

论文阅读

ST-SAM: SAM-Driven Self-Training Framework for Semi-Supervised Camouflaged Object Detection

Semi-Supervised Camouflaged Object Detection (SSCOD)：半监督伪装目标检测

Segment Anything Model (SAM)：分割一切大模型

一些概念

1. 师生 (Teacher-Student) 框架：深度学习中的一种经典模型架构，其核心思想是通过让一个较小、较简单的模型（学生模型）向一个较大、较复杂且性能优异的模型（教师模型）学习，来实现模型压缩和性能提升等目的。
2. 伪装特定数据 (camouflage-specific data)：其所有特征都围绕“支撑伪装系统实现‘目标 - 环境特征匹配’”展开，本质是“为伪装设计、优化、评估提供量化依据”，最终目标是降低目标在特定探测手段下的“可探测性”。无论是军事、安防（如反恐伪装）还是民用（如野生动物保护中的隐蔽观测设备伪装）领域，数据特征均需贴合具体场景的“环境属性”与“探测威胁”。
3. 局部前景概率 (the local foreground probability)：在处理图像或视频数据时，算法需要区分“前景”和“背景”。

前景 (Foreground)：指图像中需要重点关注、分析的目标（如行人、车辆、物体等）；

背景 (Background)：指图像中衬托前景的环境区域（如路面、天空、墙壁等）；

局部 (Local)：强调以图像的“局部区域”(而非全局) 为分析单位，通常是单个像素、像素块或小窗口；

概率 (Probability)：用 0~1 之间的数值表示“某局部区域属于前景的可能性”—— 数值越接近 1，该区域是前景的概率越高；越接近 0，则更可能是背景。

模型

现有的基于师生框架的SSCOD方法在监督信息稀缺的情况下存在严重的预测偏差和误差传播问题，且其多网络架构会导致较高的计算开销和有限的可扩展性，本文设计的ST-SAM框架，采用自训练策略，根本上规避了模型间的预测偏差。此外，论文的设计还利用SAM在特定任务中的潜力，以减轻自训练的误差累积。

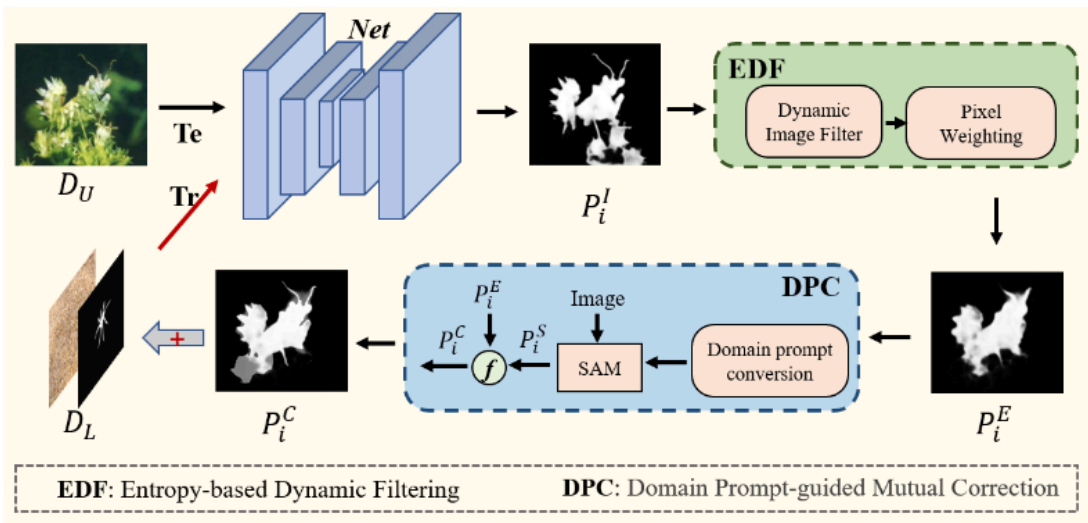
简单来说，该论文在自训练框架 (self-training) 的基础上，设计了EDF和DPC两个模块，DPC模块利用SAM强大的能力，增强了伪标签的可信度。同时，需要注意到，在论文中的伪装目标检测的任务中，

训练与微调的只有COD网络Net，SAM并不直接参与COD任务的训练循环。SAM的作用非常特定：利用Net生成的、带有领域知识的提示，来生成一个高质量的分割结果，这个结果被用来和Net的原始伪标签进行“相互校正”(mutual correction)，从而产生更可靠的伪标签。SAM是作为一个外部的、强大的分割引擎来使用的。

原文：

The COD network initializes the encoder parameters using ConvNeXt-B, while SAM adopts the parameters and settings of vit-h.

基本结构：



1. 基于熵的动态过滤策略（Entropy-based Dynamic Filtering strategy，EDF）

EDF 模块通过局部熵计算、全局熵评估、不确定性筛选、像素加权、动态扩展五个关键步骤，完成对初始伪标签的清洗与优化，具体流程如下：

（1）输入：初始伪标签的生成

首先，用少量标注数据（如 1%）预训练基础 COD（伪装目标检测）网络，再用该网络对无标签数据（ D_U ）进行预测，生成初始伪标签 P_i^I （ i 表示第 i 个无标签样本）。此时 P_i^I 因模型能力有限，包含大量噪声（如背景误判为目标、目标轮廓不完整）。

（2）步骤 1：计算局部与全局熵（量化不确定性）

熵（Entropy）是衡量概率分布不确定性的指标：熵值越高，预测结果越模糊（如像素既像目标又像背景）；熵值越低，预测越确定。EDF 通过“局部熵（ E_{local} ） + 全局熵（ E_{global} ）”双重维度评估伪标签的可靠性：

- 局部熵：评估像素级不确定性
- 全局熵：评估实例级不确定性

对初始伪标签 P_i^I 的每个像素，用 7×7 窗口计算局部前景概率，再推导局部熵，量化该像素邻域的预测模糊度：

先对 P_i^I 进行归一化（Norm），再通过均值滤波（UF）得到每个像素的局部前景概率 p_f ，背景概率 $p_b = 1 - p_f$ ：

$$p_f = UF \left(Norm \left(P_i^I \right), 7 \times 7 \right)$$

基于 p_f 和 p_b ，用信息熵公式计算每个像素的局部熵，熵值越高表示该区域预测越不确定：

$$E_{local} = -p_f \log(p_f) - p_b \log(p_b)$$

对整个伪标签 P_i^I 计算全局前景概率 \tilde{p}_f （所有像素归一化后的均值），再推导全局熵，量化整个样本的预测可信度：

$$\tilde{p}_f = \frac{1}{H \times W} \sum Norm(P_i^I)$$

$$E_{global} = -\tilde{p}_f \log(\tilde{p}_f) - (1 - \tilde{p}_f) \log(1 - \tilde{p}_f)$$

（ $H \times W$ 为伪标签的像素尺寸）

（3）步骤 2：实例级筛选（剔除高噪声样本）

基于局部熵与全局熵，定义不确定性指标 u_α ，筛选出可靠的伪标签样本：

不确定性指标计算：统计“局部熵 $> 0.5 \times$ 全局熵”的像素占比（ N 为总像素数， $\mathbb{I}(\cdot)$ 为指示函数），占比越高表示样本噪声越多：

$$u_\alpha = \frac{1}{N} \sum \mathbb{I}(E_{local} > E_{global} \times 0.5)$$

样本保留规则：设定阈值 $\tau_\alpha = 0.3$ ，仅保留 $u_\alpha < \tau_\alpha$ 的样本，直接剔除噪声严重的样本，避免早期训练引入大量误差：

Retain Sample If $u_\alpha < 0.3$, Else Discard

（4）步骤 3：像素级加权（抑制局部模糊区域）

对筛选后的样本，基于局部熵生成熵权重图，对不同可信度的像素分配不同权重，降低不确定区域对模型训练的干扰，最终得到熵加权伪标签 P_i^E ：

$$P_i^E = P_i^I \cdot (0.5 + 0.5 \times (1 - E_{local})^k)$$

权重系数设计： $k = 1$ （实验验证的最优值），局部熵 E_{local} 越低，权重越接近 1（可信区域充分贡献）； E_{local} 越高，权重越接近 0.5（模糊区域贡献减半）。

（5）动态扩展（适配模型渐进式学习）

考虑到模型训练初期能力有限，无法处理复杂样本，EDF 采用随 epoch 动态扩展的策略，逐步增加伪标签的规模，实现“从易到难”的学习：

- 一、计算所有熵加权伪标签 P_i^E 的局部熵均值 \bar{E}_{local} ，按均值从小到大排序（均值越小表示样本越简单、可信）；
- 二、初始训练阶段，仅选择排序靠前的少量低熵样本（如当前训练集规模 x 的等量样本）加入训练集；
- 三、随 epoch 增加，逐步扩大选择范围，纳入更多熵值稍高的样本（即更复杂的样本），直至所有无标签样本被充分利用。

2. 领域提示引导的相互校正策略（Domain Prompt-guided Mutual Correction strategy, DPC）

作用：注入领域知识到 SAM，生成高质量伪标签，与模型伪标签互校正。

DPC模块的核心目标是**解决SAM在伪装目标检测（COD）中的领域适配缺陷**——SAM虽经通用数据预训练具备强分割能力，但缺乏COD领域先验（如伪装目标与背景纹理高度相似、轮廓复杂），直接应用易误判；同时，自训练初期EDF模块输出的熵加权伪标签（ P_i^E ）仍存在边界模糊、小目标漏检等问题。DPC通过“领域提示转化+伪标签互校正”，既为SAM注入COD知识，又修正伪标签噪声，生成高置信监督信号。

DPC模块以EDF输出的熵加权伪标签 P_i^E 为输入，通过三步完成领域知识注入与伪标签优化：

（1）步骤1：生成混合提示（注入COD领域知识）

将 P_i^E 转化为SAM可理解的“点+框”混合提示（ $Prompt_{P-B}$ ），核心是注入COD领域三大先验知识：

- **框提示（ $Prompt_B$ ）：确保目标区域完整性**

过滤 P_i^E 中极小轮廓区域（排除噪声），保留有效目标轮廓 $Mask_i$ （ $i = 1, \dots, n$ ）；

提取每个 $Mask_i$ 的最小外接矩形，坐标表示为 $[x_{min}, y_{min}, x_{max}, y_{max}]$ ，公式如下：

$$Prompt_B = Mask_i [x_{min}, y_{min}, x_{max}, y_{max}]$$

该提示告诉SAM：“COD场景中伪装目标需完整覆盖，避免因局部相似遗漏目标区域”。

- **点提示（ $Prompt_P$ ）：精准定位目标中心**

计算每个 $Mask_i$ 的几何中心点 $c_i(x_i, y_i)$ ，验证是否在 $Mask_i$ 内部；

若 c_i 在外部，通过轴向搜索（ $AxialS(\cdot)$ ）沿 $Mask_i$ 长轴方向找到最近内部点 c'_i ；

最终点提示集为：

$$Prompt_P = \{c_i, IsInside(c_i) = 1\} \cup \{c'_i, IsInside(c_i) = 0\}$$

该提示告诉SAM：“COD场景中需以目标核心为锚点，避免被背景纹理干扰定位”。

- **混合提示融合：**将 $Prompt_B$ 与 $Prompt_P$ 结合为 $Prompt_{P-B}$ ，同时注入“区域完整性”与“中心定位”知识，适配COD复杂场景。

(2) 步骤2：SAM引导分割（生成 P_i^S ）

将混合提示（ $Prompt_{P-B}$ ）与无标签图像输入SAM，利用SAM的强分割能力生成**SAM伪标签** P_i^S ，公式如下：

$$P_i^S = SAM(Image, Prompt_{P-B})$$

此时 P_i^S 因融入COD领域知识，相比直接使用SAM的输出，在伪装目标边界完整性、小目标检出率上显著提升。

(3) 步骤3：伪标签互校正（生成 P_i^C ）

将 P_i^E 与 P_i^S 等比例融合，相互修正误差：

- P_i^E ：虽经EDF过滤，但可能存在局部边界模糊；
- P_i^S ：虽分割精度高，但可能因提示偏差导致部分区域误判；
- 融合后生成高置信伪标签 P_i^C ，公式如下：

$$P_i^C = fuse(P_i^E, P_i^S)$$

3. COD 网络和损失函数

ST-SAM 框架的核心优势之一是模型灵活性 —— COD 网络并非固定架构，可根据任务需求替换，文中实验采用基于经典 U-Net 改进的编码器 - 解码器结构（源自 Zhang 等人 2024 年的研究），该结构专为伪装目标检测设计，重点强化了多尺度特征融合与细节捕捉能力。

文中设计混合损失函数 \mathcal{L}_{total} ，公式定义为：

$$\mathcal{L}_{total} = \mathcal{L}_s + \alpha \cdot \mathcal{L}_{dice} + \beta \cdot \mathcal{L}_{UAL}$$

其中， $\alpha=4$ 、 $\beta=2$ 为权重系数，用于平衡不同损失组件的贡献。

\mathcal{L}_s 由加权二元交叉熵（wbce）与交并比损失（wiou）加权组成。

\mathcal{L}_{dice} 是基于 Dice 系数的改进版本，引入平滑项S（通常取 1e-6）避免分母为 0，公式为：

$$\mathcal{L}_{dice} = 1 - \frac{2 \cdot \sum (Pred^p \cdot Target^p) + S}{\sum Pred^p + \sum Target^p + S}$$

\mathcal{L}_{UAL} 是基于交叉熵的变体，引入动态权重系数 λ_{ual} （随学习率衰减而减小），公式为：

$$\mathcal{L}_{UAL} = \lambda_{ual} \cdot \sum (Pred \cdot \log(Pred) + (1 - Pred) \cdot \log(1 - Pred))$$

实验

对比实验

Table 1: Quantitative comparison between our method and other 11 SOTA methods on 4 benchmark datasets. '-' indicates the code or result is not available. The optimal and suboptimal results are represented in Red and Blue.

Methods	CAMO					CHAMELEON					COD10K					NC4K				
	$E_{\xi} \uparrow$	$S_{\alpha} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$M \downarrow$	$E_{\xi} \uparrow$	$S_{\alpha} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$M \downarrow$	$E_{\xi} \uparrow$	$S_{\alpha} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$M \downarrow$	$E_{\xi} \uparrow$	$S_{\alpha} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$M \downarrow$
Fully-Supervised Methods																				
DSAM [44] MM '24	0.906	0.832	0.794	0.824	0.061	-	-	-	-	-	0.913	0.846	0.760	0.761	0.033	0.928	0.871	0.826	0.845	0.040
TJNet [37] AI '24	0.890	0.841	0.779	-	0.064	0.958	0.913	0.859	-	0.024	0.907	0.844	0.738	-	0.030	-	-	-	-	-
ICEG [14] ICLR '24	0.879	0.810	-	0.789	0.068	0.950	0.899	-	0.858	0.027	0.906	0.826	-	0.747	0.030	0.908	0.849	-	0.814	0.044
DINet [50] TMM '24	0.883	0.821	-	0.790	0.068	-	-	-	-	-	0.901	0.832	-	0.744	0.031	0.910	0.856	-	0.820	0.043
Weakly-Supervised Methods																				
WS-SAM [13] NeurIPS '23	0.818	0.759	-	0.742	0.092	0.897	0.824	-	0.777	0.046	0.878	0.803	-	0.719	0.038	0.886	0.829	-	0.802	0.052
PSCOD [3] ECCV '24	0.872	0.798	0.727	-	0.074	-	-	-	-	-	0.859	0.784	0.650	-	0.042	0.889	0.822	0.748	-	0.051
CRNet [15] AAAI '23	0.815	0.735	0.641	-	0.092	0.897	0.818	0.744	-	0.046	0.832	0.733	0.576	-	0.049	-	-	-	-	-
ProMaC [19] NeurIPS '24	0.846	0.767	-	0.725	0.090	0.899	0.833	-	0.790	0.044	0.876	0.805	-	0.716	0.042	-	-	-	-	-
GenSAM [18] AAAI '24	0.775	0.719	-	0.659	0.113	0.807	0.764	-	0.680	0.090	0.838	0.775	-	0.681	0.067	-	-	-	-	-
Semi-Supervised Methods																				
CamoTeacher [25] -1%	0.669	0.621	0.456	0.545	0.136	0.714	0.652	0.476	0.558	0.093	0.788	0.699	0.517	0.582	0.062	0.779	0.718	0.599	0.675	0.090
CamoTeacher -5%	0.711	0.669	0.523	0.601	0.122	0.785	0.729	0.587	0.656	0.070	0.827	0.745	0.583	0.644	0.050	0.834	0.777	0.677	0.739	0.071
CamoTeacher -10% ECCV '24	0.742	0.701	0.560	0.635	0.112	0.813	0.756	0.617	0.684	0.065	0.836	0.759	0.594	0.652	0.049	0.842	0.791	0.687	0.746	0.068
SSCOD [11] -1%	0.804	0.708	0.583	0.653	0.110	0.771	0.683	0.574	0.629	0.063	0.805	0.725	0.537	0.578	0.057	0.844	0.767	0.652	0.700	0.073
SSCOD -10%	0.806	0.737	0.638	0.708	0.094	0.878	0.805	0.707	0.751	0.047	0.852	0.779	0.639	0.676	0.042	0.868	0.808	0.729	0.775	0.059
SSCOD -20% MM '24	0.844	0.778	0.704	0.767	0.078	0.906	0.834	0.761	0.802	0.042	0.864	0.791	0.662	0.699	0.039	0.882	0.821	0.750	0.795	0.055
Ours -1%	0.879	0.819	0.753	0.804	0.070	0.920	0.848	0.778	0.815	0.038	0.884	0.824	0.713	0.737	0.032	0.907	0.855	0.795	0.830	0.043
Ours -5%	0.886	0.831	0.761	0.807	0.066	0.921	0.855	0.776	0.798	0.036	0.871	0.834	0.709	0.716	0.031	0.909	0.870	0.803	0.823	0.038
Ours -10%	0.886	0.835	0.778	0.818	0.063	0.919	0.850	0.770	0.792	0.036	0.882	0.837	0.723	0.729	0.029	0.911	0.868	0.807	0.830	0.038
Ours -20%	0.890	0.844	0.779	0.809	0.058	0.926	0.876	0.804	0.818	0.032	0.874	0.837	0.713	0.717	0.030	0.911	0.874	0.807	0.824	0.037

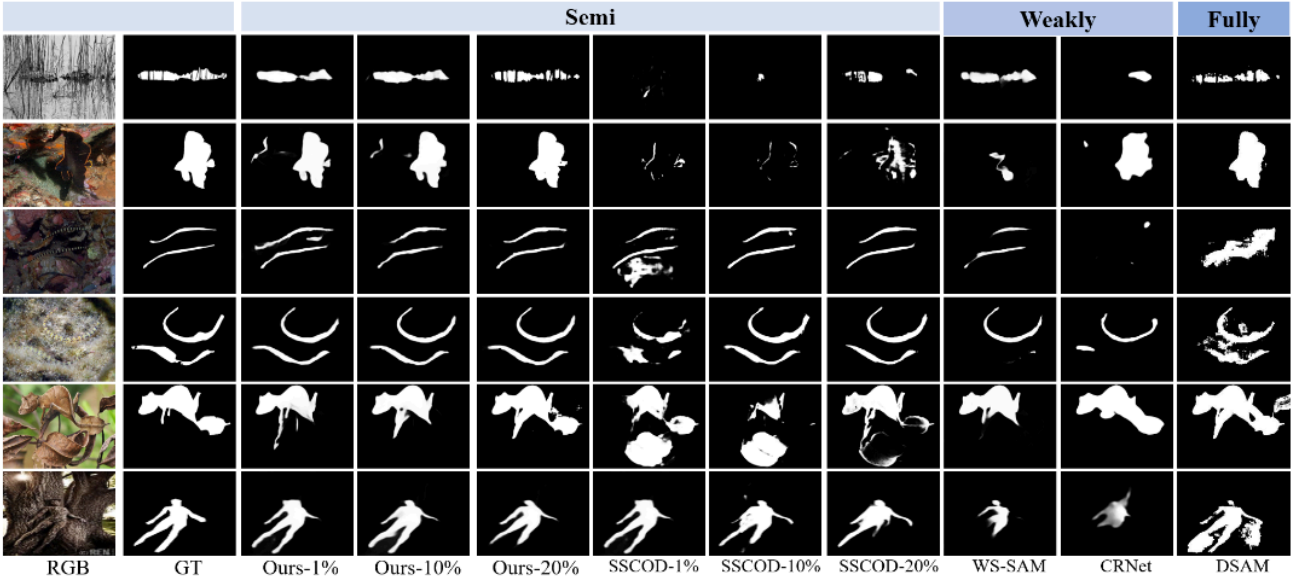


Figure 6: Qualitative comparison results between ST-SAM and SOTA COD models.

消融实验

模块的有效性：

- (1) baseline：只用少量有标签样本集 D_L
- (2) self-training: 经典的自训练策略，伪标签通过COD网络从无标签样本 D_U 获得，以直接扩展 D_L
- (3) 只加入EDF
- (4) 只加入DPC
- (5) EDF+DPC

Table 2: Ablation experiments on the effectiveness of each component, the best results are marked in Red.

ID	Variants	COD10K				NC4K			
		$E_{\xi} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$\mathcal{M} \downarrow$	$E_{\xi} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$\mathcal{M} \downarrow$
1	Baseline	0.793	0.517	0.569	0.054	0.834	0.611	0.693	0.077
2	Self-Training	0.807	0.535	0.585	0.057	0.830	0.628	0.688	0.078
3	+EDF	0.868	0.676	0.708	0.037	0.883	0.748	0.804	0.056
4	+DPC	0.856	0.655	0.684	0.038	0.879	0.740	0.778	0.052
5	+EDF+DPC	0.884	0.713	0.737	0.032	0.907	0.795	0.830	0.043

关于EDF的有效性：

EDF的动态扩展策略的有效性：

- (1) 一次性扩展所有合格样本
- (2) 每一步按同等比例（20%）扩展样本进行学习
- (3) 随着训练轮次动态扩展

关于样本学习顺序：

- (1) 先学习高熵样本
- (2) 随机选择
- (3) 先学习低熵样本

Table 3: Ablation experiments on the effectiveness of EDF, the results marked in **Red indicate the best performance.**

ID	Variants	COD10K				NC4K			
		$E_{\xi} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$\mathcal{M} \downarrow$	$E_{\xi} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$\mathcal{M} \downarrow$
1	One-shot	0.817	0.621	0.649	0.047	0.872	0.737	0.775	0.055
2	Equal Ratio	0.850	0.682	0.703	0.036	0.891	0.778	0.812	0.045
3	Epoch-Dynamic	0.884	0.713	0.737	0.032	0.907	0.795	0.830	0.043
4	H→L Entropy	0.788	0.525	0.590	0.059	0.813	0.612	0.685	0.078
5	Random Select	0.883	0.710	0.737	0.032	0.902	0.789	0.830	0.044
6	L→H Entropy	0.884	0.713	0.737	0.032	0.907	0.795	0.830	0.043

关于DPC的有效性:

关于不同提示策略:

- (1) 多点提示
- (2) 单框提示
- (3) 混合提示

关于融合方法:

- (1) 取交集
- (2) 取并集
- (3) 等比例融合

Table 4: Ablation experiments on the effectiveness of DPC, the results marked in **Red indicate the best performance.**

ID	Variants	COD10K				NC4K			
		$E_{\xi} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$\mathcal{M} \downarrow$	$E_{\xi} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$\mathcal{M} \downarrow$
1	Points	0.830	0.645	0.681	0.050	0.872	0.742	0.794	0.061
2	Box	0.876	0.701	0.731	0.035	0.905	0.795	0.827	0.044
3	P-B	0.884	0.713	0.737	0.032	0.907	0.795	0.830	0.043
4	Intersect	0.821	0.581	0.649	0.049	0.826	0.639	0.740	0.077
5	Union	0.800	0.622	0.626	0.048	0.871	0.752	0.762	0.049
6	Ratio	0.884	0.713	0.737	0.032	0.907	0.795	0.830	0.043

关于ST-SAM的可扩展性:

- (1) 用PRNet替代COD网络，PRNet是一种为全监督COD设计的轻量级网络。
- (2) 用BCE-IoU替代损失函数。

Table 5: Validation of the scalability of ST-SAM, the results marked in **Red indicate the best performance.**

ID	Variants	COD10K				NC4K			
		$E_{\xi} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$\mathcal{M} \downarrow$	$E_{\xi} \uparrow$	$F_{\beta}^{\omega} \uparrow$	$F_{\beta} \uparrow$	$\mathcal{M} \downarrow$
1	ST-SAM(Light)	0.852	0.653	0.694	0.040	0.884	0.741	0.796	0.051
2	ST-SAM(BCE-IoU)	0.864	0.698	0.717	0.034	0.904	0.798	0.826	0.040
3	SSCOD	0.805	0.537	0.578	0.057	0.844	0.652	0.700	0.073
4	ST-SAM	0.884	0.713	0.737	0.032	0.907	0.795	0.830	0.043