

S2-UniSeg: Fast Universal Agglomerative Pooling for Scalable Segment Anything without Supervision

Huihui Xu^{1,2}, Jin Ye², Hongqiu Wang¹, Changkai Ji², Jiashi Lin², Ming Hu², Ziyang Huang², Ying Chen², Chenglong Ma², Tianbin Li², Lihao Liu², Junjun He^{2*}, Lei Zhu^{1,3*}

¹The Hong Kong University of Science and Technology (Guangzhou)

²Shanghai Artificial Intelligence Laboratory

³The Hong Kong University of Science and Technology
leizhu@hkust-gz.edu.cn

Abstract

Recent self-supervised image segmentation models have achieved promising performance on semantic segmentation and class-agnostic instance segmentation. However, their pretraining schedule is multi-stage, requiring a time-consuming pseudo-masks generation process between each training epoch. This time-consuming offline process not only makes it **difficult to scale with training dataset size**, but also leads to **sub-optimal solutions** due to its discontinuous optimization routine. To solve these, we first present a novel pseudo-mask algorithm, Fast Universal Agglomerative Pooling (UniAP). Each layer of UniAP can identify groups of similar nodes *in parallel*, allowing to generate both **semantic-level and instance-level** and **multi-granular** pseudo-masks within *tens of milliseconds* for one image. Based on the fast UniAP, we propose the Scalable Self-Supervised Universal Segmentation (S2-UniSeg), which employs a student and a momentum teacher for continuous pretraining. A novel **segmentation-oriented pretext task**, Query-wise Self-Distillation (QuerySD), is proposed to pre-train S2-UniSeg to learn the local-to-global correspondences. Under the same setting, S2-UniSeg outperforms the SOTA UnSAM model, achieving notable improvements of AP+6.9 on COCO, AR+11.1 on UVO, PixelAcc+4.5 on COCOStuff-27, RQ+8.0 on Cityscapes. After scaling up to a larger 2M-image subset of SA-1B, S2-UniSeg further achieves performance gains on all four benchmarks. Our code and pretrained models shall be released upon the acceptance of this work.

Introduction

Supervised image segmentation has made significant progress in recent years (Xie et al. 2021; Cheng et al. 2022; Kirillov et al. 2023; Zou et al. 2024; Li et al. 2023; Ravi et al. 2024). However, training such segmentation models typically relies on large-scale human annotations, which are both time-consuming and labor-intensive. For example, annotating a single image in the SA-1B dataset (Kirillov et al. 2023) can take over 25 minutes of detailed effort. Moreover, human annotations are probably noisy, inconsistent, and influenced by subjective biases, leading to misalignment with fine-grained image details and object boundaries. These limitations can undermine the robustness and generalizability of

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

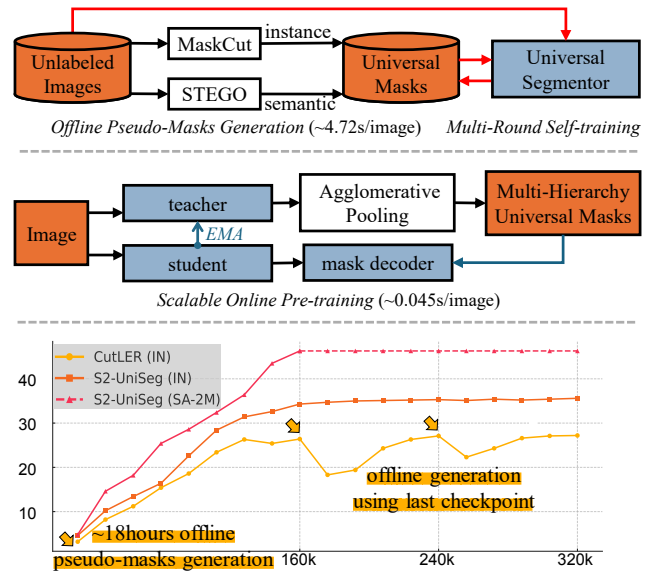


Figure 1: Previous paradigms vs. our paradigm. S2-UniSeg eliminates the need for time-consuming offline pseudo-mask generation, functioning as a single-stage pretraining framework with continuous optimization.

segmentation models across diverse visual contexts. Therefore, we propose an efficient single-stage self-supervised segmentation model leveraging intrinsic image information to reduce reliance on expensive annotation.

As shown in the top of Figure 1, existing self-supervised image segmentation methods typically adopt a two-stage pipeline (Wang et al. 2023a; Wang, Yang, and Darrell 2024; Niu et al. 2024; Arica et al. 2024). In the first stage, based on pretrained self-supervised visual representations (Caron et al. 2021; Grill et al. 2020; Chen and He 2021; Chen et al. 2020; He et al. 2020; Oquab et al. 2023), graph partitioning algorithms (Wang et al. 2023a,b) are used to generate pseudo-masks *offline* for all unlabeled images. These pseudo-masks are then used to initialize training of the segmentation model. In the second stage, segmentation performance is improved via *multi-round self-training*, where pseudo-masks generated from the model checkpoint of the

Method	Time	AP	mIoU	Universal	Multi-Granular
Divide-and-Conquer	5.27	7.6	-	✗	✓
TokenCut	2.25	3.5	-	✗	✗
MaskCut	4.72	6.7	-	✗	✗
MaskCut+STEGO	4.86	6.7	26.3	✓	✗
UniAP	0.045	6.2	21.8	✓	✓

Table 1: Quantitative efficiency and effectiveness comparison among different pseudo-masks generation algorithms. We run all methods on COCOval2017 with the same hardware settings, and evaluate their average processing time per image, zero-shot class-agnostic instance segmentation performance (AP^{mask}), unsupervised semantic segmentation performance (mIoU). All methods use the DINO pretrained ViT-base/8 features.

previous round are used to fine-tune the model at next round until convergence.

As shown in Table 1, although the pseudo-masks used in the first stage achieve a higher AP^{mask} , the generation process is time-consuming. For UnSAM (Wang, Yang, and Darrell 2024), average time for generating pseudo masks for a 512×512 image takes 5.27 seconds, which is impractical to scale with large-scale datasets. Furthermore, as shown in the bottom of Figure 1, the overall optimization is discontinuous, leading to unstable and sub-optimal convergence.

To achieve scalable and continuous training, the primary challenge is to reduce the time for pseudo-mask generation. To this end, we propose the **Fast Universal Agglomerative Pooling (UniAP)** to efficiently generate both universal and multi-granular pseudo-masks of a single image within tens of milliseconds. Our key insight is that pixels belonging to the same instance or semantic category are spatially adjacent in form of a connected region. These groups of strongly connected pixels can be identified in parallel (Pearce 2005). As shown in Table 1, UniAP achieves comparable segmentation performance while being approximately $100\times$ faster than previous methods, making it feasible for single-stage pretraining and scalable to large-scale dataset.

Equipped with fast pseudo-masks generation by UniAP, we propose the **Scalable Self-Supervised Universal Segmentation (S2-UniSeg)**, which leverages the teacher-student framework trained with a novel segmentation-oriented pretext task **Query-wise Self-Distillation (QuerySD)**. We condense each image into a set of universal object queries, and train each student query to predict the corresponding bipartite-matched teacher query. Our main contributions are:

- We propose Fast Universal Agglomerative Pooling (UniAP), which can generate both universal and multi-granular pseudo-masks within tens of milliseconds. UniAP makes it feasible for continuous optimization and scalable with large-scale dataset.
- We present the Scalable Self-Supervised Universal Segmentation (S2-UniSeg) framework, which features a novel segmentation-oriented pretext task Query-wise Self-distillation (QuerySD).

- Extensive experiments on ImageNet, SA-1B, COCO, UVO, COCOstuff, and Cityscapes demonstrate that S2-UniSeg achieves substantial performance improvements on four universal segmentation tasks. Moreover, S2-UniSeg demonstrates evident scalability, yielding better performance as the training dataset size grows.

Related Work

Self-supervised Representation Learning (SSL). Self-supervised representation learning aims to learn general-purpose (Bengio, Courville, and Vincent 2013) features from large amounts of unlabeled data samples without manual annotation. A pretext task is often defined to train the model. According to types of pretext task, they can be classified into *contrastive learning* methods and *masked image modeling* methods. Contrastive learning based methods include pretext tasks based on negative samples (Chen et al. 2020; He et al. 2020; Chen, Xie, and He 2021), clustering (Caron et al. 2020; Asano, Rupprecht, and Vedaldi 2019), self-distillation (Caron et al. 2021; Grill et al. 2020; Chen and He 2021; Oquab et al. 2023), and feature decorrelation (Zbontar et al. 2021; Bardes, Ponce, and LeCun 2021). Masked image modeling methods include pretext tasks based on low-level targets (He et al. 2022; Chen et al. 2020; Wei et al. 2022), high-level targets (Bao et al. 2021; Dong et al. 2023; Chen et al. 2024), self-distillation (Chen et al. 2022a; Baevski et al. 2022), and multi-modal teacher (Zhou et al. 2022; Peng et al. 2022).

Self-supervised Instance and Semantic Segmentation. Recent studies (Caron et al. 2021; Hamilton et al. 2022; Siméoni et al. 2021; Vo, Pérez, and Ponce 2020; Vo et al. 2021) show pretrained SSL features can capture pixel-to-pixel semantic similarity. Inspired by that, several works (Hamilton et al. 2022; Wang et al. 2023a; Wang, Yang, and Darrell 2024; Seong et al. 2023; Kim et al. 2024; Wang et al. 2023b; Van Gansbeke, Vandenhende, and Van Gool 2022; Wang et al. 2022; Liu et al. 2024; Hahn et al. 2025; Aydemir, Xie, and Guney 2023) aim to distill or self-train a segmentation model based on the pretrained SSL representations. These methods can be classified into semantic segmentation methods (Hamilton et al. 2022; Kim et al. 2024; Liu et al. 2024; Seong et al. 2024), zero-shot class-agnostic instance segmentation methods (Wang et al. 2023a; Wang, Yang, and Darrell 2024), and universal segmentation methods (Niu et al. 2024). State-of-the-art unsupervised zero-shot instance segmentation methods (Wang et al. 2023a; Arica et al. 2024; Wang, Yang, and Darrell 2024) adopt a cut and learn pipeline, in the sense that they first generate pseudo-masks of the whole dataset using pretrained SSL features, and then learn a model through multi-round self-training. Unsupervised semantic segmentation methods (Hamilton et al. 2022; Seong et al. 2023; Kim et al. 2024; Liu et al. 2024) adopts a distillation-based objective, in the sense that the projected segmentation features should preserve the pixel-to-pixel semantic correspondence in the frozen SSL representation space. Recently, U2Seg (Niu et al. 2024) proposes a self-supervised universal segmentation framework for class-aware instance and panoptic segmentation. U2Seg adopts a similar cut-and-

learn pretraining pipeline but also clusters the masks to generate their pseudo classification labels.

Graph Pooling. As an essential component of Graph Neural Networks, graph pooling (Liu et al. 2022; Buterez et al. 2022; Grattarola et al. 2022; Mesquita, Souza, and Kaski 2020) is crucial for aggregating information from multiple nodes to obtain holistic subgraph-level representations. Graph pooling can be roughly divided into flat pooling and hierarchical pooling. Flat pooling (Dwivedi et al. 2023; Xu et al. 2020; Noutahi et al. 2019), also known as Graph Readout (Buterez et al. 2022), aims to obtain a global representation. Hierarchical pooling aims to iteratively coarsen the graph into smaller size and generate multi-scale graph features. According to how new nodes are formed, it can be classified into node clustering pooling (Wu, He, and Xu 2019; Liu et al. 2021) and node drop pooling (Gao, Chen, and Ji 2019; Lee et al. 2021; Gao et al. 2021). The primary distinction lies in that node clustering pooling aggregates information from old nodes into new nodes, whereas node drop pooling directly drops unwanted ones and retains a subset of old nodes.

Scalable Self-Supervised Universal Segmentation

As shown in Figure 2, S2-UniSeg has a student branch and a teacher branch. Given an image, we follow multi-crop (Caron et al. 2020) to get a set of δ local views input to the student branch. The original image is treated as the global view input to the teacher branch. The student and teacher encoder both adopt the same architecture of a multi-scale encoder (Chen et al. 2022b).

Teacher Branch. For the global view, its feature map with the largest scale (stride 4) is input to a stack of Γ UniAP layers to generate the instance and semantic pseudo-masks along with their feature embeddings. We call these embeddings as teacher queries in later sections.

Student Branch. For each local view, a mask decoder (Cheng et al. 2022) takes its multi-scale feature maps as input and predicts its semantic and instance masks each along with the query features at the last decoder layer. During query initialization, we partition object queries into two groups—semantic queries, which are distinguished by a learnable token [SEM] and obligated to attend to scattered semantic regions, and instance queries, which are distinguished by a learnable token [INS] and obligated to identify individual instances. Compared with the previous tedious two-branch decoder design in U2Seg (Niu et al. 2024), our framework achieves task distinction using only two learnable tokens.

Fast Universal Agglomerative Pooling

As shown in Figure 2, each UniAP layer first identifies groups of strongly-connected nodes, and merges each group into one supernode. Compared with the optimization-based TokenCut (Wang et al. 2023b) and MaskCut (Wang et al. 2023a), UniAP is a heuristic nonparametric approach and does not require computation of eigenvectors. Moreover, based on the SCC (Pearce 2005) algorithm, UniAP can

merge a group of strongly-connected nodes in parallel. As shown in Table 1, UniAP is much faster than existing methods and achieves comparable performance, which can be integrated into each training step to enable single-stage pre-training and continuous optimization routine.

Moreover, UniAP can also generate pseudo-masks for scattered regions of semantic segmentation, which are not supported in MaskCut (Wang et al. 2023b), TokenCut (Wang et al. 2023a), and Divide-and-Conquer (Wang, Yang, and Darrell 2024). UniAP is nonparametric and does not require in-domain pretrained STEGO (Hamilton et al. 2022) as in U2Seg. Next, we illustrate the graph initialization, identify step, and merge step in detail. UniAP is summarized in the Appendix Algorithm 1.

Graph Initialization. We use $\mathbf{F} \in R^{HW \times d}$ to denote the L2-normalized feature map with the largest scale from the teacher encoder of the global view, where H and W is the spatial size. To initialize the graph, each token is treated as one node, and edges are formed solely between two nodes that are *directly* adjacent horizontally or vertically. Each i^{th} node of the t^{th} layer is associated with a mask $\mathbf{M}_i^t \in \{0, 1\}^{HW}$ denoting which tokens belong to its subtree. We initialize \mathbf{M}^0 as identity matrix $\mathbf{I} \in \{0, 1\}^{HW \times HW}$. The initialized *undirected, connected* graph is denoted as $\mathcal{G}^0 = \{\mathbf{V}^0, \mathbf{M}^0, \mathbf{E}^0\}$, where $\mathbf{V}^0 = \mathbf{F} \in R^{s^0 \times d}$ is node features, $\mathbf{E}^0 \in \{0, 1\}^{s^0 \times s^0}$ is adjacency matrix, $s^0 = HW$ is nodes number. We also denote $\mathbf{A} = \mathbf{F}\mathbf{F}^T \in R^{HW \times HW}$ as the spatial affinity matrix, which is of range $[-1, 1]$.

Identify Step. To identify a group of strongly-connected nodes, for each edge, we first compute the similarity of the two connected nodes. The *feature similarity* of two adjacent i^{th} and j^{th} nodes is computed as:

$$\mathcal{S}_{ij}^f = \mathbf{V}_i^t \mathbf{V}_j^{tT} \in [-1, 1]. \quad (1)$$

Since the node features are evolving at each layer due to the merge step, the pairwise feature similarity may not be consistent with the one implied from the original encoder features. To mitigate this issue, we also measure the *spatial similarity* of two nodes as:

$$\mathcal{S}_{ij}^s = 1 - \frac{1}{HW} \left| \frac{\mathbf{M}_i^t \mathbf{A}}{\mathbf{M}_i^t \mathbf{1}^T} - \frac{\mathbf{M}_j^t \mathbf{A}}{\mathbf{M}_j^t \mathbf{1}^T} \right| \mathbf{1}^T \in [-1, 1], \quad (2)$$

where $|\cdot|$ is the absolute operator, $\mathbf{1} \in R^{1 \times HW}$ is a vector of 1. According to Equation 2, if node i and j have similar affinity distribution along the original HW tokens map, their spatial similarity will be large. \mathcal{S}_{ij}^s does not require direct feature comparison, it instead assembles a voting mechanism, in the sense that each original token is a voter which gives its score on $i - j$'s similarity. The final similarity measure is formulated as:

$$\mathcal{S}_{ij} = \omega_f \mathcal{S}_{ij}^f + \omega_s \mathcal{S}_{ij}^s, \quad (3)$$

where $\omega_f + \omega_s = 1$ and $\mathcal{S}_{ij} \in [-1, 1]$. Then, given a threshold τ_t , edges with $\mathcal{S}_{ij} \geq \tau_t$ are labeled to be coarsened. We then use the SCC (Pearce 2005) algorithm to find groups of strongly-connected nodes. Nodes in each group will be merged into one supernode.

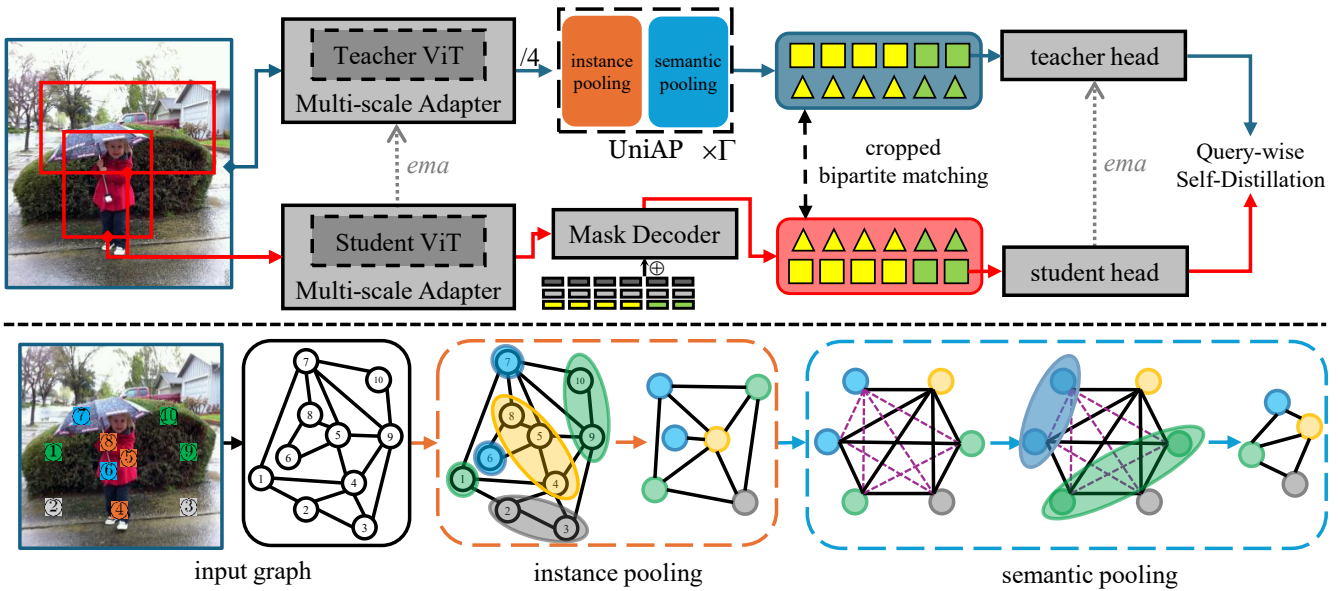


Figure 2: Model architecture for Scalable Self-Supervised Universal Segmentation (S2-UniSeg). (Top) A student and a momentum teacher is leveraged to facilitate self-distillation. The original view (global) is fed into the teacher branch, then we propose Fast Universal Agglomerative Pooling to efficiently generate universal masks (Δ) and their query features (\square). In student branch, we use mask decoder with task-specific object queries to predict the local view universal masks. During training, bipartite matching is devised to match student masks with the cropped teacher masks. (Bottom) Each UniAP layer is composed of a semantic and instance pooling obligated to generate universal pseudo-masks. UniAP can efficiently identify groups of strongly-connected nodes *in parallel*, thus enabling single-stage continuous pretraining.

Merge Step. The SCC algorithm (Pearce 2005) outputs a node-supernode assignment matrix $\Omega \in \{0, 1\}^{s^t \times s^{t+1}}$. According to the assignment matrix, the new adjacent matrix and mask matrix are updated as:

$$\mathbf{E}^{t+1} = \Omega^T \mathbf{E}^t (\Omega^T \mathbf{E}^t)^T; \mathbf{M}^{t+1} = \Omega^T \mathbf{M}^t, \quad (4)$$

where each supernode takes the union of its children nodes masks. To get supernode features, a direct solution is summarizing children features, i.e. $\mathbf{V}^{t+1} = \text{L2N}\{\Omega^T \mathbf{V}^t\}$. However, as shown in Figure 3, we empirically find that this can cause semantically-different but spatially-close nodes to merge earlier. We think this is because the supernode feature updated in this way will treat its boundary tokens and main region tokens equally. When two supernodes are neighboring to each other, their common boundary tokens will make their features have a larger similarity. To mitigate this issue, the supernode feature is computed as:

$$\mathbf{V}_i^{t+1} = \text{L2N}\{\text{softmax}(\frac{\mathbf{M}_i^{t+1} \mathbf{A}}{\sigma}) \mathbf{F}\}, i = 1, \dots, s^{t+1} \quad (5)$$

where $\text{L2N}\{\cdot\}$ denotes L2-normalization, σ is the softmax temperature. Equation 5 can also be seen as a weighted voting mechanism, where most of the softmax mass will lie on the main region of supernode, and information of boundary tokens is suppressed. As shown in Figure 3, the “grass” region are not merged with “human leg” region throughout all layers. This shows that updating supernode features using Equation 5 can make semantically-different but spatially-close regions more discriminative.

Universal Pseudo-masks Generation with Instance Pooling and Semantic Pooling. It is noteworthy that edges exist only between directly adjacent nodes in \mathcal{G}^0 . Moreover, Equation 4 ensures edges are maintained only between adjacent supernodes, i.e. sparsely connected. Consequently, each mask corresponds to a consolidated region. It is sensible for instance segmentation but not for semantic segmentation, which requires masks that encompass multiple disjoint regions belonging to the same semantic class. As shown in Figure 2, to address this limitation, we construct a *fully connected* version of the graph \mathcal{G}^* derived from instance pooling. Subsequently, the same Identify and Merge pipeline is employed to generate semantic masks, thereby combining multiple disjoint components into a single semantic-level mask. The graph derived from instance pooling is used as input of the next UniAP layer.

Multi-Granular Pseudo-masks Generation with Time-varied Thresholding. UniAP can be seen as an *iterative coarsening* procedure. After each iteration, each supernode will represent its receptive region defined by \mathbf{M}_i^t . By defining a set of decreasing thresholds $\{\tau_t | \tau_{t+1} < \tau_t\}_{t=1}^T$, we can get masks at different hierarchy. We empirically find that background tokens, where pixels are very similar to their neighbors, are merged at the very early layers, i.e. with much higher threshold close to 0.9. While tokens like “human head” and “human body” each with higher level semantics are merged at later layers with a lower threshold close to 0.5. Readers can refer to Appendix C.2, Appendix Figure

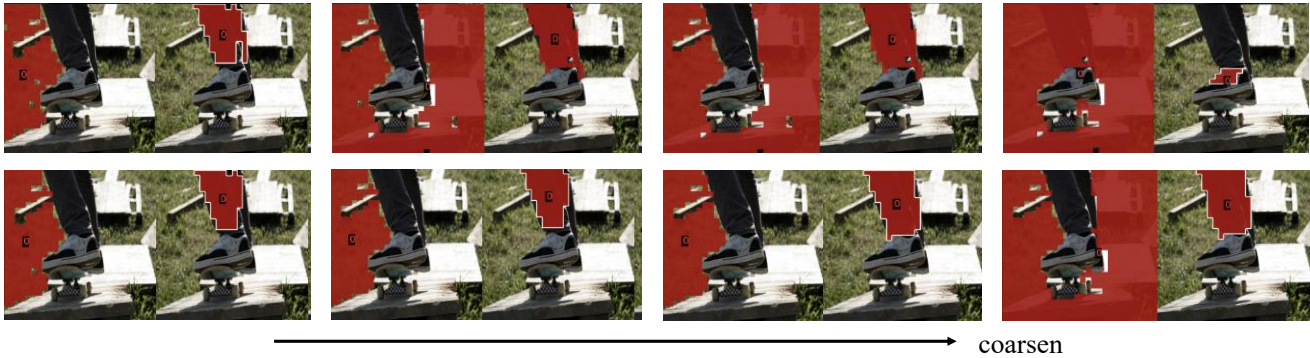


Figure 3: Mask visualization of different feature updating strategies. Each column corresponds to one UniAP layer ($t=2,3,4,5$). (Top) The supernode feature is updated using $\mathbf{V}^{t+1} = \text{L2N}\{\Omega^T \mathbf{V}^t\}$. (Bottom) The supernode feature is updated using Equation 5. Please refer to Appendix C.2 for more qualitative visualizations.

5, and Appendix Figure 6 for more detailed visualizations.

Model Training and Initialization

To deploy UniAP for scalable single-stage pretraining, we devise a teacher-student framework with momentum updating strategy. Moreover, we introduce a novel pretext task, Query-wise Self-Distillation, that self-distills in a query-wise manner to facilitate segmentation-oriented self-supervised learning.

Multi-scale encoder and zero-initialization. Existing self-supervised models (Wang et al. 2023a; Wang, Yang, and Darrell 2024; Niu et al. 2024) utilize ResNet (He et al. 2016) as backbone, whereas they use DINO (Caron et al. 2021) pretrained ViT-base/8 (Dosovitskiy 2020) to generate pseudo-masks. Our framework only devises ViT-base/8 as backbone. We also follow ViT-Adapter (Chen et al. 2022b) to augment ViT with lightweight multi-scale adapters. To save computation cost, we scale down the image input to the ViT branch by half as if using a patch size of 16. To generate sensible pseudo-masks as training begins, we zero-initialize the adapters so that UniAP can utilize the pretrained DINO features at the very start.

Task-specific Object Queries. Unlike U2Seg (Niu et al. 2024) where the decoder has two separate branches, we adopt a more streamlined approach by partitioning the object queries in mask decoder into semantic queries and instance queries with two special learnable tokens [INS] and [SEM]. It is noteworthy that both groups of queries share the same decoder parameters and projection head, enabling universal segmentation through parameter sharing.

Cropped Bipartite Matching. The teacher branch processes the original *global* view of the image, producing pseudo-masks of the entire image. In contrast, the student branch processes *local* views obtained from multi-crop augmentation. For each local view, we crop the teacher’s pseudo-masks correspondingly to match the student’s view position. Teacher queries and pseudo-masks that do not overlap with the student’s view are dropped. Only mask Dice similarity is used as the matching criterion. The matching

processes for the semantic and instance levels are independent and do not interfere with each other.

Query-wise Self-Distillation. We denote the query embeddings of the teacher and student branch after bipartite matching as $\mathbf{Q}_t \in R^{L \times d}$ and $\mathbf{Q}_s \in R^{L \times d}$, where L is the query number. A projection head of three-layer MLP is used to transform queries into distribution logits. The Query-wise Self-distillation (QuerySD) pretext is formulated as:

$$\sum_{t=1, s \in \{1, \dots, \delta\}} \sum_{l=1}^L \sum_{k=1}^d \mathbf{Q}_t^{l,k} \log \mathbf{Q}_s^{l,k}, \quad (6)$$

where δ is the number of local crops. Loss 6 can be interpreted as the sum of self-distillation loss (Caron et al. 2021) over each pair of matched teacher query and student query. Unlike other self-supervised *representation* frameworks which use the ViT [CLS] token or ResNet global pooling, self-supervised *segmentation* requires more fine-grained features for pretext training. S2-UniSeg first aggregates each image into a set of object queries, and self-distills in a query-wise manner to facilitate segmentation-oriented self-supervised learning.

Experiments

Training Settings

Model Architecture. Following previous works (Niu et al. 2024; Wang et al. 2023a; Wang, Yang, and Darrell 2024), we use the DINO pretrained ViT-base/8 and ViT-Adapter (Chen et al. 2022b) as backbone. For the mask decoder, we use the official setting of Mask2Former (Cheng et al. 2022). For the project head, we follow DINO (Caron et al. 2021) using a 3-layer MLP with hidden dimension 2048 followed by L2 normalization and a linear layer of K dimensions. After training, the final teacher encoder with the mask decoder is used for inference. **Hyper-parameter setting.** We set $\sigma = 0.07$, $\{\tau_t\}_{t=1}^T = [0.8, 0.7, 0.6, 0.5, 0.4]$, $\phi = 5$, $\omega_f = 0.6$, $\omega_s = 0.4$, $K = 512$. The number of semantic and instance queries are set to 50 and 150. A local multi-crop scale between 0.05 and 0.4 is used. **Optimization setting.**

Task	Class-agnostic Instance Seg.		Instance Seg.				Semantic Seg.		Panoptic Seg.					
Datasets	COCO		COCO		UVO		COCOStuff-27		COCO			Cityscapes		
Metric	AP	AR	AP	AR	AP	AR	PixelAcc	mIoU	PQ	SQ	RQ	PQ	SQ	RQ
STEGO (Hamilton et al. 2022) (COCOStuff)	-	-	-	-	-	-	56.9	28.2	-	-	-	-	-	-
CutLER (Wang et al. 2023a) (IN)	9.7	27.1	-	-	-	-	-	-	-	-	-	-	-	-
U2Seg (Niu et al. 2024) (IN)	7.3	19.2	6.4	18.5	6.2	21.0	63.9	30.2	16.1	71.1	19.9	17.6	52.7	21.7
S2-UniSeg (IN)	14.2	34.3	15.3	35.5	16.2	32.1	68.4	36.1	20.2	80.6	26.7	20.5	60.1	29.7
UNSAM (Niu et al. 2024) (SA-1B 0.4M)	31.4	42.0	-	-	-	-	-	-	-	-	-	-	-	-
S2-UniSeg (SA-1B 0.4M)	33.6	43.9	27.4	36.1	20.8	34.6	72.3	37.2	25.7	88.4	28.6	22.5	63.7	32.6
S2-UniSeg (SA-1B 1.2M)	35.1	44.6	29.1	38.4	22.6	35.7	74.4	38.9	28.3	89.2	29.3	23.6	64.3	33.4
S2-UniSeg (SA-1B 2.0M)	36.4	46.3	30.7	40.3	24.3	36.4	76.8	39.7	29.6	90.3	30.2	25.4	66.7	35.2

Table 2: Performance comparison with previous methods on zero-shot class-agnostic instance segmentation, unsupervised instance-segmentation, unsupervised semantic segmentation, and unsupervised panoptic segmentation. Our model outperforms other state-of-the-art methods on all tasks by a large margin. Moreover, pretrained on the large-scale 2M subset from SA-1B (Kirillov et al. 2023), S2-UniSeg achieves significant improvements on class-agnostic instance segmentation compared with UnSAM (Wang, Yang, and Darrell 2024). Please refer to Appendix B and Appendix C for full results and more qualitative visualizations.

We train the model using AdamW with a batch size of 16. The learning rate is linearly warmed up to 0.000625 for 10k iterations. Our model is trained for 160k iterations, while other CutLER models are additionally self-trained for several 80K iterations.

Datasets and Metrics Details

Following U2Seg (Niu et al. 2024) and UnSAM (Wang, Yang, and Darrell 2024), we test unsupervised instance segmentation on COCO_{val} (Lin et al. 2014), and UVO_{val} (Wang et al. 2021). We test unsupervised semantic segmentation on COCOStuff-27 (Caesar, Uijlings, and Ferrari 2018). Following U2Seg (Niu et al. 2024), we test unsupervised panoptic segmentation on Cityscapes_{val} (Cordts et al. 2016) and COCO_{val}. Following U2Seg, the pseudo-classes are mapped using Hungarian matching to class labels. We use the ImageNet-1k (1.3M images) dataset for pre-training. Please refer to Appendix D for more dataset details.

Experiment Results

Self-supervised Instance Segmentation. As shown in Table 2, for class-agnostic unsupervised instance segmentation, S2-UniSeg achieves an increase of **+6.9** in AP and **+15.1** in AR, which is 94.5% and 78.6% increase compared to U2Seg (Niu et al. 2024). As shown in Figure 1, our online pretraining framework can converge faster and achieve significant improvements over previous multi-stage alternating methods. Moreover, for our method, the performance of the class-aware instance segmentation is higher of that in class-agnostic instance segmentation. However, this is reversed for U2Seg. This shows that our online clustering QuerySD pretext task can better help model to capture the semantics information in the pretraining dataset.

Self-supervised Semantic Segmentation. As shown in Table 2, S2-UniSeg outperforms other state-of-the-art methods on COCOStuff-27 for unsupervised semantic segmentation. Specifically, our model achieves an increase of **+5.9** in PixelAcc, which is 19.5% of increase compared to U2Seg and 20.9% of increase compared to STEGO. Please refer to Appendix B and Appendix C for more qualitative comparisons.

Self-supervised Panoptic Segmentation. As shown in Table 2, S2-UniSeg also significantly outperforms U2Seg over panoptic segmentation on COCO and Cityscapes. For the zero-shot setting (solely trained on ImageNet), our method achieves an increase of **+6.8** in SQ on Cityscapes, which is 14.5% of increase compared to U2Seg, and achieves an increase of **+5.1** in PQ on COCO, which is 45.9% of increase compared to U2Seg. Please refer to Appendix B and Appendix C for more qualitative comparisons.

Scaling S2-UniSeg with SA-1B.

Since S2-UniSeg is a single-stage pretraining framework without any time-consuming offline processes, we can easily scale S2-UniSeg with larger dataset size. We pretrain S2-UniSeg on subsets of SA-1B with different scale, i.e. 0.4M, 1.2M, and 2M for 160K iterations. The optimization landscape is plotted in Figure 1. Compared with the state-of-the-art UnSAM (Wang, Yang, and Darrell 2024) framework which adopts the discontinuous CutLER routine, S2-UniSeg effectively scales with dataset size. For class-agnostic instance segmentation, S2-UniSeg achieves **+2.4** AP and **+1.9** AR using the same amount of data (0.4M). The 2M pretrained version achieves the highest performance with **46.3** AR and **36.4** AP. Moreover, the SA-1B pretrained S2-UniSeg also sets new record on other three benchmarks.

Ablation Studies

We identify four main hyper-parameters for ablation studies, which are the spatial and feature similarity weights (ω_f, ω_s), UniAP threshold schedule $\{\tau_t\}_{t=1}^T$, the number of QuerySD projection head dimension, and the local crops number. We evaluate all ablations on the unsupervised class-aware instance segmentation on UVO_{val} dataset.

Feature and spatial similarity weights. As shown in Table. 3, without considering spatial similarity, i.e. setting $\omega_s = 0$, the performance drops significantly. By setting $\omega_s = 0.4$, our model achieves an increase of **+3.8** AP. This validates our design for the Identify step.

Time-varied UniAP thresholds. As shown in Table. 3, when we use a set of much lower thresholds, the perfor-



Figure 4: Comparison between U2Seg (Niu et al. 2024), CutLER (Wang et al. 2023a), UnSAM (Wang, Yang, and Darrell 2024), and S2-UniSeg (ours) on Unsupervised Class-agnostic Instance Segmentation. Each row from top to bottom is U2Seg, CutLER, UnSAM, and S2-UniSeg. Please refer to Appendix B and Appendix C for more comparisons.

ω_s, ω_f	AP	AR100	$\{\tau_t\}_{t=1}^T$	AP	AR100
(0.6, 0.4)	15.1	27.3	(0.9-0.1)	16.7	31.8
(0.5, 0.5)	15.7	28.4	(0.5-0.1)	15.5	26.3
(0.4, 0.6)	16.2	32.1	(0.8-0.4)	16.2	32.1
(0.0, 1.0)	12.4	25.4			

δ	AP	AR100	K	AP	AR100
2	16.2	32.1	128	13.6	27.4
4	15.9	31.6	512	16.2	32.1
6	16.0	32.3	1024	16.9	33.2

Table 3: Ablation studies for the spatial and feature similarity weights, threshold schedule, local crops number, and QuerySD projection dimension.

mance drops significantly. This is because a lower threshold (0.5) will make almost every edge to be coarsened. Instead, by using a more fine-grained set of thresholds (0.9,0.8,...,0.1), the UniAP layer can identify more fine-grained groups of different semantic hierarchies. However, this would cost much time since more UniAP layers are used. Instead, by setting an intermediate set of thresholds (0.8,...,0.4), our model can have comparable performance and also cost less time.

Local crops number. As shown in Table. 3, different from DINO (Caron et al. 2021) which shows sensitive to multi-crops number. S2-UniSeg is more robust. Since more local crops mean more bipartite matching and each matching costs considerable time, we just crop 2 local views to save

computation.

QuerySD projection head dimension K . As shown in Table. 3, by using more clusters, our model can learn a finer representation granularity. This is also validated in U2Seg and previous self-supervised representation models. However, unlike DINO where a large number of 65536 is used, we find a smaller K is adequate for satisfactory self-supervised segmentation performance.

Limitation

UniAP involves a number of hyperparameters that may necessitate additional tuning. Further refinement of these parameters could improve its performance and expand its applicability to a broader set of use cases.

Conclusion

We propose the Fast Universal Agglomerative Pooling (UniAP), which can generate both universal and multi-granular masks of one image within tens of milliseconds. A novel segmentation-oriented pretext task, Query-wise Self-Distillation, is devised to train a student and a momentum teacher with single-stage online pretraining. S2-UniSeg achieves state-of-the-art performance on unsupervised zero-shot instance segmentation, semantic segmentation, and panoptic segmentation. Moreover, S2-UniSeg demonstrates evident scalability, yielding better performance as the training dataset size grows.

Acknowledgments

This work is supported by the Guangdong Science and Technology Department (No. 2024ZDZX2004), and Nansha Key Area Science and Technology Project (No. 2024ZD006). This work is also supported by Shanghai Artificial Intelligence Laboratory.

References

- Arica, S.; Rubin, O.; Gershov, S.; and Laufer, S. 2024. CuVLER: Enhanced Unsupervised Object Discoveries through Exhaustive Self-Supervised Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23105–23114.
- Asano, Y. M.; Rupprecht, C.; and Vedaldi, A. 2019. Self-labelling via simultaneous clustering and representation learning. *arXiv preprint arXiv:1911.05371*.
- Aydemir, G.; Xie, W.; and Guney, F. 2023. Self-supervised object-centric learning for videos. *Advances in Neural Information Processing Systems*, 36: 32879–32899.
- Baevski, A.; Hsu, W.-N.; Xu, Q.; Babu, A.; Gu, J.; and Auli, M. 2022. Data2vec: A general framework for self-supervised learning in speech, vision and language. In *International Conference on Machine Learning*, 1298–1312. PMLR.
- Bao, H.; Dong, L.; Piao, S.; and Wei, F. 2021. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*.
- Bardes, A.; Ponce, J.; and LeCun, Y. 2021. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*.
- Bengio, Y.; Courville, A.; and Vincent, P. 2013. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8): 1798–1828.
- Buterez, D.; Janet, J. P.; Kiddle, S. J.; Oglic, D.; and Liò, P. 2022. Graph neural networks with adaptive readouts. *Advances in Neural Information Processing Systems*, 35: 19746–19758.
- Caesar, H.; Uijlings, J.; and Ferrari, V. 2018. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1209–1218.
- Caron, M.; Misra, I.; Mairal, J.; Goyal, P.; Bojanowski, P.; and Joulin, A. 2020. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33: 9912–9924.
- Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9650–9660.
- Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.
- Chen, X.; Ding, M.; Wang, X.; Xin, Y.; Mo, S.; Wang, Y.; Han, S.; Luo, P.; Zeng, G.; and Wang, J. 2024. Context autoencoder for self-supervised representation learning. *International Journal of Computer Vision*, 132(1): 208–223.
- Chen, X.; and He, K. 2021. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 15750–15758.
- Chen, X.; Xie, S.; and He, K. 2021. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9640–9649.
- Chen, Y.; Liu, Y.; Jiang, D.; Zhang, X.; Dai, W.; Xiong, H.; and Tian, Q. 2022a. Sdae: Self-distilled masked autoencoder. In *European conference on computer vision*, 108–124. Springer.
- Chen, Z.; Duan, Y.; Wang, W.; He, J.; Lu, T.; Dai, J.; and Qiao, Y. 2022b. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*.
- Cheng, B.; Misra, I.; Schwing, A. G.; Kirillov, A.; and Girdhar, R. 2022. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1290–1299.
- Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; and Schiele, B. 2016. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3213–3223.
- Dong, X.; Bao, J.; Zhang, T.; Chen, D.; Zhang, W.; Yuan, L.; Chen, D.; Wen, F.; Yu, N.; and Guo, B. 2023. Peco: Perceptual codebook for bert pre-training of vision transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 552–560.
- Dosovitskiy, A. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.
- Dwivedi, V. P.; Joshi, C. K.; Luu, A. T.; Laurent, T.; Bengio, Y.; and Bresson, X. 2023. Benchmarking graph neural networks. *Journal of Machine Learning Research*, 24(43): 1–48.
- Gao, H.; Chen, Y.; and Ji, S. 2019. Learning graph pooling and hybrid convolutional operations for text representations. In *The world wide web conference*, 2743–2749.
- Gao, X.; Dai, W.; Li, C.; Xiong, H.; and Frossard, P. 2021. ipool—information-based pooling in hierarchical graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 33(9): 5032–5044.
- Grattarola, D.; Zambon, D.; Bianchi, F. M.; and Alippi, C. 2022. Understanding pooling in graph neural networks. *IEEE transactions on neural networks and learning systems*, 35(2): 2708–2718.
- Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.; Buchatskaya, E.; Doersch, C.; Avila Pires, B.; Guo, Z.; Gheshlaghi Azar, M.; et al. 2020. Bootstrap your own latent—a new approach to self-supervised learning. *Advances in neural information processing systems*, 33: 21271–21284.
- Hahn, O.; Reich, C.; Araslanov, N.; Cremers, D.; Rupprecht, C.; and Roth, S. 2025. Scene-Centric Unsupervised Panoptic Segmentation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 24485–24495.
- Hamilton, M.; Zhang, Z.; Hariharan, B.; Snively, N.; and Freeman, W. T. 2022. Unsupervised semantic segmentation by distilling feature correspondences. *arXiv preprint arXiv:2203.08414*.
- He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16000–16009.
- He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9729–9738.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.
- Kim, C.; Han, W.; Ju, D.; and Hwang, S. J. 2024. EAGLE: Eigen Aggregation Learning for Object-Centric Unsupervised Semantic Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3523–3533.

- Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4015–4026.
- Lee, D.; Kim, S.; Lee, S.; Park, C.; and Yu, H. 2021. Learnable structural semantic readout for graph classification. In *2021 IEEE International Conference on Data Mining (ICDM)*, 1180–1185. IEEE.
- Li, F.; Zhang, H.; Xu, H.; Liu, S.; Zhang, L.; Ni, L. M.; and Shum, H.-Y. 2023. Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3041–3050.
- Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*, 740–755. Springer.
- Liu, C.; Zhan, Y.; Wu, J.; Li, C.; Du, B.; Hu, W.; Liu, T.; and Tao, D. 2022. Graph pooling for graph neural networks: Progress, challenges, and opportunities. *arXiv preprint arXiv:2204.07321*.
- Liu, Y.; Yang, S.; Zhang, Y.; Miao, C.; Nie, Z.; and Zhang, J. 2021. Learning hierarchical review graph representations for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(1): 658–671.
- Liu, Y.; Zeng, J.; Tao, X.; and Fang, G. 2024. Rethinking Self-Supervised Semantic Segmentation: Achieving End-to-End Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Mesquita, D.; Souza, A.; and Kaski, S. 2020. Rethinking pooling in graph neural networks. *Advances in Neural Information Processing Systems*, 33: 2220–2231.
- Niu, D.; Wang, X.; Han, X.; Lian, L.; Herzig, R.; and Darrell, T. 2024. Unsupervised universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 22744–22754.
- Noutahi, E.; Beaini, D.; Horwood, J.; Giguère, S.; and Tossou, P. 2019. Towards interpretable sparse graph representation learning with laplacian pooling. *arXiv preprint arXiv:1905.11577*.
- Oquab, M.; Darcet, T.; Moutakanni, T.; Vo, H.; Szafraniec, M.; Khalidov, V.; Fernandez, P.; Haziza, D.; Massa, F.; El-Nouby, A.; et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.
- Pearce, D. J. 2005. An improved algorithm for finding the strongly connected components of a directed graph. *Victoria University, Wellington, NZ, Tech. Rep.*
- Peng, Z.; Dong, L.; Bao, H.; Ye, Q.; and Wei, F. 2022. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*.
- Ravi, N.; Gabeur, V.; Hu, Y.-T.; Hu, R.; Ryal, C.; Ma, T.; Khedr, H.; Rädle, R.; Rolland, C.; Gustafson, L.; et al. 2024. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*.
- Seong, H. S.; Moon, W.; Lee, S.; and Heo, J.-P. 2023. Leveraging hidden positives for unsupervised semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 19540–19549.
- Seong, H. S.; Moon, W.; Lee, S.; and Heo, J.-P. 2024. Progressive Proxy Anchor Propagation for Unsupervised Semantic Segmentation. *arXiv preprint arXiv:2407.12463*.
- Siméoni, O.; Puy, G.; Vo, H. V.; Roburin, S.; Gidaris, S.; Bursuc, A.; Pérez, P.; Marlet, R.; and Ponce, J. 2021. Localizing objects with self-supervised transformers and no labels. *arXiv preprint arXiv:2109.14279*.
- Van Gansbeke, W.; Vandenhende, S.; and Van Gool, L. 2022. Discovering object masks with transformers for unsupervised semantic segmentation. *arXiv preprint arXiv:2206.06363*.
- Vo, H. V.; Pérez, P.; and Ponce, J. 2020. Toward unsupervised, multi-object discovery in large-scale image collections. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16*, 779–795. Springer.
- Vo, V. H.; Sizikova, E.; Schmid, C.; Pérez, P.; and Ponce, J. 2021. Large-scale unsupervised object discovery. *Advances in Neural Information Processing Systems*, 34: 16764–16778.
- Wang, W.; Feiszli, M.; Wang, H.; and Tran, D. 2021. Unidentified video objects: A benchmark for dense, open-world segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 10776–10785.
- Wang, X.; Girdhar, R.; Yu, S. X.; and Misra, I. 2023a. Cut and learn for unsupervised object detection and instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3124–3134.
- Wang, X.; Yang, J.; and Darrell, T. 2024. Segment Anything without Supervision. *arXiv preprint arXiv:2406.20081*.
- Wang, X.; Yu, Z.; De Mello, S.; Kautz, J.; Anandkumar, A.; Shen, C.; and Alvarez, J. M. 2022. Freesolo: Learning to segment objects without annotations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14176–14186.
- Wang, Y.; Shen, X.; Yuan, Y.; Du, Y.; Li, M.; Hu, S. X.; Crowley, J. L.; and Vafreydaz, D. 2023b. Tokencut: Segmenting objects in images and videos with self-supervised transformer and normalized cut. *IEEE transactions on pattern analysis and machine intelligence*.
- Wei, C.; Fan, H.; Xie, S.; Wu, C.-Y.; Yuille, A.; and Feichtenhofer, C. 2022. Masked feature prediction for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14668–14678.
- Wu, J.; He, J.; and Xu, J. 2019. Net: Degree-specific graph neural networks for node and graph classification. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 406–415.
- Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J. M.; and Luo, P. 2021. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34: 12077–12090.
- Xu, K.; Zhang, M.; Li, J.; Du, S. S.; Kawarabayashi, K.-i.; and Jegelka, S. 2020. How neural networks extrapolate: From feedforward to graph neural networks. *arXiv preprint arXiv:2009.11848*.
- Zbontar, J.; Jing, L.; Misra, I.; LeCun, Y.; and Deny, S. 2021. Barlow twins: Self-supervised learning via redundancy reduction. In *International conference on machine learning*, 12310–12320. PMLR.
- Zhou, Q.; Yu, C.; Luo, H.; Wang, Z.; and Li, H. 2022. Mimco: Masked image modeling pre-training with contrastive teacher. In *Proceedings of the 30th ACM International Conference on Multimedia*, 4487–4495.
- Zou, X.; Yang, J.; Zhang, H.; Li, F.; Li, L.; Wang, J.; Wang, L.; Gao, J.; and Lee, Y. J. 2024. Segment everything everywhere all at once. *Advances in Neural Information Processing Systems*, 36.