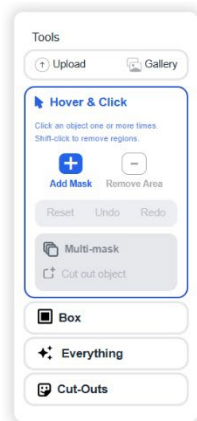


Segment Anything Model



Introduction

The Segment Anything Model (SAM) [1] by Meta AI has transformed computer vision by enabling prompt-based segmentation across varied domains. Its success has led to multiple variants—SAM2 [2], EdgeSAM [3] and MobileSAM [4,5]—aimed at improving scalability and efficiency. However, these models remain computationally demanding [6], posing challenges for real-time and edge deployment.

Background

SAM, introduced by Kirillov et al. (2023) [1], and later extensions like SAM2 [2], established a foundation for general-purpose segmentation. Lightweight variants such as MobileSAM [4,5] and EdgeSAM [3] focus on reducing model size and latency. Works like and MobileNet [7] and EfficientNet [8] further demonstrate how architecture design can balance accuracy and computation.

Problem Specification

Although SAM [1, 2] achieves remarkable segmentation performance, its large model size and heavy computational requirements hinder practical deployment on edge and mobile devices [6]. These architectures attempt to balance accuracy, generalization, and computational efficiency. However, despite their impressive segmentation capabilities, these models remain computationally expensive [6].

Suggested Method

An empirical approach will be used to profile and optimize latest SAM family of architecture, such as NanoSAM architectures. Model components, including the encoder, prompt encoder, and mask decoder—will be analyzed using tools such as caltflops and torch.profiler to measure FLOPs, parameters, memory, and latency. Optimization will focus on replacing the heavy encoder with efficient backbones like MobileNet [7] or EfficientNet [8], followed by benchmarking to compare accuracy, speed, and resource usage against the state-of-the-art architecture.

Expected Outcome

The study will provide a clearer understanding of SAM's architecture and computational flow, identifying opportunities for efficiency improvement. The student will gain skills in deep learning analysis, model profiling, and hardware-aware optimization, as well as hands-on experience in PyTorch and computer vision model design—bridging theoretical understanding and practical AI system development.

Relevant Articles

- [1] Kirillov, Alexander, et al. "Segment anything." *Proceedings of the IEEE/CVF international conference on computer vision*. 2023.
- [2] Ravi, Nikhila, et al. "Sam 2: Segment anything in images and videos." *arXiv preprint arXiv:2408.00714* (2024).
- [3] Lu, Rui, et al. "Samedge: An edge-cloud video analytics architecture for the segment anything model." *arXiv preprint arXiv:2409.14784* (2024).
- [4] Zhang, Chaoning, et al. "Faster segment anything: Towards lightweight sam for mobile applications." *arXiv preprint arXiv:2306.14289* (2023).
- [5] Zhang, Chaoning, et al. "Mobilesamv2: Faster segment anything to everything." *arXiv preprint arXiv:2312.09579* (2023).
- [6] Sun, Xiaorui, et al. "On efficient variants of segment anything model: A survey." *International Journal of Computer Vision* 133.10 (2025): 7406-7436.
- [7] Qin, Danfeng, et al. "MobileNetV4: Universal models for the mobile ecosystem." *European Conference on Computer Vision*. Cham: Springer Nature Switzerland, 2024.
- [8] Tan, Mingxing, and Quoc Le. "Efficientnetv2: Smaller models and faster training." *International conference on machine learning*. PMLR, 2021.

Useful Tools

- i. Google CoLaboratory for running Python code and training deep learning model: <https://colab.research.google.com/>
- ii. Segment Anything Model: <https://segment-anything.com/>
- iii. Source Code of Nano SAM: <https://github.com/NVIDIA-AI-IOT/nanosam>
- iv. Tensorflow Model Optimization Toolkit: https://www.tensorflow.org/lite/performance/model_optimization
- v. Image segmentation databases: <https://paperswithcode.com/datasets?task=semantic-segmentation>
- vi. CityScape Dataset: <https://www.cityscapes-dataset.com/>
- vii. CalFlops: <https://github.com/MrYxJ/calculate-flops.pytorch>

For more information, please contact Prof. Mårten Sjöström (Marten.Sjostrom@miun.se)