OSFormer：单阶段伪装实例分割 Transformers

裴佳伦 [†1], 程天阳 [†2], 范登平 [*3], 唐赫[2], 陈传波[2], Luc Van Gool[3]

[1] 计算机科学与技术学院，华中科技大学，中国
[2] 软件学院，华中科技大学，中国
[3] 计算机视觉实验室，苏黎世联邦理工学院，瑞士

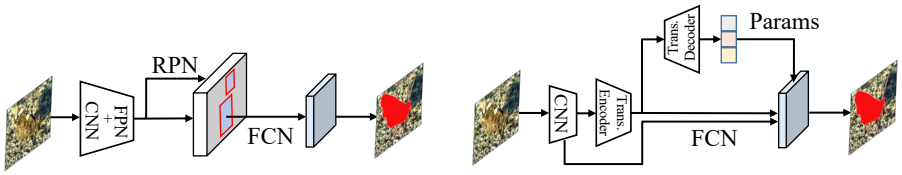
摘要 本文提出了 **OSFormer**，这是第一个用于伪装实例分割（Camouflaged Instance Segmentation, CIS）的单阶段 Transformer 模型。OSFormer 基于两个关键设计。首先，提出了一个**位置感知 Transformer**（Location-sensing Transformer, LST），通过位置引导 queries 和混合卷积的前馈网络来获得位置标签和实例感知参数。其次，设计了一个**由粗到细的融合器**（Coarse-to-fine fusion, CFF）用来合并来自 LST 编码器和 CNN 主干网络的上下文信息。耦合这两个模块可以使 OSFormer 有效地融合局部特征和长距离的上下文依赖，以预测出伪装实例。与两阶段框架相比，OSFormer 获得了 41% 的 AP，并且在不需要大量训练数据的情况下（3040 个样本，60 个 epochs）实现了良好的收敛效率。代码链接：https://github.com/PJLallen/OSFormer。

Keywords: 伪装，实例分割，Transformer

1 引言

伪装是一种强大而普遍避免被发现或被识别的手段，它源于生物学 [51]。在自然界中，伪装物体已经进化出一套隐蔽策略来欺骗猎物或捕食者的感知和认知机制，例如背景匹配、自我阴影隐蔽、湮没性阴影、破坏性着色和分散注意力的标记等 [11, 48]。这些防御行为使得伪装物体检测（COD）与一般的物体检测相比是一项非常具有挑战性的任务 [5, 32, 42, 44, 50]。COD 致力于区分与背景具有高度内在相似性的伪装物体 [16]。使用计算机视觉模型辅助人类视觉和感知系统进行 COD 是非常必要的，例如息肉分割 [17, 28]、肺部感染分割 [18]、野生动物保护和休闲艺术 [10] 等。

由于大规模和标准的基准建立，如 COD10K [16]、CAMO [31]、CAMO++ [30] 和 NC4K [37]，COD 的性能显著提升。然而，COD 只能在区域层面将伪装物体从场景中分离出来，而忽略了进一步的实例级识别。最近，Le 等



(a) Two-stage: Mask R-CNN (CNN+FPN) **(b) One-stage:** OSFormer (CNN+Transformer)

图 1: Mask R-CNN 和本文的 OSFormer 的框架比较。

人 [30] 提出了一个新的伪装实例分割（CIS）基准和一个伪装融合学习框架。在现实场景中，捕捉伪装实例可以提供更多的线索（比如，语义类别、物体的数量），因此 CIS 更具挑战性。

与一般的实例分割 [23] 相比，CIS 需要在特征相似度较高的、更复杂的场景下进行，并且获得的掩膜是类不可知的。此外，各种实例在场景中可能显示出不同的伪装策略，它们组合起来还可能形成交互伪装。这些衍生的群体伪装使 CIS 的任务更加艰巨。当人类注视一个高度伪装的场景时，视觉系统会本能地在整个场景中扫过一系列的局部范围来寻找有价值的线索 [38, 45]。受这种视觉机制的启发，本文提出了一种新颖的位置感知 CIS 方法，该方法在 *全局视角*中细致地捕捉所有位置的关键信息（即，*局部背景*），并直接生成伪装实例掩膜（即，*单阶段模型*）。

得益于 Transformer [52] 在视觉领域的兴起，研究者可以采用自注意力（self-attention）和交叉注意力（cross-attention）来捕捉长距离的依赖关系，并建立全局内容的感知交互 [5]。尽管 Transformer 模型在一些密集预测的任务上表现出高性能 [22,53,54,60]，但它需要大规模的训练数据和更长的训练时间。而作为一项全新的下游任务，目前只有有限的实例级训练标签可用。为此，本文提出了一个基于 [65] 的 **位置感知 Transformer**（Location-sensing Transformer, LST），以利用有限的训练样本来实现更快的收敛和更高的性能。为了动态地产生每个输入图像的位置引导 queries（Location-guided queries），本文将 LST 编码器输出的多尺度全局特征划分为一组包含不同局部信息的特征块。与原始 DETR [5] 中目标 query 的零初始化相比，本文的位置引导 queries 可以促进对特定位置特征的关注，并通过交叉注意力与全局特征进行交互，以获得实例感知嵌入。这种设计有效地加快了收敛速度，并显著提高了检测伪装实例的性能。为了增强局部感知和相邻 token 之间的相关性，将卷积操作引入到标准前馈网络中 [52]，命名为混合卷积前馈网络（Blend-convolution Feed-forward Network, BC-FFN）。因此，基于 LST 的

模型可以整合局部和全局的上下文信息，并有效地提供位置敏感的特征来分割伪装实例。

此外，本文还设计了一个 **由粗到细的融合模块**（Coarse-to-fine Fusion, CFF）来整合从 ResNet [24] 和 LST 中得到的多尺度低级和高级特征，以获得共享的掩膜特征。由于伪装实例的边缘难以捕捉，CFF 模块中嵌入了一个反向边缘注意力（Reverse edge attention, REA）模块，以提高对边缘特征的敏感性。最后，受 [25] 的启发，本文引入了动态伪装实例归一化（Dynamic Camouflaged Instance Normalization, DCIN）模块，通过结合高分辨率的掩码特征和实例感知嵌入来生成掩膜。基于上述两种新颖的设计（LST 和 CFF），本文提供了一个全新的单阶段框架 OSFormer 用于伪装实例分割（图1）。据本文所知，OSFormer 是第一个探索基于 Transformer 的 CIS 模型。本文的贡献如下：

1. 提出了 **OSFormer**，这是第一个为伪装实例分割任务设计的基于 Transformer 的单阶段模型。它的框架灵活，可用端到端的方式进行训练。
2. 提出了一个**位置感知 Transformer（LST）**，以动态地捕捉不同位置的实例线索。LST 包含一个带有混合卷积前馈网络（BC-FFN）的编码器来提取多尺度全局特征，以及一个带有位置引导 queries 的解码器来获得实例感知嵌入。LST 可在约 3,000 张图片的训练数据下快速收敛。
3. 提出了一个新的**由粗到细的融合模块（CFF）**，它通过融合主干网络和 LST 的多尺度低级和高级特征来获得高分辨率的掩码特征。在此模块中还嵌入了反向边缘注意力（REA）模块，来突出伪装实例的边缘信息。
4. 大量的实验表明，在具有挑战性的 CIS 任务中，OSFormer 表现良好，在 COD10K 测试集上相比 11 种主流的实例分割方法有很大的优势（*8.5% AP* 的提升）。

2 相关工作

伪装物体检测：该任务旨在识别混合在周围场景中的伪装物体 [20]。早期的研究主要采用低层手工对比度特征和一些启发式先验（比如，颜色 [27]、文本 [2, 47] 和运动边缘 [41]）来构建伪装物体检测（COD）模型。随着深度学习架构的流行和大规模像素级 COD 数据集的发布 [16, 31]，COD 的性能在过去两年得到了飞速提升。深度学习方法 [39, 43, 61, 64] 利用 CNN 提取高层次的信息特征来搜索和定位伪装目标，然后设计一个基于 FCN 的解码器来优化特征以预测伪装图。例如，Mei 等人 [39] 提出了一个定位聚焦网

络（PFNet）来模仿自然界中的动物捕食过程。PFNet 首先利用定位模块来定位潜在的目标，并使用聚焦模块来完善模糊区域。Zhai 等人 [63] 采用了交互图学习策略，对伪装物体的区域和边缘进行交互式训练。之后，Lyu 等人 [37] 提出了一个排序网络，该网络同时对伪装物体进行定位、分割和排序，以达到更好的预测效果。最近，Yang 等人 [62] 提出了一个新颖的基于不确定性引导的 Transformer 模型，旨在用贝叶斯学习推断不确定区域。尽管 COD 发展迅速，但该任务忽略了对实际应用场景至关重要的实例级预测图。因此，本文致力于将 COD 任务从区域层面推进到实例层面。

通用的实例分割：现有的工作可以大致归纳为自上而下和自下而上的模式。前一种模式表现为经典的‘检测-分割’框架，首先通过边界框检测出 ROI，然后在 ROI 中进行像素级实例分割 [49]。其中典型的模型是 Mask R-CNN [23]，它通过在 Faster R-CNN [44] 上扩展一个掩膜分支来预测实例级的掩膜。在此基础上，Mask Scoring R-CNN [26] 引入了一个 MaskIoU 头来评估实例掩膜的质量。为了增强特征金字塔并缩短信息流，PANet [35] 创建了一个自下而上的路径增强。此外，Chen 等人 [7] 提出了混合任务级联（HTC）来交互检测和分割特征进行联合处理。与上述两阶段模型不同，YOLACT [3] 是一个实时的单阶段框架，它包含了两个并行任务：产生非局部原型掩膜和预测一组掩膜系数。

与自上而下的模式相比，自下而上的方法首先学习实例感知的整体嵌入，然后通过聚类识别每个具体的实例 [8, 34]。Bai 等人 [1] 提出了一个源自经典分水岭变换的端到端边界感知的深度模型。SSAP [21] 可以通过实例感知的像素在关联金字塔的同时学习像素级语义类别和实例分割。然而，由于次优的像素分组，之前的自下而上模型的性能是不如自上而下模型的。因此，Tian 等人 [49] 提出了一个动态实例感知网络，在全卷积范式下直接输出实例掩膜。这种更简单的策略是高效的，并且比基于 Mask R-CNN 的框架表现更好。此外，SOLO [56, 57] 通过语义类别来检测实例的中心位置，并将掩膜预测解耦到动态卷积核特征学习中。受这一策略的启发，本文设计了一个基于 Transformer 的位置感知网络来动态感知伪装实例。

视觉 Transformer：Transformer [52] 诞生于自然语言处理，并已成功地扩展到计算机视觉领域 [15]。Transformer 编码器-解码器架构的核心思想是一种自注意力机制，它可以建立长距离的依赖关系，并从输入序列中捕捉全局的上下文信息。最近，Carion 等人 提出了 DETR [5]，它将 Transformer 与 CNN 主干网络相结合来聚合与目标相关的信息，并提供一组目标 queries 来

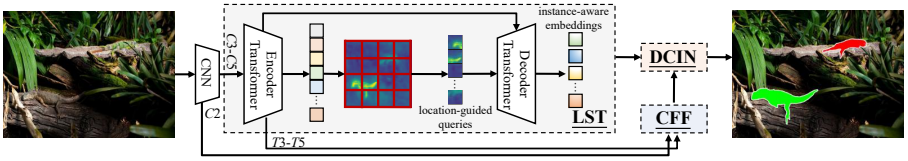


图 2: OSFormer 由位置感知 Transformer（LST）、由粗到细的融合模块（CFF）和动态伪装实例归一化（DCIN）模块组成。

输出最终的预测集合。尽管 DETR 开创了一个新颖且简洁的范式，但仍存在计算成本高和收敛速度慢的问题。针对这些问题，许多工作集中在如何设计一个更有效的 DETR 架构 [12,13,65]。Zhu 等人 [65] 引入了嵌入自注意力模块中的可变形注意力层，以降低计算成本和训练时间。UP-DETR [13] 利用一种新型的无监督 Pretest 任务来预训练 DETR 中的 Transformer 来加速收敛。然而，大多数现有的 Transformer 模型只适应于有大量训练数据的视觉任务。因此，对于只有小规模数据集的下游任务，充分利用 Transformer 的性能是一个迫切需要解决的问题。为此，本文提出了一个基于可变形 DETR [65] 的高效位置感知 Transformer（LST）用于 CIS 任务。本文的 Transformer 在 CIS 任务上仅用 3,040 个训练样本就能轻松收敛。

3 OSFormer

架构： OSFormer 包括四个重要组成部分。（1）一个 CNN 主干网络来提取物体的特征表示。（2）一个位置感知 Transformer（LST），它利用全局特征和位置引导 queries 来产生实例感知嵌入。（3）由粗到细的融合（CFF）模块，来整合多尺度的低级和高级特征，并产生高分辨率的掩膜特征。（4）动态伪装实例归一化（DCIN）模块，用于预测最终的实例掩膜。图2中阐述了整个架构。

3.1 CNN 主干网络

给定输入图像 $I \in \mathbb{R}^{H\times W\times 3}$，CNN 主干会生成多尺度特征 $\{Ci\}_{i=2}^{5}$（即，ResNet-50 [24]）。为了降低计算成本，直接将后三个特征图（$C3$、$C4$、$C5$）展平并串联成一个具有 256 个通道的序列 X_m，然后输入到本文的 LST 编码器中（§ 3.2）。对于特征 $C2$，本文将其作为高分辨率的低级特征输入到 CFF（§ 3.3）模块，以捕捉更多的伪装实例线索。

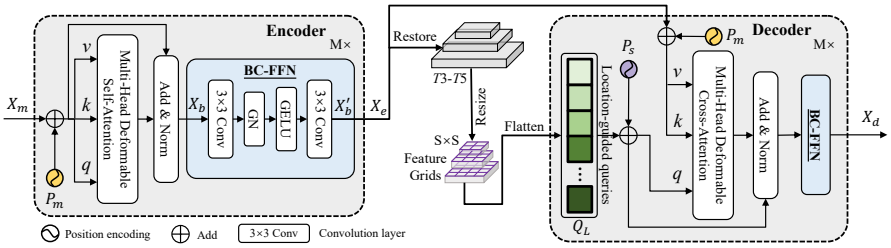


图 3: 位置感知 Transformer 的结构。

3.2 位置感知 Transformer

尽管 Transformer 可以通过自注意力层更好地提取全局信息，但它需要大规模的训练样本和高计算成本来支撑。由于 CIS 的数据规模有限，本文的目标式设计一个高效的架构，能够更快地收敛并达到有竞争力的性能。在图3中展示了位置感知 Transformer（LST）。

LST 编码器：与 DETR [5] 只有单一尺度的低分辨率特征输入到 Transformer 编码器不同，LST 编码器接收了多尺度特征 X_m 以获得更丰富的信息。在可变形自注意力 [65] 的基础上，为了更好地捕捉局部信息并增强相邻 token 之间的相关性，本文将卷积操作引入前馈网络，命名为混合卷积前馈网络（BC-FFN）。首先，根据 Ci 的形状，将特征向量恢复到空间维度。然后，使用卷积核为 3×3 的卷积层来学习归纳偏置。最后，添加一个组归一化 (GN) 和一个 $GELU$ 激活层来构成本文的前馈网络：BC-FFN。在一个 3×3 的卷积之后，将特征展平回序列。与 mix-FFN [60] 相比，BC-FFN 不包含 MLP 操作和残差连接。与 [59] 在每个阶段开始时嵌入卷积 token 并在 Transformer 块中采用深度可分离卷积不同，BC-FFN 中只添加了两个卷积层。具体来说，给定一个输入特征 X_b，$BC\text{-}FFN$ 的流程可以表述为：

$$X_b' = Conv^3(GELU(GN(Conv^3(X_b)))), \tag{1}$$

其中 $Conv^3$ 是一个 3×3 的卷积运算。总之，一个 LST 编码器层描述如下：

$$X_e = BC\text{-}FFN(LN((X_m + P_m) + MDAttn(X_m + P_m))), \tag{2}$$

其中 P_m 是位置编码。$MDAttn$ 和 LN 为多头可变形自注意力和层归一化。

位置引导 Queries：目标 query 在 Transformer 架构中起着至关重要的作用 [5]，它被用作解码器的初始输入，并通过解码器层获得输出嵌入。然而，query 的零初始化是原始 DETR 收敛缓慢的原因之一。为此，本文提出了另

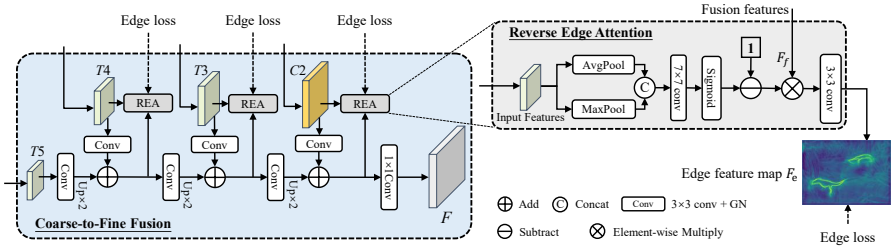


图 4: CFF 模块的结构。

一种 query 策略，它利用 LST 编码器的多尺度特征图 $Ti, i = 3, 4, 5$ 进行位置引导。[4] 值得注意的是，DETR 中的每个 query 都专注于特定区域。受 SOLO 模型的启发，首先将复原的特征图 $T3$-$T5$ 调整为 $S_i \times S_i \times D, i = 1, 2, 3$。然后，将调整后的特征划分为 $S_i \times S_i$ 个特征网格，并将其展平来产生位置引导 queries $Q \in \mathbb{R}^{L \times D}, L = \sum_{i=1}^{3} S_i^2$。在这种情况下，位置引导 queries 可以利用不同位置的可学习的局部特征来优化初始化过程，并有效地聚合伪装区域的特征。与零初始化或随机初始化 [5,65] 相比，这种 query 生成策略提高了 Transformer 解码器中 query 迭代的效率，并加速了收敛。更多的讨论，请参考 § 4.2。

LST 解码器： LST 解码器对于融合 LST 编码器产生的全局特征和位置引导 queries 产生实例感知嵌入是至关重要的。空间位置编码也被添加到位置引导 queries Q_L 和编码器特征 X_e 中。然后，可变形交叉注意力用来将它们融合。与一般的 Transformer 解码器不同，因为位置引导 queries 已经包含了可学习的全局特征，本文在解码器中直接使用交叉注意力而不使用自注意力。与 LST 编码器类似，BC-FFN 也是在可变形注意力操作之后使用的。给定位置引导 queries Q_L，LST 解码器的流程可以总结为：

$$X_d = \mathit{BC\text{-}FFN}(\mathit{LN}((Q_L + P_s) + \mathit{MDCAttn}((Q_L + P_s), (X_e + P_m)))), \quad (3)$$

其中 P_s 表示基于特征网格的位置编码。$\mathit{MDCAttn}$ 表示多头可变形交叉注意力操作。X_d 是用于实例感知表示的输出嵌入。最后，X_d 被恢复然后输入到下面的 DCIN 模块（§ 3.4）中来预测掩膜。

3.3 由粗到细融合模块（CFF）

作为一个自下而上的基于 Transformer 的模型，OSFormer 利用 LST 编码器输出的多级全局特征，从而产生一个共享的掩膜特征表示。为了合并不

[4] X_e 拆分并还原为 2D 表征 $T3 \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times D}$, $T4 \in \mathbb{R}^{\frac{H}{16} \times \frac{W}{16} \times D}$, and $T5 \in \mathbb{R}^{\frac{H}{32} \times \frac{W}{32} \times D}$。

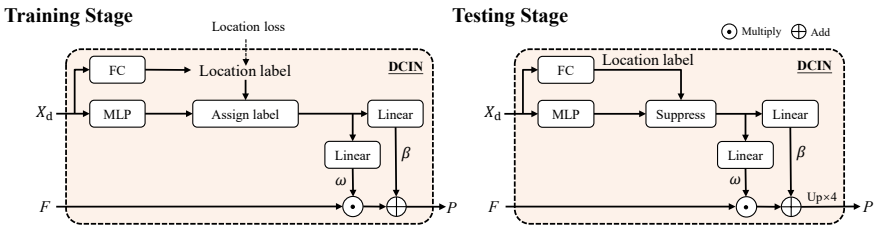


图 5: 动态伪装实例归一化（DCIN）的结构。

同的上下文信息，还融合了来自 CNN 主干网络的低级特征 $C2$ 作为补充，以产生一个统一的高分辨率特征图 $F \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times D}$。CFF 模块的详细结构见图4。这里将多级特征 $C2$、$T3$、$T4$ 和 $T5$ 作为级联融合的输入。从输入的 1/32 尺度的 $T5$ 开始，通过 3×3 卷积、GN 和 $2\times$ 的双线性上采样，并加入更高分辨率的特征（1/16 比例的 $T4$）。在以 1/4 的尺度融合 $C2$ 后，特征通过 1×1 的卷积、GN 和 RELU 操作，生成掩膜特征 F。值得注意的是，每个输入特征在第一次卷积后将通道数从 256 减少到 128，然后在最终输出时返回到 256 个通道。

考虑到伪装实例的边缘特征更难捕捉，还设计了一个反向边缘注意力（REA）模块嵌入到 CFF 中，用来在迭代过程中监督边缘特征。与之前的反向注意力 [9,17] 不同，REA 操作面向的是边缘特征而不是预测的二元掩码。此外，用于监督的边缘标签是通过膨胀侵蚀实例掩码标签获得的，不需要任何额外的人工标注。受卷积块注意力模块（CBAM）的启发 [58]，输入特征经过平均池化（AvgPool）和最大池化（MaxPool）操作。然后，把它们串联起来并通过一个 7×7 的卷积和一个 Sigmoid 激活函数。之后，反转注意权重，并通过元素相乘将它们与特征 F_f 进行融合。最后，使用 3×3 的卷积来预测边缘特征。假设输入特征为 Ti，每个 REA 模块的过程表述如下：

$$F_e = Conv^3(F_f \otimes (1 - Sigmoid(Conv^7([AvgPool(Ti); MaxPool(Ti)])))), \tag{4}$$

$Conv^7$ 为 7×7 的卷积层，$[;]$ 是通道层上的连接。总之，CFF 模块提供了一个共享的掩码特征 F，以输入到 DCIN 预测最终的每一个伪装实例掩膜。

3.4 动态伪装实例归一化

受风格转换领域中实例归一化操作的启发 [25,46]，模型引入了动态伪装实例归一化（DCIN）模块来预测最终的掩膜。当 DCIN 收到来自 LST 解码器的输出嵌入 $X_d \in \mathbb{R}^{S^2 \times D}$ 时，采用一个全连接层（FC）来获得位置标

签。同时，一个多层感知机（MLP）被用来获得大小为 D（即, 256）的实例感知参数。在训练阶段，根据真实标签分配正负位置。正位置的实例感知参数被用于生成分割掩码。在测试阶段，利用位置标签的置信度来过滤（见图5中的 Supress）无效的参数（比如, Threshold > 0.5）。随后，两个线性层对过滤后的位置感知参数进行操作，来获得仿射权重 $\omega \in \mathbb{R}^{N\times D}$ 和偏置 $\beta \in \mathbb{R}^{N\times 1}$。最后，它们与掩膜特征 $F \in \mathbb{R}^{\frac{H}{4}\times\frac{W}{4}\times D}$ 结合用于预测伪装实例，该过程可以描述为：

$$P = U_{\times 4}(\omega F + \beta), \tag{5}$$

其中 $P \in \mathbb{R}^{H\times W\times N}$ 是预测的掩膜。N 是预测的实例数量。最后，使用 Matrix NMS [57] 来确定最终的伪装实例。

3.5 损失函数

在训练过程中，总的损失函数可以写成：

$$L_{total} = \lambda_{edge}L_{edge} + \lambda_{loc}L_{loc} + \lambda_{mask}L_{mask}, \tag{6}$$

L_{edge} 用来监督 CFF 中不同层级特征的边缘损失。边缘损失定义为 $L_{edge} = \sum_{j=1}^{J} L_{dice}^{(j)}$，其中 J 代表用于监督的边缘特征的总级数，可以参考图4。λ_{edge} 是边缘损失的权重，默认设置为 1。由于 CIS 任务是不区分类别的，因此这里使用每个位置存在伪装实例的置信度（L_{loc}）来对比一般实例分割中的分类置信度。此外，L_{loc} 由 Focal loss [32] 实现，L_{mask} 由 Dice loss [40] 来计算分割损失。λ_{loc} 和 λ_{mask} 分别设置为 1 和 3 来平衡总损失。

4 实验

4.1 实验设置

数据集：作为一项全新的、具有挑战性的 CIS 任务，到目前为止还没有特定的数据集。令人欣慰的是，Fan 等人 贡献了一个 COD 数据集 [16]，即 COD10K，它同时提供了高质量的实例级注释可以用来训练 CIS 模型。具体来说，COD10K 包含 3,040 张带有实例级标签的伪装图像用于训练，2,026 张图像用于测试。最近，Le 等人 [30] 提供了一个更大的 CIS 数据集，称为 CAMO++，其中包括总共 5,500 个带有分层像素级注释的样本。此外，Lyu 等人 [37] 贡献了一个包含 4121 张图像的 CIS 测试集，称为 NC4K。本文使用 COD10K 中的实例级注释来训练 OSFormer，并在 COD10K 和 NC4K 测试集上对其进行评估。

评价指标： OSFormer 采用 COCO 形式的评价指标，包括 AP_{50}、AP_{75} 和 AP 指标 [33] 来评价分割结果。与实例分割中的 mAP 指标相比，从隐蔽区域检测到的每个伪装实例是类不可知的。因此，只需要考虑伪装实例的存在性而不需要考虑类别的平均值。

技术细节： OSFormer 是在单个 RTX 3090 GPU 上使用 PyTorch 实现的，模型使用随机梯度下降法（SGD）训练。为了进行公平的比较，模型采用 ResNet-50 主干网络，它通过 ImageNet 的预训练权重初始化的。如果没有特别说明，本文实验中使用的其他主干网络也是在 ImageNet 上预训练的。在训练过程，所有的模型都被迭代了 90K 次（60 个 epochs），Batch Size 为 2，初始学习率为 $2.5e-4$，Warm Up 在前 1K 次迭代中使用。随后，学习率在 60K 和 80K 时分别除以 10。此外，权重衰减被设置为 10^{-4}，动量为 0.9。输入图像的大小进行了调整，最短边的尺寸为 480 到 800，而最长边的尺寸最多为 1,333。模型还使用 scale jittering 技术进行了数据增强。在 LST 中，S_1、S_2 和 S_3 分别被设置为 36、24 和 16。请注意，在 BC-FFN 的整个过程中，特征的维度保持在 256。模型总共嵌入了六个编码层，并依次堆叠。为了达到更好的性能，只使用了三次 LST 解码层来聚合与 query 相关的伪装线索。

4.2 消融实验

本文在实例级 COD10K 数据 [16] 上进行一系列消融实验，验证 OSFormer 的有效性并确定超参数。实验主要包括以下几个部分：LST 中编码器和解码器的层数、多尺度特征输入数量、位置引导 queries 的设计、CFF 模块中的特征融合、主干网络结构、实时模型设置以及不同组件的贡献。

LST 中编码器和解码器的层数： Transformer 的深度是影响模型性能和效率的一个关键因素。该实验在 LST 中尝试了不同数量的编码器和解码器层的多种组合来优化 OSFormer 的性能。如表1的前三行所示，三层的 LST 不足以使 OSFormer 的性能最大化。此外，可以观察到 LST 对编码器比解码器更敏感。当编码器和解码器的层数分别为 6 层和 3 层时，AP 的值达到最高。当增加更多的层数时，精度没有进一步提高，并且推理时间下降到

表 1: LST 中不同数量的编码器和解码器的效果。

编码器	解码器	AP	AP_{50}	AP_{75}	FPS
1	3	37.0	68.0	35.4	**21.8**
3	1	39.2	69.1	38.5	20.0
3	3	39.4	70.2	39.3	18.8
3	6	38.9	68.6	37.9	17.2
6	3	**41.0**	**71.1**	40.8	14.5
6	6	40.6	70.3	**41.2**	13.4
9	6	40.7	70.6	40.4	11.3

表 2: 输入到 LST 的多尺度特征的不同组合的消融实验。

尺度	数量	AP	AP_{50}	AP_{75}	参数量	Memory
$C3$-$C5$	3	**41.0**	**71.1**	**40.8**	**46.58M**	**6.4G**
$C2$-$C5$	4	39.9	70.5	38.7	46.80M	9.3G
$C3$-$C6$	4	40.8	70.6	40.9	47.39M	6.7G
$C2$-$C6$	5	40.2	69.9	40.3	47.62M	17.7G

表 3: 不同的 query 设计在 OSFormer 上的比较。

Queries	AP	AP_{50}	AP_{75}
零初始化 [5]	34.7	64.1	33.1
可学习嵌入 [65]	35.0	64.8	33.2
位置引导 queries (**本文**)	**41.0** $_{+6.0}$	**71.1** $_{+6.3}$	**40.8** $_{+7.6}$

$14fps$ 以下。因此，模型采用 6 个编码器层和 3 个解码器层作为默认设置，以平衡模型的性能和效率。

多尺度特征输入的数量：模型利用从 ResNet-50 中提取的多层特征作为 LST 的输入。为了更准确地捕捉不同尺度的伪装特征，同时保持模型的效率，该实验从主干网络中组合了不同的特征，包括 $C3$-$C5$、$C2$-$C5$、$C3$-$C6$ 以及 $C2$-$C6$。在表2中，可以看到 $C3$-$C5$ 的组合以最低的参数量和训练内存实现了较强的性能。

位置引导 queries 的设计：在用于密集预测任务的 Transformer 架构中，目标 queries 是必不可少的。为了验证位置引导 queries 的有效性，本实验比较了两种典型的目标 query 设计，包括原始 DETR [5] 中的零初始化和可变形 DETR [65] 中的可学习输入嵌入。这里将 queries 的数量统一设置为多尺度特征网格的默认数量以进行公平的比较。OSFormer 中的其他设置保持不变。简而言之，Transformer 解码器中的目标 queries 包括两部分：query 特征和位置嵌入。在原始 DETR 中，由一组可学习的位置嵌入加上全零矩阵被当作目标 queries，然后通过解码器生成相应的输出嵌入。相比之下，可变形 DETR 直接由可学习的嵌入作为 query 特征进行初始化，并与可学习的位置嵌入相耦合。从表3中可以看出，位置引导 queries 明显优于其他 query 设计。这说明在 query 中加入有监督的全局特征对于回归不同的伪装线索和有效定位实例是至关重要的。此外，实验还比较了三种策略的训练效率。可以发现，位置引导 queries 在早期的训练阶段具有更快的收敛率，最终的收敛率也优于其他两种模型。这也证明了位置引导 queries 能够有效地利用全局特征，并通过交叉注意力捕捉不同位置的伪装信息。

CFF 中的特征融合：在本文的 CFF 模块中，多尺度输入特征直接影响到通过融合生成的掩膜特征 F 的质量。为了探究 ResNet-50 和 LST 编码器的最佳融合方案，表4中尝试了不同的组合。其中，只使用单尺度特征 $T2$ 而不

表 4: 输入到 CFF 模块的不同特征组合的比较。

特征	AP	AP_{50}	AP_{75}
Single $T2$	38.0	69.2	36.8
$C2, C3, C4, C5$	35.4	64.3	34.6
$C2, C3, C4, T5$	40.0	69.7	40.1
$C2, C3, T4, T5$	39.5	69.9	39.0
$T2, T3, T4, T5$	40.0	70.1	40.0
$C2, T3, T4, T5$	**41.0**	**71.1**	**40.8**

表 5: 不同主干网络下 OSFormer 的性能。

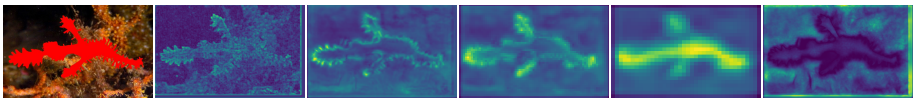
主干网络	AP	AP_{50}	AP_{75}	FPS
ResNet-50 [24] (**Default**)	41.0	71.1	40.8	**14.5**
ResNet-101 [24]	42.0	71.3	42.8	12.9
PVTv2-B2-Li [55]	47.2	74.9	**49.8**	13.2
Swin-T [36]	**47.7**	**78.6**	49.3	12.6

进行多尺度融合是不合适的。第 2^{nd} 行的结果说明，只融合主干网络特征的效率很低。最后，通过将 $C2$、$T3$、$T4$ 和 $T5$ 送入 CFF 模块达到了最佳结果。这可以解释为 LST 编码器的特征具有更详细的全局信息。此外，$C2$ 的特征也提供了一些低级特征作为补充。此外，图6中对输入到 CFF 模块的每个尺度的特征和掩膜特征 F 进行了可视化。

主干网络：在该实验中，使用不同的主干网络 ResNet-50 [24]、ResNet-101 [24]、PVTv2-B2-Li [55] 和 Swin-T [36] 来训练本文的模型。它们都是在 ImageNet [14] 上预训练的。从表5中，可以看到 OSFormer 只用 ResNet-50 就可以达到 41% 的 AP。此外，使用更强大的主干网络可以进一步激发该模型的潜力，使结果提高到 47.7%AP。

实时模型设置：为了提高 OSFormer 的应用价值，本文还提供了一个名为 OSFormer-550 的实时版本。具体来说，将输入的短边大小调整为 550，同时将 LST 编码器层数减少到 3。如表6所示，尽管 AP 下降到了 36.0%，但推理时间增加到了 25.8fps，参数和 FLOPs 也得到了显著改善。本文希望 OSFormer-550 能够扩展到更多的实际应用场景。

不同组件的贡献：本实验在 COD10K 测试集 [16] 上进行了充分的消融研究，包括 LST 编码器、位置引导 queries（LGQ）、BC-FFN、CFF 模块和反向边缘注意力（REA）模块。这里采用控制变量的方式，即只去掉当前模块而



(a) Image (b) $C2$ (c) $T3$ (d) $T4$ (e) $T5$ (f) F

图 6: 特征可视化。(a) 图像上覆盖真实掩膜；(b) - (e) 是由 CNN 主干网络和 LST 编码器产生的 CFF 的输入特征；(f) 是 CFF 输出的掩码特征 F。

表 6: 实时设置的 OSFormer 的性能。

模型	主干网络	AP	AP_{50}	FPS	参数量	FLOPs
OSFormer (**Default**)	ResNet-50	**41.0**	**71.1**	14.5	46.6M	324.7G
OSFormer-550	ResNet-50	36.0	65.3	**25.8**	**42.4M**	**138.7G**

表 7: OSFormer 中不同组件的消融研究。

编码器	LGQ	BC-FFN	CFF	REA	AP	AP_{50}	AP_{75}
	✓	✓	✓	✓	33.7	63.4	32.0
✓		✓	✓	✓	34.7	64.1	33.1
✓	✓		✓	✓	37.2	67.3	35.8
✓	✓	✓			38.0	69.2	36.8
✓	✓	✓	✓		39.3	69.7	38.5
✓	✓	✓	✓	✓	**41.0**	**71.1**	**40.8**

表 8: 与 11 种代表性方法的定量比较。

	方法	主干网络	参数量	FLOPs	COD10K-Test			NC4K-Test		
					AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
两阶段	Mask R-CNN [23]	ResNet-50	43.9M	186.3G	25.0	55.5	20.4	27.7	58.6	22.7
	Mask R-CNN [23]	ResNet-101	62.9M	254.5G	28.7	60.1	25.7	36.1	68.9	33.5
	MS R-CNN [26]	ResNet-50	60.0M	198.5G	30.1	57.2	28.7	31.0	58.7	29.4
	MS R-CNN [26]	ResNet-101	79.0M	251.1G	33.3	61.0	32.9	35.7	63.4	34.7
	Cascade R-CNN [4]	ResNet-50	71.7M	334.1G	25.3	56.1	21.3	29.5	60.8	24.8
	Cascade R-CNN [4]	ResNet-101	90.7M	386.7G	29.5	61.0	25.9	34.6	66.3	31.5
	HTC [7]	ResNet-50	76.9M	331.7G	28.1	56.3	25.1	29.8	59.0	26.6
	HTC [7]	ResNet-101	95.9M	384.3G	30.9	61.0	28.7	34.2	64.5	31.6
	BlendMask [6]	ResNet-50	35.8M	233.8G	28.2	56.4	25.2	27.7M	56.7	24.2
	BlendMask [6]	ResNet-101	54.7M	302.8G	31.2	60.0	28.9	31.4	61.2	28.8
	Mask Transfiner [29]	ResNet-50	44.3M	**185.1**G	28.7	56.3	26.4	29.4	56.7	27.2
	Mask Transfiner [29]	ResNet-101	63.3M	253.7G	31.2	60.7	29.8	34.0	63.1	32.6
单阶段	YOLACT [3]	ResNet-50	-	-	24.3	53.3	19.7	32.1	65.3	27.9
	YOLACT [3]	ResNet-101	-	-	29.0	60.1	25.3	37.8	70.6	35.6
	CondInst [49]	ResNet-50	**34.1**M	200.1G	30.6	63.6	26.1	33.4	67.4	29.4
	CondInst [49]	ResNet-101	53.1M	269.1G	34.3	67.9	31.6	38.0	71.1	35.6
	QueryInst [19]	ResNet-50	-	-	28.5	60.1	23.1	33.0	66.7	29.4
	QueryInst [19]	ResNet-101	-	-	32.5	65.1	28.6	38.7	72.1	37.6
	SOTR [22]	ResNet-50	63.1M	476.7G	27.9	58.7	24.1	29.3	61.0	25.6
	SOTR [22]	ResNet-101	82.1M	549.6G	32.0	63.6	29.2	34.3	65.7	32.4
	SOLOv2 [57]	ResNet-50	46.2M	318.7G	32.5	63.2	29.9	34.4	65.9	31.9
	SOLOv2 [57]	ResNet-101	65.1M	394.6G	35.2	65.7	33.4	37.8	69.2	36.1
	OSFormer (本文)	ResNet-50	46.6M	324.7G	41.0	71.1	40.8	42.5	72.5	42.3
	OSFormer (本文)	ResNet-101	65.5M	398.2G	**42.0**	**71.3**	**42.8**	**44.4**	**73.7**	**45.1**

将其他部分保持默认设置。在验证 LGQ 时，使用可学习嵌入 [65] 进行代替。同样，BC-FFN 被原始的 FFN [52] 取代。对于 CFF 模块，直接使用单尺度特征 $T2$ 作为 CFF 的输出。正如 表7所示，如果没有编码器，AP 直接下降了约 7%。这表明 LST 编码器对于提取高层级的全局特征是必不可少的。此外，第 2^{nd} 行再次验证了 LGQ 设计的有效性。值得注意的是，BC-FFN 在 LST 的编码器和解码器中起着至关重要的作用，因为 3×3 卷积层可以加强自注意力中全局特征的局部关联性。此外，CFF 有效地融合了多尺度特征，并通过嵌入 REA 模块加强了伪装实例的边缘特征。通过组合所有模块，OSFormer 达到了最佳性能。

4.3 与最先进方法的对比

本文将 OSFormer 与多个著名的实例分割模型（即，两阶段和单阶段模型）在实例级的 COD10K [16] 数据集上重新训练进行比较。为了更公平的比较，统一采用官方代码来训练每个模型，并在 COD10K 和 NC4K [37] 测

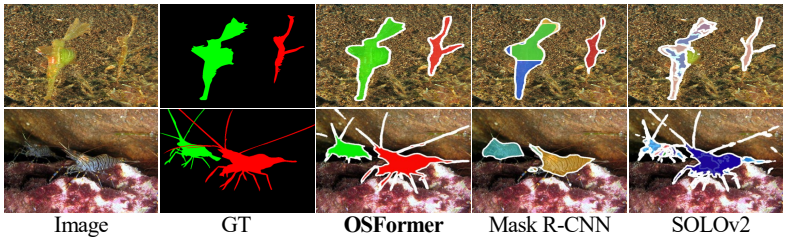


图 7: OSFormer 与 Mask R-CNN 和 SOLOv2 的定性比较。

试集上进行评估。此外，还展示了基于不同主干网络（ImageNet [14] 预训练的 ResNet-50 和 ResNet-101）的结果。

量化比较： 正如在表8中显示的那样，尽管 CIS 任务具有挑战性，但 OSFormer 与其他竞争对手相比在所有指标上仍然表现良好。特别是在使用 ResNet-50 时，OSFormer 的 AP 值比排名第二的 SOLOv2 [57] 要高得出约 8.5%。这个令人满意的结果应该归功于 LST，因为它提供了更高层次的全局特征，并与 LST 解码器中不同位置的伪装线索进行了交互。通过利用更强大的主干网络，即 Swin-T，OSFormer 可以继续将性能提升到 47.7%AP（表5）。表8中的参数和 FLOPs 也证明了 OSFormer 在不增加额外参数的情况下实现了更好的性能。

定性比较： 为了验证 OSFormer 的有效性，还在图7中展示了两个具有代表性的可视化结果。具体来说，顶部的样本说明了 OSFormer 可以在多个实例的情况下轻松地划分伪装实例。底部结果显示，本文的方法在捕捉细长边界方面表现出色，这可以归功于 REA 模块对边缘特征的增强。总的来说，与其他方法的可视化结果相比，OSFormer 有能力克服更具挑战性的情况并取得良好的性能。

5 结论

本文为伪装实例分割（CIS）贡献了一个全新的位置感知的单阶段 Transformer 模型，称为 **OSFormer**。OSFormer 包含了一个高效的位置感知 Transformer，以捕捉全局特征并动态回归伪装实例的位置和形状。作为第一个自下而上的单阶段 CIS 框架，模型还嵌入了一个由粗到细的融合模块，以整合多尺度特征并突出伪装的边缘，并输出全局特征。大量的实验结果表明，OSFormer 的性能优于目前其他知名的模型。此外，OSFormer 只需要训练大约 3,000 张图像，并且收敛迅速。它还可以被灵活地扩展到其他具有较少训练样本的下游视觉任务。

参考文献

1. Bai, M., Urtasun, R.: Deep watershed transform for instance segmentation. In: IEEE CVPR (2017)
2. Bhajantri, N.U., Nagabhushan, P.: Camouflage defect identification: a novel approach. In: IEEE ICIT (2006)
3. Bolya, D., Zhou, C., Xiao, F., Lee, Y.J.: Yolact: Real-time instance segmentation. In: IEEE CVPR (2019)
4. Cai, Z., Vasconcelos, N.: Cascade r-cnn: high quality object detection and instance segmentation. IEEE TPAMI **43**(5), 1483–1498 (2019)
5. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV (2020)
6. Chen, H., Sun, K., Tian, Z., Shen, C., Huang, Y., Yan, Y.: Blendmask: Top-down meets bottom-up for instance segmentation. In: IEEE CVPR (2020)
7. Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Shi, J., Ouyang, W., et al.: Hybrid task cascade for instance segmentation. In: IEEE CVPR (2019)
8. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE TPAMI **40**(4), 834–848 (2017)
9. Chen, S., Tan, X., Wang, B., Hu, X.: Reverse attention for salient object detection. In: ECCV (2018)
10. Chu, H.K., Hsu, W.H., Mitra, N.J., Cohen-Or, D., Wong, T.T., Lee, T.Y.: Camouflage images. ACM TOG **29**(4), 51–1 (2010)
11. Cuthill, I.: Camouflage. JOZ **308**(2), 75–92 (2019)
12. Dai, X., Chen, Y., Yang, J., Zhang, P., Yuan, L., Zhang, L.: Dynamic detr: End-to-end object detection with dynamic attention. In: IEEE CVPR (2021)
13. Dai, Z., Cai, B., Lin, Y., Chen, J.: Up-detr: Unsupervised pre-training for object detection with transformers. In: IEEE CVPR (2021)
14. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: IEEE CVPR (2009)
15. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
16. Fan, D.P., Ji, G.P., Sun, G., Cheng, M.M., Shen, J., Shao, L.: Camouflaged object detection. In: IEEE CVPR (2020)

17. Fan, D.P., Ji, G.P., Zhou, T., Chen, G., Fu, H., Shen, J., Shao, L.: Pranet: Parallel reverse attention network for polyp segmentation. In: MICCAI (2020)
18. Fan, D.P., Zhou, T., Ji, G.P., Zhou, Y., Chen, G., Fu, H., Shen, J., Shao, L.: Inf-net: Automatic covid-19 lung infection segmentation from ct images. IEEE TMI **39**(8), 2626–2637 (2020)
19. Fang, Y., Yang, S., Wang, X., Li, Y., Fang, C., Shan, Y., Feng, B., Liu, W.: Instances as queries. In: IEEE CVPR (2021)
20. Fennell, J.G., Talas, L., Baddeley, R.J., Cuthill, I.C., Scott-Samuel, N.E.: The camouflage machine: Optimizing protective coloration using deep learning with genetic algorithms. Evolution **75**(3), 614–624 (2021)
21. Gao, N., Shan, Y., Wang, Y., Zhao, X., Yu, Y., Yang, M., Huang, K.: Ssap: Single-shot instance segmentation with affinity pyramid. In: IEEE CVPR (2019)
22. Guo, R., Niu, D., Qu, L., Li, Z.: Sotr: Segmenting objects with transformers. In: IEEE ICCV (2021)
23. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: IEEE ICCV (2017)
24. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE CVPR (2016)
25. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: IEEE ICCV (2017)
26. Huang, Z., Huang, L., Gong, Y., Huang, C., Wang, X.: Mask scoring r-cnn. In: IEEE CVPR (2019)
27. Huerta, I., Rowe, D., Mozerov, M., Gonzàlez, J.: Improving background subtraction based on a casuistry of colour-motion segmentation problems. In: Iberian PRIA (2007)
28. Ji, G.P., Chou, Y.C., Fan, D.P., Chen, G., Fu, H., Jha, D., Shao, L.: Progressively normalized self-attention network for video polyp segmentation. In: MICCAI (2021)
29. Ke, L., Danelljan, M., Li, X., Tai, Y.W., Tang, C.K., Yu, F.: Mask transfiner for high-quality instance segmentation. In: IEEE CVPR (2022)
30. Le, T.N., Cao, Y., Nguyen, T.C., Le, M.Q., Nguyen, K.D., Do, T.T., Tran, M.T., Nguyen, T.V.: Camouflaged instance segmentation in-the-wild: Dataset, method, and benchmark suite. IEEE TIP **31**, 287–300 (2022)
31. Le, T.N., Nguyen, T.V., Nie, Z., Tran, M.T., Sugimoto, A.: Anabranch network for camouflaged object segmentation. CVIU **184**, 45–56 (2019)
32. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: IEEE ICCV (2017)

33. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)

34. Liu, S., Jia, J., Fidler, S., Urtasun, R.: Sgn: Sequential grouping networks for instance segmentation. In: IEEE ICCV (2017)

35. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: IEEE CVPR (2018)

36. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: IEEE CVPR (2021)

37. Lyu, Y., Zhang, J., Dai, Y., Li, A., Liu, B., Barnes, N., Fan, D.P.: Simultaneously localize, segment and rank the camouflaged objects. In: IEEE CVPR (2021)

38. Matthews, O., Liggins, E., Volonakis, T., Scott-Samuel, N., Baddeley, R., Cuthill, I.: Human visual search performance for camouflaged targets. Journal of Vision **15**(12), 1164–1164 (2015)

39. Mei, H., Ji, G.P., Wei, Z., Yang, X., Wei, X., Fan, D.P.: Camouflaged object segmentation with distraction mining. In: IEEE CVPR (2021)

40. Milletari, F., Navab, N., Ahmadi, S.A.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: IEEE 3DV (2016)

41. Mondal, A.: Camouflaged object detection and tracking: A survey. IJIG **20**(04), 2050028 (2020)

42. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: IEEE CVPR (2016)

43. Ren, J., Hu, X., Zhu, L., Xu, X., Xu, Y., Wang, W., Deng, Z., Heng, P.A.: Deep texture-aware features for camouflaged object detection. IEEE TCSVT (2021)

44. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: NeurIPS (2015)

45. Sandon, P.A.: Simulating visual attention. Journal of Cognitive Neuroscience **2**(3), 213–231 (1990)

46. Sofiiuk, K., Barinova, O., Konushin, A.: Adaptis: Adaptive instance selection network. In: IEEE CVPR (2019)

47. Song, L., Geng, W.: A new camouflage texture evaluation method based on wssim and nature image features. In: ICMT (2010)

48. Stevens, M., Merilaita, S.: Animal camouflage: current issues and new perspectives. PTRS B: BS **364**(1516), 423–427 (2009)

49. Tian, Z., Shen, C., Chen, H.: Conditional convolutions for instance segmentation. In: ECCV (2020)
50. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. In: IEEE ICCV (2019)
51. Troscianko, J., Nokelainen, O., Skelhorn, J., Stevens, M.: Variable crab camouflage patterns defeat search image formation. Communications biology **4**(1), 1–9 (2021)
52. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
53. Wang, H., Zhu, Y., Adam, H., Yuille, A., Chen, L.C.: Max-deeplab: End-to-end panoptic segmentation with mask transformers. In: IEEE CVPR (2021)
54. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: IEEE CVPR (2021)
55. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pvtv2: Improved baselines with pyramid vision transformer. CVMJ (2022)
56. Wang, X., Kong, T., Shen, C., Jiang, Y., Li, L.: Solo: Segmenting objects by locations. In: ECCV (2020)
57. Wang, X., Zhang, R., Kong, T., Li, L., Shen, C.: Solov2: Dynamic and fast instance segmentation. In: NeurIPS (2020)
58. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: Convolutional block attention module. In: ECCV (2018)
59. Wu, H., Xiao, B., Codella, N., Liu, M., Dai, X., Yuan, L., Zhang, L.: Cvt: Introducing convolutions to vision transformers. In: IEEE CVPR (2021)
60. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. In: NeurIPS (2021)
61. Yan, J., Le, T.N., Nguyen, K.D., Tran, M.T., Do, T.T., Nguyen, T.V.: Mirrornet: Bio-inspired camouflaged object segmentation. IEEE Access **9**, 43290–43300 (2021)
62. Yang, F., Zhai, Q., Li, X., Huang, R., Luo, A., Cheng, H., Fan, D.P.: Uncertainty-guided transformer reasoning for camouflaged object detection. In: IEEE CVPR (2021)
63. Zhai, Q., Li, X., Yang, F., Chen, C., Cheng, H., Fan, D.P.: Mutual graph learning for camouflaged object detection. In: IEEE CVPR (2021)
64. Zhu, J., Zhang, X., Zhang, S., Liu, J.: Inferring camouflaged objects by texture-aware interactive guidance network. In: AAAI (2021)

65. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. In: ICLR (2020)