Computational Visual Media
DOI 10.1007/s41095-xxx-xxxx-x Vol. x, No. x, mouth year, xx–xx

Article Type (Research)

序列性交互式图像分割

林铮, 张钊, 朱子悦, 范登平 (✉), 刘夏雷

Abstract 交互式图像分割 (IIS) 是获取像素级标注的重要技术。在许多情况下，目标对象共享相似的语义，例如语义分割、实例分割和人体解析任务等。先前的对象表征、用户交互和预测结果可以为当前标注提供适当的先验信息，然而，IIS 方法忽略了这些联系和这些新获得的信息。本文提出了一种序列性交互式图像分割 (SIIS) 任务，以最小化用户在序列数据上的交互代价; 同时提出了一个具有两种相关设计的实用解决方案。首先是一个新的交互模式。在标注一个新样本时，我们的方法可以根据先前的标注自动提出一个初始交互点推荐。它极大地减少了用户的交互负担。其次是密集性在线优化策略。为了减少标注特定目标时的语义差距，我们进一步优化了模型参数，利用之前标注的样本进行密集性监督。实验证明了我们的方法的有效性和所提出的 SIIS 任务的重要性。

Keywords 交互式分割, 用户交互, 对象分割.

1 介绍

在图像编辑和数据标注领域，以最少的人力和时间成本获得高质量的像素级掩膜是至关重要的。因此，学术社区对交互式图像分割 (IIS) 技术进行了大量的关注，该技术通过用户参与到分割过程和反复提供交互信息来获得更好的掩膜。为了减轻用户的负担，对 IIS 的研究主要集中在两个方面。一是精心设计交互模式 [19, 26, 36, 38, 48, 51], 使用户能够以最小的交互成本提供更多的信息。二是精心设计后端算法 [20, 28, 30, 33, 35, 37, 41] 来最大限度地利用用户提供的信息。

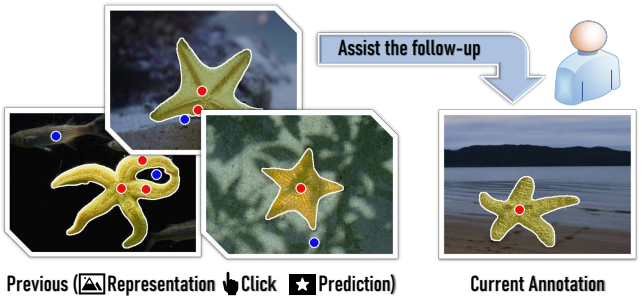


图 1 序列性交互式图像分割动机。在标注过程中，先前的对象表征、用户交互以及最终的预测结果可以为当前的交互和预测提供帮助。

在实践中，用户通常会对多个相关图像进行标注，例如在语义分割任务中对具有相同类别的图像进行标注，在人/场景解析任务中对具有相同子结构的图像进行标注。同时，在推理阶段，IIS 模型将获得一个几乎准确的掩膜，而不像大多数计算机视觉任务中的不确定掩膜。以上观察结果启发我们思考是否可以利用之前的标注信息来辅助当前的标注信息，见 图 1. 然而，这个想法在很大程度上被当前的 IIS 方法忽略了，这些方法独立处理每个图像，没有考虑以前标注中有用的先验信息。一个最近的工作 [25] 首先尝试将交互分割作为一个序列任务，通过用户点击进行参数优化。但这个工作仅利用点击信息，采用非精确掩膜作为正则化。在改工作的基础上我们进一步提出在序列任务中探索交互逻辑层次。此外我们建议在特定的任务中获得一个精确的掩膜层以更好地优化参数。

本文针对 SIIS 问题，围绕交互模式和后端算法这两个研究点，提出了系统的解决方案，对应初始交互点推荐和在线参数提纯优化两个模块。在交互方式上，

我们设计了一种新的交互逻辑，通过初始交互点推荐 (ICP)，大大降低了用户的交互负担，见 图 2 和 图 3. 特别地，ICP 维护了一个语义目标的初始交互点特征库。当处理一个新的目标时，通过相似性测量，ICP 将提出一个最可能的位置的初始交互点作为一个真实的交互。如果采纳了推荐，用户可以进一步进行交互以改进分割; 否则，直接用新点击修正推荐。在后端算法方面，我们基于之前的交互结果，提出了一种序列性交互式分割的在线参数提纯优化 (OPO) 策略，见 图 2. OPO 为每个语义目标保留一组参数，以缩小语义差距。随着用户标注的增加，我们的流程框架对于特定的语义目标将变得更加高效。

我们的贡献可以概括如下:

- 本文从交互方式和后端算法两方面定义并阐述了序列性交互式图像分割 (SIIS) 任务。
- 为了提高交互效率，我们为 SIIS 设计了初始交互点推荐 (ICP)，以推荐初始交互点来代替真实的用户输入。
- 为了更好地利用交互信息，我们提出了在线参数提纯优化 (OPO)，利用先前的标注信息使模型适应特定的语义目标。

2 相关工作

2.1 交互式图像分割

交互式图像分割领域已经被人们探索了近二十年，研究主要集中在两个视角。*1) 交互模式* 交互模式的研究旨在让用户以最少的交互提供最多的信息。传统方法使用涂鸦 [3, 15, 22, 24, 44, 46] 来指示前景和背景区域。此外还有许多变体在社区中被研究，如跨实例涂鸦 [47]，带容错涂鸦 [2]，包围盒 [40] 和自动边界套索 [27, 39]。最近深度学习技术带来了更强的感知能力，使得更轻量的交互模式成为可能。例如用户可以直接点击目标对象来选择对象，点击背景来删除错误预测 [49]。此外人们还研究了一些较为轻量的交互模式例如极值点 [38] 和边界点击 [19, 26]。IOG 方法 [51] 作为一种新颖的方式，将外边框和内点击相结合，也取得了很好的效果。*2) 后端算法*. 后端算法的研究旨在最大限度地利用用户提供的交互信息进行准确预测。传统的方法主要是基于颜色特征 [5, 6, 14, 23]。最近，基于卷积神经网络 [49]、循环神经网络 [1, 7]、图卷积神经网络 [34]、强化学习 [42] 的方法在 IIS 任务中涌现. 相关研究也涉及了各种体系结构, 例如区域细化块 [29] 和双流融合 [18]。一些研究 [28, 30] 试图解决交互式分割中的歧义性分割问题。此外，一些关于交互式分割的重要信息也得到了关注，例如训练策略 [35]，交互图 [4]，和用户的意图 [32, 33, 37] 等.

2.2 使用在线学习的交互式分割

在线学习已经在许多与细分相关的工作中使用 [8, 50]。对于 IIS 任务，用户交互可以作为预测的参考也可以作为微调模型的监督信号。BRS [20] 首先将这个想法用于个体交互分割根据它的假设，用户点击位置模型预测的置信度可能不够高, 这些不确定性为模型优化提供了可能, BRS 以错误预测的被点击像素作为惩罚，对输入的距离图进行微调，更加针对目标对象，确保预测能够很好地覆盖交互点。f-BRS [41] 通过反向传播部分模型而不是整个模型，改进了 BRS 算法。通过这种方式使在线学习更加高效。最近 Kontogianni *et al.* [25] 初步尝试将点击位置的监督引入到图像序列分割中，取得了很好的效果。它采用了只有几个正负点击的稀疏监督，并在稀疏优化中使用这些不完整的掩膜作为正则化约束。然而，交互式分割的特殊性在于，用户在与先前的图像交互后会得到完整的掩膜。我们的方法更进一步，直接使用所有先前的预测作为密集性监督，而不是用户的几次点击。在此基础上，我们提出了 OPO，它利用之前的最终标注结果来帮助后续的图片处理。

3 提出的方法

在本节中，我们将分三个部分介绍所提出的方法。章节 3.1介绍了我们改进的 DeepLab v3+ [9]，它是专门为序列性交互式分割设计的。章节 3.2描述了初始交互点推荐, 它根据图像序列中先前的初始交互点，为用户提供初始交互点推荐。在 章节 3.3我们提出了在线参数提纯优化（OPO）。它在修改后的 DeepLab v3+ 中根据先前的标注进行在线训练，优化提纯参数。

3.1 网络结构

交互式分割是对象分割中的一个特定任务。对于大多数以前基于点击的交互式分割工作 [25, 30, 33, 35,

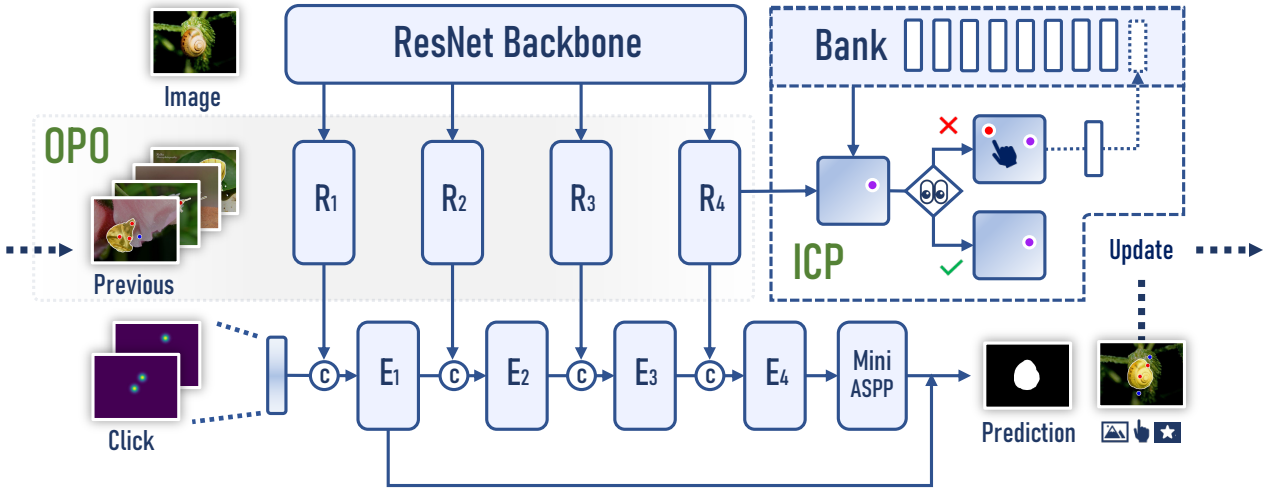


图 2 我们提出的序列性交互式分割方法的整体流程框架。ICP 是初始交互点推荐，详见 章节 3.2。它的目的是自动获得一个减少用户负担的初始交互点推荐。OPO 是在线参数提纯优化，详见章节 3.3。它根据先前的标注不断优化语义目标的专用参数，以保证推荐结果空间的语义自适应和更高效的分割。标志 © 的含义是两个卷积层。红点 "•" 和蓝点 "•" 表示在前景和背景中的点击，紫点 "•" 表示推荐的点击。

41], 他们通常采用 DeepLab v3+ [9] 作为分割网络，5 个通道 (RGB 图像 + 正/负点击图) 作为输入。这种网络结构在大多数时候都运行良好。然而，在序列性交互式分割中，原始结构有两个问题。第一，我们需要利用特定类别图像之间的特征相关性。传统的 5 通道输入由于标注输入会干扰语义特征，不能满足我们的要求。换句话说，交互点之间的相关性将显著增强，语义相似性将显著减弱。第二，在序列性交互式分割中参数需要不断优化。我们需要为每个类别保存特定的参数。优化全局参数将显著增加实际应用环境中硬件存储 (如内存或显存) 的负担。基于以上原因，我们将原有的架构分为两个部分，特征提取部分和交互分割部分，如 图 2所示. 我们称之为改进的 DeepLab v3+。在特征提取方面，我们也采用输出步长 16 的 ResNet-101 [17] 作为主干网络。特征提取最后四层的通道为 $\{256, 512, 1024, 2048\}$. 这些特征被送入一个简单的净化模块，该模块包含几个 $\times 1$ 的卷积对特定类别的特征进行简化和净化。通道缩减特征定义为 $\{\mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3, \mathbf{R}_4\}$，通道为 $\{32, 64, 128, 256\}$，仅仅只有原始的 $\frac{1}{8}$。有了标注指导图（点击图）和这些通道缩减特征，我们设计了一个 Mini-DeepLab v3+ 模块，其结构与原来的模块相似。编码器模块的输入是标注指导图 $\mathbf{E}_0$，它是两个基于点的高斯图。输入特征与通道缩减特征逐渐联合，表示为:

$$\mathbf{E}_i = \mathcal{C}\left(\mathbf{R}_i \oplus \mathcal{D}\left(\mathbf{E}_{i-1}\right)\right), i \in \{1, \ldots, 4\}, \tag{1}$$

其中 $\mathcal{D}(\cdot)$ 指下采样，$\oplus$ 指的是特征拼接，$\mathcal{C}$ 表示内核大小为 3×3 的两个卷积层. 对于微型编码器模块输出 $\mathbf{E}_4$，它将被送入微型 ASPP 模块。与原来不同的是，下采样是 $\frac{1}{4}$ 而不是 $\frac{1}{8}$. 带有 64 个通道的微型 ASPP 的输出最终将通过 $\mathbf{E}_1$ 拼接并且卷积到最终的结果虽然修改后的 DeepLab v3+ 在原始版本的基础上增加了一些部件，但由于通道减少，这个网络比原始网络更轻便。由于特征提取和交互分割的分离，可以实现更多的序列操作，我们可以更好地探索序列性交互式分割，如 ICP 和 OPO。

3.2 初始交互点推荐

在序列性交互式分割中，如何减轻用户在交互逻辑上的负担是一个重要的问题。我们提出初始交互点推荐 (ICP)。它维护了一个初始点击库，该库记录了每个类别之前的初始交互点所在像素上的特征向量。它初始为空。当用户对某一特定类别进行交互式分割时，将计算所有像素点的特征向量与该库之前初始交互点的特征向量的相似度。对于初次的分割，将最相似的像素标记为初始交互点推荐。如果用户对初始交互点推荐不满意

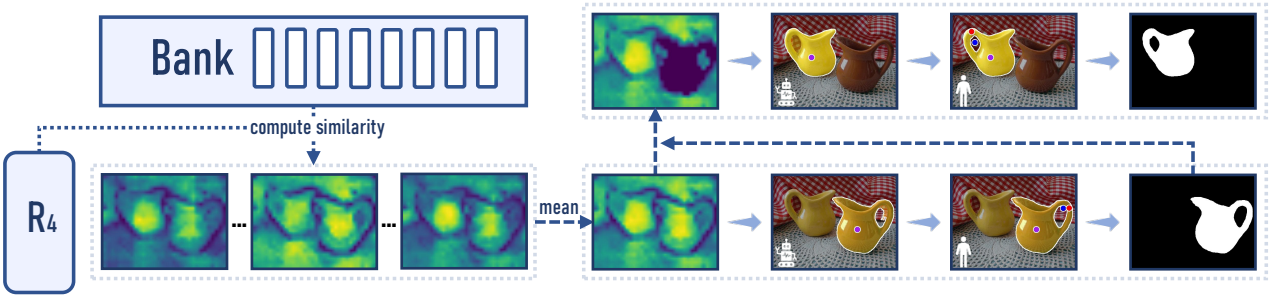


图 3 初始交互点推荐 (ICP) 的细节. ICP 给出了一个基于置信度图的初始交互点。置信图是使用图像特征和储存的所有先前初始交互点的特征向量计算出来的相似度图的平均。得到的交互点结果将作为一个来自用户的真实的点击交互并进行分割。当得到推荐的结果正确时，用户可以通过提供更多的交互来进一步细化它。在多目标场景中，将先前的掩膜从置信度图中删除，以便 ICP 可以为下一个目标提出新的初始交互点。

或推荐有错误，用户可以手动选择初始交互点，继续后续的交互分割。当用户手动选择初始点或者采用正确的初始交互点推荐，对应的特征向量会存储在初始点击库中。

如何推荐初始交互点？我们使用余弦相似度 (见公式 (2) 中的 ϕ) 找到推荐点。假设目标图像是 $\mathcal{T}$ 且我们选择的图像特征定义为 $\mathbf{F}$。$\mathbf{F}(p)$ 表示对应像素的特征向量 p。然后计算出推荐点 $\hat{p}$ 的公式如下:

$$\hat{p} = \underset{p_n \in (\mathcal{T}-\mathcal{I})}{\arg\max} \frac{\sum_{i=1}^{n-1} \phi\left(\mathbf{F}(p_n), \mathbf{F}(p_i)\right)}{n-1}, \tag{2}$$

$$\phi(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\|\|\mathbf{b}\|}, \tag{3}$$

其中 p_n 表示推荐点，$p_1 \ldots p_{n-1}$ 表示之前的初始点击，$\mathcal{I}$ 表示忽略性掩膜，这些被初始化为 $\varnothing$ 且被用于多个实例的分割。在实践中，我们可以在选择最大值点之前对置信度图进行均值滤波，以防止偶尔出现极值点。我们选择最后一层 $\mathbf{R}_4$ 为初始交互点推荐生成通道缩减的特征。它将提供更多语义信息。关于选择不同层特征的消融实验见 表 3.

在真实的场景中，选择初始交互点推荐可以采用多种交互方式。例如对于用户不满意的推荐点，用户可以通过鼠标中键重新选择初始点，并继续使用鼠标左键和右键进行交互式分割。对于多种语义标注, ICP 将为每个类别保留一个初始点击库。当用户选择要标注的语义标签时，会生成相应的初始交互点推荐并显示出来。如果同一个图片中有多个相同类别的实例，ICP 还可以推荐多个实例。如图 3, 每次用户为一个实例完成标注 $\mathcal{A}$，忽略性掩膜 $\mathcal{I}$ 将会更新到 $\mathcal{I} \cup \mathcal{A}$. 接下来的推荐点将不会和之前的重复。

3.3 在线参数提纯优化

作为基于先前标注结果的在线训练的首次探索，我们采用了最简洁的训练设置，这与训练基准模型时的设置类似。核心区别在于，对于基准模型的训练，前景点和背景点的个数分别在 [1 10] 和 [0 10] 之间，更好地模拟用户操作。对于在线训练，为减少交互点的影响，则在 [1, 5] 和 [0, 5] 范围内。批大小设为 8，经过完整的交互分割后训练 4 次迭代。图像和实例是从所有带标注的图像和实例中选择的。在在线训练过程中，我们也使用随机梯度下降法进行优化，学习率固定为 5×10^{-3}。与其他在线学习方法不同，我们只优化了提纯模块中的一小部分参数，这被称为在线参数提纯优化，如图 2所示。提纯模块由多个 1×1 的卷积组成，其参数起到提取原始特征的作用。对于序列性交互式分割，每个类别所依赖的特征往往是不同的。通过提纯模块，对特征进行重组整合，通过在线学习改变净化模块参数。因为它有利于从原始的复杂图像特征中提取特定类别的参数，所以被称为提纯模块。在该模块之前，特征是不纯的，因为它们代表了各种图像特征。经过这个模块简化后的特征可以更好地表示这类特定的对象。如 图 6所示，初始交互点击的分割结果反了映参数符合该类别对应特征。

用户每完成一次实例分割，就会在真实场景中进行一轮该类别的在线学习。对于每个类别，分割系统会保存一组提纯模块的参数。因为参数非常小, 如表 7所示，所需的存储空间小，不会造成负担。当用户选择要标注

#	ICP	OPO	PASCAL		COCO		CoSOD3k		CoCA		Fashionpedia		LeedsButterfly	
			@85%	@90%	@85%	@90%	@85%	@90%	@85%	@90%	@85%	@90%	@85%	@90%
(a)			3.18	4.07	5.81	8.25	4.86	7.24	6.81	9.61	13.87	16.47	2.16	2.89
(b)	✓		2.82	3.70	5.40	7.85	4.34	6.70	6.50	9.32	13.68	16.30	1.37	2.07
(c)		✓	3.11	3.98	5.36	7.88	4.47	6.82	5.88	8.76	11.34	14.29	1.25	1.48
(d)	✓	✓	2.74	3.60	4.98	7.51	3.93	6.29	5.57	8.48	11.17	14.14	0.29	0.52

表 1 在所有 6 个数据集上进行关于我们提出的初始交互点推荐 (ICP) 和在线参数提纯优化 (OPO) 的核心消融研究，使用 NoC 指标 (@85% 和 @90%)。

的语义标签时，会采用相应的参数。使用在线学习的任务通常面临着难以实时运行的问题，而交互式分割任务则完全不存在这种问题。在大多数情况下，交互式分割过程中用于思考的时间要比计算机处理的时间长得多。所以只要你取一个特定的区间 δ, 并在分割第 n 个对象时使用第 1 到第 $(n-\delta)$ 个样本训练的模型，就可以充分实现用户实时交互的效果。一般情况下，只要 δ 大于 1，就可以满足实时性要求。

4 实验

4.1 实验设置

数据集 *Augmented PASCAL VOC* [11, 16]，一个含 20 个类别的广泛使用的语义分割数据集。与之前的一些工作一样，我们使用训练集 (25832 个实例) 进行训练，使用验证集 (3427 个实例) 进行测试。*COCO* [31]，一个由微软提供的大规模数据集。我们采用三种设置进行测试。为了与单独的交互式分割进行比较，我们与 [33] 使用相同的设置. 为了与序列性交互分割进行比较，我们与 [25] 采用了相同的设置, 其中包括 COCO (Unseen 6k), COCO (Donut, Bench, Umbrella, Bed). *CoSOD3k* [12, 13] 是一个类别丰富的用于协同显著性目标检测的数据集。我们为测试选择了 160 个类别中的 4874 个实例。*CoCA* [52] 是另一个用于协同显著性目标检测的数据集，具有特殊的类别。这些类别不是典型的，并且未出现在其他数据集中，非常适合研究独立的语义任务。我们为测试选择了 80 个类别中的 2143 个实例。*Fashionpedia* [21] 是一个关于时装照片的数据集。我们使用验证集中 46 个类别的 8781 个部件掩膜进行测试。我们将其应用于序列性交互式分割中对目标部件的分割。*LeedsButterfly* [45]，该数据集包含 832 张蝴蝶图像。

评测指标 为了评估交互式分割，我们采用了与大多数交互式分割工作相同的评测指标。采用机器模拟的用户，每次都在最大错误区域的中心选取下一个点。平均点击次数（NoC）表示交互过程中的平均点击次数，直到每个实例达到指定的交并比（IoU）得分。(表示为 @XX%). 数值越小，性能越好。值得一提的是，使用在线训练时的 NoC 值为 5 次实验的平均值。

实现细节 我们使用在 ImageNet 预训练 [10] 的 ResNet-101 [17] 作为主干网络。我们将批大小设置为 8，并训练 30 轮。我们在基准训练中使用二值交叉熵损失函数。我们采用指数学习率衰减策略，每代的初始学习率为 7×10^{-3}，gamma 为 0.95。对于参数优化，我们采用动量为 0.9 的随机梯度下降和 5×10^{-4} 的权重衰减。通过随机翻转和随机裁剪来进行数据增强, 我们裁剪和调整图像大小为 384 × 384。对于交互点模拟，我们与 [33] 中使用类似的策略，也与 [35] 中采用相同的迭代训练策略。所有的实验都是用 PyTorch [43] 框架实现的且运行在一个单独的 NVIDIA Titan XP GPU 上。

4.2 消融实验和讨论

我们在 6 个选定的数据集上进行了充分的关于我们核心部分的消融实验。表 1显示了所有数据集上不同目标阈值上的 NoC 指标。从整体数据观察，无论哪个数据集，无论哪个目标阈值 (@85% 或 90%)，我们的 ICP 和 OPO 都能起到提高性能的作用，这充分证明了我们所提方法的有效性。本节将以 @85% 的数据为例，分析两个核心模块对不同数据集的影响。

Name	Params (M)	FLOPs (G)	SPC (s)
DeepLab v3+	59.345	50.149	0.024
Ours	45.743	32.364	0.016

表 2 我们修改后的 DeepLab v3+ 与原始版本之间的网络比较。(见问题 1)

Dataset	R1	R2	R3	R4	R-MS
CoSOD3k	7.00	7.04	6.84	6.70	6.64
CoCA	9.43	9.46	9.37	9.32	9.23

表 3 ICP 模块使用主干网络不同层的特征时的 NoC(@90%)。"R-MS" 的意思是使用多尺度特征。(见问题 2)

#	Setting	C1	C2	C3	C4	All
(a)	Provide Initial Click	1.62	1.64	1.28	1.48	1.51
(b)	Judge Positive Sample	0.48	0.46	0.47	0.43	0.46
	Judge Negative Sample	0.58	0.56	0.54	0.55	0.56

表 4 ICP 模块的用户调研。(见问题 3)

对于 PASCAL 中的验证数据集 (其类别与训练集相同)，提纯模块中的参数已完全符合这些可见的类别。我们可以发现 ICP 是非常有效的，有 11.37% 的改善。然而，OPO 的改进非常有限。这也是合理的，因为完全拟合的特征更适合提供交互点推荐，而通过有限的样本和可见类别的在线训练很难改善这些参数。COCO 和 CoSOD3k 有少量与训练集重叠的类别，同时 CoCA 中的类是独一无二的。这三个数据集的类别丰富，但每个类别的数量有限，加入 ICP 组的提高分别为 7.04%、10.61%、4.67%，OPO 组的提高分别为 7.70%、7.97%、13.66%。结合 ICP 和 OPO，性能提高可达 14.32%、19.09%、18.21%。这些数据充分反映了我们的方法可以在每个类别样本较少的情况下带来明显的改进。Fashionpedia 是最困难的，因为分割目标是时装的部分，而训练样本是所有的实例，特别是整个人体。采用 ICP 只能改善 1.33%，而采用 OPO 可以达到 18.25%。我们推测，这种现象是因为神经网络有很高的概率将人体作为一个统一的类别，这导致计算特征相似度会因为服装配饰等物品而出现无法匹配的情况。但参数优化仍然有助于提高这种局部对象的性能。在只包含少量蝴蝶的数据集 LeedsButterfly 上 ICP 和 OPO 的改进是显著的。与基准相比，OPO 带来了 42.04% 的改进。对于配备 ICP 的基准模型，改善达到 36.62%，ΔNoC 达到 0.79。加入 OPO 后，ICP 的上涨更显著，为 76.52%，ΔNoC 达到 0.96，其最大值为 1.0。这反映了经过参数优化后获得的特征更加合适初始交互点推荐也就是说，OPO 可以帮助 ICP 获得更好的性能。ICP 和 OPO 结合的 NoC 指标仅为 0.29。这意味着该框架可以在近乎半自动的交互下完成令人满意的标注，这大大减轻了标注的负担。

以下是一些附加的消融实验和一些问题的讨论:

问题 1：这个网络和原来的 DeepLab v3+ 有什么不同? 表 2展示了两种网络架构的主要指标，包括参数的数量 (Params)，每秒浮点运算 (FLOPs)，和每次点击秒数 (SPC)。我们可以看到改良后版本更加轻量。值得一提的是，这并不意味着我们的网络比原版更好，但由于我们的 ICP 和 OPO 的独特设计，我们必须采取这样的改变。

问题 2：ICP 模块如果选择骨干网络的其它层特征会如何？ 表 3探讨了使用提纯模块的其他层特征的情况。我们可以发现使用 $\mathbf{R}_4$ 特征的时候表现最好, 之后依次是 $\mathbf{R}_3$, $\mathbf{R}_1$, $\mathbf{R}_2$. 这与我们的直觉是一致的; 使用最高层的特征信息更有利于初始交互点推荐。我们还对初始交互点推荐进行了额外的多尺度特征实验。我们可以发现，性能还可以进一步提高。

问题 3：ICP 真的能为用户节省时间，减轻交互负担吗? 按照表 4所示，我们对提出的 ICP 模块进行了用户调研，以验证其有效性。选取 COCO [25, 31] 数据集中的 4 个类别的 40 张图像作为调研集。一半的数据为每个类别提供了正确的推荐点，另一半则相反。我们邀请了 20 名志愿者参与我们的用户调研。他们被要求完成两项测试。(a) 一种是在提供的随机图片中找到并点击对应类别的物体。(b) 另一种是通过提供的随机图片和相应的推荐点来判断推荐点是否正确。从表中, 我们可以发现，判断推荐点为错误的时间比判断推荐点为正确的时间要多。两者都少于直接点击对象的时间。这反

Method	PASCAL @85%	COCO @85%	CoSOD3k @90%	CoCA @90%	Fashionpedia @85%	LeedsButterfly @90%
CVPR - DOS [49]	6.88	9.07	11.04	13.04	16.27	5.32
ICCV - RIS [29]	5.12	N/A	N/A	N/A	N/A	N/A
CVPR - LD [28]	N/A	7.86	8.73	11.94	16.41	3.66
BMVC - ITIS [35]	3.80	6.51	8.67	11.42	16.77	3.43
ICCV - MS [30]	3.88	N/A	N/A	N/A	N/A	N/A
CVPR - BRS [20]	N/A	5.16	N/A	N/A	N/A	N/A
CVPR - CMG [37]	3.62	5.90	N/A	N/A	N/A	N/A
CVPR - FCA [33]	2.98	5.28	6.31	9.51	13.31	2.44
CVPR - f-BRS [41]	N/A	5.75	6.93	9.46	14.40	2.86
Ours	2.74	4.98	6.29	8.48	11.17	0.52

表 5 本文方法与其他交互式图像分割方法的 NoC 指标比较。

Dataset	Ours	Full	Foreground
CoSOD3k	4.34 / 6.70	4.77 / 7.15	4.50 / 6.86
CoCA	6.50 / 9.32	6.77 / 9.56	6.55 / 9.38

表 6 当 ICP 模块中采用从全图或前景中随机点击时的 NoC (@85%/@90%)。 (见问题 4)

Name	Params (M)	SPB (s)	CoSOD3k	CoCA
Purification	0.697	0.098	6.82	8.76
Global	45.743	0.277	6.67	8.38

表 7 优化提纯参数与全局参数的比较。最后两列中的值是 NoC@90%。 (见问题 5)

映了 ICP 模块在实际应用中可以节省用户的交互时间。

问题 4：初始交互点的质量如何影响最终的分割？ 我们对随机的初始点击建议进行了额外的实验。我们选择两种随机策略进行比较。一种是从整幅图像中选择随机点。另一种方法是，当推荐正确时，将该初始点推荐替换为该类别对象上的一个随机点。五个实验的平均值结果如表 6所示。我们发现，如果在全图或前景中选择随机的初始点，性能会下降。由于 ICP 模块主要用于定位这类物体，与整幅图像相比，从前景进行随机选择时，性能下降相对较小。

Method	Donut	Bench	Umbrella	Bed	Unseen 6k
CA [25]	6.50	13.30	10.20	5.00	9.30
Ours	5.65	12.56	9.63	4.56	9.18

表 8 在相同的五组 COCO 数据集上，比较我们的用于序列交互分割的解决方案与另一个基于序列的工作 [25] 之间的 NoC@85% 指标。

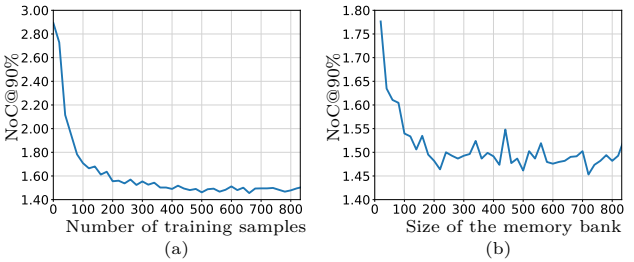


图 4 当增加训练数据 (a) 和在线训练内存大小 (b) 后的性能趋势 (见问题 6, 问题 7)

问题 5：为什么我们只优化提纯参数？ 表 7将优化后的提纯参数与全局参数进行比较。我们发现，与整个网络相比净化模块的参数量 (Params) 很小，甚至不到它的 2%。但是，性能差距并不显著。对于序列性交互式分割，需要为每个类别保存唯一的参数。这样小的参数量无疑是合适的。少量参数带来的优化速度 (秒/批) 更快，对任务也有帮助。

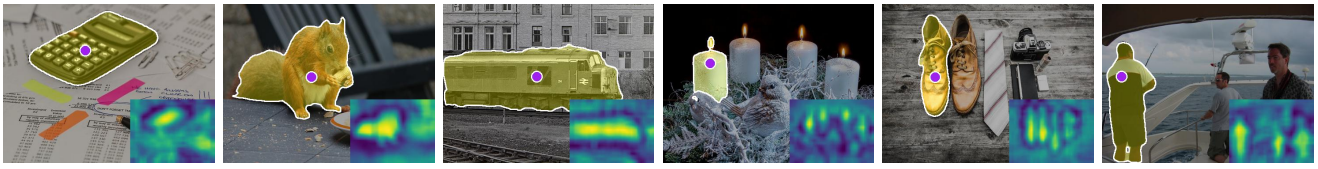
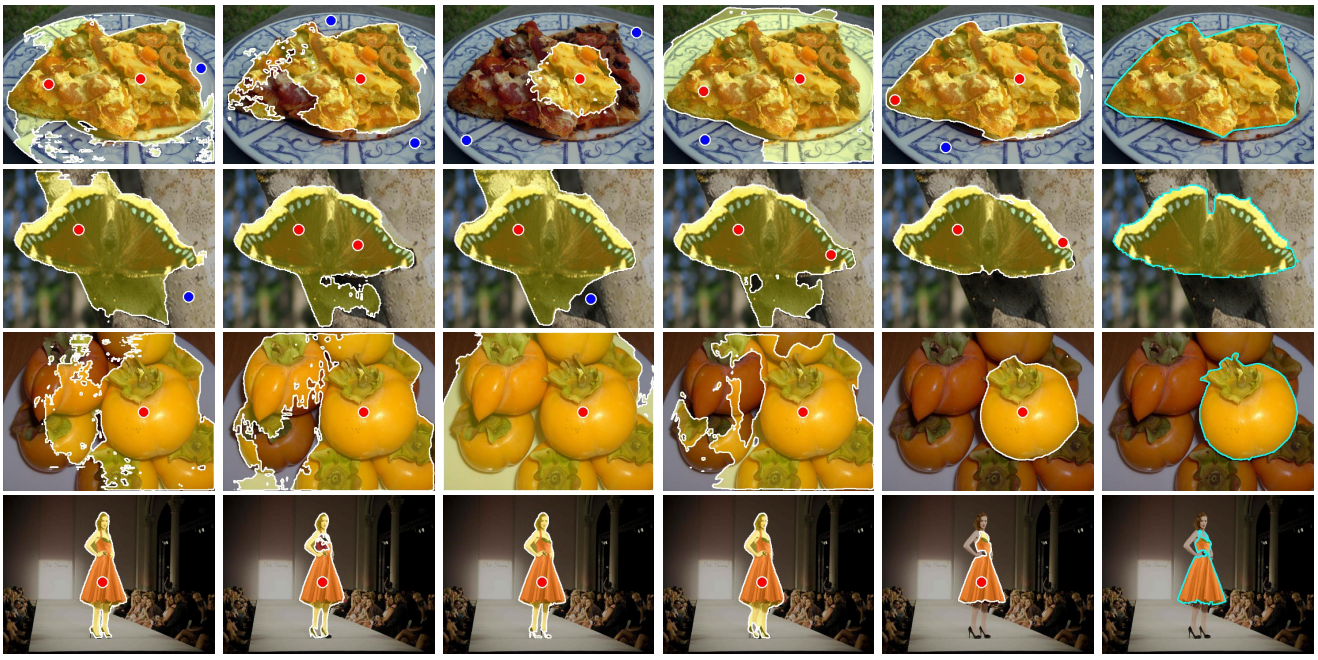


图 5 ICP 的定性结果，包括点击推荐、置信度图和预测结果。



LD[28] FCA[33] f-BRS[41] B B+OPO GT

图 6 其他方法和基准模型 (B) 以及加入 OPO 的定性结果比较。

问题 6：随着在线训练数据的增加，性能会提高吗? 图 4(a) 说明了 LeedsButterfly 在经过指定数量的样本 (横坐标) 后停止在线训练时的 NoC 度量指标的趋势。随着在线训练数据的增加，性能不断提高，后期达到稳定状态。

问题 7：该方法是否需要在在线学习期间访问整个训练数据集? 该方法不需要访问整个训练数据集。我们可以设置一个存储库，训练样本总是在这个存储库中选择。图 4(b) 说明了 LeedsButterfly 在采用不同大小的存储库时的 NoC 度量趋势。我们可以看到如果储存库的规模太小，会影响表现。然而，在特定的大小后，表现是稳定的，适合实际应用。

4.3 与最先进方法的比较

与单个 IIS 的比较 表 5显示了这些具有丰富类别的数据集中的 IIS 方法的 NoC 指标。由于个体性和序列性交互式分割的侧重点不同，我们提供性能只是为了进行直观的比较。这些方法都是精心设计的，并以 IIS 为重点。我们的方法主要解决了序列性交互式分割的问题，并对基本网络做了一些修改。它的性能可以媲美甚至超过这些先进方法。这反映出将交互分割看作是一个循序渐进的过程是有益的。

与序列 IIS 的比较如表 8中所示，我们还比较了最先进的与序列性交互分割有关的方法 [25]。该方法利用稀疏点击进行在线训练，我们的方法更进一步，可以取得更好的效果。因为我们的方法更关注语义对象，我们在实验中比较了大多数语义数据。通过更密集性的监督，我

们让模型对这类对象更敏感，这样才能表现得更好。

定性结果 图 5和图 6给出了我们提出方法的一些定性可视结果。从图 5中，我们可以发现初始的交互点击推荐恰恰位于不同类别物体的内部，比如动物和交通工具，可以减少用户的交互负担。图 6显示点击 1、2、3 后，基准模型和其他方法相比的分割结果。通过优化后的参数，无论是实例 (前三行) 还是部分的对象 (第四行)，在相同数量的交互点上，我们的方法都可以得到更准确的结果。这也有利于用户从具有多个实例的场景中分割目标 (第三行)。

5 结论

在这篇文章中, 我们提出了序列性交互式图像分割 (SIIS) 的任务。为了解决 SIIS 问题，我们从交互模式和后端算法的角度对其进行了系统的探索。具体来说，对于交互逻辑，我们设计了初始点推荐 (ICP)，它利用目标对象之前的语义特征来推荐一个初始交互点作为当前标注的实际输入。我们对算法逻辑提出了在线参数提纯优化 (OPO)，使用前面的精确标注将模型参数微调到目标类别。大量实验证明了序列性交互式图像分割的重要性和该方法的有效性。

References

[1] D. Acuna, H. Ling, A. Kar, and S. Fidler. Efficient interactive annotation of segmentation datasets with polygon-rnn++. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 859–868, 2018.

[2] J. Bai and X. Wu. Error-tolerant scribbles based interactive image segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 392–399, 2014.

[3] X. Bai and G. Sapiro. Geodesic matting: A framework for fast interactive image and video segmentation and matting. *Int. J. Comput. Vis.*, 82(2):113–132, 2009.

[4] R. Benenson, S. Popov, and V. Ferrari. Large-scale interactive object segmentation with human annotators. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11700–11709, 2019.

[5] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE T. Pattern Anal. Mach. Intell.*, 26(9):1124–1137, 2004.

[6] Y. Y. Boykov and M.-P. Jolly. Interactive grrrraph cuts for optimal boundary & region segmentation of objects in nd images. In *Int. Conf. Comput. Vis.*, pages 105–112, 2001.

[7] L. Castrejon, K. Kundu, R. Urtasun, and S. Fidler. Annotating object instances with a polygon-rnn. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5230–5238, 2017.

[8] F. Cermelli, M. Mancini, S. R. Bulo, E. Ricci, and B. Caputo. Modeling the background for incremental learning in semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9233–9242, 2020.

[9] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Eur. Conf. Comput. Vis.*, pages 801–818, 2018.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 248–255, 2009.

[11] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.*, 88(2):303–338, 2010.

[12] D.-P. Fan, T. Li, Z. Lin, G.-P. Ji, D. Zhang, M.-M. Cheng, H. Fu, and J. Shen. Re-thinking co-salient object detection. *IEEE T. Pattern Anal. Mach. Intell.*, 2021.

[13] D.-P. Fan, Z. Lin, G.-P. Ji, D. Zhang, H. Fu, and M.-M. Cheng. Taking a deeper look at co-salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2919–2929, 2020.

[14] L. Grady. Random walks for image segmentation. *IEEE T. Pattern Anal. Mach. Intell.*, 28(11):1768–1783, 2006.

[15] V. Gulshan, C. Rother, A. Criminisi, A. Blake, and A. Zisserman. Geodesic star convexity for interactive image segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3129–3136, 2010.

[16] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik. Semantic contours from inverse detectors. In *Int. Conf. Comput. Vis.*, pages 991–998, 2011.

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 770–778, 2016.

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

[19] S. D. Jain and K. Grauman. Click carving: Interactive object segmentation in images and videos with point clicks. *Int. J. Comput. Vis.*, 127(9):1321–1344, 2019.

[20] W.-D. Jang and C.-S. Kim. Interactive image segmentation via backpropagating refinement scheme. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5297–5306, 2019.

[21] M. Jia, M. Shi, M. Sirotenko, Y. Cui, C. Cardie, B. Hariharan, H. Adam, and S. Belongie. Fashionpedia: Ontology, segmentation, and an attribute localization dataset. In *Eur. Conf. Comput. Vis.*, pages 316–332, 2020.

[22] M. Jian and C. Jung. Interactive image segmentation using adaptive constraint propagation. *IEEE T. Image Process.*, 25(3):1301–1311, 2016.

[23] T. H. Kim, K. M. Lee, and S. U. Lee. Generative image segmentation using random walks with restart. In *Eur. Conf. Comput. Vis.*, pages 264–275, 2008.

[24] T. H. Kim, K. M. Lee, and S. U. Lee. Nonparametric higher-order learning for interactive segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3201–3208, 2010.

[25] T. Kontogianni, M. Gygli, J. Uijlings, and V. Ferrari. Continuous adaptation for interactive object segmentation by learning from corrections. In *Eur. Conf. Comput. Vis.*, pages 579–596, 2020.

[26] H. Le, L. Mai, B. Price, S. Cohen, H. Jin, and F. Liu. Interactive boundary prediction for object selection. In *Eur. Conf. Comput. Vis.*, pages 18–33, 2018.

[27] Y. Li, J. Sun, C.-K. Tang, and H.-Y. Shum. Lazy snapping. *ACM T. Graph.*, 23(3):303–308, 2004.

[28] Z. Li, Q. Chen, and V. Koltun. Interactive image segmentation with latent diversity. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 577–585, 2018.

[29] J. Liew, Y. Wei, W. Xiong, S.-H. Ong, and J. Feng. Regional interactive image segmentation networks. In *Int. Conf. Comput. Vis.*, pages 2746–2754, 2017.

[30] J. H. Liew, S. Cohen, B. Price, L. Mai, S.-H. Ong, and J. Feng. Multiseg: Semantically meaningful, scale-diverse segmentations from minimal user input. In *Int. Conf. Comput. Vis.*, pages 662–670, 2019.

[31] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*, pages 740–755, 2014.

[32] Z. Lin, Z.-P. Duan, Z. Zhang, C.-L. Guo, and M.-M. Cheng. Focuscut: Diving into a focus view in interactive segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022.

[33] Z. Lin, Z. Zhang, L.-Z. Chen, M.-M. Cheng, and S.-P. Lu. Interactive image segmentation with first click attention. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 13339–13348, 2020.

[34] H. Ling, J. Gao, A. Kar, W. Chen, and S. Fidler. Fast interactive object annotation with curve-gcn. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5257–5266, 2019.

[35] S. Mahadevan, P. Voigtlaender, and B. Leibe.

Iteratively trained interactive segmentation. In *Brit. Mach. Vis. Conf.*, 2018.

[36] S. Majumder, A. Rai, A. Khurana, and A. Yao. Two-in-one refinement for interactive segmentation. In *Brit. Mach. Vis. Conf.*, 2020.

[37] S. Majumder and A. Yao. Content-aware multi-level guidance for interactive instance segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11602–11611, 2019.

[38] K.-K. Maninis, S. Caelles, J. Pont-Tuset, and L. Van Gool. Deep extreme cut: From extreme points to object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 616–625, 2018.

[39] E. N. Mortensen and W. A. Barrett. Intelligent scissors for image composition. In *Annual Conf. on Comput. Graph. and Intera. Tech.*, pages 191–198, 1995.

[40] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM T. Graph.*, 23(3):309–314, 2004.

[41] K. Sofiiuk, I. Petrov, O. Barinova, and A. Konushin. f-brs: Rethinking backpropagating refinement for interactive segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8623–8632, 2020.

[42] G. Song, H. Myeong, and K. Mu Lee. Seednet: Automatic seed generation with deep reinforcement learning for robust interactive segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1760–1768, 2018.

[43] B. Steiner, Z. DeVito, S. Chintala, S. Gross, A. Paszke, F. Massa, A. Lerer, G. Chanan, Z. Lin, E. Yang, et al. Pytorch: An imperative style, high-performance deep learning library. In *Adv. Neural Inform. Process. Syst.*, volume 32, pages 8024–8035, 2019.

[44] V. Vezhnevets and V. Konouchine. Growcut: Interactive multi-label nd image segmentation by cellular automata. *Proc. of Graph.*, 1(4):150–156, 2005.

[45] J. Wang, K. Markert, and M. Everingham. Learning models for object recognition from natural language descriptions. In *Brit. Mach. Vis. Conf.*, 2009.

[46] T. Wang, J. Yang, Z. Ji, and Q. Sun. Probabilistic diffusion for interactive image segmentation. *IEEE T. Image Process.*, 28(1):330–342, 2018.

[47] J. Wu, Y. Zhao, J.-Y. Zhu, S. Luo, and Z. Tu. Milcut: A sweeping line multiple instance learning paradigm for interactive image segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 256–263, 2014.

[48] N. Xu, B. Price, S. Cohen, J. Yang, and T. Huang. Deep grabcut for object selection. In *Brit. Mach. Vis. Conf.*, 2017.

[49] N. Xu, B. Price, S. Cohen, J. Yang, and T. S. Huang. Deep interactive object selection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 373–381, 2016.

[50] C.-B. Zhang, J. Xiao, X. Liu, Y. Chen, and M.-M. Cheng. Representation compensation networks for continual semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022.

[51] S. Zhang, J. H. Liew, Y. Wei, S. Wei, and Y. Zhao. Interactive object segmentation with inside-outside guidance. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12234–12244, 2020.

[52] Z. Zhang, W. Jin, J. Xu, and M.-M. Cheng. Gradient-induced co-saliency detection. In *Eur. Conf. Comput. Vis.*, pages 455–472, 2020.