基于 CNN 和 Twin Transforemr 的实例分割

摘要

本研究论文介绍了一种名为 SOTR（Segmenting Objects with Transformers）的创新型、灵活而高效的基于 Transformer 的模型，专为高精度实例分割而设计。SOTR 通过采用基于残差网络（Resnet）的骨干网络模型，优化了分割过程，并行执行两个子任务：(1) 利用 Transformer 预测每个实例的类别；(2) 通过多级上采样（Multi-level Upsampling Module）模块，动态地生成分割掩膜。SOTR 能够从特征金字塔网络（FPN）中高效提取低层特征，这是其 CNN 骨干网络的一个重要组成部分，并通过 Twin Transformer 捕捉长距离的上下文依赖关系。与传统 Transformer 相比，SOTR 中引入的 Twin Transformer 结构只涉及到对像素的行和列的注意力，从而在时间和资源消耗上更具优势。最后，本文复现工作对 SOTR 模型进行了改进，将其骨干网络从 ResNet101 替换 ResNetXt101，增强模型对目标对象低层次特征的表达能力，并在 MS COCO 数据集的实验中得到验证。

关键词： 实例分割; 卷积神经网络; Twin Transformers; 多级上采样

1 引言

1.1 研究背景和挑战

实例分割，作为计算机视觉领域的一个关键任务，涉及到在像素级别上对图像中的每个对象进行精确识别和分割。它对于理解和解析复杂的视觉场景至关重要，尤其在自动驾驶、医疗图像分析等领域中具有广泛应用。然而，现有的实例分割方法大多基于卷积神经网络（CNN），遵循“先检测对象然后分割”的策略。这些方法在处理大型对象、对象遮挡、以及精确的像素级分割时面临着重大挑战。尤其是在复杂场景中，传统的 CNN 方法往往难以捕捉到足够的上下文信息，导致分割结果不理想。

1.2 研究动机和方法

针对现有方法的局限性，研究者们开始探索更为高效的实例分割策略。一种新兴的趋势是采用自底向上的方法，通过学习图像中每个像素的嵌入表示，然后基于这些表示将像素聚类成独立的对象实例。这种方法能够更好地处理对象的遮挡和边界问题。然而，它们在聚类过程中的不稳定性以及在不同场景下的泛化能力依然是一个挑战。此外，Transformer 架构在自然语言处理领域取得了巨大的成功，其在捕捉长距离依赖关系方面的能力激发了将其应用于视觉任务的兴趣。特别是在实例分割等高层次视觉任务中，Transformer 有望提供比 CNN 更好的性能。

1.3 SOTR 的创新和方法

本研究提出的 SOTR（Segmenting Objects with TRansformers）框架，是对现有实例分割方法的重大创新。SOTR 结合了 CNN 的强大的低层次特征提取能力和 Transformer 在处理长距离依赖关系方面的优势。通过特征金字塔网络（FPN）增强了特征的表达能力，并引入了一个双 Transformer 结构，这个结构在进行像素编码时，专注于行和列的注意力，从而提高了计算效率。这一独特的结构不仅降低了计算和资源消耗，还提高了模型在处理复杂场景和大型对象时的性能。

1.4 研究贡献

SOTR 的主要贡献在于：首先，提出了一个新型的 CNN-Transformer 混合框架，有效地结合了两者的优势，为实例分割提供了一个强大的新途径。其次，设计了一种创新的双注意力机制，显著提高了模型的效率和性能。此外，SOTR 展示了出色的泛化能力，能够在不同的数据集和场景下保持稳定的性能。在 MS COCO 数据集上的实验结果表明，SOTR 实现了超过 39% 的平均精度（AP），特别是在中等和大型对象的实例分割任务上，SOTR 的性能表现优于现有的最先进方法。这一成果不仅证明了 SOTR 在实例分割任务中的有效性，也为将来的研究提供了一个新的方向。此外，SOTR 的设计简化了实例分割的流程，使其在计算资源有限的环境下也能高效运行。最后，SOTR 在不需要大规模预训练的情况下表现出良好的泛化能力，这一点对于在数据受限的应用场景中尤为重要，为实例分割的研究和应用开辟了新的可能性。

2 相关工作

2.1 传统的实例分割方法

传统的实例分割方法主要基于卷积神经网络（CNN），通过在检测之后进行分割来处理问题。作为基于锚点 (anchor-based) 和两阶段方法的代表，Mask R-CNN [1] 在 Faster R-CNN [2] 的基础上增加了一个额外的分支，用于在提议的潜在边界框中进行实例分割。同样作为基于锚点的方法之一，YOLACT [3] 在单阶段中分割实例，但有两个并行的子任务：生成原型掩模和预测每个实例的掩模系数。最终的实例掩模是这两者的线性组合。另一方面，一些工作致力于在无锚点 (anchor-free) 框架内生成分割掩模。其中许多衍生自 FCOS [4]。例如，CenterMask [5] 向 FCOS 添加了一个新颖的空间注意力引导的掩模分支，用于预测每个检测到的框的分割掩模。

2.2 自底向上的实例分割方法

近年来，自底向上的实例分割方法受到了越来越多的关注。这些方法与基于 CNN 的“检测后分割”策略不同，通过将像素聚类到图像中的每个实例中来生成掩模。典型的方法包括 SSAP、SGN 等。SSAP 通过深度学习捕获每个像素的特征，并使用谱聚类来形成独立的对象实例。SGN [6] 通过三个子网络来解决实例聚类的问题。此外，最新的自下而上方法 [7,8] 更直接地进行了实例分割。SOLO 没有利用像素对之间的关系，而是通过分类来处理聚类问题。它对每个网格进行分类，并在不进行聚类的情况下，端到端地预测每个网格的掩模。当场景

非常复杂且一幅图像中存在密集的对象时，大量的计算和时间不可避免地会浪费在背景像素上。

2.3 Transformer 在视觉任务中的应用

受到 Transformer 在自然语言处理中巨大成功的启发，研究人员提出将 Transformer 应用于解决计算机视觉问题 [9–11]。遵循标准 Transformer 范式，Dosovitskiy 等人 [11] 提出了一个 Pure Transformer 模型，称为视觉 Transformer（ViT），在图像分类任务中实现了最先进的结果。为了使 ViT 的架构尽可能类似于原始 Transformer，输入图像被重塑成一系列扁平化的 2D 补丁，并通过可训练的线性投影和位置嵌入映射到相应的嵌入向量。这种 Pure Transformer 模型可以自然地泛化，通过添加基于 FCN 的掩模头来产生语义分割。Segmentation Transformer（SETR）[12] 中，该框架在 ViT 的基础上进行了最小化修改，并应用了一种渐进式上采样策略作为解码器来生成最终掩模。

然而，Transformer 在提取低级特征方面遇到了困难，并且缺乏一些归纳偏差，因此 Pure Transformer 模型过度依赖于大型数据集上的预训练。该问题可以通过与 CNN 主干的结合有效解决。Detection Transformer（DETR）[13] 由标准 CNN 主干和用于对象检测的编码器-解码器 Transformer 组成。前者学习输入图像的 2D 表示并生成较低分辨率的特征图。后者从上述平铺特征并带有位置信息并行预测 N 个对象（框坐标和类别标签）。然而，DETR 存在两个问题。由于在 Transformer 中的关系建模之前进行特征映射，DETR 不仅承受高计算成本，而且在小对象上表现不佳。此外，DETR 需要更长的训练时间表来调整注意力权重并关注有意义的稀疏位置。

对于实例分割，DETR 可以通过在解码器输出顶部附加一个掩模塔来扩展。与这些方法不同，本文以不同的方式重新思考实例分割，提出了一种结合 CNN 和 Transformer 的新颖实例分割方法，称为 SOTR。首先，SOTR 遵循标准 FCN 设计，并利用可学习的卷积来通过位置直接以无框方式分割每个对象区域。其次，我们采用双重注意力，这是一种替代性的自注意力自回归块，通过将全局空间注意力分解为独立的垂直和水平注意力，显著减少了计算和内存需求。

3 本文方法

3.1 本文方法概述

SOTR 是一种 CNN-Transformer 混合实例分割模型，它能够同时学习 2D 表示并轻松捕获远程信息。它遵循直接分割范式，首先将输入特征图划分为补丁，然后预测每个补丁的类别，同时动态地分割每个实例。具体来说，模型主要由三部分组成：1) 骨干网络 Backbone，用于从输入图像中提取图像特征，特别是低级别和局部特征；2) Transformer，用于建模全局和语义依赖关系，附加了功能性头部 Functional Head，分别预测每个补丁的类别和卷积核；以及 3) 多级上采样模块 multi-level upsampling module，通过在生成的特征图和相应的卷积核之间进行动态卷积操作来生成最终的分割掩模。整个框架在图 1中展示。

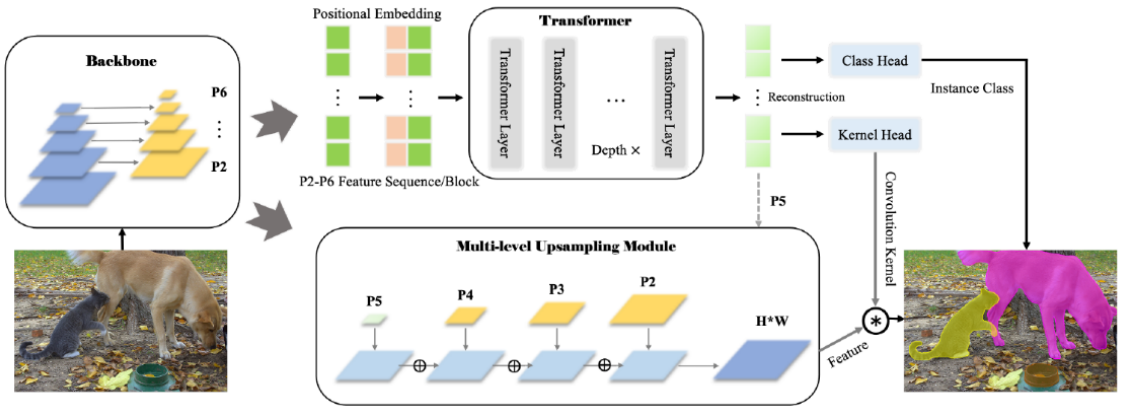

图 1. 模型框架图

3.2 Backbone

3.2.1 功能

Backbone 的主要功能是从输入图像中提取特征。这些特征包括但不限于图像的低层次特征（如边缘、纹理等）和局部特征。这些特征对于理解图像的基本组成和局部内容非常重要。在实例分割等视觉任务中，不同尺度的特征对于捕获不同大小对象的细节至关重要。背景提取器通过提取多尺度的特征，有助于模型更好地理解和分割图像中的不同尺寸的对象。

3.2.2 实现

FPN 通常建立在标准的卷积神经网络（如 ResNet）之上，通过融合网络不同层次的特征来构建一个特征金字塔。每一层的特征都具有不同的空间分辨率，使模型能够捕获从粗糙到细致的多层次信息。FPN 通过上采样和下采样的方式，结合不同深度的特征图。例如，更深层次的特征图（捕获更高层次的语义信息，但空间分辨率较低）会通过上采样与较浅层次（具有更高空间分辨率的细节信息）的特征图融合。

在 SOTR 模型中，FPN 作为背景提取器，为后续的 Transformer 模块和多层次上采样模块提供了丰富的、多尺度的特征图，这对于模型准确地进行实例分割至关重要。

3.3 Twin transformer

3.3.1 Twin Attention

Self-attention 机制是 Transformer 的关键组成部分，能够捕获全图的上下文并学习输入序列中每个元素之间的长距离交互。然而，在处理如图像这样的高维序列时，自我注意力具有二次方的时间和内存复杂度，导致更高的计算成本，限制了模型在不同设置下的可扩展性。

为了解决这些问题，本文提出了 Twin Attention 机制，通过稀疏表示简化注意力矩阵。该策略将感受野限制在固定步长的设计块模式中。首先计算每一列内的注意力，保持不同列的元素独立，然后在每一行内执行类似的注意力，以全面利用垂直尺度上的特征交互。这两个尺度上的注意力顺序连接，形成具有全局感受野的最终注意力，覆盖两个维度上的信息。Twin Attention 机制将内存和计算复杂度变化：

$$O((H \times W)^2) \rightarrow O(H \times W^2 + W \times H^2) \tag{1}$$

3.3.2 Transformer Layer

Transformer Layer 基于编码器作为基础构建块。原始的 Transformer 层类似于在自然语言处理中使用的编码器 [35]，其包括两部分：1）在层归一化 [1] 之后的多头自注意力机制，以及 2）在层归一化之后的多层感知器。此外，采用残差连接 [17] 来连接这两部分。最终，可以将这些 Transformer 层的 K 次串联连接的输出作为多维序列特征，用于后续在不同功能性头部中的预测。

为了在计算成本和特征提取效果之间取得最佳平衡，Pure Twin Transformer 层遵循原始 Transformer 层设计，只是将多头注意力替换为 Twin Attention。

为了进一步提高 Twin Transformer 的性能，设计了 Hybrid Twin Transformer 层。它在每个 Twin Attention 模块中增加了两个 3x3 卷积层，连接一个 Leaky ReLU 层。这种额外的卷积操作补充了注意力机制，更好地捕获局部信息，增强特征表示。

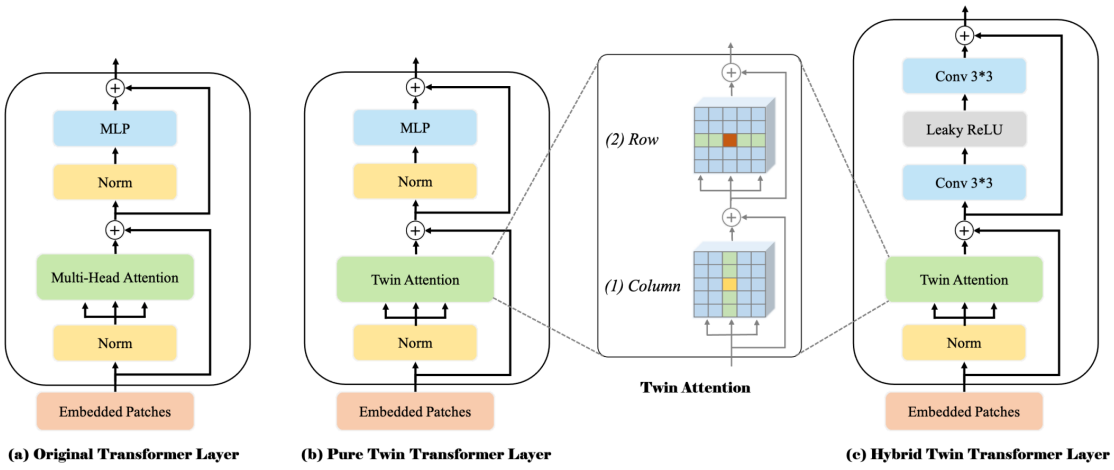

图 2. 不同 Transformer 层设计

3.3.3 Functional Head

来自 Transformer 模块的特征图被输入到不同的功能性头部以进行后续预测。类别头部包含一个单一的线性层，用以输出一个 $N \times N \times M$ 的分类结果，其中 M 是类别的数量。由于每个补丁像 YOLO[32] 一样只为一个中心落在该补丁内的单个对象分配一个类别，我们利用多级预测，并在不同特征层级间共享头部，以进一步提高模型在不同规模对象上的性能和效率。核心头部也由一个线性层组成，与类别头部并行，输出一个 $N \times N \times D$ 的张量，用于后续的掩模生成，其中张量表示具有 D 个参数的 $N \times N$ 卷积核。在训练期间，对分类应用 Focal Loss[26]，而这些卷积核的所有监督来自最终的掩模损失。

4 复现细节

4.1 与已有开源代码对比

本章提供的源代码搭建基于 ResNet101 的骨干网络结构作为特征提取器。但在某些情况下，它可能存在一些局限性。特别是，ResNet 的传统设计在低层次特征提取方面可能不够强大，这可能导致模型在处理一些细节丰富的目标对象时性能下降。ResNetXt 是对 ResNet 的改进，引入了分组卷积等技术，以提供更强大的低层次特征提取能力。分组卷积有助于更好地捕获细节信息，并且具有更强的特征表示能力。因此，本次复现工作将 ResNet 替换为 ResNetXt，旨在增强模型对目标对象低层次特征的表达能力，从而有望提高分割任务的性能。

4.2 实验环境搭建

本次前言技术复现使用的编程语言是 Python3.8,深度学习框架使用的是 PyTorch (1.10.1) 和 (Detectron2 0.6+cu113)。并在一个 TITAN X 的 GPU 上训练了 2500 个周期，批量大小为 8。对于优化器，本研究采用随机梯度下降 (SGD)。初始学习率设置为 0.01，并采用 1k 迭代的恒定预热，使用 10^{-4} 的权重衰减和 0.9 的动量。

4.3 实验数据

本研究在具有挑战性的 MS COCO 公共数据集上进行实验，该数据集包含有 80 类实例标签的 123K 图像。所有模型均在 train2017 子集上进行训练，并在 test-dev 子集上进行评估。本研究采用的实例分割评估指标包括平均精度 (AP)，在 IoU 0.5 (AP50)，0.75 (AP75) 的 AP，以及不同尺寸对象的 AP（APS, APM, 和 APL)。

4.4 创新点

将 ResNet 替换为 ResNetXt，提高模型对目标对象的低层次特征提取能力。

5 实验结果分析

Model	mask AP	AP50	AP75	APS	APM	APL
SOTR_ResNet50	38.723	59.740	41.365	17.492	42.016	57.813
SOTR_ResNet101	39.726	60.299	42.701	18.041	43.414	59.796
SOTR_ResNetXt101	41.532	63.300	63.300	20.184	45.425	61.774

表 1. 模型性能对比

本研究对 SOTR 模型进行了改进，将其骨干网络从 ResNet101 替换为 ResNetXt101, 实验结果如表1所示。相比原始的 SOTR_ResNet101 模型，SOTR_ResNetXt101 在所有评价指标上都显示出显著的提升。具体而言，mask AP 从 39.726 提升至 41.532，AP50 从 60.299 提升至 63.300，AP75 从 42.701 提升至 63.300，APS 从 18.041 提升至 20.184，APM 从 43.414

提升至 45.425，APL 从 59.796 提升至 61.774。这些结果表明，ResNetXt101 的引入显著增强了模型在各种尺寸目标的检测能力，尤其是在处理细节丰富的目标时。图 3 为实验结果的可视化对比。

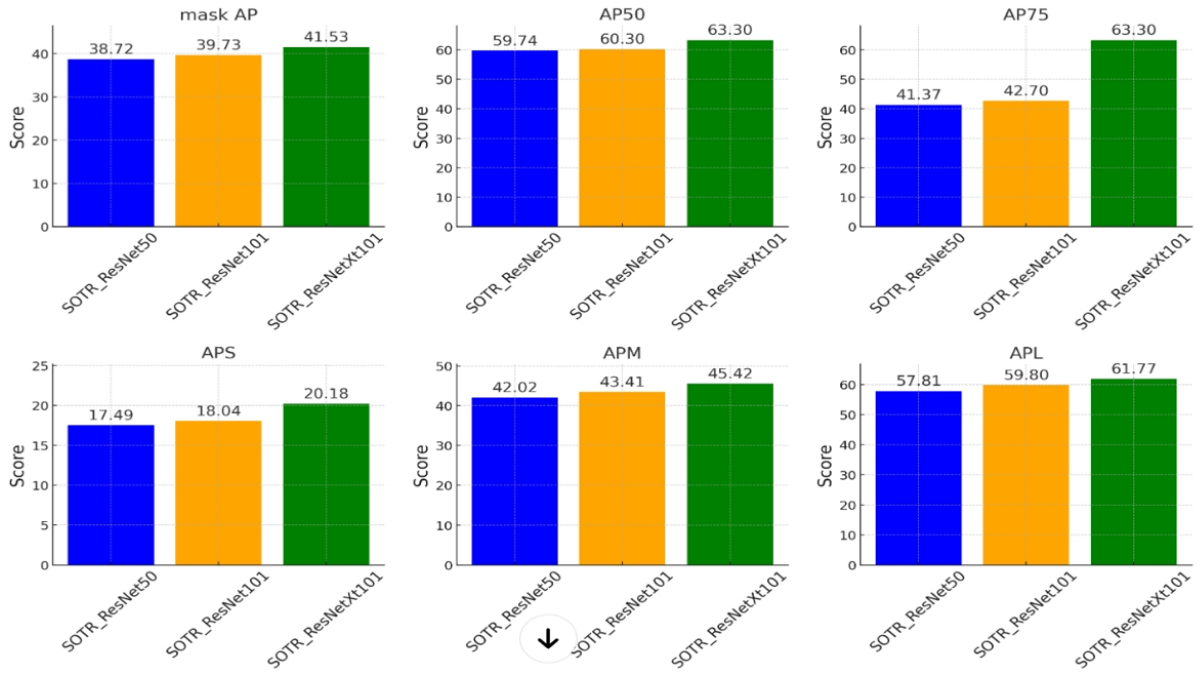

图 3. 模型性能对比

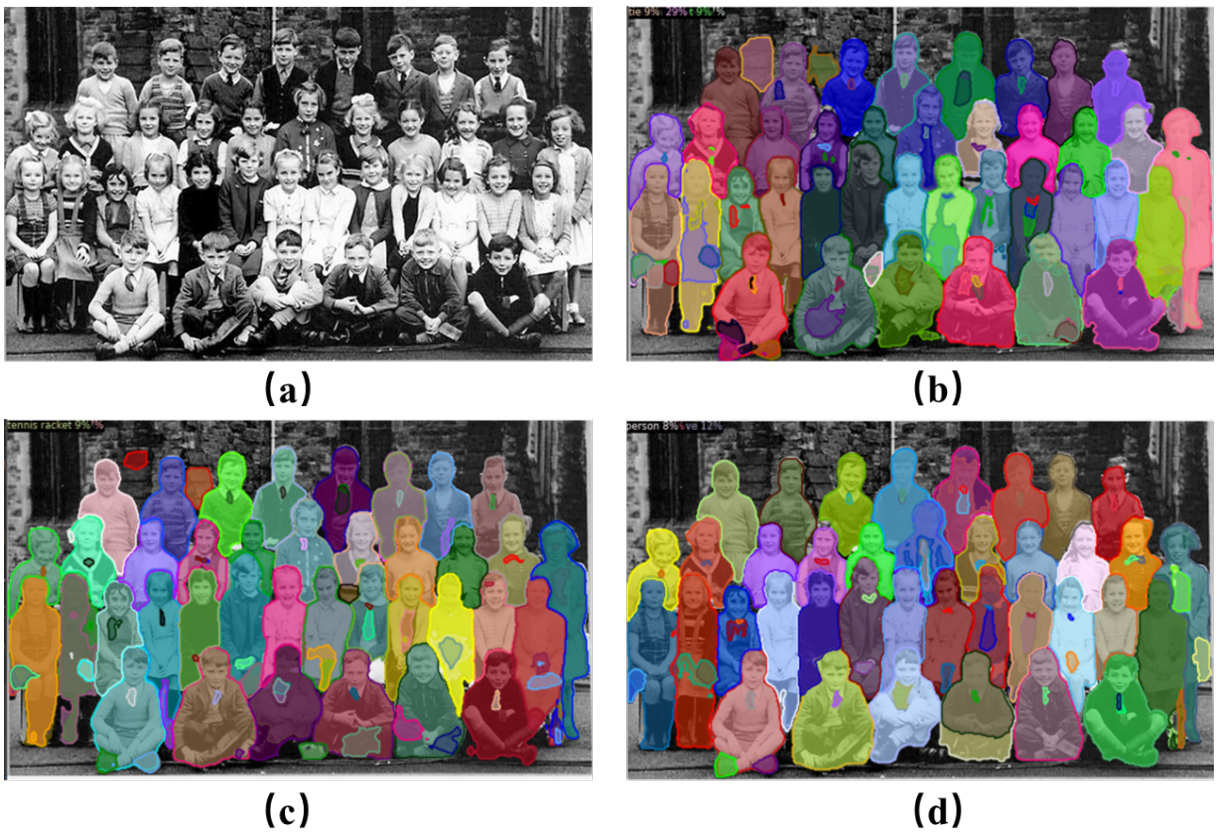

图 4. 分割结果对比

图 4 展现了使用 SOTR 系列不同模型的图像分割效果。其中，子图 a 显示原始图像；子图 b、c 和 d 对应 SOTR_ResNet50、SOTR_ResNet101 和 SOTR_ResNetXt101 模型的分割结果。随着骨干网络的逐步增强，模型在分割精度上有所提升，尤其在细节边缘和小尺寸物体的处理上更为准确。SOTR_ResNetXt101 因采用分组卷积技术，在细节表现上尤其优秀，说明选择适合的骨干网络对提高实例分割性能至关重要。

6 总结与展望

在本次复现和改进的工作中，SOTR 框架经过对骨干网络的优化，尤其是在 ResNetXt 的引入和分组卷积技术的应用后，表现出对复杂细节的更强捕捉能力，特别是对复杂边缘细节处理。展望未来，研究将致力于进一步提升 Transformer 在复杂视觉任务中的应用效率，同时，将探索模型结构的优化，以满足实时处理的需求，并适应不同规模和复杂度的图像分割任务。这将为实例分割技术的发展开辟新的可能性，特别是在动态环境下的应用。

参考文献

[1] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[2] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[3] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9157–9166, 2019.

[4] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019.

[5] Youngwan Lee and Jongyoul Park. Centermask: Real-time anchor-free instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13906–13915, 2020.

[6] Shu Liu, Jiaya Jia, Sanja Fidler, and Raquel Urtasun. Sgn: Sequential grouping networks for instance segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 3496–3504, 2017.

[7] Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. Solo: Segmenting objects by locations. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 649–665. Springer, 2020.

[8] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on vision transformer. *IEEE transactions on pattern analysis and machine intelligence*, 45(1):87–110, 2022.

[9] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM computing surveys (CSUR)*, 54(10s):1–41, 2022.

[10] Jonathan Ho, Nal Kalchbrenner, Dirk Weissenborn, and Tim Salimans. Axial attention in multidimensional transformers. *arXiv preprint arXiv:1912.12180*, 2019.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arxiv 2020. *arXiv preprint arXiv:2010.11929*, 2010.

[12] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890, 2021.

[13] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.