于细小部分的分割质量。在细小结构相似性模块中，该方法采用 $\mathcal{L}_{\text{line}}$ 来监督切割线内部预测结果:

$$\mathcal{L}_{\text{line}} = \ell(\mathbf{P}_{\text{line}}, \mathbf{G}_{\text{thin}}, \mathbf{L}). \tag{5.13}$$

对于分割精细化模块中的精细化结果，此权重图 $\tilde{\mathbf{W}}$ 可以使用如下公式计算：

$$\tilde{\mathbf{W}}_{\text{branch}} = 1 - \mathbf{G}_{\text{non-thin}} + \mathbf{M}_{\text{branch}}. \tag{5.14}$$

对于局部和全局两个分支的最终输出，该方法同样采用二值交叉熵损失函数来监督。每个分支的真值标注图可以使用如下公式表示：

$$\tilde{\mathbf{G}}_{\text{branch}} = \begin{cases} \mathbf{P}' \times (1 - \mathbf{E}^*) + \mathbf{G} \times \mathbf{E}^*, & \text{branch} = \text{local} \\ \mathbf{P}' \times (1 - \mathbf{E}) + \mathbf{G} \times \mathbf{E}, & \text{branch} = \text{global} \end{cases}, \tag{5.15}$$

其中，$\mathbf{E}^*$ 指的是交互针对的细小部分的扩展区域，**E** 指的是所有细小部分的扩展区域。因此，两个分支的损失可以用如下公式计算：

$$\mathcal{L}_{\text{branch}} = \ell(\mathbf{R}_{\text{branch}}, \mathbf{G}, \tilde{\mathbf{W}}_{\text{branch}}) + \ell(\mathbf{P}_{\text{branch}}, \tilde{\mathbf{G}}_{\text{branch}}, \mathbf{A}), \tag{5.16}$$

其中，**A** 指的是一个全 1 矩阵。

最终网络模型的总体损失 $\mathcal{L}_{\text{total}}$ 可以用如下公式表示：

$$\mathcal{L}_{\text{total}} = \alpha \mathcal{L}_{\text{line}} + \beta \mathcal{L}_{\text{local}} + \gamma \mathcal{L}_{\text{global}}, \tag{5.17}$$

其中，$\mathcal{L}_{\text{local}}$ 和 $\mathcal{L}_{\text{global}}$ 是通过把 `branch` 设置为 `local` 或者 `global` 并根据公式 5.16计算得到的。本章的实验对切割线内部的分割施加了硬性约束。因此，α、β 和 γ 分别被设置为 10、1 和 1。

5.4 实验结果与分析

本章节包括三个部分。首先，章节5.4.1介绍了该方法的实验设置，包括使用的数据集、评测指标、实现细节和模型推理。然后，章节5.4.2介绍了该方法的消融实验，包括相似性模块、线距离图和切割线交互的消融实验。之后，章节5.4.3介绍了该方法的性能分析，包括与其他方法的对比和分割结果的展示和分析。最后，章节5.4.4介绍了该方法中切割线交互模式的用户调研。

5.4.1 实验设置

数据集。 本章采用以下数据集进行实验：

- **ThinObject-5K [46]**：该数据集包含 4748、500 和 500 张图像，分别用于训练、验证和测试。其中所有图像都是通过在背景图像中加入前景对象合成而成的。本工作在其训练集上训练网络模型，并在测试集上对其进行评测。
- **HRSOD [188]**：该数据集最初是为显著性物体检测任务提出的。遵循先前的工作 [46]，本章使用相同的包含 305 个细小结构物体的 287 张图像进行评测。
- **COIFT [189,190]**：该数据集包含 280 张图像，它结合了 [189, 189] 中的 3 个鸟类和昆虫数据集。需要注意的是，此数据集中的图像的分辨率比其他两个数据集低得多。本章遵循先前的工作 [46]，也采用了此数据集进行评测。

评测指标。 由于本章的方法是专门针对细小结构提出的，并且细小结构区域的像素数量通常非常少，因此无法在交并比（Intersection over Union，IoU）指标上明显反映出性能变化。遵循 [46]，本章在扩展的细小部分区域 $\mathbf{E}$（详见章节5.2.2）上计算 IoU 分数，得到细小 IoU 指标（$\mathrm{IoU}_{\mathrm{thin}}$），以更好地反映细小结构的分割性能。因此，$\mathrm{IoU}_{\mathrm{thin}}$ 可以表示为：

$$\mathrm{IoU}_{\mathrm{thin}} = \frac{\sum(\mathbf{P}_{\mathrm{thin}} \cap \mathbf{G}_{\mathrm{thin}})}{\sum(\mathbf{P}_{\mathrm{thin}} \cup \mathbf{G}_{\mathrm{thin}})}, \quad \mathbf{P}_{\mathrm{thin}} = \mathbf{P} \times \mathbf{E}, \tag{5.18}$$

此外，本章还采用细小部分的边界度量 $\mathcal{F}$ [193] 来评估分割掩膜的边缘质量，表示为 $\mathcal{F}_{\mathrm{thin}}$。首先通过形态学算子计算出细小部分轮廓图 $c(\mathbf{P}_{\mathrm{thin}})$ 和 $c(\mathbf{G}_{\mathrm{thin}})$ 的轮廓点之间的基于轮廓的准确率 P_c 和召回率 R_c。之后 $\mathcal{F}_{\mathrm{thin}}$ 可以用如下公式表示：

$$\mathcal{F}_{\mathrm{thin}} = \frac{2P_cR_c}{P_c + R_c}. \tag{5.19}$$

对于这两个指标，数值越大，表明模型的细小结构分割性能越好。

实现细节。 本章实验中使用的 ResNet [176] 和 HRNet [192] 分别是在 ImageNet [178] 上预训练过的 ResNet-50 和 HRNet-18。网络模型在 ThinObject-5K 的训练集上进行训练。训练过程持续 30 个周期，批大小设置为 4。训练时候采用了初始学习率为 7×10^{-3}，每个周期衰减比例 γ 为 0.9 的指数学习率衰减策略。本实验采用了动量值为 0.9、权重衰减为 5×10^{-4} 的随机梯度下降算法进行参数优化。训练阶段，本实验采用随机翻转来增强数据。为了帮助相似性模块识别细小结构间的差异，本章还采用了一种将两张图像拼接成为一张图像的随机拼接策略。测试阶段，使用章节5.2.2介绍的模拟算法生成固定切割线进行评测。值得注意的是，所有评测在根据 0.5 阈值对预测图进行二值化后的结果上进行。

模型推理。 本章在单个 NVIDIA Titan XP GPU 上测试 KnifeCut 的推理时间。采用 HRNet、基于 ResNet 的 DeepLab v3+ 类结构作为特征提取器时，处理一张分辨率为 1024×1024 的输入图像分别需要花费 0.057 秒和 0.281 秒。这两个推理速度都足以应对实际应用中实时计算的需求。

5.4.2 消融实验

本章节开展消融实验来证明局部和全局精细化模式所采用的每个模块的有效性。实验结果展示在表 5.1中。该实验使用 $\mathbf{G}_{\text{non-thin}}$ 作为预分割掩膜，并选择 IoU_{thin} 和 $\mathcal{F}_{\text{thin}}$ 作为评测指标。相似性模块、线距离图等将逐渐从 KnifeCut 方法中移除以证明它们的有效性。此外，该实验用位于细小部分处的一个点击替换切割线交互来展示新交互模式的优越性。基于 HRNet 和 ResNet 的实验都将在 ThinObject-5K、HRSOD 和 COIFT 三个数据集上进行。

相似性模块。 如表 5.1所示，本实验首先移除局部和全局分支中的由相似性模块生成的相似度图。对于局部分支，以 ResNet 为特征提取器的 KnifeCut 方法的 IoU_{thin} 和 $\mathcal{F}_{\text{thin}}$ 指标在三个数据集上大约减少了 3%。这说明相似度图带来的提升效果明显。但是，对于以 HRNet 为特征提取器的 KnifeCut 方法，该提升则相对有限。这可能是因为 HRNet 保持了高分辨率的特征，而 ResNet 在下采样过程中丢失了一些细小部分的信息，所以相似度图中细小部分的激活产生了更大的影响。以上实验结果表明相似度图有助于网络更好地识别细小部分，特别是对于具有低分辨率特征的网络。对于全局分支，去掉相似性模块产生的影响是更加糟糕的。较差的性能证明了相似度图在全局精细化中的重要性。此点也符合

表 5.1 KnifeCut 方法的消融实验。该实验使用 $\mathrm{IoU}_{\mathrm{thin}}$ 和 $\mathcal{F}_{\mathrm{thin}}$ 指标来评价细小部分的分割性能。符号↑表示指标数值越高，模型的性能越好。目标对象的非细小结构部分的真值标注图 $\mathbf{G}_{\text{non-thin}}$ 被采用作为预分割结果。符号 → 表示交互模式的替换操作。

		候选项	ThinObject-5K		HRSOD		COIFT	
			$\mathrm{IoU}_{\mathrm{thin}}\uparrow$	$\mathcal{F}_{\mathrm{thin}}\uparrow$	$\mathrm{IoU}_{\mathrm{thin}}\uparrow$	$\mathcal{F}_{\mathrm{thin}}\uparrow$	$\mathrm{IoU}_{\mathrm{thin}}\uparrow$	$\mathcal{F}_{\mathrm{thin}}\uparrow$
		预分割结果（$\mathbf{G}_{\text{non-thin}}$）	.000	.000	.000	.000	.000	.000
ResNet	局部模式	最终	.637	.776	.400	.674	.448	.667
		无相似度图	.604	.738	.371	.629	.416	.640
		无相似度图和线距离图	.554	.692	.164	.283	.279	.447
		切割线 → 点击	.584	.692	.361	.575	.386	.524
	全局模式	最终	.798	.926	.497	.809	.671	.903
		无相似度图	.780	.910	.285	.497	.600	.838
		切割线 → 点击	.793	.916	.489	.788	.667	.894
HRNet	局部模式	最终	.661	.785	.413	.659	.493	.713
		无相似度图	.655	.778	.403	.659	.482	.702
		无相似度图和线距离图	.618	.749	.206	.339	.403	.602
		切割线 → 点击	.615	.716	.379	.578	.439	.578
	全局模式	最终	.821	.932	.504	.792	.688	.922
		无相似度图	.803	.915	.297	.488	.628	.850
		切割线 → 点击	.822	.931	.498	.779	.690	.918

本章的预期假设，在图像上所有相似的细小部分被激活后，网络可以准确地估计细小部分范围并专注于对它们进行精细化。表中模型性能下降程度不同可以归因于数据集分布的不同。HRSOD 数据集由环境复杂的真实图像组成，因此增加了分割难度。而 ThinObject-5K 测试集的数据分布与网络训练数据相同，并且仅包含具有明显目标边界的合成图像。因此，前者数据集上的性能提升能更好地说明相似度图的必要性。相似度图的另一个不能被数据所反映的优点是可以区分不同类型的细小部分。如果忽视类型特征而激活所有细小部分，网络会混淆要精细化的位置，并可能将它们全部分割，这在大多数情况下会违反用户的意图。图 5.5 展示了相似度图用于区分各类型细小结构的可视化图。顶部图像中，切割线在笼子上时，只有笼子被激活，鸟尾巴也是如此。底部图像同样反映了不同目标的相似性区别，弓和箭根据切割线位置不同分别被单独激活。

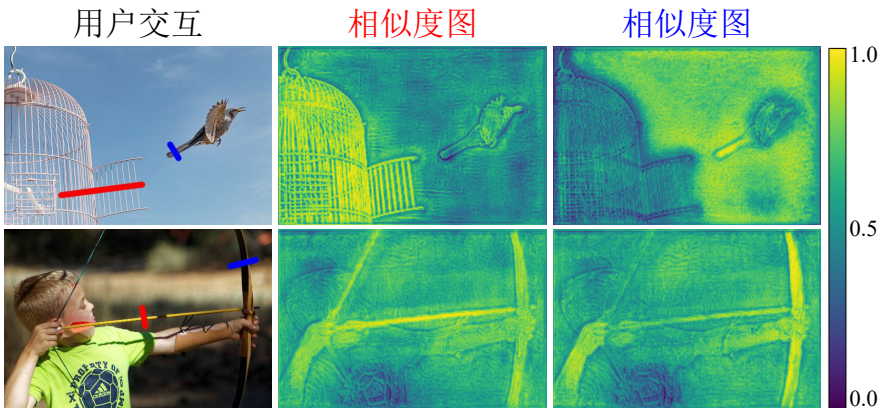


图 5.5　切割线作用在不同位置的相似度图可视化。第二列和第三列的相似度图分别对应红线和蓝线。相似度图的激活区域与交互位置的细小部分高度相关。

线距离图。　对于局部分割精细化模块，该实验进一步去除了线距离图。如表 5.1所示，模型的性能在三个数据集中不同程度地出现了下降。在不同数据集上的下降比例不同的原因与相似性模块类似。因此，这一实验也证明了线距离图在局部精细化中的主导作用。在该图指导下，网络模型可以判断细小结构部分的局部范围，并专注于掩膜区域的分割精细化。

切割线交互。　作为 KnifeCut 方法的一项重要贡献，本章还开展了交互模式的消融实验。由于可以将点击视为线条的退化形式，因此本章在 KnifeCut 的消融实验中使用一次点击替换切割线。遵循前背景点击的交互模式的惯例 [19]，点击被放置在最大错误区域的中心。表 5.1 中展示了对应的实验结果。可以看到，模型在 $\mathrm{IoU}_{\mathrm{thin}}$ 和 $\mathcal{F}_{\mathrm{thin}}$ 两个指标上的性能都在一定程度上变差了。与点击相比，切割线更容易穿过多个细小部分，而不仅仅是一个。由于切割线同时穿过前景和背景，因此它还可以提供对比先验信息。此外，绘制切割线对用户来说方便直观，对大多数设备，例如鼠标、触摸板和移动设备，也都很友好，而对于点击来说，仔细瞄准针对每个细小部分既费时又费力。因此，切割线交互作为针对细小结构的交互模式，能够有效减轻用户负担，并且达到良好的性能。

5.4.3　性能分析

本章节将 KnifeCut 方法的实验结果与其他方法进行了比较并对可视结果进行了分析。具体包括使用 $\mathrm{IoU}_{\mathrm{thin}}$ 和 $\mathcal{F}_{\mathrm{thin}}$ 的细小结构分割指标进行对比，针对多种细小结构情况的分割结果分析，以及更多分割结果的展示和描述。

表 5.2 KnifeCut 方法和其他方法的细小分割指标对比。IoU_{thin} 和 $\mathcal{F}_{\text{thin}}$ 指标被用来评测细小结构的分割性能。*R* 和 *H* 分别表示基于 ResNet 的 DeepLab v3+ 类结构和 HRNet 作为特征提取器。符号 † 和 § 分别指的是局部模式和全局模式。

方法	交互模式	ThinObject-5K		HRSOD		COIFT	
		$\text{IoU}_{\text{thin}}\uparrow$	$\mathcal{F}_{\text{thin}}\uparrow$	$\text{IoU}_{\text{thin}}\uparrow$	$\mathcal{F}_{\text{thin}}\uparrow$	$\text{IoU}_{\text{thin}}\uparrow$	$\mathcal{F}_{\text{thin}}\uparrow$
FCA-Net [186]	4 交互点	0.645	0.776	0.372	0.618	0.498	0.726
(*R*) KnifeCut†	2 交互点 + 1 切割线	0.758	0.887	0.488	0.814	0.601	0.869
(*R*) KnifeCut§	2 交互点 + 1 切割线	0.811	0.927	0.519	0.850	0.678	0.932
(*H*) KnifeCut†	2 交互点 + 1 切割线	0.770	0.888	0.501	0.800	0.623	0.888
(*H*) KnifeCut§	2 交互点 + 1 切割线	0.826	0.927	0.532	0.843	0.694	0.939
f-BRS [39]	4 交互点	0.784	0.872	0.455	0.713	0.631	0.855
(*R*) KnifeCut†	2 交互点 + 1 切割线	0.812	0.914	0.502	0.823	0.666	0.914
(*R*) KnifeCut§	2 交互点 + 1 切割线	0.828	0.929	0.523	0.850	0.692	0.939
(*H*) KnifeCut†	2 交互点 + 1 切割线	0.823	0.916	0.512	0.805	0.679	0.921
(*H*) KnifeCut§	2 交互点 + 1 切割线	0.840	0.931	0.535	0.847	0.704	0.943
TOS-Net [46]	4 交互点	0.865	0.938	0.651	0.916	0.764	0.962
(*R*) KnifeCut†	1 包围框 + 1 切割线	0.874	0.944	0.666	0.916	0.772	0.965
(*R*) KnifeCut§	1 包围框 + 1 切割线	0.873	0.946	0.664	0.917	0.786	0.968
(*H*) KnifeCut†	1 包围框 + 1 切割线	0.875	0.945	0.668	0.917	0.775	0.966
(*H*) KnifeCut§	1 包围框 + 1 切割线	0.877	0.946	0.670	0.917	0.793	0.969

细小结构分割指标对比。 如表 5.2所示，本章在 ThinObject-5K、HRSOD 和 COIFT 三个数据集上对比了 KnifeCut 方法和其他方法的性能。由于本章的方法是专门为细小结构而设计的，所以本章只使用关于细小部分的评测指标。这些指标已在章节5.4.1 中进行了详细介绍。至于一般物体的交互式分割方法，本章选择了代码维护良好的 FCA-Net [186] 和 f-BRS [39] 方法进行对比。为了保正这些方法的公平性，这些方法的网络模型使用了 ThinObject-5K 训练集进行了微调。本章对这些方法使用两次点击，并采用它们的结果作为 KnifeCut 方法的预分割结果。由于切割线可以被视为两次点击（两个端点），KnifeCut 方法同这些方法在 4 次点击时的分割结果进行了对比。可以看到，无论是局部还是全局的细小部分精细化，KnifeCut 方法的提升都相当明显。这不仅证明了该方法作为精细化修复工具的可行性，而且表明了切割线处理细小部分分割的适用性和有效性。

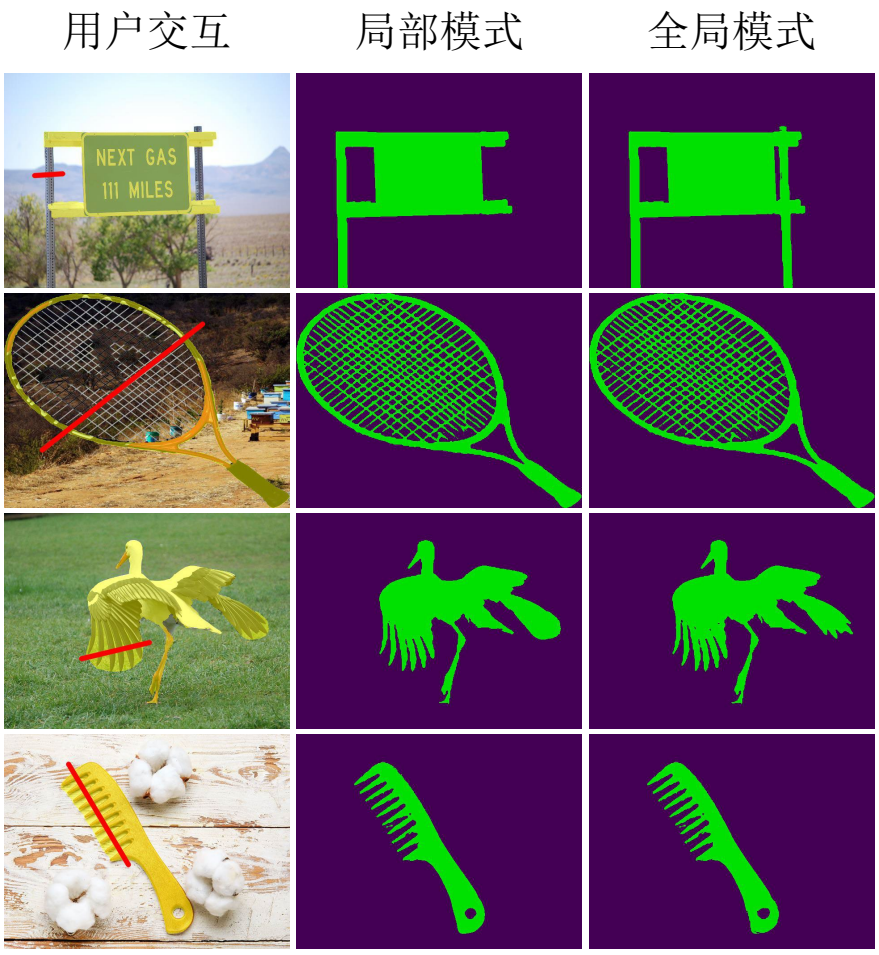


图 5.6 KnifeCut 方法的可视结果。左列的黄色掩膜展示了预分割结果。前两行样例主要针对分割缺失情况，后两行样例主要针对分割过度情况。

对于交互式细小结构分割，本章还将 KnifeCut 方法与 TOS-Net [46] 方法进行了比较。同样为了控制相同的点击次数，本章使用边界框（对角线位置的两次外部点击）作为交互模式重新训练了与 TOS-Net 相同网络结构的模型。模型得到的结果将作为预分割结果，这样总的交互次数仍然可以控制为 4。经过 KnifeCut 精细化修复后，细小部分的分割得到了改进，并且超过了 TOS-Net 方法。

分割结果分析。 图 5.2中展示了预分割结果可能面临的分割缺失和分割过度情况的样例。与这些情况对应，图 5.6展示了 KnifeCut 方法面对这些情况时的局部修复与全局修复的结果图。当细小部分未被分割时，KnifeCut 方法可以精细化预分割结果。用户通过绘制一条穿过未分割细小部分的切割线可以获得出色的

修复结果。例如图中第一行，标志牌的左右支撑杆的分割都缺失了，用户绘制一条切割线穿过左支撑杆。作为局部结果，左支撑杆的分割将被修复。而右支撑杆的修复则会包括在全局修复结果中。更复杂的情况如第二行所示，球拍缺少了网线的分割，使用现有的工具来进行修复是负担极大的。幸运的是，即使在这种困难的情况下，KnifeCut 方法仍可以采用同样的方式解决，而无需太多交互时间成本。在另一种情况下，细小部分由于太靠近而导致了分割过度。第三行以鸟的翅膀为例来说明情况，用户只要绘制切割线穿过分割过度的一部分翅膀，然后就可以在局部模式下获得目标翅膀细小结构精细化后的结果，在全局模式获得双翅精细化后的分割结果。第四行中的梳子的锯齿过于紧密，此情况下 KnifeCut 方法的修复效果依然良好。由于图中第二行的球拍和第四行的梳子不存在多个细小部分，因此在局部模式和全局模式下，二者的修复结果相差不大。

更多分割结果展示。 为了更加直观地说明 KnifeCut 方法的分割修复效果以及本章提出的切割线模拟算法的有效性，在图 5.7和图 5.8中，本章展示了更多基于该方法的样例修复结果，其中都包含了局部模式和全局模式的结果。这些样例均选自 ThinObject-5K、HRSOD 和 COIFT 数据集，且它们的预分割结果对于细小结构都表现不佳。图中的切割线是由模拟算法根据真值标注图和预分割结果生成的。从图上的切割线位置与形态可以看出，这些切割线都以较为垂直的角度穿过了分割差的细小结构区域。由此可以看出本章提出的模拟算法较为合理，且符合人类的直觉。图 5.7展示了采用非细小结构标注作为预分割的实验结果图。图中目标的细小结构往往整体缺失。可以看出，使用了 KnifeCut 方法后，局部模式下的部分细小结构得到了修复，全局模式下所有的细小结构都被分割出来。为了更符合现实场景中的实际情况，图 5.7展示了采用真实分割结果作为预分割的实验结果图。其真实分割结果来自于在 ThinObject-5K 训练集微调后的 FCA-Net [186] 和 f-BRS [39] 方法. 该实验选取了这两个方法两次点击后的结果作为预分割结果。这些预分割结果往往已经分割出一小部分的细小结构，此时模拟的切割线则更加针对未分割的细小结构部分。从实验结果可以看出，这些真实分割结果中的细小结构区域也得到了有效的修复。这说明该方法可以用作真实场景中的图像分割修复。由于本章提出的方法只针对细小结构的分割进行修复，而不会影响原始图像中非细小结构的分割结果，所以即使在某些情况下 KnifeCut 方法的结果不能令人满意，对原先的分割也几乎没有影响。

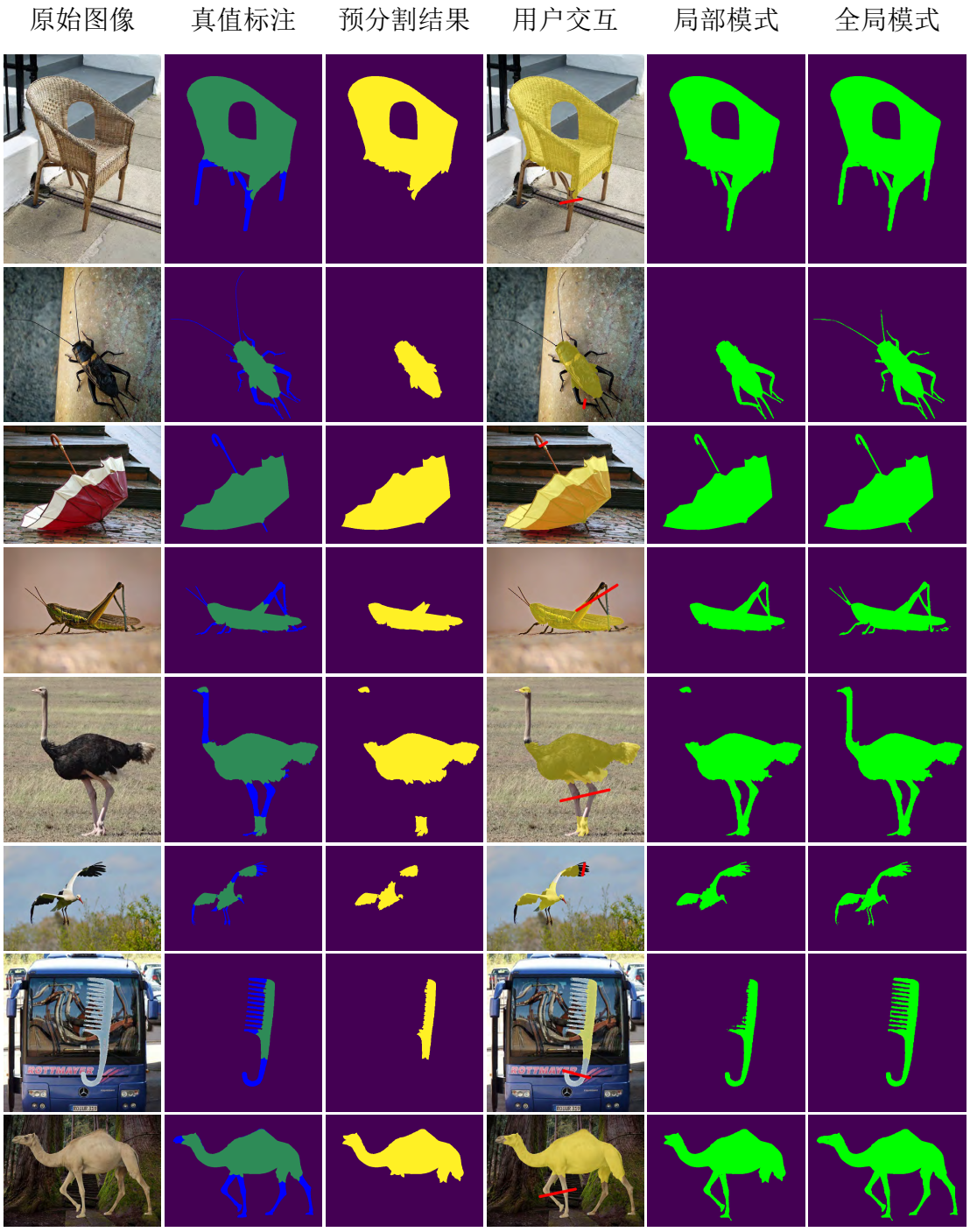


图 5.7　KnifeCut 方法的以非细小结构标注作为预分割的实验结果图。

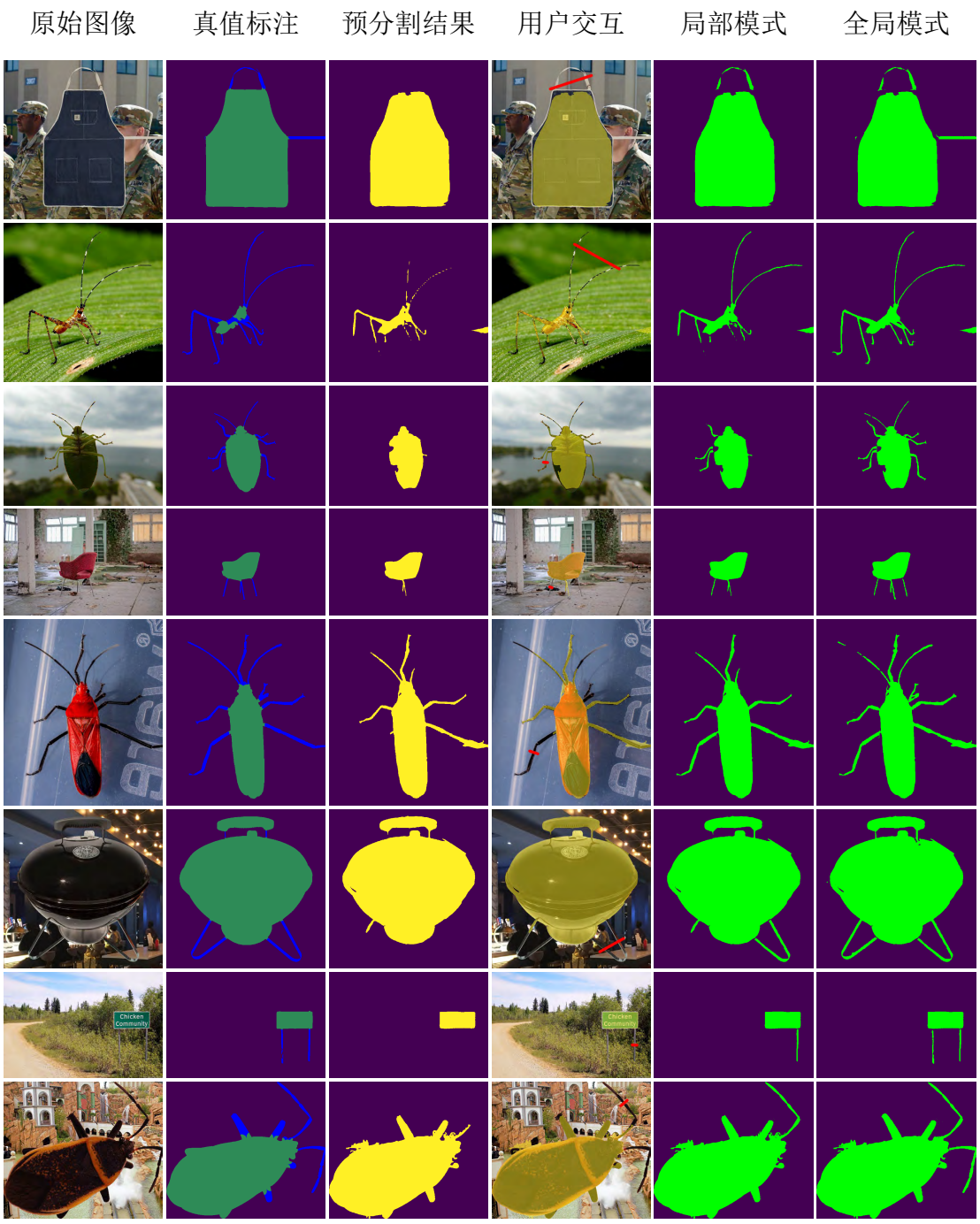


图 5.8　KnifeCut 方法的以真实分割结果作为预分割的实验结果图。

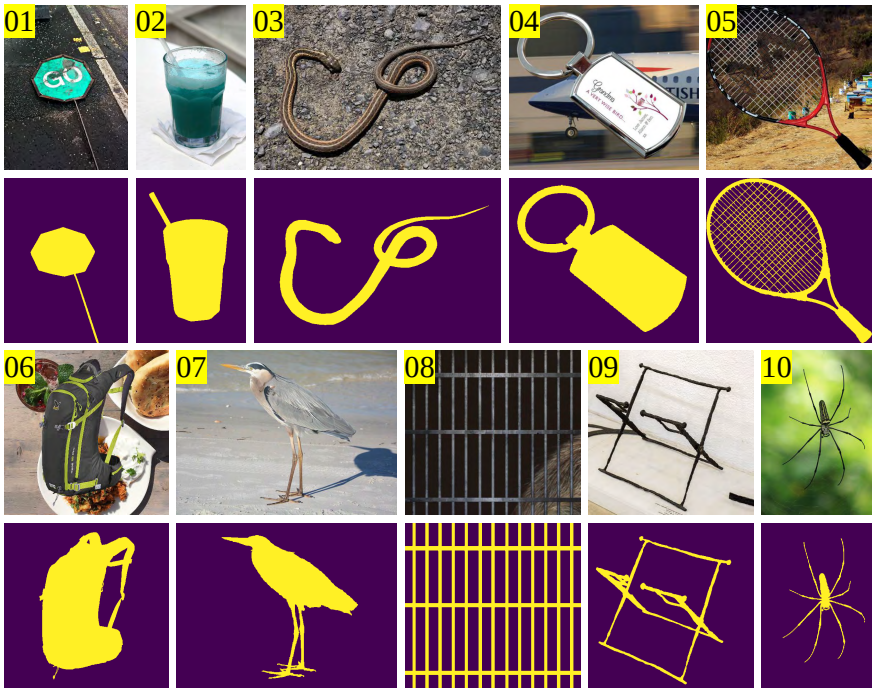


图 5.9　KnifeCut 方法的用户调研选取的样本图像展示。其中涵盖了各类型的细小结构物体。

5.4.4　用户调研

由于本章提出的方法是一种新的交互模式，进行真实的用户调研有助于评判该方法的实际使用效果。本章节首先描述了用户调研的设置，然后将切割线交互模式与其他交互模式进行了交互用时的对比，最后对于本章提出的切割线模拟算法和真实用户绘制的切割线进行了比较与分析。

用户调研设置。　为了进一步证明本章提出的 KnifeCut 方法的友好性和便利性，本章邀请了二十个人参加用户调研。本章精心挑选了十个物体样本，其中包含各种各样的细小结构物体，同时利用这些样本的真值标注图获得非细小结构的真值标注图作为预分割结果。选定的十个物体样本展示在图 5.9中。可以看到，前四幅图像对应于单个细小结构的情况，而且它们包含不同的形状，例如棒形、曲线形和环形等。后六幅图像是更复杂的情况，它们在细小结构的分布上各有不同。例如，存在类似于样本 05 和样本 08 的网格拓扑形状的细小结构，也存在如样本 06 和样本 07 那样细小结构部分相距太近而无法分离的情况。还存在如样本 09 和样本 10 那样细小部件具有相似的特征但彼此分散分布的情况。除了

表 5.3 KnifeCut 方法的用户调研中多种交互模式的交互时间对比。表中展示了四种交互工具用时的均值和方差。60.00 秒代表用户使用对应的交互工具无法达到指定的目标 IoU_{thin}。

样本	切割线	点击	多边形	笔刷
01	1.03 ± 0.52	9.61 ± 6.92	45.85 ± 19.81	60.00 ± 0.00
02	0.81 ± 0.48	0.13 ± 0.19	9.81 ± 10.97	11.35 ± 5.65
03	6.90 ± 4.18	12.36 ± 5.25	59.42 ± 1.90	56.56 ± 8.57
04	0.80 ± 0.36	0.08 ± 0.01	44.39 ± 14.29	32.69 ± 11.94
05	1.18 ± 1.11	60.00 ± 0.00	60.00 ± 0.00	60.00 ± 0.00
06	0.81 ± 0.31	2.86 ± 2.32	48.05 ± 12.58	49.28 ± 15.53
07	0.68 ± 0.19	60.00 ± 0.00	60.00 ± 0.00	60.00 ± 0.00
08	1.11 ± 0.90	1.91 ± 1.08	60.00 ± 0.00	60.00 ± 0.00
09	3.16 ± 3.21	9.08 ± 3.97	60.00 ± 0.00	60.00 ± 0.00
10	2.09 ± 1.34	1.63 ± 0.61	60.00 ± 0.00	60.00 ± 0.00
全部	1.85 ± 2.54	15.77 ± 22.75	50.75 ± 17.61	50.99 ± 17.05

切割线交互外，本实验还让用户使用其他三种交互式分割方法对目标样本进行标注，分别为点击、多边形和笔刷交互模式。点击交互模式采用了 FCA-Net 方法 [186]；多边形交互模式是让用户可以绘制多个多边形包围细小区域进行分割；笔刷交互模式指用户可以直接对细小区域进行涂鸦。当用户在 60 秒内能使目标物体的 IoU_{thin} 指标超过 70% 则停止交互并记录时间，否则则将时间记作 60 秒。

交互模式对比。 本章记录了 20 名参与者分别使用四种交互工具达到目标 IoU_{thin} 所需的时间。考虑到每张图像对应于不同的细小结构情况，本章分别计算了每个样本的平均时间和方差，结果显示在表 5.3中。如图 5.10所示，根据记录的数据，本章还绘制了对应的箱线图来直观反映不同交互模式的时间分布。可以看到，对于单个细小结构的情况，笔刷和多边形只能勉强处理，但对于更复杂的情况，它们无能为力，几乎都以失败告终。基于前背景点击的方法比这两个工具更好。值得注意的是，本实验从参与者的第一次鼠标点击开始计时。因此，用时不包括用鼠标瞄准细小部分所消耗的时间。在这种情况下，基于点击的方法对于那些可以通过一次点击进行精细化的图像（如样本 02 和样本 04）来说，可以获得出色的性能。但是，当物体前景和背景相似且难以区分时，如样本 01、样本 04、样本 05 和样本 07，基于点击的方法耗时更长，甚至会以失败告

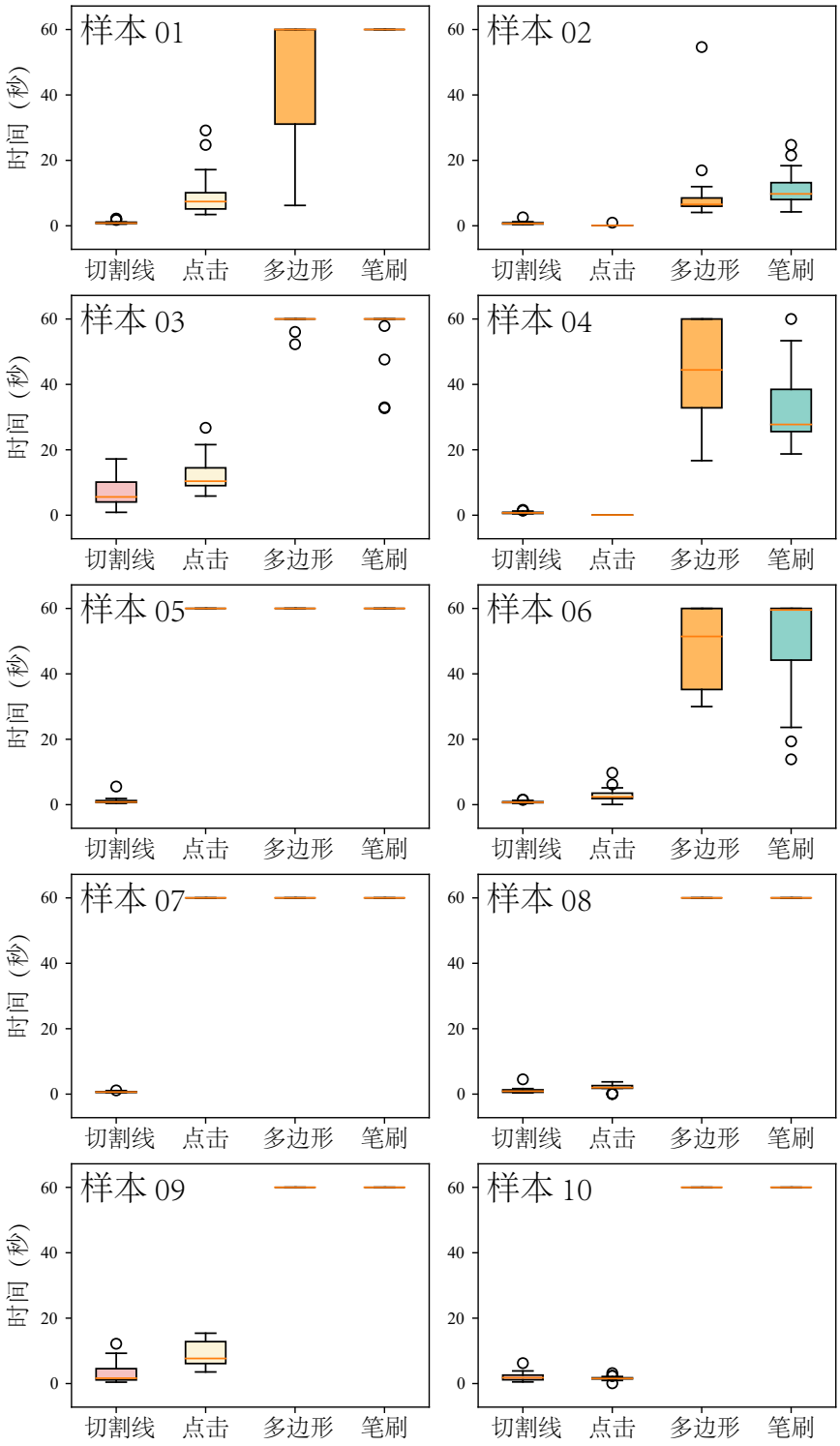


图 5.10　KnifeCut 方法的用户调研中不同交互模式所用时间的箱线图。

表 5.4 KnifeCut 方法的用户调研中关于模拟和真实切割线的性能对比。

	模拟算法生成的切割线		真实用户绘制的切割线	
	$\mathrm{IoU}_{\mathrm{thin}}\uparrow$	$\mathcal{F}_{\mathrm{thin}}\uparrow$	$\mathrm{IoU}_{\mathrm{thin}}\uparrow$	$\mathcal{F}_{\mathrm{thin}}\uparrow$
样本 01	0.702	0.940	0.801	0.986
样本 02	0.960	0.996	0.957	0.992
样本 03	0.664	0.731	0.502	0.581
样本 04	0.949	0.995	0.942	0.993
样本 05	0.787	0.999	0.779	0.998
样本 06	0.733	0.760	0.719	0.805
样本 07	0.724	0.933	0.679	0.936
样本 08	0.871	0.988	0.850	0.976
样本 09	0.725	0.855	0.696	0.830
样本 10	0.803	0.963	0.740	0.908
全部样本	0.792	0.916	0.767	0.900

终。因此，这种方法平均耗时 15.77 秒，但波动幅度很大，为 22.75 秒。关于本章提出的 KnifeCut 方法，从表 5.3中可以看出，该交互适用于所有细小结构的情况，平均只需要 1.85 秒。对于像样本 01 和样本 02 这样的简单情况，一条切割线足以修复其细小结构。即使是像样本 05 和样本 07 这样复杂的情况，虽然它们不能通过基于点击的方法来达到目标性能，但通过 KnifeCut 方法则可以很好处理，而且不会显著增加时间。这些在图 5.10中反映得更加直观。其中 KnifeCut 方法对 10 个样本的交互时间相对较短且波动较小。总而言之，用户调研表明，与其他三个交互工具相比，KnifeCut 方法具有更好的适应性、稳定性和友好性。

模拟算法与真实用户结果对比。 除了与其他交互工具的比较外，该用户调研还对本章提出的切割线模拟算法的合理性进行了研究。对于相同的 10 个样本，用户调研记录了参与者用 KnifeCut 方法进行第一次切割线交互得到的 $\mathrm{IoU}_{\mathrm{thin}}$ 和 $\mathcal{F}_{\mathrm{thin}}$ 的值。这些指标的数值经过平均计算后显示在表 5.4中。使用模拟算法生成的切割线得到的性能结果也展示在该表中。根据 10 个样本的总体平均值，可以看到模拟算法生成的切割线比真实用户绘制的切割线具有更好一点的性能，$\mathrm{IoU}_{\mathrm{thin}}$ 和 $\mathcal{F}_{\mathrm{thin}}$ 分别提高了约 0.3 和 0.2。这是可以理解的，因为切割线模拟算法被设置为切割细小结构的最大的错误区域，这将对细小结构的分割结果产生很

大的影响。特别是当面对像样本 01 和样本 03 那样的细小部分过长或者像样本 10 那样的细小部分分散的情况时，模拟算法生成的切割线和真实用户绘制的切割线可能存在较大不同。但这也一定程度上说明了 KnifeCut 方法对交互多样性的容忍度。除了这些情况外，模拟算法生成的切割线和用户绘制的切割线几乎达到了相同的效果，这证明了在测试过程中应用该算法的合理性。这同时也说明了本章提出的该模拟算法迎合了用户操作习惯，很好地模拟了真实用户的交互。

5.5 本章小结

针对交互式图像分割任务中，复杂场景下的细小结构交互难的问题，本章提出了一种新颖的交互模式，即切割线交互模式，来分割复杂场景下遇到的细小结构物体。该模式只需要用户绘制一条穿过细小部分错误区域的切割线来进行交互。与目前的交互模式相比，用户易于通过鼠标、触摸板和移动设备来进行控制，使交互负担得到了有效的减轻。为了充分探索该交互模式的优越性，本章基于该交互模式提出了一种交互式分割方法，命名为 KnifeCut。它不仅可以从头分割细小结构的物体，还可以进一步精细化其他分割方法获得的预分割结果。在三个数据集上的实验均证明了 KnifeCut 方法作为标注工具和修复工具的优越性。有了该方法，复杂场景下的细小结构物体分割负担将得到有效减轻。

6 面向医学图像的多模式交互式分割

对于交互式图像分割任务，用户有时候会遇到一些特殊图像，比如医学图像。这些图像由于非自然场景，常常具有低对比度特征。而针对这类图像中的目标对象使用一般的交互式分割方法难以准确预测分割掩膜。由于面向自然图像的不同交互方式各有优点，要想对医学图像中的目标进行精确预测，则需要设计一个结合多种交互方式的统一模型。面对复杂场景中的医学图像分割差这一难点，以针对低对比度的医学图像分割为目标，本章提出了面向医学图像的多模式交互式分割。该工作设计了一个统一的框架，命名为 MMIIS，它结合了多种初始交互模式和修复交互模式，使用户可以根据不同医学目标的特征，自由选择交互模式并高效率分割出目标对象。实验证明，本章提出的方法对于交互式图像分割任务中针对低对比度的医学图像分割具有显著作用。在本章中，首先，章节6.1对该工作的背景、动机、贡献等进行了介绍。其次，章节6.2展示了该工作提出的多模式的交互方式及其各种模式的模拟算法。之后，章节6.3详细描述了该工作基于多模式交互方式提出的 MMIIS 方法的模型结构。然后，章节6.4描述了实验设置，进行了消融实验，结合其他方法进行了性能对比与分析，并且对 MMIIS 方法进行了用户调研。最后，章节6.5对该工作进行了总结。

6.1 本章引言

交互式医学图像分割旨在使用较少的交互来标识出医学图像中用户感兴趣的区域。该任务在目前的计算机辅助诊断系统中起着至关重要的作用。它可以帮助医护人员从头开始或基于粗略的自动分割结果精确地突出特定区域来进行疾病进展的定量测量 [25]。除此之外，它对于各种计算机辅助诊断模型 [194–196] 的训练数据标注也是必不可少的。但考虑到医学影像的特殊性，只有训练有素、经验丰富的工作人员才能胜任对应的标注工作，往往这样的专业人员非常稀缺。尤其是在发生如新冠肺炎大流行等紧急医疗事故时，在短期内，寻找专业人员为计算机辅助诊断系统标注足够的数据既困难又昂贵。以较小的交互成本，快速、准确地完成像素级标注的交互式医学图像分割任务对于当今医疗行业是非常重要的。因此该任务值得科研人员进行相关的研究和探索。

几十年来，科学界一直致力于这项研究 [197]。到目前为止，为了高效地获取用户的交互意图，研究人员提出了多种方法来接收用户输入的包围框 [51, 198]、极值点 [152]、点击与涂鸦 [52, 53, 55] 等交互信息。与自然图像不同，很多医学图像存在低对比度特性，这也造成了医学目标中存在的各种歧义性 [26]，比如不规则的形状 [199]、模糊的边界 [200] 等。对于目前的交互式医学图像分割方法，单一的交互模式不能有效地应对这些歧义性。例如，很难使用包围框来精确定位不规则形状的区域，也很难使用区域点击或涂鸦来定位模糊边界下的目标轮廓。理想的交互式医学图像分割框架应该能灵活地选择多种交互模式，并允许它们相互协作以克服多种医学目标的歧义性，从而高效地进行标注。然而这种潜在的有效方法直到现在还没有得到很好的研究。

为了解决交互式医学图像分割的问题，本章通过探索不同的交互模式，提出了一个统一的多模式交互式分割框架，称作 MMIIS，以实现灵活、准确地分割医学目标。图 6.2中展示了该框架的网络结构图。其中初始分割模块接收包围框、包围多边形或包围涂鸦来定位目标并生成初始预测。用户可以根据目标区域的结构复杂程度灵活选择初始交互模式。在这些模式中，包围框的时间成本较低，而包围多边形和包围涂鸦通过引入更多交互，使用户交互更接近目标轮廓，可以有效降低歧义性。如果用户对初始分割结果不满意，该框架的细节修复模块将协同使用区域和边界的点击或涂鸦来修复错误标记的分割结果。将区域和边界的交互相结合以相互作用，改进了现有的基于区域 [51–53] 和边界 [20, 55, 149] 的局部修复方法无法解决的多重歧义性问题。除了修复初始分割结果，细节修复模块还支持修复其他自动分割算法生成的粗糙分割图，这将是一个减少用户交互负担的重要特性。为了验证 MMIIS 方法的有效性，本章将其应用于多个医学图像分割任务，如新冠肺炎感染和皮肤病变等。六种不同类型医学图像上的实验结果和用户调研表明，该框架在面对不同医学目标时候的有效性。

该工作的贡献可以总结如下：

1. 提出了一种多模式交互方式的思想，用户可以自由选择并联合使用多种交互模式来处理医学目标分割中存在的多种歧义性问题。
2. 基于该思想，本章提出了 MMIIS 方法，其提供了不同的初始交互模式用来准确定位目标，区域-边界协同的修复交互模式用以快速修复分割结果。
3. 六个不同类型的医学数据集上的实验结果和真实的用户调研证明了 MMIIS 方法对于交互式医学图像分割的便捷性和有效性。

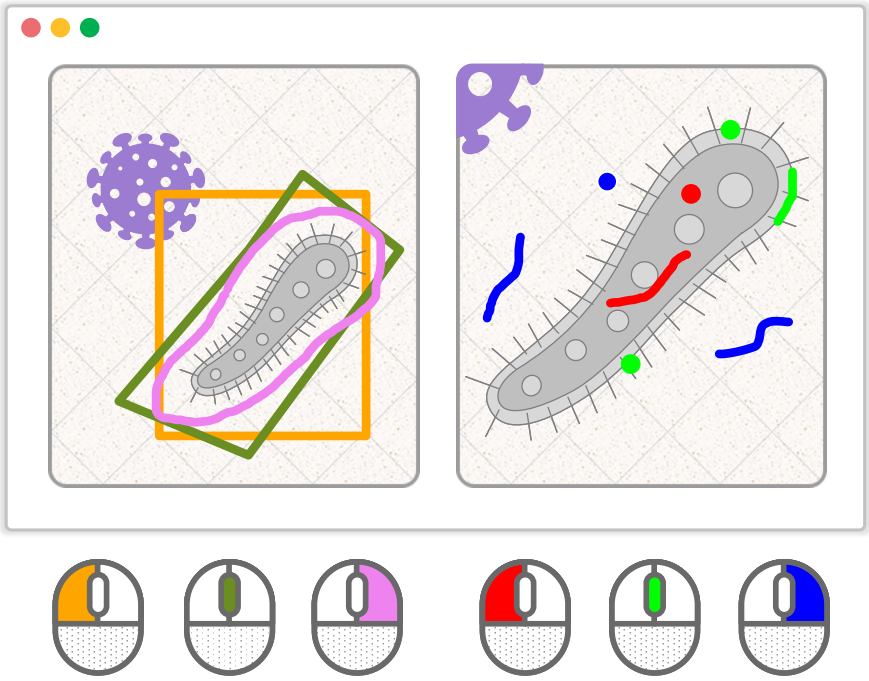


图 6.1 多模式交互方式及其用户界面原型。用户首先在左侧窗口提供初始交互。包围框/包围多边形/包围涂鸦可以通过鼠标左键/中键/右键进行点击或拖动获得。然后，右侧窗口将提供放大后的局部区域。前景/边界/背景的修复交互可以通过鼠标左键/中键/右键进行点击或拖动绘制涂鸦。修复过程可以不断迭代进行，直到得到用户满意的结果。

6.2 多模式交互方式

本章节包括两个部分。章节6.2.1介绍了提出的多模式交互方式及其用户界面原型。章节6.2.2则详细描述了训练和测试中各种交互模式的模拟算法。

6.2.1 交互模式介绍

如图 6.1和图 6.2所示，MMIIS 方法的交互模式主要分为两部分，分别为初始交互模式和修复交互模式。初始交互模式用来对医学目标进行定位和得到粗糙的分割结果。修复交互模式则用来对初始交互模式得到的或者用户及其他方法提供的粗糙分割结果进行修复以得到最终的精细标注掩膜。

初始交互模式。 如图 6.1中的左侧界面及图 6.2（a）中所示，MMIIS 方法为用户提供了便捷的方式来定位目标区域以分割医学对象。它支持包围框、包围多边形和包围涂鸦。初始指导图可以由这些交互方式生成。它由三个通道组成，会根据交互模式激活其中一个通道。具体地说，当用户采用某一种交互模式时，

根据绘制方式的内外区域，将对应通道内的像素设为 1 和-1，而其他不相关的通道设为 0。这些通道将被送入初始分割模块。初始交互模式的介绍如下：

- **包围框**：包围框是一个围绕目标区域的矩形。作为一种低成本的交互模式，用户可以通过拖拽来确定矩形对角线上的两个端点。这种交互模式可以处理大多数拥有圆形或矩形类形状的目标。
- **包围多边形**：包围多边形是对包围框的自然扩展方式，其将矩形替换为任意多边形。多边形比矩形更容易贴合目标区域。由于减少了无关背景的干扰，所以它比包围框更容易起到引导作用。但这些好处是以更多的交互时间为代价的，因为它需要用户确定多边形的所有顶点。
- **包围涂鸦**：包围涂鸦是对包围多边形的进一步扩展，可以更灵活地包围目标。当目标区域轮廓很复杂时，包围涂鸦可以更好地包裹区域，并提供目标形状的先验信息。此种交互比包围多边形需要更大的交互时间代价。

图 6.1中的左侧窗口展示了初始交互模式的界面原型。在该用户界面中，用户可以用鼠标左键拖拽出一个包围框，也可以单击中键来提供包围多边形的顶点，或者长按右键并拖动来绘制包围涂鸦。之后用户界面将在初始交互模式的基础上在右侧窗口显示局部放大的图像和初始分割结果。

修复交互模式。　如图 6.1中的右侧界面及图 6.2（b）中所示，在使用初始交互模式或者通过自动类图像分割算法获得粗糙分割结果后，MMIIS 方法为用户提供了多样的方式来迭代修复分割结果。它支持前景区域、背景区域和边界上的点击或涂鸦。这三种位置的交互会分别激活对应的三个不同的修复指导图通道。具体地说，当用户采用某一种交互模式时，会在对应的通道内绘制高斯距离图。对于前景和背景区域的交互，高斯距离图的半径为 80 像素，而对于边界的交互，半径则为 10 像素。在图 6.2（b）中，本章展示了这个三通道修复指导图的可视化图像，可以看到，前背景区域交互会比边界交互的影响范围更大。这些通道将随着粗糙分割结果被送入细节修复模块。修复交互模式介绍如下：

- **区域点击/涂鸦**：用户可以根据初始分割模块或其他方法得到的结果，使用区域点击/涂鸦纠正错误的区域。前景区域的交互能分割出更完整的目标，背景区域的交互能消除错误预测的局部掩膜。与涂鸦相比，点击错误标记的前景和背景是一种低成本的交互模式。而涂鸦则比点击包含更多的指导信息。对于前景和背景的交互，两者的指导图是相互独立的。

- **边界点击/涂鸦**：医学图像经常面临严重的边界模糊问题。对于目标边界，区域的点击和涂鸦无法给出有效的约束。只有通过不断使用前景、背景区域的交互来收紧边界，才能实现边界的精确定位，这在实际的交互中是费时费力的。边界点击和涂鸦可以直接绘制在目标边界上，以达到准确定位边界的作用。这在交互过程中对于目标形状的准确分割具有重要作用。

图 6.1中的右侧界面展示了修复交互模式的界面原型。在该用户界面中，用户可以使用鼠标左键和右键进行前景和背景的区域交互，还可以使用鼠标中键进行边界交互。单击对应按键可以产生点击交互，而长按拖拽可以产生涂鸦交互。用户可以在右侧窗口中反复添加交互来修复分割结果，直到满足用户的需求。

6.2.2 交互模拟算法

为了对模型进行训练和评估，MMIIS 方法需要一个机器人用户来模拟真实用户的交互。在本章中，$\mathcal{I}$ 代表所有图像像素; $\mathcal{P}$ 代表预测图的前景像素; $\mathcal{G}$ 代表真值标注图的前景像素。本章使用 $\phi(p,S)$ 来表示点 p 到点集 S 的最短距离。本节将简要介绍不同交互模式下的机器人用户模拟策略。

初始交互模式。 对于初始交互模式，当用户使用鼠标包围较小的物体区域时，由于操作误差，包围框/包围多边形/包围涂鸦会有较大的松弛度；而当目标区域较大时，操作误差相对较小，此时包围交互将更接近目标对象。为了在模拟和实际使用中更符合用户习惯，本章在训练和测试中设置了一个参数 ξ ：

$$\xi_\gamma = \gamma \cdot \sqrt{|\mathcal{G}|} \cdot \ln\left(|\mathcal{G}|/|\mathcal{I}|\right), \tag{6.1}$$

其中 γ 是一个在不同交互模式中变化的调节参数（对于包围框是 0.3，对于包围多边形是 0.2，以及对于包围涂鸦是 0.1）。偏移量 ξ 在训练过程中被设置成一个 $[0,\xi_\gamma]$ 区间内的随机数字，在测试过程中为了稳定评估被设置成一个固定值 $\xi_{\frac{\gamma}{2}}$。具体的初始交互模式模拟算法如下：

- **包围框**：在包围目标的紧致矩形基础上生成带有偏移量 ξ 的松弛矩形。
- **包围多边形**：首先使用凸包算法 [201] 寻找到包围目标的最小凸包多边形。然后将这些顶点和随机偏移 ξ 后的多边形顶点作为一个新的点集。通过重新计算新点集的凸包，可以得到一个不规则包围多边形。
- **包围涂鸦**：对目标对象掩膜进行膨胀操作，膨胀迭代次数设置成 ξ。使用膨胀后的掩膜的轮廓边界作为包围涂鸦。

修复交互模式。 对于修复交互模式，训练时采用 0.5 的概率进行随机采样策略和迭代采样策略，测试时只采用迭代采样策略。对于随机采样策略，区域交互将分别以 0.2 和 0.8 的概率对真值标注图或先前预测结果的错误区域进行随机采样；边界交互则总是采用真值标注图的轮廓边界进行随机采样。对于迭代采样策略，区域交互总是采用先前预测结果的错误预测区域进行固定采样；边界交互总是根据先前预测结果的轮廓边界在目标对象轮廓边界上进行固定采样。具体的修复交互模式模拟算法如下：

- **区域点击**：对于随机采样策略，从前景和背景区域或之前的错误预测区域中随机选择 0 ∼ 5 个前背景点来模拟点击。它们之间相距 25 个像素以上。对于迭代采样策略，在保持上一轮交互点不变的基础上，根据先前预测结果，，新添加最大错误预测区域的中心 p_{region} 作为交互点：

$$p_{\text{region}} = \underset{p \in (\mathcal{P} \Delta \mathcal{G})}{\arg\max}\, \phi\left(p, \overline{\mathcal{P} \Delta \mathcal{G}}\right), \tag{6.2}$$

  其中 Δ 表示集合之间的异或运算，即二者交集减去并集所剩下的集合。
- **区域涂鸦**：对于随机采样策略，本方法首先采用 [191] 中的算法获得目标真值标注图或错误预测区域的骨架，然后通过模板匹配，获取骨架上的端点集合。本方法之后选择骨架上的一个随机端点到另一个随机端点，并使用迪杰斯特拉算法获得这两端点间的最短八连通路径，之后采用其中间段作为涂鸦。对于迭代采样策略，本方法只采用错误预测区域进行该算法，并选取所有端点之间的路径距离的最长者对应的路径来作为涂鸦。
- **边界点击**：对于随机采样策略，边界点会从目标轮廓边界的像素中随机选取。它们之间和区域点击一样相距 25 个像素以上。对于迭代采样策略，边界点将会是距离先前预测结果的轮廓边界最远的目标轮廓边界点 p_{boundary}，其计算公式如下：

$$p_{\text{boundary}} = \underset{p \in \mathcal{G}^b}{\arg\max}\, \phi\left(p, \mathcal{P}^b\right), \tag{6.3}$$

  其中上标 b 代表对应集合的边界像素集合。
- **边界涂鸦**：对于随机采样策略，边界涂鸦是目标轮廓边界的一定比例部分。涂鸦中点在轮廓边界上，且涂鸦长度是整体轮廓长度的 0 ∼ 0.1 倍。对于迭代采样策略，本方法将确保涂鸦的中点与边界点击在该策略下选取的点相同，长度固定为整体轮廓长度的 0.05 倍。

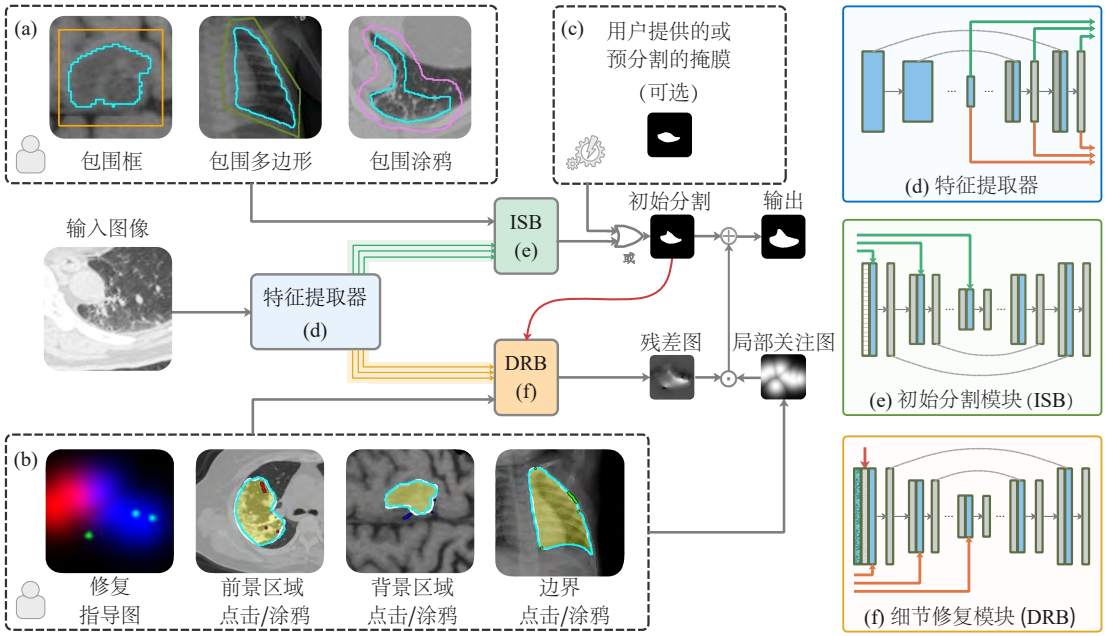


图 6.2　MMIIS 方法的网络结构图。（a）展示了初始交互模式，图中省略了输入初始分割模块中的初始指导图。如（c）所示，初始分割模块的分割结果可以被其他方法自动生成或用户手动标记的区域掩膜所取代。如（b）所示，用户可以在区域和边界提供进一步的交互，然后同初始结果一同输送到细节修复模块进行迭代修复。对于图上的可视图像，黄色掩膜是预测结果，青色轮廓线是目标真值标注图的轮廓边界。图中各交互针对的图像样例不同。

6.3　交互框架与网络模型

本章提出的 MMIIS 方法不仅包含多模式的交互方式，还包含基于多种交互模式设计的统一的交互式医学图像分割模型。图 6.2展示了该模型的网络结构。该模型可以分为三个模块，即特征提取器、初始分割模块（Initial Segmentation Block，ISB）和细节修复模块（Detail Refinement Block，DRB）。对应以上三个模块，本章节分为三个部分。首先，章节6.3.1介绍了特征提取器，然后，章节6.3.2介绍了初始分割模块。最后，章节6.3.3介绍了细节修复模块。

6.3.1　特征提取器

该特征提取器是一种类似 FPN [61] 的编码器-解码器架构。它以裁剪后的医学图像为输入，从解码器中提取多个 32 通道特征。具体来说，该模型采用了 VGG-16 主干网络 [202] 的 5 个卷积层，并额外增加了一个 32 通道的卷积层。该模型将这些特征从底层到高层分别表示为 $\{\mathbf{F}_i\}_{i=1}^{5}$ 和 $\mathbf{G}_6$。对应的解码器特征

$\{\mathbf{G}_i\}_{i=1}^{5}$ 可以从以下公式得到：

$$\mathbf{G}_i = D_i\left((\mathbf{G}_{i+1})\uparrow \oplus R_i(\mathbf{F}_i)\right), \quad i \in \{5,4,3,2,1\}, \tag{6.4}$$

其中，$(\cdot)\uparrow$ 是上采样操作，$\oplus$ 是拼接操作，$R_i(\cdot)$ 代表卷积模块，它将卷积 $\mathbf{F}_i$ 以生成 32 通道的特征，$D_i(\cdot)$ 是解码器中一个包含两个 32 核卷积层的模块。

6.3.2 初始分割模块

初始分割模块接受用户输入的包围框、包围多边形或包围涂鸦来输出粗糙分割结果。如图 6.2（d）所示, 它是一个高效的沙漏形结构，通过相同分辨率的层之间的跨层连接来共享表征。与特征提取器相连接的编码器可以形式化为：

$$\mathbf{F}_i^{\text{ISB}} = \begin{cases} E_i^{\text{ISB}}\left(\mathbf{G}_i \oplus \mathbf{A}^{\text{ISB}}\right), & i = 1 \\ E_i^{\text{ISB}}\left(\mathbf{G}_i \oplus \left(\mathbf{F}_{i-1}^{\text{ISB}}\right)\downarrow\right), & i \in \{2,3,4,5\} \end{cases}, \tag{6.5}$$

其中，$\mathbf{A}^{\text{ISB}}$ 是初始指导图，$(\cdot)\downarrow$ 是下采样操作，$E_i^{\text{ISB}}(\cdot)$ 同样是一个包含两个 32 核卷积层的模块。为了恢复到原始的分辨率，解码器可以表示为：

$$\mathbf{G}_i^{\text{ISB}} = \begin{cases} D_i^{\text{ISB}}\left(\mathbf{G}_i \oplus \mathbf{F}_{i-1}^{\text{ISB}}\right), & i = 6 \\ D_i^{\text{ISB}}\left(\left(\mathbf{G}_{i+1}^{\text{ISB}}\right)\uparrow \oplus \left(\mathbf{F}_i^{\text{ISB}}\right)\right), & i \in \{5,4,3,2,1\} \end{cases}, \tag{6.6}$$

其中，$D_i^{\text{ISB}}(\cdot)$ 同样是一个包含两个 32 核卷积层的模块。本方法在 $\mathbf{G}_1^{\text{ISB}}$ 上使用 3×3 卷积，然后对预测图进行阈值化处理，得到二值的初始分割结果 $\mathbf{M}^{\text{ISB}}$。

6.3.3 细节修复模块

如图 6.2（f）所示，细节修复模块的设计类似于初始分割模块，它们的区别在于输入输出部分。细节修复模块将医学图像特征拼接上初始分割结果 $\mathbf{M}^{\text{ISB}}$ 和修复指导图作为输入。初始分割结果来自初始分割模块、其他自动类分割方法或用户提供。修复指导图由前景/背景区域交互图和边界交互图组成。细节修复模块生成残差来修复粗糙分割。它使用与公式 6.5和公式 6.6相同的方法获得 $\{\mathbf{F}_i^{\text{DRB}}\}_{i=1}^{5}$ 和 $\{\mathbf{G}_i^{\text{DRB}}\}_{i=1}^{6}$。残差 $\mathbf{R}^{\text{DRB}}$ 可以对 $\mathbf{G}_1^{\text{DRB}}$ 使用 1×1 卷积得到。最终结果 $\mathbf{M}$ 可以从下式获得：

$$\mathbf{M} = T\left(\mathbf{M}^{\text{ISB}} + \mathbf{U} \odot \mathbf{R}^{\text{DRB}}\right), \tag{6.7}$$

其中，$T(\cdot)$ 是二值化函数，$\odot$ 是逐元素乘法。如图 6.2所示，$\mathbf{U}$ 表示局部焦点图，一个基于标注的半径为 80 的高斯距离图，目的是减少局部修复交互的全局影响。

6.4 实验结果与分析

本章节包括三个部分。首先，章节6.4.1介绍了该方法的实验设置，包括使用的数据集、评测指标、实现细节和模型推理。然后，章节6.4.2介绍了该方法的消融实验，包括初始交互模式和修复交互模式的消融实验。之后，章节6.4.3介绍了该方法的性能分析，包括与其他方法的对比和分割结果的展示和分析。最后，章节6.4.4介绍了该方法中多模式交互方式的用户调研。

6.4.1 实验设置

数据集。 针对 MMIIS 方法，本章实验采用了六种不同类型的医学图像分割数据集，表示为（疾病或器官组织 & 图像类型），各数据集详细介绍如下：

- **COVID19-CT [195]（新冠肺炎 & 电子计算机断层扫描图像）**：该数据集包含来自 200 名新冠肺炎患者的 3855 张带标注的 CT 扫描图像。总共可以得到 8313 个病变区域的实例，其中 6186 个在训练集，2187 个在测试集。
- **COVID19-Xray [203]（新冠肺炎 & X 光图像）**：该数据集中有 6500 张带有分割掩膜的胸部 X 光射线图像，其中包含 12789 个肺部区域的实例样本。本章选择其中 8951 个样本进行训练，3838 个样本进行测试。
- **BraTS-T1 [204–206]（脑肿瘤 & 核磁共振图像）**：该数据集来自于 2020 年脑肿瘤分割挑战赛，从提供的 T1 加权的核磁共振扫描图像和对应的真值标注图中。本章选择了 6314 例样本进行训练，2541 例样本进行测试。
- **Nerve [207]（臂丛 & 超声图像）**：该数据集来自于超声神经分割挑战赛，由 5635 张超声图像组成，其中包含 2323 张带有像素级掩膜标注的图像样本。本章随机抽取 1634 个样本进行训练，689 个样本进行测试。
- **Polyp [208]（息肉 & 内窥镜图像）**：该数据集为息肉的内窥镜图像，训练集中包含 1519 个息肉样本，测试集中包含 835 个息肉样本。
- **ISIC [209]（皮肤病变 & 病灶照片图像）**：该数据集来自于 2017 年国际皮肤成像合作挑战赛，其提供了带有像素级分割掩膜的皮肤病变图像。训练集中有 2000 张图像样本，测试集中有 600 张图像样本。

评测指标。 针对医学图像的特性，本章选取了三个指标进行评测，分别为 Dice 指标、ASSD 指标和 mNoI 指标。它们分别用来反映目标分割的整体质量、边界精细度和交互效率，以下是各指标的详细介绍：

- **Dice 指标**：和大多数医学图像分割工作类似，本章利用 Dice 评分来评价交互式医学图像分割得到的分割掩膜的质量，其计算公式如下：

$$\text{Dice} = 2|\mathcal{P} \cap \mathcal{G}| / (|\mathcal{P}| + |\mathcal{G}|). \tag{6.8}$$

  该指标表示的是医学图像中目标的整体分割效果。Dice 指标的数值越高，代表着网络模型生成的分割掩膜结果的整体质量越好。
- **ASSD 指标**：与 [52] 相同，本章使用像素的平均对称表面距离（Average Symmetric Surface Distance，ASSD）来评价医学图像的分割结果的边界质量，其中 ASSD 的计算如下：

$$\text{ASSD} = \frac{1}{|\mathcal{P}_b| + |\mathcal{G}_b|} \left(\sum_{p \in \mathcal{P}_b} \phi(p, \mathcal{G}_b) + \sum_{p \in \mathcal{G}_b} \phi(p, \mathcal{P}_b) \right),$$

  其中，$\mathcal{P}_b$ 表示分割结果的边界，$\mathcal{G}_b$ 表示真值标注的边界，$\phi(p, \mathcal{X})$ 表示点 p 到点集 $\mathcal{X}$ 的最短距离，$|\cdot|$ 表示像素数。边界越精确则该指标数值越小。
- **NoI 指标**：本章使用平均的交互数量（Number of Interaction，NoI）来衡量交互式分割方法的交互性能。它被定义为修复阶段的平均交互次数，直到每个实例达到指定的 Dice 数值 (@95%)。最大交互次数设置为 20 次。

实现细节。 MMIIS 方法以 VGG-16 [202] 作为特征提取器中的主干网络并设置批大小为 16。本方法初始学习率设置为 1×10^{-4}，并使用权重衰减为 5×10^{-4} 的 Adam 方法 [210] 进行模型参数优化。本方法在训练过程中采用多项式学习率衰减策略并使用二值交叉熵损失进行监督。所有的实验都是用 PyTorch [179] 深度学习框架实现的，并运行在单个 NVIDIA Titan XP GPU 上。

模型推理。 本章提出的模型在 Intel i7-8700K 3.70GHz CPU 和单个 NVIDIA Titan XP GPU 上测试了推理时间。每次交互的推理时间为 0.015 秒，该速度对于现实世界中的应用来说已经满足实时操作的需求。

6.4.2 消融实验

本章节将开展消融实验来证明 MMIIS 方法中初始交互模式和修复交互模式的有效性。实验结果如表 6.1、表 6.2 和图 6.6所示。对于这两类交互模式，本章节展示了面对不同环境的不同样本下的性能结果和可视效果。不同的性能和效果说明了提供多种交互模式对于交互式医学图像分割的必要性。

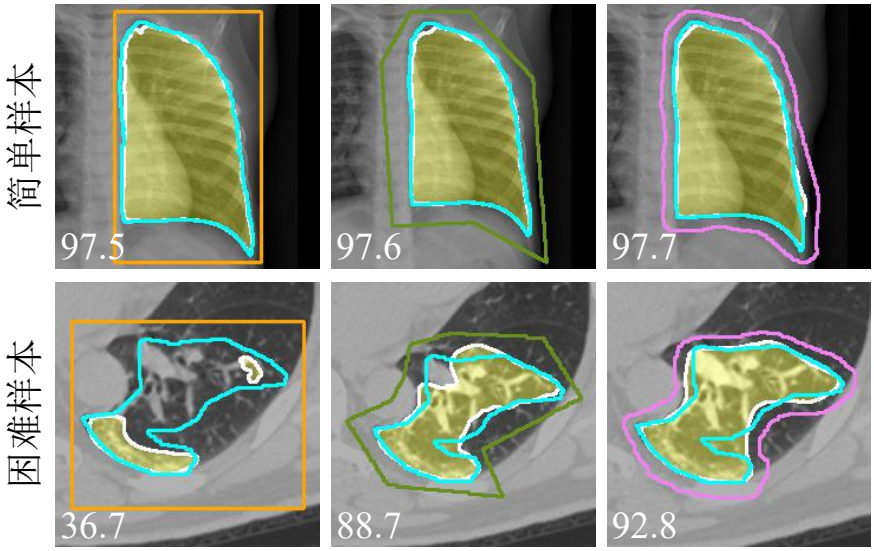


图 6.3　初始交互模式在不同场景下的效果对比。左下角的白色数字为 Dice 指标的数值。第一行为简单样本，目标对象具有规则的形状，所有交互模式都能很好地分割对象。第二行为复杂样本，目标对象具有不规则的形状，包围框交互会面临歧义性从而导致不准确的分割。

初始交互模式。　对于三种初始交互模式，总的来说，包围框、包围多边形、包围涂鸦的交互负担是逐渐增加的。同时，它们带来的定位效果和初始分割性能也会相应地更加准确。本章首先在一个简单和一个困难的样本中研究初始交互模式。绘制包围框，用户只需要确定两个关键点。这种轻量级交互模式可以处理规则形状的区域。如图 6.3第一行所示，利用包围框可以获得与包围多边形、包围涂鸦相似的结果。当场景变得复杂时，包围框就不能胜任目标的分割了。例如，第二行中目标对象是不规则形状的。包围框不可避免地包含了背景区域，这反过来导致了不准确的预测。在这种情况下，包围多边形和包围涂鸦可以灵活地靠近目标边界，同时带来更多的形状先验。因此，它们可以得到更准确的结果。表 6.1和表 6.2 中显示了不同初始交互模式获得的分割结果的 Dice 得分（表中为修复交互数为 0 所对应的列）。可以看到，在所有数据集上，包围框、包围涂鸦和包围多边形的性能都是逐步上升的，这也和本章的期望相符。MMIIS 方法在最初的分割阶段允许使用多种交互。用户可以根据场景和目标的复杂程度选择合适的交互模式，避免了特定交互模式能力不足或交互成本过高的问题。

修复交互模式。　对于修复交互模式，总体来说分为两种，区域交互与边界交互。区域交互使用前背景的点击和涂鸦来标识区域，能更好地处理区域分割差

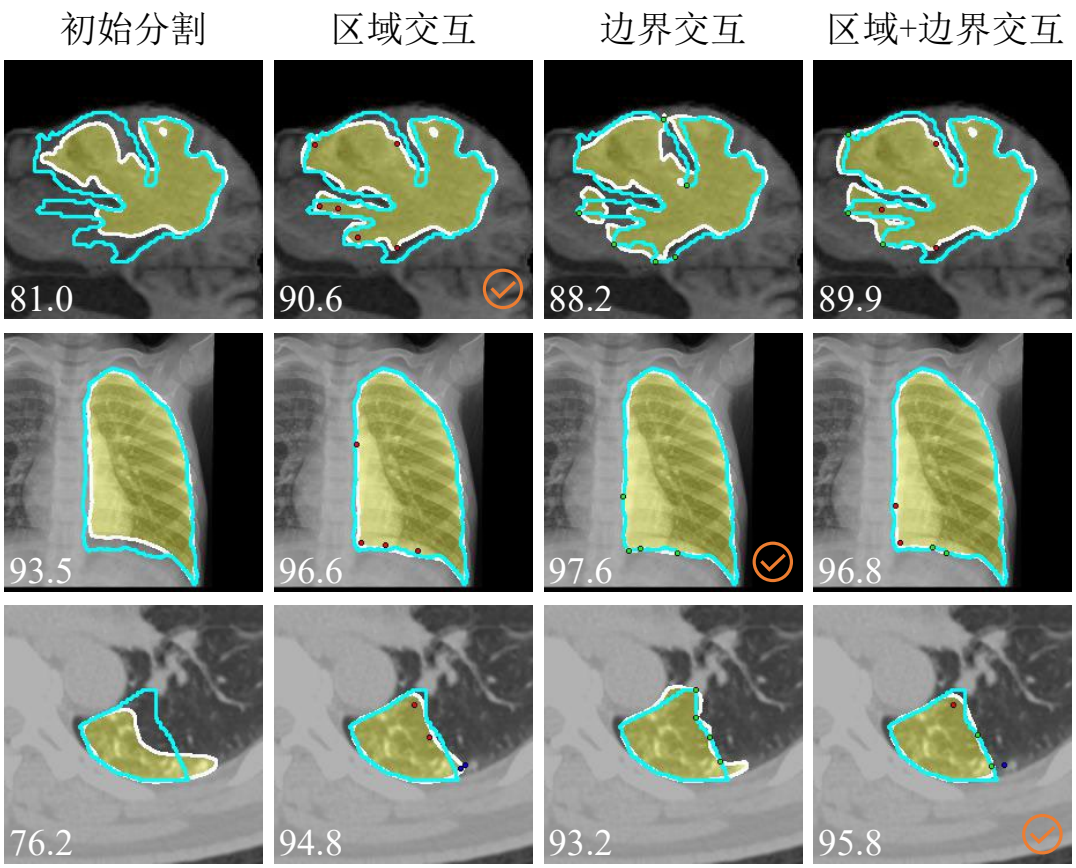


图 6.4　区域和边界修复交互模式独自工作或协同工作的对比。左下角的白色数字为 Dice 指标的数值，打勾的为性能最佳的情况。不同的医学图像场景和目标面临不同的歧义性。第一行样例的初始分割结果有很大区域被错误预测。第二行样例包含不准确的边界。第三行样例面临着更有挑战性的情况，目标边界模糊且区域难以区分。不同的场景和目标适用不同的交互模式，一些情况下区域交互和边界交互协同作用能取得更好的效果。

的问题。边界交互使用边界上的点击和涂鸦来标识边界，能更好地处理分割边界不准确的问题。为什么细节修复需要多个交互模式？图 6.4中列举了一些实际样例。第一行的样例由于错误区域大，所以更适合使用区域交互。第二行的样例由于边界存在细小偏差，所以更适合使用边界交互。而第三行样例，同时使用两种交互能获得更好的性能。以上例子说明了在不同场景下适合使用不同的交互模式，有时候多种交互模式的协同作用更有利于分割修复。表 6.1和表 6.2 中显示了不同修复交互模式获得的分割结果的 Dice 得分（表中为修复交互数为 5 和 10 所对应的列）。在有的数据集上，区域交互的性能大于边界交互，有时候则相反。说明二者在不同情况下是各具优势的。两张表格定量评估了细节修

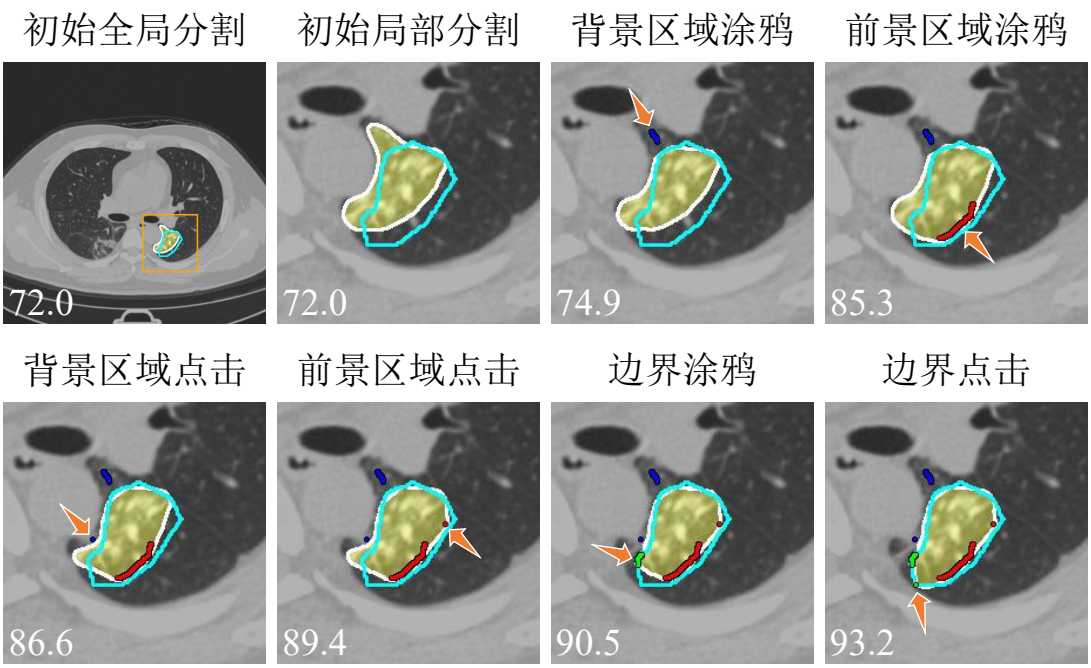


图 6.5　不同修复交互模式协同工作的可视样例。左下角的白色数字为 Dice 指标的数值。

复模块对于修复初始分割结果的效果。可以看到修复交互模式在初始分割结果的基础上带来了普遍的性能提高。图 6.6中直观地展示了区域交互与边界交互的 ASSD 指标曲线图。由于 ASSD 指标主要用来评判目标边界的预测性能，可以发现，区域交互的曲线在大多数情况下不如边界交互。这也说明了边界交互对于准确预测目标边界的有效性。MMIIS 方法在分割修复阶段允许使用多种修复交互模式。用户可以根据目标的不同分割情况选择合适的交互模式，从而更好地提高修复效率。最后，在图 6.5中，本章通过联合使用六种修复交互模式来进行了一个实际的修复演示。可视结果反映出，涂鸦比点击能获得更精确的效果，这是因为涂鸦以更高的交互成本为代价提供了更多的指导信息。通过比较不同类型的交互模式可以发现，前景区域的交互可以很容易地修复错误标记的前景预测，而背景区域的交互则可以很好地消除背景上的错误预测。用户可以利用这些交互来解决前景分布不均匀和背景相似区域干扰造成的歧义性问题。边界交互与区域交互不同，所带来的效果也不同。可以看到，无论是边界点击还是涂鸦，预测的边界都会固定到交互位置，并延伸到两边寻找可能的轮廓。由于医学图像经常面临模糊不清的轮廓，这一特性非常利于医学图像的标注。

表 6.1　初始交互与点击修复交互模式的消融实验及和其他方法的对比。这些交互模式缩写为：包围框（**Bbox**），包围多边形（**Bpoly**），包围涂鸦（**Bscri**），区域点击（**RC**），边界点击（**BC**）。在比较方法中，“-C”意味着基于“点击”进行交互。“0、5、10”表示当每个实例上有 0、5 或 10 个修复交互时的 Dice 得分。mNoI（@95%）是修复交互的平均数量。

基于点击修复交互	COVID19-CT				COVID19-Xray				BraTS-T1			
	0	5	10	mNoI	0	5	10	mNoI	0	5	10	mNoI
Bbox+RC	84.8	93.4	95.2	8.4	96.0	98.0	98.3	0.4	81.0	89.7	92.4	14.8
Bbox+BC	84.8	93.9	96.2	6.2	96.0	98.2	98.6	0.3	81.0	88.9	91.8	13.5
Bpoly+RC	86.6	94.0	95.8	6.9	96.8	98.2	98.5	0.1	85.4	91.3	93.5	12.8
Bpoly+BC	86.6	94.2	96.5	5.8	96.8	98.3	98.7	0.1	85.4	90.5	92.5	12.7
Bscri+RC	91.6	95.8	96.8	3.2	98.7	98.7	98.7	0.0	91.8	93.9	94.9	8.4
Bscri+BC	91.6	96.1	97.0	2.7	98.7	98.9	98.9	0.0	91.8	93.8	94.6	8.5
GrabCut-C	14.6	20.3	35.0	19.9	12.9	13.3	13.8	20.0	43.6	44.2	45.1	20.0
GraphCut-C	1.0	5.2	11.7	20.0	15.1	15.8	12.9	20.0	3.5	2.6	6.6	20.0
RW-C	4.5	64.7	85.8	19.3	48.9	78.8	85.1	19.7	9.4	65.9	85.1	19.7
FCA-Net	81.0	92.4	94.0	3.0	96.0	97.9	98.4	0.4	80.2	83.7	84.9	19.1
f-BRS	80.0	90.2	92.1	11.8	95.6	97.2	97.9	0.5	79.7	67.7	85.7	18.2
基于点击修复交互	Nerve				ISIC				Polyp			
	0	5	10	mNoI	0	5	10	mNoI	0	5	10	mNoI
Bbox+RC	81.9	93.6	96.0	7.2	88.5	92.9	93.8	9.9	85.9	91.1	92.1	9.7
Bbox+BC	81.9	94.4	96.9	5.1	88.5	94.9	96.0	5.1	85.9	93.7	95.3	4.0
Bpoly+RC	87.2	94.8	96.7	5.7	91.6	94.7	95.5	6.6	91.4	94.2	94.7	8.5
Bpoly+BC	87.2	95.9	97.7	3.9	91.6	96.0	96.9	3.9	91.4	96.6	97.8	2.6
Bscri+RC	92.5	95.9	97.2	3.6	92.5	95.1	95.8	6.3	93.5	95.2	95.6	6.5
Bscri+BC	92.5	97.3	98.1	1.8	92.5	96.3	97.1	3.7	93.5	97.0	98.0	1.9
GrabCut-C	0.1	0.2	0.4	20.0	72.2	73.0	73.9	18.2	39.9	40.3	41.7	19.7
GraphCut-C	2.6	3.2	2.6	20.0	13.8	20.0	14.0	19.9	5.1	6.8	5.4	20.0
RW-C	10.0	66.5	88.1	19.0	40.5	78.3	88.2	18.1	20.8	72.0	88.7	18.2
FCA-Net	79.4	90.5	93.8	11.8	83.2	94.7	96.1	4.9	82.3	94.7	96.5	4.7
f-BRS	78.1	90.5	90.6	13.3	83.0	94.4	95.9	4.6	85.4	94.3	96.5	4.3

表 6.2　初始交互与涂鸦修复交互模式的消融实验及和其他方法的对比。这些交互模式缩写为：包围框（**Bbox**），包围多边形（**Bpoly**），包围涂鸦（**Bscri**），区域涂鸦（**RS**），边界涂鸦（**BS**）。在比较方法中，“-S”意味着基于“涂鸦”进行交互。“0、5、10”表示当每个实例上有 0、5 或 10 个修复交互时的 Dice 得分。mNoI（@95%）是修复交互的平均数量。

基于涂鸦修复交互	COVID19-CT				COVID19-Xray				BraTS-T1			
	0	5	10	mNoI	0	5	10	mNoI	0	5	10	mNoI
Bbox+RS	84.8	95.2	96.3	5.6	96.0	98.4	98.6	0.2	81.0	93.0	94.5	11.0
Bbox+BS	84.8	95.7	97.8	4.3	96.0	98.6	98.9	0.3	81.0	92.0	95.0	8.8
Bpoly+RS	86.6	95.8	97.1	4.0	96.8	98.5	98.7	0.1	85.4	93.9	95.4	8.5
Bpoly+BS	86.6	96.2	98.3	3.7	96.8	98.7	99.0	0.1	85.4	93.2	96.0	7.4
Bscri+RS	91.6	97.0	97.5	1.7	98.7	98.9	98.9	0.0	91.8	95.6	96.2	5.3
Bscri+BS	91.6	97.4	98.6	1.7	98.7	99.0	99.0	0.0	91.8	95.6	97.2	4.1
GrabCut-S	14.6	83.1	92.0	13.8	12.9	88.0	95.2	9.2	43.6	73.9	87.5	18.3
GraphCut-S	35.7	84.8	92.3	13.9	30.9	84.1	94.8	9.5	30.7	79.3	89.2	17.9
RW-S	44.6	88.3	94.3	10.9	57.7	87.7	94.7	10.8	25.6	85.7	92.0	15.2
基于涂鸦修复交互	Nerve				ISIC				Polyp			
	0	5	10	mNoI	0	5	10	mNoI	0	5	10	mNoI
Bbox+RS	81.9	96.0	97.2	4.1	88.5	95.5	95.7	5.7	85.9	94.4	94.6	7.8
Bbox+BS	81.9	96.5	98.3	3.8	88.5	96.1	97.0	4.5	85.9	96.6	98.1	2.8
Bpoly+RS	87.2	96.9	97.9	2.8	91.6	96.5	96.6	3.3	91.4	95.7	95.9	5.9
Bpoly+BS	87.2	97.2	98.6	3.1	91.6	96.6	97.4	3.5	91.4	97.4	98.7	2.0
Bscri+RS	92.5	97.8	98.1	1.2	92.5	96.8	96.9	2.8	93.5	96.4	96.4	4.3
Bscri+BS	92.5	98.0	98.7	1.5	92.5	96.8	97.5	3.4	93.5	97.8	98.7	1.4
GrabCut-S	0.1	83.9	93.2	13.1	72.2	91.5	95.1	8.5	39.9	82.7	93.8	10.5
GraphCut-S	29.0	83.9	93.3	13.6	19.2	83.3	93.3	12.4	16.2	87.2	95.1	9.5
RW-S	44.0	88.2	95.6	9.0	52.6	91.3	95.6	8.9	61.9	91.2	96.2	7.8

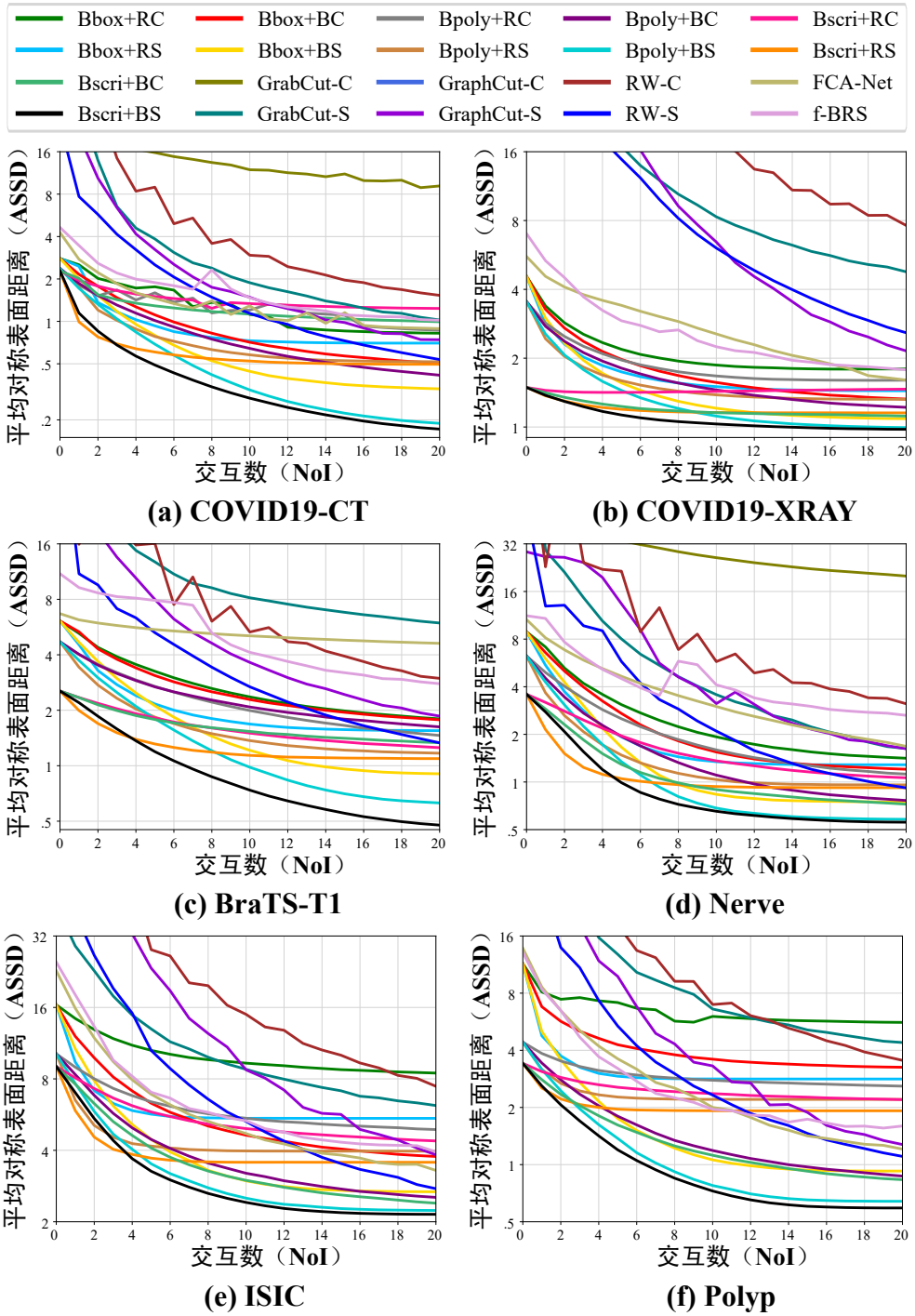


图 6.6　MMIIS 方法和其他方法的 NoI-ASSD 曲线图。图例中的交互模式缩写为：包围框（**Bbox**），包围多边形（**Bpoly**），包围涂鸦（**Bscri**），区域点击（**RC**），边界点击（**BC**），区域涂鸦（**RS**），边界涂鸦（**BS**）。在比较方法中，“-C”意味着基于“点击”进行交互，“-S”意味着基于“涂鸦”进行交互。FCA-Net [186] 和 f-BRS [39] 为基于点击的交互方法。

表 6.3 初始分割结果和自动类分割方法结果的 Dice 指标对比。

	FCN [61]	U-Net [69]	DSS [77]	包围框	包围多边形	包围涂鸦
ISIC	79.4	74.9	80.5	88.5	91.6	92.4
Nerve	47.1	50.7	60.3	81.9	87.2	92.5
BraTS-T1	40.7	59.9	54.5	81.0	85.4	91.8

6.4.3 性能分析

本章节将该框架的两个阶段，即初始分割阶段和细节修复阶段，与现有的相关方法进行了比较。该实验主要比较了四种开源的交互式图像分割方法：传统算法 GrabCut [16] 和 Random Walks（RW）[18]，基于深度学习的方法 FCA-Net [186] 和 f-BRS [39]。性能结果同样展示在表 6.1、表 6.2 和图 6.6中。除此之外，本章节还展示了初始分割模块的分割结果和自动类分割方法的分割结果的性能对比。以及基于自动类分割方法的结果使用修复交互模式的效果。

初始交互模式性能对比。 对于初始交互模式，从表 6.1和表 6.2中可以看出，本章的方法远远超出了传统方法，如 GrabCut 和 Random Walks。此外，初始交互模式的目标定位作用也使 MMIIS 方法性能优于 FCA-Net [186] 和 f-BRS [39]。在表 6.3中，本章将初始分割模块和自动类分割方法的结果进行了比较。这些方法包括：U-Net [69]、FCN [61] 和 DSS [77]。可以看出，MMIIS 方法提供的三种初始交互模式带来了有用的目标指导信息，从而获得了更好的分割性能。

修复交互模式性能对比。 对于修复交互模式，从表 6.1和表 6.2中可以看出，在相同的交互次数下，无论是点击还是涂鸦，本章提出的修复交互模式取得了更好的分割效果。对于反映整体修复效率的平均交互次数（NoI）指标，修复交互模式也大多超过了其他方法。图 6.6反映了不同修复交互模式的分割性能趋势以及和其他方法的对比。可以看出，相对于其他方法，各种修复交互模式能更好地到达较小的 ASSD 值，这表示这些交互模式可以更快地对目标分割进行修复。

修复自动类分割结果。 MMIIS 方法除了可以对初始分割模块提供的结果进行修复，还可以用于修复给定的粗糙分割，如自动类分割方法的结果等。在表 6.4中，本章展示了在 COVID19-CT 和 COVID19-Xray 数据集上对自动分割结

表 6.4 修复交互模式用以修复 U-Net 模型生成的粗糙掩膜的性能。评测指标为 ASSD 指标。0、1 和 5 表示对应的修复交互模式的交互数。

基于 U-Net [69] 结果添加修复交互模式	COVID19-CT			COVID19-Xray		
	0	1	5	0	1	5
+ 区域点击	26.7	9.5	7.9	5.7	5.4	4.3
+ 区域涂鸦	26.7	9.3	7.9	5.7	5.0	3.6
+ 边界点击	26.7	8.7	6.7	5.7	5.5	4.3
+ 边界涂鸦	26.7	8.5	6.5	5.7	5.3	3.7

果进行修复的效果。该实验使用经典的医疗图像分割模型 U-Net [69] 来提供初始分割。可以看出，使用了各种修复交互模式后，分割结果的 ASSD 指标得到了大幅度下降。这说明了本章提出的修复交互模式可以对已有的预测进行稳定的修复。在现实世界的标注任务中，如新冠肺炎标注 [71,211,212]，使用基于自动预测的结果进行交互修复，将大大减少用户的交互负担。在临床诊断上配合自动预测算法，有助于产生更准确的分割掩膜，并提高疾病的定量评估性能。

6.4.4 用户调研

由于本章提出的方法是一种集合多种交互模式的框架，进行真实的用户调研有助于评判该方法的实际使用效果。本章节首先描述了用户调研的设置，然后将 MMIIS 方法与其他方法的交互时间进行对比并展示在表 6.5中。本章节还探索了现实场景中多种交互模式的使用情况，并将相关数据展示在表 6.6中。用户在进行分割修复过程中使用不同交互的比例曲线图被记录在图 6.7中。

用户调研设置。 和 [37] 一样，本章从 COVID19-CT、BraTS-T1 和 Nerve 数据集中随机抽取 50 张图像进行该用户调研。本章为这项用户调研招募了五名参与者。要求每个参与者使用 FCA-Net 方法 [186]、f-BRS 方法 [39] 和本章提出的 MMIIS 方法来对这些医学样本进行标注。当某个样本 Dice 分数超过 90% 或达到 30 秒的最大限制时，停止对该样本的标注并记录所用时间。

方法时间对比。 表 6.5中记录了本章提出的 MMIIS 方法以及其他两个方法的平均标注时间。MMIIS 方法在每个数据集上使用的时间都最少。总体来看，它平均为每个图像节省了大约 5 秒的时间，快了近 40%，有效地减少了交互负担。

表 6.5 MMIIS 方法的用户调研中多种方法的交互时间对比。表中含模型推理时间（秒）。

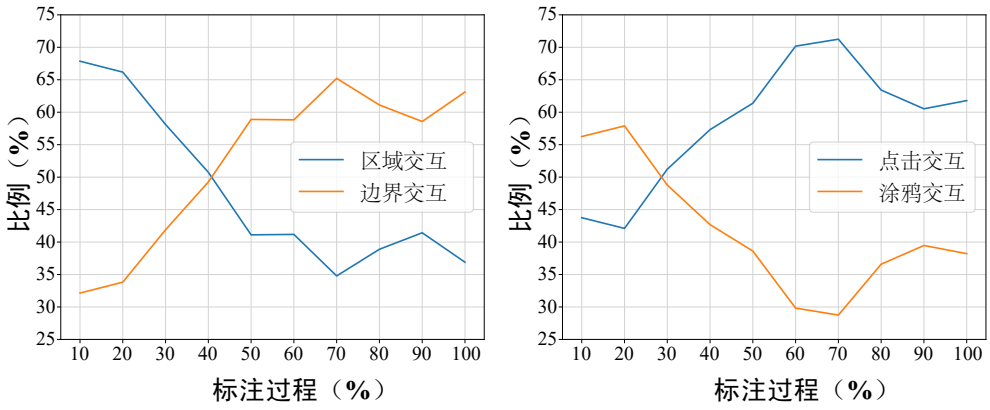
	COVID19-CT	BraTS-T1	Nerve	全部
f-BRS [39]	12.26	14.82	9.29	12.12
FCA-Net [186]	10.77	19.93	6.14	12.28
MMIIS	6.11	10.39	5.74	7.41

初始交互模式。 在初始交互模式中，包围框是最受真实用户欢迎的选择，据统计占比高达 81%。它的平均交互时间为 1.44 秒。包围多边形和包围涂鸦的平均交互时间分别为 3.64 秒和 2.25 秒，用户用它们来定位较复杂的物体。从表 6.6中可以看出，17% 的样本仅使用初始交互模式就实现了 90% 的 Dice 指标。

修复交互模式。 对于修复交互模式，可以根据作用位置划分为区域交互和边界交互。也可以根据交互形式，划分为点击交互和涂鸦交互。表 6.6给出了两种划分的真实用户统计数据。图 6.7则绘制了两种划分在真实用户交互过程中的占比曲线。以下，本章节将针对这两种不同的划分进行分析:

（1）区域交互与边界交互：从表 6.6中可以看出，28% 的情况下，仅区域交互就能对样本进行修复，33% 的情况下，仅边界交互就能对样本进行修复。此外，22% 的情况下用户选择了同时使用区域交互和边界交互来协同修复分割结果。对于真实用户，区域交互和边界交互的选择比例相差不多。区域交互所用时间比边界交互略多一些，但区域交互由于经常用于快速修复大的错误标记区域，它所带来的性能的提升也相对较大。边界交互主要是用来固定边界位置以获得更精确的结果。在实际交互过程中综合使用二者更能满足用户的需求。图 6.7（a）也反映了这一点，该曲线统计了用户标注过程中区域交互和边界交互的比例变化。从图中可以看出，在错误标注区域较大的前期，用户更倾向于选择区域交互修复大块错误区域，而在后期，用户更倾向于使用边界交互来固定物体轮廓。综上所述，MMIIS 方法提供的区域交互与边界交互这两类交互模式在实际标注中是实用的，可以让用户有效修复初始分割结果的错误区域与边界。

（2）点击交互与涂鸦交互：从表 6.6中可以看出，36% 的情况下，仅点击交互就能对样本进行修复，22% 的情况下，仅涂鸦交互就能对样本进行修复。此外，25% 的情况下用户选择了同时使用点击交互和涂鸦交互来协同修复分割结果。对于



(a) 区域交互对比边界交互 **(b) 点击交互对比涂鸦交互**

图 6.7 MMIIS 方法的用户调研中各类交互比例随标注过程的变化曲线。

表 6.6 MMIIS 方法的用户调研中关于真实场景下各类交互的统计数据。“无”指的是初始分割阶段就已经达到了目标指标。“提升”指的是 Dice 指标的提升。

	整体交互过程							
	仅区域	仅边界	区域 + 边界	无	仅点击	仅涂鸦	点击 + 涂鸦	无
比例	28%	33%	22%	17%	36%	22%	25%	17%
	修复交互过程							
	区域		边界		点击		涂鸦	
时间	1.55		1.30		1.21		1.82	
比例	50.7%		49.3%		64.1%		35.9%	
提升	3.4%		2.9%		2.4%		4.5%	

真实用户，点击交互所用时间比涂鸦交互少，所造成的交互负担也相对小，所以点击交互的选择比例远大于涂鸦交互。但由于涂鸦交互能提供更丰富的指导信息，其所带来的性能提升也相对较大。在实际交互过程中综合使用二者更能满足用户的需求。图 6.7（b）也反映了这一点，该曲线统计了用户标注过程中点击交互和涂鸦交互的比例变化。前期用户主要进行区域交互，而较大的错误区域更好进行涂鸦，所以用户在早期倾向于选择区域涂鸦，而后期则倾向于选择区域点击，这造成了涂鸦比例前期不断下降。而最后涂鸦比例的略微上升很可能是由于用户使用边界涂鸦来固定目标轮廓。综上所述，MMIIS 方法提供的点击交互与涂鸦交互这两类交互模式有利于用户从整体到局部完成分割的修复。

6.5 本章小结

针对交互式图像分割任务中，复杂场景下的医学图像分割差的问题，本章提出了一种多模式交互式图像分割框架，以解决医学图像分割中的各种歧义性问题，如不规则形状、模糊边界和相似背景干扰等。与目前的交互式医学图像分割方法相比，该方法系统地将多种交互模式集成到一个统一的网络框架中。用户可以轻松地利用各种初始交互模式进行医学目标的初始预测，然后用各种区域和边界的修复交互模式对目标分割进行修复。充分的实验结果和用户调研表明，该方法能灵活有效地解决现实中医学图像标注任务。有了该方法，复杂场景下的低对比度医学图像中的医学目标可以更好地被交互分割出来。

7 总结与展望

交互式图像分割任务是计算机视觉中一个重要的研究方向。用户通过使用不同的交互模式，分割出目标对象的二值掩膜。交互式分割所面对的是复杂场景下的不同目标物体，其可能存在不同的图像类型、不同的图像构成、不同的目标结构等情况。现有的方法对于复杂场景下的交互式分割存在一定的缺陷，表现在对于一些复杂的目标，无法有效地进行精确分割或者用户交互负担过大等。因此，研究面向复杂场景的交互式图像分割方法能更高效地处理不同的目标对象。这对于如今需要大量标注数据为基础的深度学习模型的发展具有重要意义，能有效促进计算机视觉相关的人工智能研究。本章将对本文的工作进行一个总结，并对未来可能的改进和探索方向进行展望。

7.1 本文工作总结

本文首先介绍了研究背景和意义，描述了交互式图像分割任务以及其实际价值。然后对于该任务的研究现状和难点进行了梳理，提出了循序渐进的四大难点与挑战。紧接着针对这些难点提出了本文的研究目标和贡献。随后在相关工作综述章节中，本文详细介绍了交互式分割及其相关任务，并对它们进行了归类。在之后的章节中，本文对四个渐进的研究工作进行了详细的介绍。最后，本文对该工作进行了总结并对未来的研究方向进行了展望。

面对目标定位不准确这一难点，以针对目标定位的全局物体分割为目标，本文提出了基于初始交互点注意力的交互式分割。不同于其他方法将所有交互点同等对待，本文将初始交互点的目标定位作用加以突出，提出了初始交互点注意力网络。该模型在基础分割网络上添加了一个初始交互点注意力模块，有效地利用了初始交互点的指导信息。该方法使得交互式图像分割在面对复杂场景下分割目标周围干扰物体过多或同类目标过于紧密的情况，网络模型能更好地判断哪个目标是用户分割对象。即使在简单的图像中，该方法也能标识出物体中心，从而使其他交互点更好地实现修复的目的。在多个通用物体数据集上的实验和可视结果分析都证明了该方法对于物体主体分割的有效性。有了该工作，交互式图像分割任务中首要的定位物体来分割主体这一目标得以更好地实现。

面对局部区域精度低这一难点，以针对精确细节的局部区域分割为目标，本文提出了深入聚焦视角的交互式分割。不同于其他方法利用所有交互点共同分割出对象主体，而忽略了许多交互点用来修复局部细节的目的，本文从交互点的聚焦视角出发，在整体分割的基础上，对交互点周围的局部分割进行精细化修复。通过提出一个聚焦分割的流程框架并训练一个可以进行全局分割和局部分割的共享网络，在尽量不改变网络参数和结构的基础上，有效地提升了交互式分割方法对于细节的分割性能。在多个通用物体数据集上的性能结果和可视结果中的精确细节都证明了该方法对于细节分割的有效性。有了该工作，交互式图像分割任务中在主体分割这一基础上的细节分割得以更好地实现。

面对细小结构交互难这一难点，以针对细小结构的复杂拓扑分割为目标，本文提出了修复细小结构的切割线交互式分割。当面对一些拥有细小结构的特殊目标时，常用的一些交互模式往往不能很好地起到效果。本文提出了一个切割线交互模式，用户只需要绘制一条线穿过错误预测的细小结构区域，就能有效地修复该处的分割。该交互符合用户直觉，且需要的交互量较小，能有效减轻用户的负担。基于该交互模式，本文还提出了一个针对细小结构的复杂拓扑分割的网络模型，利用细小区域的局部相似性和全局相似性，让用户可以自由地对局部和全局的细小结构分割进行修复。在多个细小结构物体数据集上的交互性能和可视结果表明了用户可以低负担地利用该方法获得较高的性能。有了该工作，交互式图像分割任务中低负担地进行细小结构物体的分割得以更好地实现。

面对医学图像分割差这一难点，以针对低对比度的医学图像分割为目标，本文提出了面向医学图像的多模式交互式分割。当面对计算机图像中的低对比度医学图像时，单一交互模式的方法往往不能取得良好的分割效果。本文将多种高效的交互模式进行整合，使用一个共享的网络，对医学图像中的目标进行分割。该方法包含的三种初始交互模式，能对医学目标进行很好的初始分割。随后可以利用后续提供的基于区域和边界的多种修复交互模式，对医学目标进行精细化的修复。在多种医学图像数据集上的实验和可视结果表明了用户可以综合地利用该方法高效地对不同的医学目标进行分割。有了该工作，交互式图像分割任务中面对低对比度医学图像中的目标进行高效分割得以更好地实现。

以上就是对本文所涉及的研究内容的工作总结。首先是针对目标定位，其次是精确细节分割，然后是特殊细小结构的处理，最后是面向更复杂的低对比度医学图像。从主到次，对面向复杂场景的交互式图像分割进行了系统的研究。

7.2 未来工作展望

对于复杂场景下交互式图像分割任务存在的四大难点，本文提出的方法进行了较为深入的尝试，但其仍然存在一定的改进空间。这里对这些研究方向的未来可改进点以及可探索点进行了进一步的展望。

首先是目标定位问题。本文提供了基于初始交互点来定位这一方法，而在交互式分割过程中，即使是后续的交互点，也可能起到一定的定位作用，比如有的不规则物体需要多个定位中心。如何判断交互点的定位作用并令其对网络模型进行指导，也是值得未来进行探索和研究的。除此之外，本文的方法提供了一个简单有效的模型架构来利用初始交互点的指导信息，如何设计更为复杂的网络模型进一步利用初始交互点也值得探索。

其次是精确细节分割问题。本文方法从交互点的聚焦视角出发进行局部修复，该视角的范围是利用该交互点对全局的影响，通过算法产生的。如何设计一定的网络结构，能合理判断交互点影响范围也值得研究。除此之外，对物体精细分割的研究仍然有探索空间。对于低分辨率图像，未来可以考虑引入超分辨率重建，再进行目标的分割，可以获得更好的分割效果。

然后是细小结构处理问题。本文提出了切割线交互模式，使用户可以低负担地分割细小结构物体。但有些图像可能存在切割线交互也无法进行精确分割的情况，在该交互后加入精确细节修复的交互模式可能能进一步提升效果，这一点值得未来进行探索。而且针对细小结构的交互模式可以有许多种，对于各种交互模式的探索和研究空间仍然很大。除此之外，如何设计相应的网络模型能更好地对细小结构的边缘细节进行处理也值得研究。

最后是低对比度医学图像分割问题。本文使用多模式的交互式分割方法使用户可以更高效地分割医学目标。而针对医学图像的交互模式还有许多值得研究的空间，如何整合更多样化的交互模式值得后续进行研究。除此之外，由于医学目标结构的复杂性，如果缺乏对特定医学目标的结构感知，多种交互模式下也可能分割效率有限。因此，如何加入有限的医学标注数据对网络进行参数微调使其能更有效地分割目标也值得后续进行探索。

参考文献

[1] 李小薪, 梁荣华, 有遮挡人脸识别综述: 从子空间回归到深度学习, 计算机学报 41(177-207) (2018).

[2] 张燕咏, 张莎, 张昱, 吉建民, 段逸凡, 黄奕桐, 张宇翔, 基于多模态融合的自动驾驶感知及计算, 计算机研究与发展 57(1781-1799) (2020).

[3] 宁凯, 张东波, 印峰, 肖慧辉, 基于视觉感知的智能扫地机器人的垃圾检测与分类, 中国图象图形学报 24(1358-1368) (2019).

[4] 田萱, 王亮, 丁琪, 基于深度学习的图像语义分割方法综述, 软件学报 30(2) 2019, 29.

[5] 王子愉, 袁春, 黎健成, 利用可分离卷积和多级特征的实例分割, 软件学报 30(954-961) (2019).

[6] 毛琳, 任凤至, 杨大伟, 张汝波, 基于卷积神经网络的全景分割 Transformer 模型, 软件学报 34(3408-3421) (2023).

[7] 李岳云, 许悦雷, 马时平, 史鹤欢, 深度卷积神经网络的显著性检测, 中国图象图形学报 21(53-59) (2016).

[8] 何淼楹, 崔宇超, 面向自动驾驶的交通场景语义分割, 计算机应用 41(25-30) (2021).

[9] 杨坚伟, 严群, 姚剑敏, 林志贤, 基于深度神经网络的移动端人像分割, 计算机应用 40(3644-3650) (2020).

[10] 周涛, 董雅丽, 霍兵强, 刘珊, 马宗军, U-net 网络医学图像分割应用综述, 中国图象图形学报 26(2058-2077) (2021).

[11] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, S.-M. Hu, Sketch2Photo: Internet image montage, ACM Transactions on Graphics 28(5) 2009, 1–10.

[12] X. Wang, W. Yang, H. Peng, G. Wang, Shape-aware skeletal deformation for 2D characters, The Visual Computer 29(6-8) 2013, 545–553.

[13] J. Singh, L. Zheng, C. Smith, J. Echevarria, Paint2Pix: interactive painting based progressive image synthesis and editing, in: European Conference on Computer Vision, 2022.

[14] H. Ramadan, C. Lachqar, H. Tairi, A survey of recent interactive image segmentation methods, Computational Visual Media 6(4) 2020, 355–384.

[15] X. Bai, G. Sapiro, Geodesic matting: A framework for fast interactive image and video segmentation and matting, International Journal of Computer Vision 82(2) 2009, 113–132.

[16] C. Rother, V. Kolmogorov, A. Blake, GrabCut: Interactive foreground extraction using iterated graph cuts, ACM Transactions on Graphics 23(3) 2004, 309–314.

[17] Y. Y. Boykov, M.-P. Jolly, Interactive graph cuts for optimal boundary & region segmentation of objects in ND images, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2001.

[18] L. Grady, Random walks for image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 28(11) 2006, 1768–1783.

[19] N. Xu, B. Price, S. Cohen, J. Yang, T. S. Huang, Deep interactive object selection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016.

[20] S. D. Jain, K. Grauman, Click carving: Interactive object segmentation in images and videos with point clicks, International Journal of Computer Vision 127(9) 2019, 1321–1344.

[21] K.-K. Maninis, S. Caelles, J. Pont-Tuset, L. Van Gool, Deep extreme cut: From extreme points to object segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[22] R. Benenson, S. Popov, V. Ferrari, Large-scale interactive object segmentation with human annotators, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[23] 桂彦, 郭林, 曾光, 单幅图像训练深度神经网络的编辑传播方法, 计算机辅助设计与图形学学报 31(1391-1402) (2019).

[24] 李月龙, 高云, 闫家良, 邹佰翰, 汪剑鸣, 基于深度神经网络的图像缺损修复方法综述, 计算机学报 44(2295-2316) (2021).

[25] F. Shan, Y. Gao, J. Wang, W. Shi, N. Shi, M. Han, Z. Xue, Y. Shi, Lung infection quantification of COVID-19 in CT images with deep learning, arXiv preprint arXiv:2003.04655 (2020).

[26] F. Zhao, X. Xie, An overview of interactive medical image segmentation, Annals of the British Machine Vision Association 2013(7) 2013, 1–22.

[27] L. Castrejon, K. Kundu, R. Urtasun, S. Fidler, Annotating object instances with a Polygon-RNN, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2017.

[28] D. Acuna, H. Ling, A. Kar, S. Fidler, Efficient interactive annotation of segmentation datasets with Polygon-RNN++, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[29] H. Ling, J. Gao, A. Kar, W. Chen, S. Fidler, Fast interactive object annotation with curve-gcn, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[30] S. Zhang, J. H. Liew, Y. Wei, S. Wei, Y. Zhao, Interactive object segmentation with inside-outside guidance, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

[31] S. Mahadevan, P. Voigtlaender, B. Leibe, Iteratively trained interactive segmentation, in: British Machine Vision Conference, 2018.

[32] K. Sofiiuk, I. A. Petrov, A. Konushin, Reviving iterative training with mask guidance for interactive segmentation, in: IEEE International Conference on Image Processing, 2022.

[33] S. Majumder, A. Yao, Content-aware multi-level guidance for interactive instance segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[34] Y. Hu, A. Soltoggio, R. Lock, S. Carter, A fully convolutional two-stream fusion network for interactive image segmentation, Neural Networks 109 2019, 31–42.

[35] B. Faizov, V. Shakhuro, A. Konushin, Interactive image segmentation with transformers, in: IEEE International Conference on Image Processing, 2022.

[36] Z. Li, Q. Chen, V. Koltun, Interactive image segmentation with latent diversity, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[37] J. H. Liew, S. Cohen, B. Price, L. Mai, S.-H. Ong, J. Feng, MultiSeg: Semantically meaningful, scale-diverse segmentations from minimal user input, in: Pro-

ceedings of the IEEE/CVF International Conference on Computer Vision, 2019.

[38] W.-D. Jang, C.-S. Kim, Interactive image segmentation via backpropagating refinement scheme, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[39] K. Sofiiuk, I. Petrov, O. Barinova, A. Konushin, f-BRS: Rethinking backpropagating refinement for interactive segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

[40] T. Kontogianni, M. Gygli, J. Uijlings, V. Ferrari, Continuous adaptation for interactive object segmentation by learning from corrections, in: European Conference on Computer Vision, 2020.

[41] X. Chen, Z. Zhao, F. Yu, Y. Zhang, M. Duan, Conditional diffusion for interactive segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021.

[42] J. Liew, Y. Wei, W. Xiong, S.-H. Ong, J. Feng, Regional interactive image segmentation networks, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2017.

[43] Y. Hao, Y. Liu, Z. Wu, L. Han, Y. Chen, G. Chen, L. Chu, S. Tang, Z. Yu, Z. Chen, et al., EdgeFlow: Achieving practical interactive segmentation with edge-guided flow, in: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshop, 2021.

[44] S. Vicente, V. Kolmogorov, C. Rother, Graph cut based image segmentation with connectivity priors, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2008.

[45] X. Dong, J. Shen, L. Shao, L. Van Gool, Sub-markov random walk for image segmentation, IEEE Transactions on Image Processing 25(2) 2015, 516–527.

[46] J. H. Liew, S. Cohen, B. Price, L. Mai, J. Feng, Deep interactive thin object selection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021.

[47] K. Han, J. H. Liew, J. Feng, H. Tian, Y. Zhao, Y. Wei, Slim scissors: Segmenting thin object from synthetic background, in: European Conference on Computer Vision, 2022.

[48] S. Andrews, G. Hamarneh, A. Saad, Fast random walker with priors using precomputation for interactive medical image segmentation, in: International Conference on Medical Image Computing and Computer Assisted Intervention, 2010.

[49] G. Wang, M. A. Zuluaga, R. Pratt, M. Aertsen, T. Doel, M. Klusmann, A. L. David, J. Deprest, T. Vercauteren, Dynamically balanced online random forests for interactive scribble-based segmentation, in: International Conference on Medical Image Computing and Computer Assisted Intervention, 2016.

[50] G. Wang, M. A. Zuluaga, R. Pratt, M. Aertsen, T. Doel, M. Klusmann, A. L. David, J. Deprest, T. Vercauteren, S. Ourselin, Slic-Seg: A minimally interactive segmentation of the placenta from sparse and motion-corrupted fetal MRI in multiple views, Medical Image Analysis 34 2016, 137–147.

[51] G. Wang, W. Li, M. A. Zuluaga, R. Pratt, P. A. Patel, M. Aertsen, T. Doel, A. L. David, J. Deprest, S. Ourselin, et al., Interactive medical image segmentation using deep learning with image-specific fine tuning, IEEE Transactions on Medical Imaging 37(7) 2018, 1562–1573.

[52] G. Wang, M. A. Zuluaga, W. Li, R. Pratt, P. A. Patel, M. Aertsen, T. Doel, A. L. David, J. Deprest, S. Ourselin, et al., DeepIGeoS: A deep interactive geodesic framework for medical image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 41(7) 2018, 1559–1572.

[53] X. Liao, W. Li, Q. Xu, X. Wang, B. Jin, X. Zhang, Y. Wang, Y. Zhang, Iteratively-refined interactive 3D medical image segmentation with multi-agent reinforcement learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

[54] G. Wang, M. Aertsen, J. Deprest, S. Ourselin, T. Vercauteren, S. Zhang, Uncertainty-guided efficient interactive refinement of fetal brain segmentation from stacks of MRI slices, in: International Conference on Medical Image Computing and Computer Assisted Intervention, 2020.

[55] G. Aresta, C. Jacobs, T. Araújo, A. Cunha, I. Ramos, B. van Ginneken, A. Campilho, iW-Net: an automatic and minimalistic interactive lung nodule segmentation deep network, Scientific Reports 9(1) 2019, 1–9.

[56] X. Gong, L. Wang, L. Miao, N. Chen, J. Li, PIMedSeg: Progressive interactive medical image segmentation, Computer Methods and Programs in Biomedicine 241 2023, 107776.

[57] N. Otsu, A threshold selection method from gray-level histograms, IEEE Transactions on Systems, Man, and Cybernetics 9(1) 1979, 62–66.

[58] J. Canny, A computational approach to edge detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 8(6) 1986, 679–698.

[59] R. Adams, L. Bischof, Seeded region growing, IEEE Transactions on Pattern Analysis and Machine Intelligence 16(6) 1994, 641–647.

[60] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8) 2000, 888–905.

[61] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2015.

[62] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: European Conference on Computer Vision, 2018.

[63] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, N. Sang, BiSeNet: Bilateral segmentation network for real-time semantic segmentation, in: European Conference on Computer Vision, 2018.

[64] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2017.

[65] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[66] D. Bolya, C. Zhou, F. Xiao, Y. J. Lee, YOLACT: Real-time instance segmentation, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019.

[67] R. Girshick, Fast R-CNN, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2015.

[68] A. Kirillov, K. He, R. Girshick, C. Rother, P. Dollár, Panoptic segmentation,

in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[69] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer Assisted Intervention, 2015.

[70] S. Hu, E. A. Hoffman, J. M. Reinhardt, Automatic lung segmentation for accurate quantitation of volumetric X-ray CT images, IEEE Transactions on Medical Imaging 20(6) 2001, 490–498.

[71] D.-P. Fan, T. Zhou, G.-P. Ji, Y. Zhou, G. Chen, H. Fu, J. Shen, L. Shao, Inf-Net: Automatic COVID-19 lung infection segmentation from CT images, IEEE Transactions on Medical Imaging 39(8) 2020, 2626–2637.

[72] S. Zhou, D. Nie, E. Adeli, Q. Wei, X. Ren, X. Liu, E. Zhu, J. Yin, Q. Wang, D. Shen, Semantic instance segmentation with discriminative deep supervision for medical images, Medical Image Analysis 82 2022, 102626.

[73] J. Yuan, D. Wang, R. Li, Remote sensing image segmentation by combining spectral and texture features, IEEE Transactions on Geoscience and Remote Sensing 52(1) 2013, 16–24.

[74] H. Zhang, X. Hong, S. Zhou, Q. Wang, Infrared image segmentation for photovoltaic panels based on Res-UNet, in: Chinese Conference on Pattern Recognition and Computer Vision, 2019.

[75] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, S.-M. Hu, Global contrast based salient region detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 37(3) 2014, 569–582.

[76] A. Borji, M.-M. Cheng, H. Jiang, J. Li, Salient object detection: A benchmark, IEEE Transactions on Image Processing 24(12) 2015, 5706–5722.

[77] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, P. Torr, Deeply supervised salient object detection with short connections, IEEE Transactions on Pattern Analysis and Machine Intelligence 41(4) 2019, 815–828.

[78] T. Zhou, D.-P. Fan, M.-M. Cheng, J. Shen, L. Shao, RGB-D salient object detection: A survey, Computational Visual Media 7(1) 2021, 37–69.

[79] D.-P. Fan, Z. Lin, Z. Zhang, M. Zhu, M.-M. Cheng, Rethinking RGB-D salient

object detection: Models, data sets, and large-scale benchmarks, IEEE Transactions on Neural Networks and Learning Systems 32(5) 2021, 2075–2089.

[80] M. Zhang, W. Ji, Y. Piao, J. Li, Y. Zhang, S. Xu, H. Lu, LFNet: Light field fusion network for salient object detection, IEEE Transactions on Image Processing 29 2020, 6276–6287.

[81] K. Fu, Y. Jiang, G.-P. Ji, T. Zhou, Q. Zhao, D.-P. Fan, Light field salient object detection: A review and benchmark, Computational Visual Media 8(4) 2022, 509–534.

[82] G. Li, Y. Xie, L. Lin, Y. Yu, Instance-level salient object segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2017.

[83] D.-P. Fan, J. Zhang, G. Xu, M.-M. Cheng, L. Shao, Salient objects in clutter, IEEE Transactions on Pattern Analysis and Machine Intelligence 45(2) 2023, 2344–2366.

[84] D.-P. Fan, G.-P. Ji, M.-M. Cheng, L. Shao, Concealed object detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 44(10) 2021, 6024–6042.

[85] Q. Zhai, X. Li, F. Yang, C. Chen, H. Cheng, D.-P. Fan, Mutual graph learning for camouflaged object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021.

[86] M. Xiang, J. Zhang, Y. Lv, A. Li, Y. Zhong, Y. Dai, Exploring depth contribution for camouflaged object detection, arXiv preprint arXiv:2106.13217 (2021).

[87] J. Pei, T. Cheng, D.-P. Fan, H. Tang, C. Chen, L. Van Gool, OSFormer: One-stage camouflaged instance segmentation with transformers, in: European Conference on Computer Vision, 2022.

[88] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, I. Sutskever, Zero-shot text-to-image generation, in: International Conference on Machine Learning, 2021.

[89] G. Mittal, S. Agrawal, A. Agarwal, S. Mehta, T. Marwah, Interactive image generation using scene graphs, arXiv preprint arXiv:1905.03743 (2019).

[90] X. Guo, H. Wu, Y. Cheng, S. Rennie, G. Tesauro, R. Feris, Dialog-based in-

teractive image retrieval, Advances in neural information processing systems 31 (2018).

[91] Y. Cheng, Z. Gan, Y. Li, J. Liu, J. Gao, Sequential attention gan for interactive image editing, in: ACM International Conference on Multimedia, 2020.

[92] X. Zhang, L. Wang, J. Xie, P. Zhu, Human-in-the-loop image segmentation and annotation, Science China-Information Sciences 63(11) 2020, 1–3.

[93] Y. Cao, H. Wang, C. Wang, Z. Li, L. Zhang, L. Zhang, MindFinder: interactive sketch-based image search on millions of images, in: ACM International Conference on Multimedia, 2010.

[94] T. Milliron, R. J. Jensen, R. Barzel, A. Finkelstein, A framework for geometric warps and deformations, ACM Transactions on Graphics 21(1) 2002, 20–51.

[95] T. Igarashi, T. Moscovich, J. F. Hughes, As-rigid-as-possible shape manipulation, ACM Transactions on Graphics 24(3) 2005, 1134–1141.

[96] W. Yu, J. Du, R. Liu, Y. Li, Y. Zhu, Interactive image inpainting using semantic guidance, in: International Conference on Pattern Recognition, 2022.

[97] P. Pérez, M. Gangnet, A. Blake, Poisson image editing, ACM Transactions on Graphics 22(3) 2003, 313–318.

[98] Z. Lin, Z. Zhang, K.-R. Zhang, B. Ren, M.-M. Cheng, Interactive style transfer: All is your palette, arXiv preprint arXiv:2203.13470 (2022).

[99] X. Pan, A. Tewari, T. Leimkühler, L. Liu, A. Meka, C. Theobalt, Drag your GAN: Interactive point-based manipulation on the generative image manifold, in: ACM SIG International Conference on Computer Graphics and Interactive Techniques, 2023.

[100] Y. Shi, C. Xue, J. Pan, W. Zhang, V. Y. Tan, S. Bai, DragDiffusion: Harnessing diffusion models for interactive point-based image editing, arXiv preprint arXiv:2306.14435 (2023).

[101] Y. Yang, M. Z. Hossain, T. Gedeon, S. Rahman, S2FGAN: semantically aware interactive sketch-to-face translation, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022.

[102] Y. Li, X. Yu, X. Han, N. Jiang, K. Jia, J. Lu, A deep learning based interactive sketching system for fashion images design, arXiv preprint arXiv:2010.04413

(2020).

[103] Y. Boykov, G. Funka-Lea, Graph cuts and efficient N-D image segmentation, International Journal of Computer Vision 70(2) 2006, 109–131.

[104] Y. Boykov, V. Kolmogorov, An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, IEEE Transactions on Pattern Analysis and Machine Intelligence 26(9) 2004, 1124–1137.

[105] A. Blake, C. Rother, M. Brown, P. Perez, P. Torr, Interactive image segmentation using an adaptive GMMRF model, in: European Conference on Computer Vision, 2004.

[106] B. L. Price, B. Morse, S. Cohen, Geodesic graph cut for interactive image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2010.

[107] T. H. Kim, K. M. Lee, S. U. Lee, Nonparametric higher-order learning for interactive segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2010.

[108] P. A. De Miranda, A. X. Falcão, J. K. Udupa, Synergistic arc-weight estimation for interactive image segmentation using graphs, Computer Vision and Image Understanding 114(1) 2010, 85–99.

[109] V. Gulshan, C. Rother, A. Criminisi, A. Blake, A. Zisserman, Geodesic star convexity for interactive image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2010.

[110] O. Veksler, Star shape prior for graph-cut image segmentation, in: European Conference on Computer Vision, 2008.

[111] J. Bai, X. Wu, Error-tolerant scribbles based interactive image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2014.

[112] T. Wang, S. Qi, Z. Ji, Q. Sun, P. Fu, Q. Ge, Error-tolerant label prior for interactive image segmentation, Information Sciences 538 2020, 384–395.

[113] V. Vezhnevets, V. Konouchine, GrowCut: Interactive multi-label nd image segmentation by cellular automata, in: Proceedings of Graphicon, 2005.

[114] A. Protiere, G. Sapiro, Interactive image segmentation via adaptive weighted

distances, IEEE Transactions on Image Processing 16(4) 2007, 1046–1057.

[115] S. Xiang, F. Nie, C. Zhang, C. Zhang, Interactive natural image segmentation via spline regression, IEEE Transactions on Image Processing 18(7) 2009, 1623–1632.

[116] L. Ding, A. Yilmaz, Interactive image segmentation using probabilistic hypergraphs, Pattern Recognition 43(5) 2010, 1863–1873.

[117] J. Ning, L. Zhang, D. Zhang, C. Wu, Interactive image segmentation by maximal similarity based region merging, Pattern Recognition 43(2) 2010, 445–456.

[118] L. Zhang, Q. Ji, A bayesian network model for automatic and interactive image segmentation, IEEE Transactions on Image Processing 20(9) 2011, 2582–2593.

[119] T. N. A. Nguyen, J. Cai, J. Zhang, J. Zheng, Robust interactive image segmentation using convex active contours, IEEE Transactions on Image Processing 21(8) 2012, 3734–3743.

[120] A. Noma, A. B. Graciano, R. M. Cesar Jr, L. A. Consularo, I. Bloch, Interactive image segmentation by matching attributed relational graphs, Pattern Recognition 45(3) 2012, 1159–1179.

[121] C. Panagiotakis, H. Papadakis, E. Grinias, N. Komodakis, P. Fragopoulou, G. Tziritas, Interactive image segmentation based on synthetic graph coordinates, Pattern Recognition 46(11) 2013, 2940–2952.

[122] C. Jung, M. Jian, J. Liu, L. Jiao, Y. Shen, Interactive image segmentation via kernel propagation, Pattern Recognition 47(8) 2014, 2745–2755.

[123] E. Hu, S. Chen, D. Zhang, X. Yin, Semisupervised kernel matrix learning by kernel propagation, IEEE Transactions on Neural Networks and Learning Systems 21(11) 2010, 1831–1841.

[124] J. Sourati, D. Erdogmus, J. G. Dy, D. H. Brooks, Accelerated learning-based interactive image segmentation using pairwise constraints, IEEE Transactions on Image Processing 23(7) 2014, 3057–3070.

[125] Y. Chen, A. B. Cremers, Z. Cao, Interactive color image segmentation via iterative evidential labeling, Information Fusion 20 2014, 292–304.

[126] M. Jian, C. Jung, Interactive image segmentation using adaptive constraint propagation, IEEE Transactions on Image Processing 25(3) 2016, 1301–1311.

[127] E. Zemene, M. Pelillo, Interactive image segmentation using constrained dominant sets, in: European Conference on Computer Vision, 2016.

[128] T. Wang, Z. Ji, Q. Sun, Q. Chen, X.-Y. Jing, Interactive multilabel image segmentation via robust multilayer graph constraints, IEEE Transactions on Multimedia 18(12) 2016, 2358–2371.

[129] T. Wang, Z. Ji, Q. Sun, Q. Chen, Q. Ge, J. Yang, Diffusive likelihood for interactive image segmentation, Pattern Recognition 79 2018, 440–451.

[130] T. Wang, J. Yang, Z. Ji, Q. Sun, Probabilistic diffusion for interactive image segmentation, IEEE Transactions on Image Processing 28(1) 2019, 330–342.

[131] G. Song, H. Myeong, K. Mu Lee, SeedNet: Automatic seed generation with deep reinforcement learning for robust interactive segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[132] X. Chen, Z. Zhao, Y. Zhang, M. Duan, D. Qi, H. Zhao, FocalClick: Towards practical interactive image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022.

[133] Q. Liu, M. Zheng, B. Planche, S. Karanam, T. Chen, M. Niethammer, Z. Wu, PseudoClick: Interactive image segmentation with click imitation, in: European Conference on Computer Vision, 2022.

[134] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in Neural Information Processing Systems (2017).

[135] Y. Gui, B. Zhou, J. Zhang, C. Sun, L. Xiang, J. Zhang, Learning interactive multi-object segmentation through appearance embedding and spatial attention, IET Image Processing 16(10) 2022, 2722–2737.

[136] H. Li, J. Ni, Z. Li, Y. Qian, T. Wang, Enhanced spatial awareness for deep interactive image segmentation, in: Chinese Conference on Pattern Recognition and Computer Vision, 2022.

[137] M. Zhou, H. Wang, Q. Zhao, Y. Li, Y. Huang, D. Meng, Y. Zheng, Interactive segmentation as gaussion process classification, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023.

[138] F. Du, J. Yuan, Z. Wang, F. Wang, Efficient mask correction for click-based in-

teractive image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023.

[139] Q. Wei, H. Zhang, J.-H. Yong, Focused and collaborative feedback integration for interactive image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023.

[140] L. Yang, W. Zi, H. Chen, S. Peng, DRE-Net: A dynamic radius-encoding neural network with an incremental training strategy for interactive segmentation of remote sensing images, Remote Sensing 15(3) 2023, 801.

[141] Q. Wei, H. Zhang, J.-H. Yong, Boosting interactive image segmentation by exploiting semantic clues, in: International Conference on Multimedia and Expo, 2023.

[142] J. Zhang, Y. Shi, J. Sun, L. Wang, L. Zhou, Y. Gao, D. Shen, Interactive medical image segmentation via a point-based interaction, Artificial Intelligence in Medicine 111 2021, 101998.

[143] Q. Liu, Z. Xu, Y. Jiao, M. Niethammer, iSegFormer: interactive segmentation via transformers with application to 3D knee MR images, in: International Conference on Medical Image Computing and Computer Assisted Intervention, 2022.

[144] V. Lempitsky, P. Kohli, C. Rother, T. Sharp, Image segmentation with a bounding box prior, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2009.

[145] M.-M. Cheng, V. A. Prisacariu, S. Zheng, P. H. Torr, C. Rother, DenseCut: Densely connected CRFs for realtime GrabCut, Computer Graphics Forum 34(7) 2015, 193–201.

[146] J. Wu, Y. Zhao, J.-Y. Zhu, S. Luo, Z. Tu, MILcut: A sweeping line multiple instance learning paradigm for interactive image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2014.

[147] H. Yu, Y. Zhou, H. Qian, M. Xian, S. Wang, LooseCut: Interactive image segmentation with loosely bounded boxes, in: IEEE International Conference on Image Processing, 2017.

[148] N. Xu, B. Price, S. Cohen, J. Yang, T. Huang, Deep GrabCut for object selection,

in: British Machine Vision Conference, 2017.

[149] H. Le, L. Mai, B. Price, S. Cohen, H. Jin, F. Liu, Interactive boundary prediction for object selection, in: European Conference on Computer Vision, 2018.

[150] D. P. Papadopoulos, J. R. Uijlings, F. Keller, V. Ferrari, Extreme clicking for efficient object annotation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2017.

[151] Z. Wang, D. Acuna, H. Ling, A. Kar, S. Fidler, Object instance annotation with deep extreme level set evolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[152] S. Khan, A. H. Shahin, J. Villafruela, J. Shen, L. Shao, Extreme points derived confidence map as a cue for class-agnostic interactive segmentation using deep neural network, in: International Conference on Medical Image Computing and Computer Assisted Intervention, 2019.

[153] K. B. Girum, G. Créhange, R. Hussain, A. Lalande, Fast interactive medical image segmentation with weakly supervised deep learning method, International Journal of Computer Assisted Radiology and Surgery 15(9) 2020, 1437–1444.

[154] E. N. Mortensen, W. A. Barrett, Intelligent scissors for image composition, in: ACM SIG International Conference on Computer Graphics and Interactive Techniques, 1995.

[155] E. N. Mortensen, W. A. Barrett, Interactive segmentation with intelligent scissors, Graphical Models and Image Processing 60(5) 1998, 349–384.

[156] A. Mishra, A. Wong, W. Zhang, D. Clausi, P. Fieguth, Improved interactive medical image segmentation using enhanced intelligent scissors, in: Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2008.

[157] M. Pizenberg, A. Carlier, E. Faure, V. Charvillat, Outlining objects for interactive segmentation on touch devices, in: ACM International Conference on Multimedia, 2017.

[158] J. Liang, T. McInerney, D. Terzopoulos, Interactive medical image segmentation with united snakes, in: International Conference on Medical Image Computing and Computer Assisted Intervention, 1999.

[159] T. McInerney, SketchSnakes: Sketch-line initialized snakes for efficient interactive medical image segmentation, Computerized Medical Imaging and Graphics 32(5) 2008, 331–352.

[160] W. Zhou, Y. Xie, et al., Interactive medical image segmentation using snake and multiscale curve editing, Computational and Mathematical Methods in Medicine 2013 2013, 325903.

[161] P. Karasev, I. Kolesov, K. Fritscher, P. Vela, P. Mitchell, A. Tannenbaum, Interactive medical image segmentation using PDE control of active contours, IEEE Transactions on Medical Imaging 32(11) 2013, 2127–2139.

[162] Y. Li, J. Sun, C.-K. Tang, H.-Y. Shum, Lazy snapping, ACM Transactions on Graphics 23(3) 2004, 303–308.

[163] T. V. Spina, P. A. de Miranda, A. X. Falcao, Hybrid approaches for interactive image segmentation using the live markers paradigm, IEEE Transactions on Image Processing 23(12) 2014, 5756–5769.

[164] S. Majumder, A. Rai, A. Khurana, A. Yao, Two-in-one refinement for interactive segmentation, in: British Machine Vision Conference, 2020.

[165] J.-L. Jones, X. Xie, E. Essa, Combining region-based and imprecise boundary-based cues for interactive medical image segmentation, International Journal for Numerical Methods in Biomedical Engineering 30(12) 2014, 1649–1666.

[166] B. Zhou, L. Chen, Z. Wang, Interactive deep editing framework for medical image segmentation, in: International Conference on Medical Image Computing and Computer Assisted Intervention, 2019.

[167] T. Zhou, L. Li, G. Bredell, J. Li, J. Unkelbach, E. Konukoglu, Volumetric memory network for interactive medical image segmentation, Medical Image Analysis 83 2023, 102599.

[168] Y. Liu, Y. Yu, Interactive image segmentation based on level sets of probabilities, IEEE Transactions on Visualization and Computer Graphics 18(2) 2011, 202–213.

[169] E. Agustsson, J. R. Uijlings, V. Ferrari, Interactive full image segmentation by considering all regions jointly, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[170] D.-P. Fan, M.-M. Cheng, J.-J. Liu, S.-H. Gao, Q. Hou, A. Borji, Salient objects in clutter: Bringing salient object detection to the foreground, in: European Conference on Computer Vision, 2018.

[171] D.-P. Fan, G.-P. Ji, G. Sun, M.-M. Cheng, J. Shen, L. Shao, Camouflaged object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

[172] K. McGuinness, N. E. O’ connor, A comparative evaluation of interactive segmentation algorithms, Pattern Recognition 43(2) 2010, 434–444.

[173] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, A. Zisserman, The PASCAL visual object classes (VOC) challenge, International Journal of Computer Vision 88(2) 2010, 303–338.

[174] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, A. Sorkine-Hornung, A benchmark dataset and evaluation methodology for video object segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016.

[175] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft COCO: Common objects in context, in: European Conference on Computer Vision, 2014.

[176] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016.

[177] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, J. Malik, Semantic contours from inverse detectors, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2011.

[178] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: A large-scale hierarchical image database, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2009.

[179] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An imperative style, high-performance deep learning library, in: Ad-

vances in Neural Information Processing Systems, 2019.

[180] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, P. Torr, Res2Net: A new multi-scale backbone architecture, IEEE Transactions on Pattern Analysis and Machine Intelligence 43(2) 2019, 652–662.

[181] F. Xia, P. Wang, L.-C. Chen, A. L. Yuille, Zoom better to see clearer: Human and object parsing with hierarchical auto-zoom net, in: European Conference on Computer Vision, 2016.

[182] W. Chen, Z. Jiang, Z. Wang, K. Cui, X. Qian, Collaborative global-local networks for memory-efficient segmentation of ultra-high resolution images, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[183] H. Tokunaga, Y. Teramoto, A. Yoshizawa, R. Bise, Adaptive weighting multi-field-of-view CNN for semantic segmentation in pathology, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[184] H. K. Cheng, J. Chung, Y.-W. Tai, C.-K. Tang, CascadePSP: Toward class-agnostic and very high-resolution segmentation via global and local refinement, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

[185] C. Huynh, A. T. Tran, K. Luu, M. Hoai, Progressive semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021.

[186] Z. Lin, Z. Zhang, L.-Z. Chen, M.-M. Cheng, S.-P. Lu, Interactive image segmentation with first click attention, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

[187] B. Cheng, R. Girshick, P. Dollar, A. C. Berg, A. Kirillov, Boundary iou: Improving object-centric image segmentation evaluation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021.

[188] Y. Zeng, P. Zhang, J. Zhang, Z. Lin, H. Lu, Towards high-resolution salient object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019.

[189] L. A. Mansilla, P. A. Miranda, Oriented image foresting transform segmentation:

Connectivity constraints with adjustable width, in: SIBGRAPI Conference on Graphics, Patterns and Images, 2016.

[190] L. A. Mansilla, P. A. Miranda, F. A. Cappabianco, Oriented image foresting transform segmentation with connectivity constraints, in: IEEE International Conference on Image Processing, 2016.

[191] T. Y. Zhang, C. Y. Suen, A fast parallel algorithm for thinning digital patterns, Communications of the ACM 27(3) 1984, 236–239.

[192] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, et al., Deep high-resolution representation learning for visual recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 43(10) 2020, 3349–3364.

[193] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, A. Sorkine-Hornung, A benchmark dataset and evaluation methodology for video object segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016.

[194] G. Gaál, B. Maga, A. Lukács, Attention U-Net based adversarial architectures for chest X-ray lung segmentation, arXiv preprint arXiv:2003.10304 (2020).

[195] Y.-H. Wu, S.-H. Gao, J. Mei, J. Xu, D.-P. Fan, R.-G. Zhang, M.-M. Cheng, JCS: An explainable COVID-19 diagnosis system by joint classification and segmentation, IEEE Transactions on Image Processing 30 2021, 3113–3126.

[196] J. Born, G. Brändle, M. Cossio, M. Disdier, J. Goulet, J. Roulin, N. Wiedemann, Pocovid-net: automatic detection of COVID-19 from a new lung ultrasound imaging dataset (pocus), arXiv preprint arXiv:2004.12084 (2020).

[197] J. F. Brinkley, Spatial anatomic knowledge for 2-D interactive medical image segmentation and matching, in: Proceedings of the Annual Symposium on Computer Application in Medical Care, 1991.

[198] M. Rajchl, M. C. Lee, O. Oktay, K. Kamnitsas, J. Passerat-Palmbach, W. Bai, M. Damodaram, M. A. Rutherford, J. V. Hajnal, B. Kainz, et al., DeepCut: Object segmentation from bounding box annotations using convolutional neural networks, IEEE Transactions on Medical Imaging 36(2) 2017, 674–683.

[199] H. J. Lee, J. U. Kim, S. Lee, H. G. Kim, Y. M. Ro, Structure boundary preserving

segmentation for medical image with ambiguous boundary, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

[200] Q. Zhu, B. Du, P. Yan, Boundary-weighted domain adaptive neural network for prostate MR image segmentation, IEEE Transactions on Medical Imaging 39(3) 2019, 753–763.

[201] J. Sklansky, Finding the convex hull of a simple polygon, Pattern Recognition Letters 1(2) 1982, 79–83.

[202] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations, 2015.

[203] V7Labs, Covid-19 xray dataset, Accessed Aug. 1 2023. https://www.github.com/v7labs/covid-19-xray-dataset (2020).

[204] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, et al., The multimodal brain tumor image segmentation benchmark (BRATS), IEEE Transactions on Medical Imaging 34(10) 2014, 1993–2024.

[205] S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycki, J. S. Kirby, J. B. Freymann, K. Farahani, C. Davatzikos, Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features, Scientific Data 4(1) 2017, 1–13.

[206] S. Bakas, M. Reyes, A. Jakab, S. Bauer, M. Rempfler, A. Crimi, R. T. Shinohara, C. Berger, S. M. Ha, M. Rozycki, et al., Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge, arXiv preprint arXiv:1811.02629 (2018).

[207] Kaggle, Ultrasound nerve segmentation | kaggle, Accessed Aug. 1 2023. https://www.kaggle.com/c/ultrasound-nerve-segmentation (2020).

[208] D.-P. Fan, G.-P. Ji, T. Zhou, G. Chen, H. Fu, J. Shen, L. Shao, PraNet: Parallel reverse attention network for polyp segmentation, in: International Conference on Medical Image Computing and Computer Assisted Intervention, 2020.

[209] N. C. Codella, D. Gutman, M. E. Celebi, B. Helba, M. A. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. Mishra, H. Kittler, et al., Skin lesion analysis to-

ward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic), in: IEEE International Symposium on Biomedical Imaging, 2018.

[210] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: International Conference on Learning Representations, 2015.

[211] Q. Yan, B. Wang, D. Gong, C. Luo, W. Zhao, J. Shen, Q. Shi, S. Jin, L. Zhang, Z. You, COVID-19 chest CT image segmentation–a deep convolutional neural network solution, arXiv preprint arXiv:2004.10987 (2020).

[212] G. Wang, X. Liu, C. Li, Z. Xu, J. Ruan, H. Zhu, T. Meng, K. Li, N. Huang, S. Zhang, A noise-robust framework for automatic segmentation of COVID-19 pneumonia lesions from CT images, IEEE Transactions on Medical Imaging 39(8) 2020, 2653–2663.