《大模型原理与技术》
课程大作业

题目：基于 SAM2 微调的显著性目标检测

学生：张超

学号：SX2516009

时间：2026 年 1 月 7 日

南京航空航天大学
计算机科学与技术学院/软件学院

选题理由

本人研究方向计算机视觉/图像处理，因此想采取使用视觉大模型来观察并解决当前研究领域的问题。这能够对自己未来工作提供非常好的帮助，而且也能够提升自己的代码水平，除此以外，SAM2 并非简单通过调库就可以实现的工具，通常需要对模型进行，更改，添加额外的模块，又或是删除不需要的结构，因此，本次实验个人认为是一次对自己非常有帮助的实验，不仅掌握了视觉中大模型的技巧，而且也更加了解了自己研究领域存在的一些问题。

显著性目标检测是一种基础的计算机任务，它模拟人类视觉的注意力机制，以像素级精度识别和分割图像中最具感知意义的目标。作为一种预注意力视觉任务，显著性目标检测抑制了背景杂乱和无关信息，因此有利于众多下游的计算机视觉应用，例如语义分割、目标识别和检测、以及目标跟踪。SAM2 作为强大的视觉大模型，具有“分割一切”的能力。

相关工作

采用多种不同大小和类型的数据集进行评估

采用随机旋转、随机反转、缩放进行数据预处理

采用适配器微调图像编码器、全量微调提示词编码器和解码器

采用掩码、矩形框、点作为提示词帮助定位

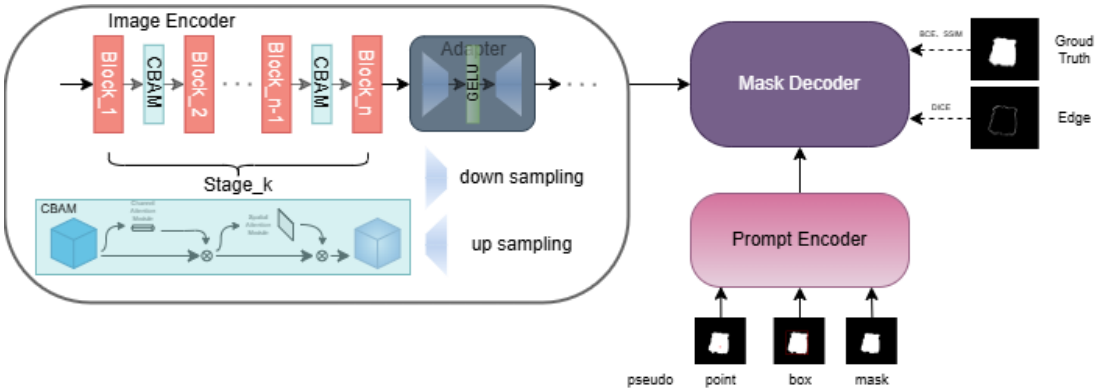
采用 CBAM 模型进行精细化特征图

采用 BCE、DICE、SSIM 作为损失进行监督

测试集评估采用 Fmax、WeightedF、MAE

模型架构

模型整体架构如图所示，整体架构分为三部分，分别是图像编码器（Image Encoder）、提示词编码器（Prompt Encoder）、掩码解码器（Mask Decoder）。



图像编码器负责从输入图像种提取多尺度高级语义特征。未来提升特征的判别能力，精细化特征图，在编码器内部的每一个 Block 后面引入 CBAM 卷积注意力模块，由于原来的 SAM2 图像编码器是用于分割任意图像，而非图像中的某一特定要素，因此通过自定义 Adapter 对编码器的能力进行微调。适配器结构通过采取下采样后降低分辨率，提升感受野，然后经过 GELU 激活函数后再通过上采样还原图像分辨率。

卷积注意力模块（Convolutional Block Attention Module, CBAM）位于每一个 Block 之后，其内部结构采取通道注意力和空间注意力，首先，特征图通过通道注意力（Channel Attention）自适应的关注“哪些特征通道更重要”，然后通过残差连接增强网络的表达能力。然后通过空间注意力（Spatial Attention）自适应的关注“特

征图中哪些空间位置更加关键”，同样的，这里也使用了残差结构。最终实现了编码器所能够提取到的语义特征质量，提升细粒度表现，即获得精细化的特征图。CBAM 代码如下所示：

```
class CBAMBlock(nn.Module):
    def __init__(self, channel=512, reduction=16, kernel_size=7):
        super().__init__()

        self.ca = ChannelAttention(channel, reduction)
        self.sa = SpatialAttention(kernel_size)

    def forward(self, x):
        residual = x
        out = x * self.ca(x)
        out = out * self.sa(out)
        return out + residual
```

采用适配器(Adapter)微调图像编码器，在编码器的每一个 Stage 之后添加一个轻量级的 Adapter，使用 GELU 激活。只要训练 Adapter 内部少量的参数就可以降低整体的微调成本并保留 SAM2 强大的分割能力，并适配于显著目标检测的任务中。适配器主要代码以及模型微调如下所示：

```
self.adapter_layer = nn.Sequential(
    nn.Linear(dim, hidden_dim),
    nn.GELU(),
    nn.Linear(hidden_dim, dim)
)
```

```python
def freeze_and_unfreeze(self):
    for p in self.model.parameters():
        p.requires_grad = False

    # Mask Decoder
    for p in self.model.sam_mask_decoder.parameters():
        p.requires_grad = True

    # Prompt Encoder
    for p in self.model.sam_prompt_encoder.parameters():
        p.requires_grad = True

    adapter_count = 0
    for name, p in self.model.named_parameters():
        if "adapter" in name:
            p.requires_grad = True
            adapter_count += 1

    if adapter_count == 0:
        print("[Warning] No parameters found with 'adapter' in name. Please check your Adapter implementation!")
    else:
        print(f"Unfrozen {adapter_count} adapter parameters.")

    # Custom Attention
    if hasattr(self.model, "my_attention"):
        print("Unfreezing cu
        #'''                (function) my_attention: Any
        for p in self.model.my_attention.parameters():
            p.requires_grad = True
            break
        #'''
    else:
        print("Warning: my_attention not found")
```

提示词编码器（Prompt Encoder）负责将各种辅助信息编码成统一的提示嵌入（Prompt Embedding），包括三种提示，分别是伪掩码、点提示和框提示，其中，伪掩码旨在伪造粗糙且不精确的预标注，通过将真实掩码下采样后进行腐蚀或膨胀来添加噪声，制造不精确的掩码；点提示通过在真实掩码中随机采样一个前景像素生成其坐标作为点提示；框提示利用前景区域的最小外接矩形作为提示框，并加入随机扰动以增强泛化性。提示词编码器的使用如下所示：

```python
sparse_embeddings, dense_embeddings = self.model.sam_prompt_encoder(
    points=(points, labels),
    boxes=box,
    masks=mask_inputs,
)
```

掩码编码器（Mask Decoder）将图像特征与提示特征进行融合，预测最终的目标掩码与边缘掩码，模型采用二元交叉熵（Binary Cross Entropy）监督主体区域，采用 DICE 损失监督边缘细节，在这

样的多任务监督下，提升了模型对于轮廓细节的刻画能力。掩码解码器的使用如下所示：

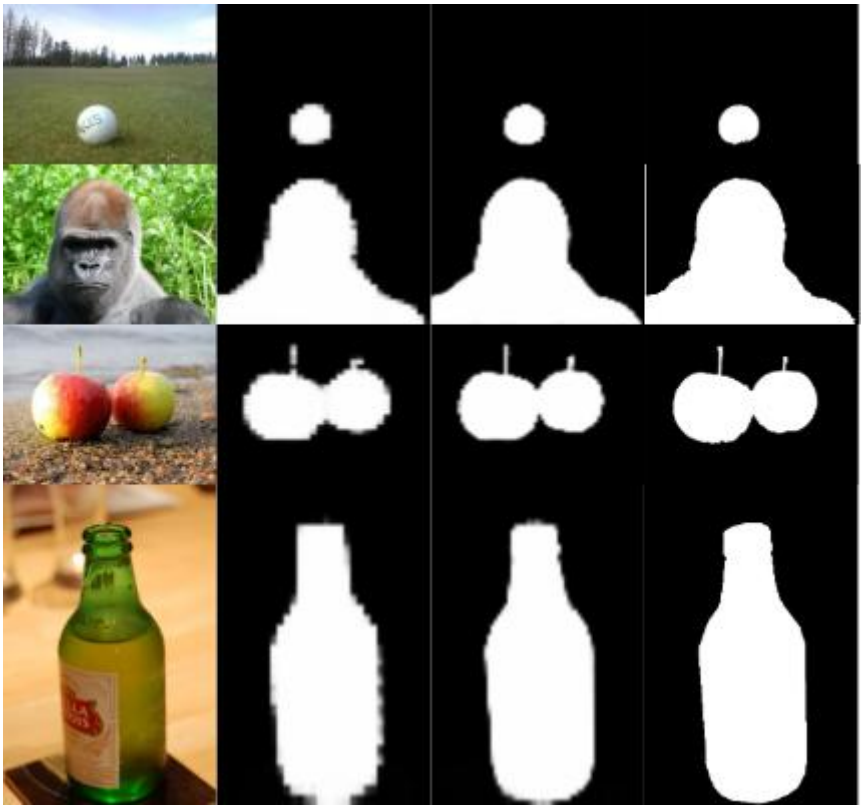
```
low_res_masks, _, _, _ = self.model.sam_mask_decoder(
    image_embeddings=image_embed,
    image_pe=image_pe,
    sparse_prompt_embeddings=sparse_embeddings,
    dense_prompt_embeddings=dense_embeddings,
    multimask_output=False,
    repeat_image=False,
    high_res_features=high_res_features,
)
```

相关工作对比

基于 SAM2 微调的方法其评估指标以及实验结果图如下所示：

可以发现评估值取得了较好的结果，以及对于一部分图像可以识别并分割

Dataset	DUTS-TE					ECSSD					HKU-IS					PASCAL-S				
Metric	Mae	MF	WF	Sm	Em	Mae	MF	WF	Sm	Em	Mae	MF	WF	Sm	Em	Mae	MF	WF	Sm	Em
SAM2	.028	.937	.862	.924	.938	.030	.970	.917	.943	.954	.029	.954	.899	.935	.953	.048	.924	.847	.899	.922



图表 1 从上至下分别是来自于 DUTS-TE、ECSSD、HKU-IS 以及 PASCAL-S 的图，从左到右分别是原始图像、epoch10、epoch20 以及 ground truth

总结

本次实验以 SAM2 作为基础框架，通过轻量级微调实现了一个基础的计算机视觉任务——目标检测任务。为了进一步提升模型对目标区域与边缘细节的表达能力，网络模型在图像编码器中引入了 CBAM 注意力模块，并结合自定义 Adapter 进行微调。此外，通过构建点、矩形框、伪掩码三类提示词生成策略，使模型能够在多种提示下拥有强大的分割性能。另外，由于实验没有训练很多轮 epoch，应该是达不到收敛的程度，所以关于评估结果，不会这么高，因此推测实际上存在些问题，代码中应该有地方写错了。对于结果图，实际上选取了较好的几张图片，同时也可以看出，图像具有很明显的毛边，其边缘完全

就不光滑，未来会进一步调整模型超参数，以及更改模型结构。