DIFFUSE2SEG: DIFFUSION MODELS CAN SEGMENT ANYTHING WITHOUT SUPERVISION

Anonymous authors
Paper under double-blind review

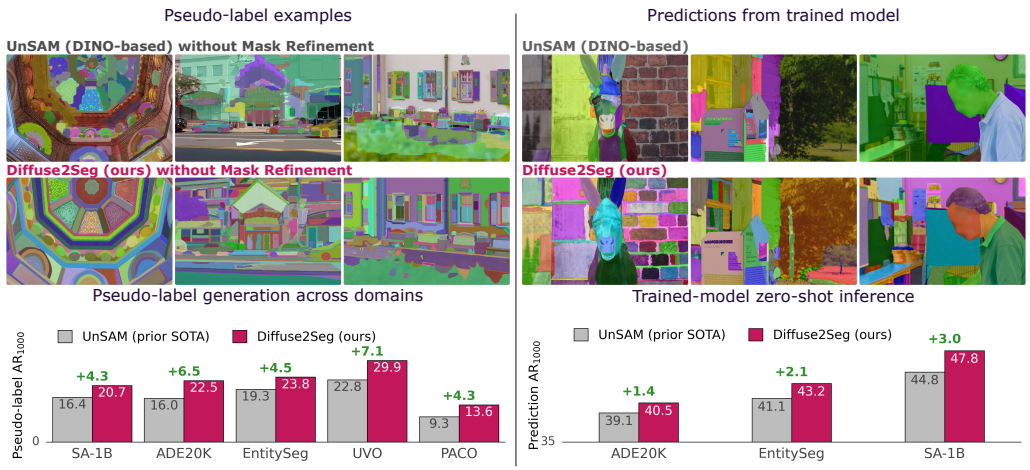


Figure 1: **Diffuse2Seg repurposes diffusion models to generate multi-granular instance masks without supervision.** Left: Our pseudo-labels are smoother and cover the image more densely than those of the DINO-based UnSAM (both shown without mask refinement), improving pseudo-label AR_{1000} by 4.3–7.1 p.p. across five domains. Right: A segmentation model trained on these pseudo-labels outperforms its UnSAM-trained counterpart in zero-shot AR_{1000} on densely labeled entity segmentation benchmarks, by 1.4–3.0 p.p.

ABSTRACT

Open-world entity segmentation aims to predict masks for arbitrary visual entities across domains and at multiple granularities, from parts to whole objects. In this setting, SAM sets a strong standard: trained on SA-1B, comprising 11M images and over 1B carefully annotated masks, it achieves remarkable zero-shot performance. Collecting such labels is expensive and time-consuming, however, which limits how far this recipe can scale. Text-to-image diffusion models offer a way around this. Their intermediate features transfer well across perception tasks, and since object structure emerges as the model denoises a noise sample into an image conditioned on a text prompt, that structure is already encoded in these representations. It should therefore be possible to exploit them for open-world entity segmentation *without retraining or supervision*. To realize this, we present *Diffuse2Seg*, which repurposes generative diffusion models for automatic mask generation. This is achieved by propagating a grid of point prompts through their self-attention representations in an edge-preserving manner. Diffuse2Seg produces multi-granular instance masks and outperforms prior state-of-the-art label generators by 4.3-7.1 p.p. in AR_{1000} across five domains. Training an instance segmentation model on these generated masks advances detector-free open-world segmentation by 7.4 and 7.7 p.p. on "things" and "stuff+things" datasets and surpasses the detector-based UnSAM on "stuff+things" by 2.1 p.p. in AR_{1000}. Finally, we show that a model trained on Diffuse2Seg labels provides a strong initialization for semi-supervised learning, matching its fully supervised counterpart with only 5k labeled images and outperforming it with 10k. Code will be made publicly available.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Segmenting *anything* involves partitioning an image into all coherent visual entities, including countable "thing" objects such as persons and cars and amorphous "stuff" regions such as sky and road, without restriction to a fixed set of class labels (Qi et al., 2022). This approach enables vision systems to operate in open-world settings without retraining. SAM (Kirillov et al., 2023) demonstrated this capability by training over multiple rounds on SA-1B, a large dataset of class-agnostic masks that span the hierarchy from part to whole object. However, densely annotating such a dataset is costly and limits scalability, increasing demand for multi-granular labels for things, stuff, and parts that can be obtained without manual supervision.

Existing work meets this demand with unsupervised mask generation from self-supervised (SSL) Vision Transformers (ViTs) (Wang et al., 2023; Arica et al., 2024; Wang et al., 2024; Sick et al., 2025; Yang et al., 2025). However, most of these methods are designed for foreground instances, and only a few works target truly dense entity annotation (Cao et al., 2024; Wang et al., 2024). Dense annotation requires representations that capture the entire scene, not only its salient objects. Text-to-image diffusion models are a promising source of such representations: they have emerged as universal representation learners (Gabeur et al., 2026; Wang et al., 2026), and learning to synthesize images leads them to encode global scene composition, semantic relationships, and fine-grained spatial layout. When a diffusion model generates entities such as sky or cars from a prompt, these entities should be extractable from its internal representations. Prior work has already exploited diffusion features for unsupervised semantic segmentation at various granularities (Tian et al., 2024; Couairon et al., 2024; Namekata et al., 2024; Kim et al., 2025; Zhang et al., 2026; Nakhli et al., 2026), interactive segmentation (Karmann & Urfalioglu, 2025; Karmann et al., 2025), and for enhancing SSL-based instance segmentors (Jo et al., 2026). Yet none of these works address the more complex task of extracting masks for all visual entities in an image. To fill this gap, we propose *Diffuse2Seg*, which repurposes a frozen diffusion model to generate class-agnostic multi-granularity instance labels without segment annotations. Specifically, for a given image, we first extract a denoising-based self-attention affinity matrix. Next, we propagate a dense grid of point prompts over the affinity matrix with non-linear p-Laplacian smoothing, merge the resulting soft object maps into segmentation maps at several granularities, and deduplicate the candidates into instance masks. Finally, we show that these labels are an effective supervision signal for training a lightweight segmentation model with strong zero-shot generalization. Moreover, the resulting model supports data-efficient semi-supervised learning through decoder-only fine-tuning. In summary, we make three contributions:

(i) Training-free multi-granularity instance segmentation: We show that a frozen diffusion model can act as an automatic mask generator. By propagating a grid of point prompts through diffusion-model self-attention, we recover dense instance masks at multiple granularities, entirely without supervision. Our generated labels surpass the previous unsupervised state-of-the-art UnSAM by **4.3 - 7.1** p.p. in AR_{1000} across a diverse set of five domains, as illustrated in Figure 1.

(ii) State-of-the-art pseudo-label training: We propose training a lightweight instance segmentation model solely on our diffusion-based pseudo-labels, outperforming the strongest detector-free baseline SOHES by **+7.7/+7.4/+10.0** p.p. AR on stuff+things/things/parts datasets, at half of the training image budget and using a much smaller backbone and improve on UnSAM in "stuff+things" by **+2.1** p.p. in AR in a detector-augmented setting.

(iii) Semi-supervised Diffuse2Seg-Semi: Fine-tuning only the decoder of our unsupervised model recovers 93.5% of a 100k-image fully-supervised reference with only 100 labeled images and surpasses it by +1.4 p.p. with 10k, staying ahead of UnSAM under the same protocol.

2 RELATED WORK

Unsupervised Instance Segmentation Early SSL-based methods like TokenCut (Wang et al., 2022), LOST (Siméoni et al., 2021), FOUND (Siméoni et al., 2023) and MOST (Rambhatla et al., 2023) partition feature graphs for foreground masks. CutLER (Wang et al., 2023) extends to multi-object segmentation via self-training, whereas CuVLER (Arica et al., 2024) improves robustness via DINO ensembling. Later methods add prompt-based similarities (Li & Shin, 2024), depth (Sick et al., 2025), or boundary reasoning (Yang et al., 2025). In contrast, HASSOD (Cao et al., 2023) generates

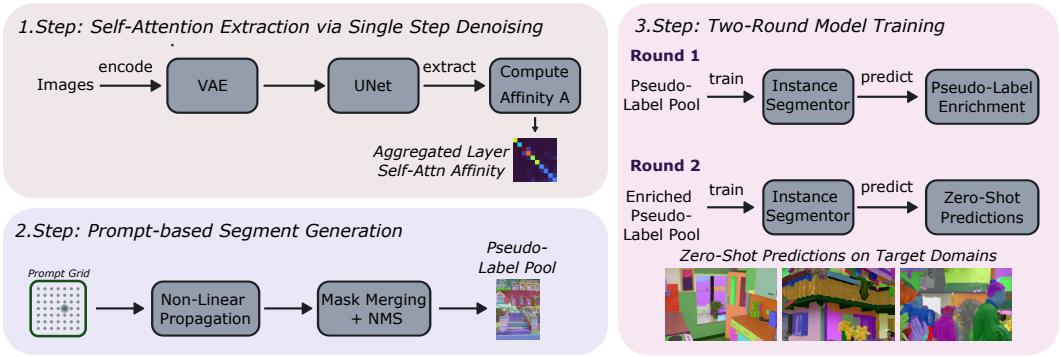


Figure 2: **Schematic depiction of our unsupervised Diffuse2Seg method.** Step 1 aggregates the multi-layer self-attention of a single StableDiffusion2 (SD2) denoising step into one affinity matrix $\mathbf{A}$. Step 2 propagates a grid of equidistant one-hot prompts over $\mathbf{A}$ with non-linear p-Laplacian propagation and merges the resulting soft object maps into multi-granular pseudo-labels. Step 3 trains a lightweight Mask2Former on those labels, followed by another round of self-training on crop-based prediction refinement.

hierarchical masks via multi-threshold agglomerative clustering, refined by SOHES (Cao et al., 2024). The SOTA UnSAM (Wang et al., 2024) combines CutLER proposals with HASSOD-based multi-granularity clustering.

Training-Free Segmentation with Generative Models. In addition to SSL-based representations, diffusion-based features were shown to also segment without retraining, via image-text cross-attention for open-vocabulary settings (Kim et al., 2025; Helbling et al., 2025; Sun et al., 2026; Meng et al., 2026) or via internal features for the class-agnostic setting we target (Ivanova et al., 2026). DiffSeg (Tian et al., 2024) merges self-attention into semantic clusters, DiffCut (Couairon et al., 2024) and Falcon (Zhang et al., 2026) apply NCut variants to feature affinities, TRACE (Jo et al., 2026) uses edges to post-process masks, and M2N2 (Karmann & Urfalioglu, 2025) converts interactive clicks into masks. Our work, Diffuse2Seg, however, differs from these approaches by leveraging generative models for automatic mask generation to obtain multi-granular instance masks with dense coverage. Moreover, we address limitations of existing works, such as requiring user input and focusing on a single level of granularity (Karmann & Urfalioglu, 2025).

3 Diffuse2Seg: Diffusion-Based Instance Segmentation Without Supervision

This section presents Diffuse2Seg. Sec. 3.1 summarizes its three-step pipeline, and Sec. 3.2 to Sec. 3.4 detail self-attention extraction, prompt-based segment generation, and two-round student training. Preliminaries on entity segmentation and diffusion-based feature extraction are given in Appendix Sec. A.

3.1 Overview of Diffuse2Seg

Diffuse2Seg (Figure 2) extends the ideas of SAM and generative diffusion models to label images densely without supervision. It mimics SAM's mask generation and self-training in an entirely unsupervised way. The method has three main steps: 1) We extract self-attention from a pre-trained diffusion model to create an affinity matrix. 2) We use the affinity matrix with point prompts to generate soft object maps by edge-stopping propagation, then merge masks and apply non-maximum suppression to create multi-granular pseudo-labels. We also refine the masks to improve their boundaries. 3) We train a segmentation network on the pseudo-labels in two rounds: initial training, then refinement using improved pseudo-labels.

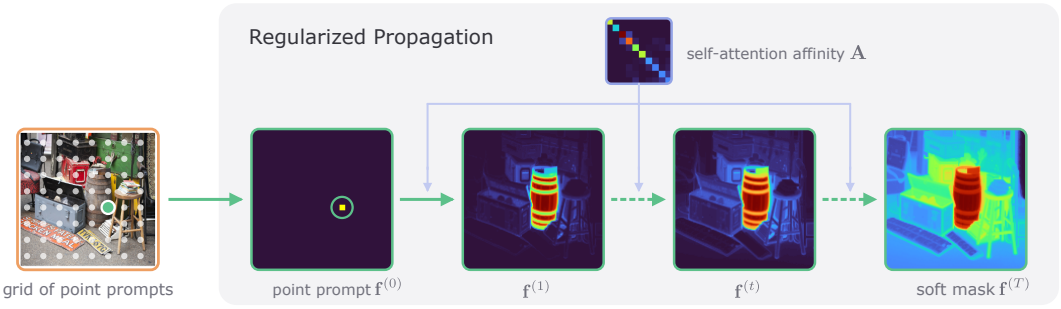


Figure 3: **Generation of soft object maps by prompt propagation.** An equidistant grid of one-hot point prompts is placed across the latent token grid ($H_f \times W_f$) as initial seeds (grey dots). Each prompt is propagated over the self-attention affinity graph $\mathbf{A}$ via edge-preserving non-linear p-Laplacian propagation (Equation (2)) until it reaches $\|\mathbf{f}^{t+1} - \mathbf{f}^t\|_2^2 \leq \tau_{\text{prop}}$, yielding continuous soft object maps. In this example, we show the propagation of a single prompt (green dot).

3.2 Step 1: Self-Attention Extraction via Single-Step Denoising

For our pseudo-label generation pipeline, we start by extracting self-attention matrices from the StableDiffusion2 (SD2) UNet (Rombach et al., 2022). To do so, we encode an input image using the SD2 VAE encoder to obtain its latent representation $\mathbf{z}$, run a single step of denoising and cache the obtained self-attention matrices. Specifically, for each layer $l \in \{1, ..., n_L\}$ and attention head $h \in \{1, ..., n_H\}$, we store one $N \times N$ matrix $\mathbf{A}^{(l,h)}$ that represents the affinity between the N patch tokens. Following Karmann & Urfalioglu (2025), we use only the self-attention tensors from the first two layers at the highest resolution of the UNet decoder. We first aggregate each layer over its heads and then combine layers via weights w_l. The weights w_l sum to 1 and are treated as hyperparameters (see Sec. B.1). The aggregated self-attention matrix then takes the form

$$\mathbf{A} = \frac{1}{n_h} \sum_{l=1}^{n_L} w_l \sum_{h=1}^{n_H} \mathbf{A}^{(l,h)} \,. \tag{1}$$

Due to the row-wise softmax, this affinity is asymmetric. We therefore symmetrize it and also apply temperature sharpening to amplify differences in affinity values (see Sec. B.1 for details).

3.3 Step 2: Prompt-Based Segment Generation

Self-attention in generative diffusion models concentrates on semantically similar regions and encodes object boundaries (Tian et al., 2024; Karmann & Urfalioglu, 2025; Erel et al., 2025; Jo et al., 2026; Zhang & Zhang, 2026), which are two useful properties for isolating instances. However, simply thresholding the affinity maps, as done in Li & Shin (2024), is insufficient because the self-attention matrices are sparse. Instead, we treat the aggregated affinity matrix $\mathbf{A}$ as a graph for prompt propagation. This yields dense and coherent object maps as shown in the following.

Prompt Propagation We utilize the non-local graph regularization of Elmoataz et al. (2008), which adaptively smooths an input signal while preserving edges. In our case, these discontinuities represent the attention differences on object edges. The smoothing process is controlled by the affinity extracted in step 1. It enforces neighboring tokens to take similar values for high-affinity neighbors while throttling the smoothing effect for low-affinity neighbors. This amplifies propagation within objects and limits it at edges. To segment the whole image, we generate a grid of equidistant prompts with fixed spacing on the latent representation. Each prompt covers a different image region and is represented by a one-hot vector $\mathbf{f}^0 \in \mathbb{R}_+^N$. Then we smooth these prompts into soft object maps $\mathbf{f} \in \mathbb{R}_+^N$ by minimizing

$$E_A^p(\mathbf{f}, \mathbf{f}^0, \lambda) \;=\; \frac{1}{p} \sum_{i=1}^{N} \left(\sum_{j=1}^{N} A_{ij} (f_j - f_i)^2 \right)^{p/2} \;+\; \frac{\lambda}{2} ||\mathbf{f} - \mathbf{f}^0||_2^2 \tag{2}$$

with respect to $\mathbf{f}$. This is done for all prompts from the grid independently and in parallel (see Figure 3 for a propagation example). The first term smooths a prompt $\mathbf{f}$ using the affinity matrix, and the second term regularizes the smoothing by anchoring it to the initial prompt. The exponent p controls how strongly discontinuities are preserved, where $p = 2$ is equivalent to a linear propagation, and λ governs the anchoring to the initial prompt. Following Elmoataz et al. (2008), we solve this optimization problem efficiently with Gauss-Jacobi iterations (see Sec. B.1 for details) and stop when the squared norm of the iteration update $\|\mathbf{f}^{t+1} - \mathbf{f}^t\|_2^2$ falls below τ_{prop}, producing our final set of soft object maps. Our propagation differs from the interactive segmentation in Karmann & Urfalioglu (2025), where each user click is turned into a discrete map of iteration counts, from which a flood-fill and a truncated nearest-neighbor rule extract a single segment per click. We instead keep the maps continuous and do not reduce each one to a segment; comparing and clustering them across prompts lets us infer multiple granularities.

Map Merging These soft object maps are often redundant or capture the same object at different granularities. Thus, we consolidate them into a set of complementary multi-granularity masks with the following steps: First, we normalize the maps into valid probability distributions $\mathbf{p}_k = \mathbf{f}_k / \sum_i f_{k,i}$. We then measure the dissimilarity of a prompt k with another prompt k' using the symmetric Kullback-Leibler (KL) divergence

$$d_{\text{KL}} = \frac{1}{2}\Big(D_{\text{KL}}\big(\mathbf{p}_k \,\|\, \mathbf{p}_{k'}\big) + D_{\text{KL}}\big(\mathbf{p}_{k'} \,\|\, \mathbf{p}_k\big) \Big) . \tag{3}$$

Based on these distances, we cluster the prompts with agglomerative average linkage clustering on the $K \times K$ matrix $\mathbf{D}$ of pairwise distances. Cutting the resulting clustering dendrogram at different threshold values h groups the prompts into clusters $\mathcal{C}^h$. Small thresholds h result in a larger number of clusters and are thus suited to isolate small objects, whereas larger h values yield fewer clusters and are thus favorable for isolating whole or coarse objects. In contrast to DiffSeg (Tian et al., 2024) which iteratively merges raw attention maps sampled from different locations with the same distance criterion, we infer multiple granularities from a single clustering run by setting different cut heights. To infer high-resolution instance masks from the merged clusters, we average all maps $\mathbf{p}_c$ belonging to a cluster C in the set of clusters $\{\mathcal{C}^h\}_h$ at a threshold h with $\bar{\mathbf{p}}_C = \sum_{c \in \mathcal{C}} \mathbf{p}_c / |C|$. Then we upsample the merged maps and perform argmax on them to obtain a cluster segmentation map $\mathbf{M}^h$ for each level h. Finally, we isolate instances out of the high-resolution segmentation by connected components (Bolelli et al., 2020). Consequently, we obtain connected regions forming our instance masks. To reduce noise, we filter out components with an area below $a_{\min}$ as noise (for pseudocode, see Sec. B.1). We repeat this procedure for all thresholds to add multi-granularity segmentation proposals to the candidate pool $\mathcal{P}$.

Mask Post-Processing Since adjacent clustering thresholds h produce similar segmentation masks, we remove redundant masks from the candidate pool $\mathcal{P}$ via a simple area-descending non-maximum suppression: candidate masks are sorted in descending order of pixel area, and a mask m is rejected if any already-accepted mask $m' \in \mathcal{M}$ satisfies $\text{IoU}(m, m') > \tau_{\text{IoU}}$, yielding at most $N_{\max} = 1000$ complementary masks per image. Like UnSAM (Wang et al., 2024) and SOHES (Cao et al., 2024), we employ CascadePSP (Cheng et al., 2020) as a refinement step on the pseudo-labels before training.

3.4 Step 3: Two-Round Model Training on Generated Segments

Because generating labels requires diffusion inference and still contains errors, we use the generated labels as supervision for a lightweight model trained on approximately 1%–4% of SA-1B for two consecutive rounds, as in SAM's multi-round training, letting the model improve label quality by self-training on its own predictions. Following UnSAM, we use Mask2Former (Cheng et al., 2022), optimized class-agnostically with $\mathcal{L} = \lambda_{\text{dice}}\mathcal{L}_{\text{dice}} + \lambda_{\text{bce}}\mathcal{L}_{\text{bce}} + \lambda_{\text{cls}}\mathcal{L}_{\text{cls}}$, where the first two terms optimize mask quality, $\mathcal{L}_{\text{cls}}$ learns an objectness probability, and $\lambda_{\text{dice}} = \lambda_{\text{bce}} = 5$, $\lambda_{\text{cls}} = 2$ (Sec. B.3). We present two configurations of this procedure: one using only the diffusion-based labels, and one that follows UnSAM by augmenting them with masks from a pre-trained CutLER segmentation model.

Round 1: Training on Generated Segments For the first round, both configurations take the same pool of previously generated segments as labels. As the latent resolution of SD2 UNet is limited, we lack some small-scale segments and thus need to increase the frequency and diversity of these

segments during training. This is done by applying copy-and-paste augmentation (Ghiasi et al., 2021). In our detector-augmented setting, we combine small-object pasting with predictions from CutLER, with a confidence above τ_{CutLER}, set to 0.1 throughout experiments. This ensures a sufficient number of labels across coarse instances, parts, and sub-parts. For further details on the training strategy see Appendix Sec. B.3.

Round 2: Training with Predict-and-Conquer Enrichment In the second round, we use the model trained in round 1 to enrich the pseudo-label pool. Round-1 labels are spatially dense, but they miss fine details because of the limited latent resolution of the SD2 UNet. Inspired by UnSAM's divide-and-conquer strategy, we therefore combine a full-image inference pass with a second pass on crops of previously predicted parent segments. We first run the round-1 model on the whole image and keep every prediction with a score of at least θ. From this set, we discard near-full-image predictions ($> a_{\mathrm{high}}$) and very small predictions ($< a_{\mathrm{low}}$), which are unlikely to contain smaller parts. A child is kept only if it is contained in its parent, as $\mathrm{area}(\mathrm{child} \cap \mathrm{parent})/\mathrm{area}(\mathrm{child}) \geq \tau_{\mathrm{cont}}$, so that we do not simply re-detect the parent. The surviving global and crop predictions $\hat{m}$ are then merged into the initial round-1 pool: any original mask m with $\mathrm{IoU}(m, \hat{m}) > \tau_{\mathrm{merge}}$ is replaced, and the remaining predictions are appended. This merge is the same for both training configurations. They differ only in whether CutLER masks are added afterward. In the detector-free setting, round 2 trains on the merged pool above (initial Diffuse2Seg labels plus predict-and-conquer predictions). In the detector-augmented setting, we additionally append CutLER masks to that pool and enable hierarchical copy-paste of CutLER instances during training. Crop refinement itself does not depend on CutLER.

4 Experiments

After describing the experimental setup in Sec. 4.1, we assess the quality of our training-free pseudo-labels against DINO- and diffusion-based labels in Sec. 4.2 and the zero-shot transfer of students trained on them in Sec. 4.3. We then study label-efficient semi-supervised fine-tuning in Sec. 4.4 and ablate the main design choices of our method in Sec. 4.5.

4.1 Experimental Setup

In this subsection, we first describe the datasets used to train and evaluate Diffuse2Seg. Then, we outline the SSL-based and diffusion-based benchmark methods against which we compare Diffuse2Seg. Finally, we present the details of pseudo-label generation and training and the metrics used to evaluate segment quality.

Datasets and Metrics We generate pseudo-labels on random SA-1B subsets of 100k, 200k and 400k images (1%, 2% and 4% of SA-1B) and hold out 1000 further images for validation. For zero-shot evaluation we add ADE20K (Zhou et al., 2017), EntitySeg (Qi et al., 2023) and UVO (Wang et al., 2021) as entity benchmarks, PACO (Ramanathan et al., 2023) for parts, and COCO/LVIS (Lin et al., 2014; Gupta et al., 2019) for "things". Compared to prior work, we intentionally do not evaluate on PartImageNet, as its images stem from ImageNet, the pre-training dataset of both DINO and CutLER. For evaluation of both pseudo-labels and trained model predictions we mainly consider AR_{1000} and recall per size AR_S, AR_M, and AR_L as the primary metrics, as object definitions vary across datasets. For further discussion on the metrics, see Sec. C.2.

Baselines We compare our label generation against the DINO-based entity segmenters SOHES (Cao et al., 2024) and UnSAM (Wang et al., 2024), against the diffusion-based DiffSeg (Tian et al., 2024) and M2N2 (Karmann & Urfalioglu, 2025), which we adapt to our protocol, and against the trained CutLER (Wang et al., 2023) as a reference; Sec. C.1 details the adaptations.

Pseudo-Label Settings We extract from SD2 with a single denoising step at $t = 150$ following Karmann & Urfalioglu (2025) at an input resolution of 1120×1120. Prompts are equidistant with a spacing of six latent tokens and propagate with $p = 1.6$; merging uses $L = 6$ log-spaced thresholds and $\tau_{\mathrm{IoU}} = 0.9$. All generator parameters are selected on the held-out SA-1B split and frozen across transfer datasets (see analyses in Secs. B.1 and B.2).

Method	Backbone	SA-1B				AR_{1000} across domains			
		AR_S	AR_M	AR_L	AR_{1000}	ADE20K	EntitySeg	UVO	PACO
Training-based labeling									
CutLER (Wang et al., 2023)	DINO R50	5.4	14.0	28.0	17.0	24.8	22.1	32.3	10.7
DINO-based labeling									
SOHES (Cao et al., 2024)	DINO ViT-B/8	2.8	10.6	15.7	11.0	11.6	11.6	19.3	5.2
UnSAM (Wang et al., 2024)	DINO ViT-B/8	**5.7**	13.9	26.0	16.4	16.0	19.3	22.8	9.3
Diffusion-based labeling									
DiffSeg (Tian et al., 2024)	SD2	0.3	12.4	29.9	16.0	14.1	17.5	18.9	9.8
M2N2 (Karmann & Urfalioglu, 2025)	SD2	0.0	2.8	26.5	9.7	18.9	19.7	29.7	9.6
Diffuse2Seg (**ours**)	SD2	2.6	**15.3**	**38.9**	**20.7**	**22.5**	**23.8**	**29.9**	**13.6**

Table 1: **Pseudo-label quality on SA-1B and across domains.** Class-agnostic mask average recall with 1000 proposals per image, without mask refinement. Each method retains its own SA-1B settings across datasets; CutLER is a trained reference and exceeds our recall on ADE20K and UVO.

Training Settings We train Mask2Former with a DINO pre-trained ResNet-50 backbone using AdamW with a learning rate of 5×10^{-5}, a batch size of 16 and a weight decay of 0.05. Both rounds train for about eight epochs. Round 2 always fine-tunes on 0.1M crop-refined images (Sec. B.3).

4.2 Evaluation of Generated Pseudo-Labels

Table 1 compares our pseudo-labels against the baselines from Sec. 4.1 on SA-1B and four additional domains, with identical hyperparameters throughout and without mask refinement. Labeling high-resolution SA-1B images is challenging because most methods work on far lower-resolution features. Nevertheless, our approach reaches the highest recall on SA-1B, 4.7 p.p. above the strongest diffusion-based baseline DiffSeg (Tian et al., 2024) and 4.3 p.p. above the prior DINO-based state-of-the-art UnSAM (Wang et al., 2024), and it transfers best across domains, gaining 6.5 p.p. on ADE20K and 4.5 p.p. on EntitySeg over UnSAM and 2.9 p.p. on PACO over the trained CutLER. UnSAM retains an advantage on small objects (5.7 vs. 2.6 AR_S) due to its divide-and-conquer procedure, while Diffuse2Seg improves medium- and large-object recall. This deficit does not carry over to the trained models (see Table 6). Moreover, we report results with refinement in the appendix (Sec. D.2), lifting both methods (UnSAM $16.4 \rightarrow 23.8$, Diffuse2Seg $20.7 \rightarrow 29.2$) while maintaining a similar margin.

4.3 Evaluation of Zero-Shot Instance Segmentation

We now evaluate the final Diffuse2Seg instance segmentation models trained with pseudo-label supervision and compare them with unsupervised segmentation baselines trained either with or without detector supervision (Table 2). All models are optimized on SA-1B and inferred zero-shot on previously unseen datasets. When comparing Diffuse2Seg to SOHES, the strongest detector-free baseline, our method provides large gains on every benchmark: +7.4 p.p. AR on things datasets (37.2 vs. 29.8), +7.7 p.p. on stuff+things (40.3 vs. 32.6), and +10.0 p.p. on parts benchmarks (27.1 vs. 17.1). These results show that our diffusion-based approach is significantly more data-efficient in training, requiring only half the training images and a much smaller image encoder (ResNet50 for Diffuse2Seg vs. ViT-B for SOHES). For training with detector supervision, we compare to the prior state-of-the-art technique UnSAM, which bases its label generation on ranked proposals from a pre-trained CutLER object detector. If we augment our segment label set with the same proposals, we consistently surpass UnSAM on stuff+things datasets. Across all training image budgets ranging from 1% over 2% to 4% of the SA-1B dataset, we improve the mAR by +3.4 p.p., 2.6 p.p., and 2.1 p.p., respectively. On the things and parts datasets, we perform on par with UnSAM, which has an advantage on things-focused datasets due to its deep integration and learned proposal ranking of the CutLER detector. As expected, a gap to supervised SAM remains, as SAM was trained on many more images (11M) and for many more rounds. Nevertheless, our method approaches SAM on generalizable stuff+things segmentation, being just 2.7 p.p. worse on EntitySeg.

Method	Detector Supervision	# Images	Things			Stuff + Things				Parts
			COCO	LVIS	Avg.	ADE	Entity	SA-1B	Avg.	PACO
Supervised										
SAM	✗	11M	49.6	46.1	47.9	45.8	45.9	60.8	50.8	18.1
Unsupervised - label generation without a pre-trained detector										
CutLER	✗	1.3M	28.1	20.2	24.2	26.3	23.1	17.0	22.1	8.9
SOHES	✗	0.2M	30.5	29.1	29.8	31.1	33.5	33.3	32.6	17.1
Diffuse2Seg (**ours**)	✗	0.1M	**37.4**	**36.9**	**37.2**	**37.6**	**39.7**	**43.5**	**40.3**	**27.1**
Unsupervised - label generation with a pre-trained detector (CutLER proposals)										
UnSAM	✓	0.1M	40.5	37.7	39.1	35.7	39.6	41.9	39.1	27.5
Diffuse2Seg (**ours**)	✓	0.1M	**41.0**	**38.7**	**39.9**	**39.8**	**42.2**	**45.5**	**42.5**	**28.9**
UnSAM	✓	0.2M	**41.2**	**39.7**	**40.5**	36.8	40.3	43.6	40.2	29.1
Diffuse2Seg (**ours**)	✓	0.2M	40.7	39.0	39.9	**39.6**	**42.2**	**46.6**	**42.8**	29.1
UnSAM	✓	0.4M	42.0	**40.5**	**41.3**	39.1	41.1	44.8	41.7	29.7
Diffuse2Seg (**ours**)	✓	0.4M	**42.2**	40.0	41.1	**40.5**	**43.2**	**47.8**	**43.8**	**29.8**

Table 2: **Comparison of instance segmentation methods with and without auxiliary detector supervision.** While the previous state-of-the-art method UnSAM relies on an ImageNet pre-trained detector, our method can also be trained without auxiliary supervision purely on diffusion-model-based labels. Combining our method with CutLER supervision positions it as complementary to the object-biased labels of UnSAM. In fact, dense supervision leads to significant improvements on stuff+things datasets while performing on par on things-only datasets.

4.4 Semi-Supervised Diffuse2Seg-Semi

Unsupervised segmentation avoids costly labeling, yet in practice annotating a small set of images is often feasible. This raises the question of whether our pseudo-labels remain valuable in this setting, as an initialization for label-efficient fine-tuning. Therefore, we study semi-supervised learning by fine-tuning our best unsupervised model and UnSAM's best unsupervised model with a simple recipe, as indicated in Figure 4. We freeze the encoder to preserve generalization and unfreeze the decoder to align its mask predictions with the ground truth. As a strong supervised reference, we additionally train the same UnSAM recipe from scratch on 100k labeled samples. This fully supervised model reaches 51.1 mAR, exceeding the SAM reference at 44.4 mAR, because its design is tailored towards automatic mask generation. With 100 labeled images, fine-tuning recovers 93.5% of the supervised reference, compared to 90.5% for UnSAM; 5k images reach the reference level, and 10k exceed it by 1.4 p.p. in the mean of AR_{1000} over the six benchmarks of Table 2.

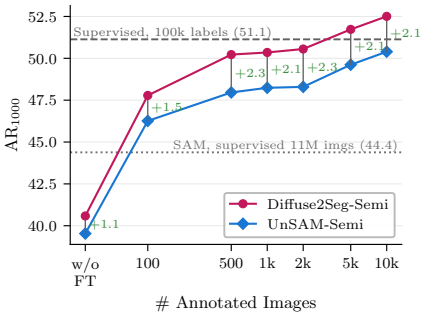
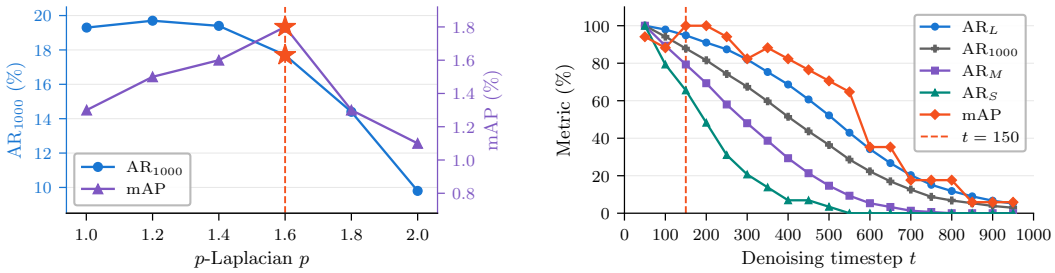


Figure 4: **Semi-supervised extensions of UnSAM and Diffuse2Seg (3 seeds avg.).** Mean average recall vs. the number of labeled fine-tuning images. 5k images suffice for Diffuse2Seg to match the 100k supervised UnSAM reference.

4.5 Ablation Study

For affinity extraction, Figure 5b shows that Diffuse2Seg's recall degrades across timesteps on the SA-1B validation set. This is because earlier time steps focus more on fine texture, which favors our objective of segmenting objects at varying granularities, whereas later steps capture coarse semantics. We select $t = 150$ for our experiments, which maximizes mAP while keeping strong mAR. For propagation, we show different p values in Figure 5a from linear propagation ($p = 2$) to increasingly edge-preserving smoothing ($p < 2$) on the SA-1B validation set. We show that non-linear propagation



(a) Influence of propagation exponent p.

(b) Behavior of the timestep t for extraction.

Figure 5: **Impact of propagation exponent and denoising timestep on Diffuse2Seg performance**. Panel (a): AR_{1000} peaks at $p = 1.2$, while mAP peaks at the selected $p = 1.6$. Panel (b): each metric is normalized to its own best timestep; recall, particularly for small objects, degrades earlier than mAP. Dashed vertical lines mark the selected $p = 1.6$ and $t = 150$.

Configuration	Things		Stuff + Things			Parts
	COCO	LVIS	ADE20K	EntitySeg	SA-1B	PACO
Diffuse2Seg labels	35.6	35.0	36.0	38.2	39.8	25.8
+ Crop labels	37.4	36.9	37.6	39.7	43.5	27.1
+ CutLER mixing	41.0	38.7	39.8	42.2	45.5	28.9

(a) **Impact of the training supervision on mAR.** Starting from global Diffuse2Seg pseudo-labels, the rows add crop-refined child masks and CutLER masks, respectively.

Supervision	Stage	# Images	AR_{1000}		
			Things	Stuff+Things	Parts
CutLER mixing	Stage 1	0.1M	37.9	40.3	27.0
	Stage 2	0.1M	38.7	40.7	27.1
+ Crop labels	Stage 2	0.1M	39.9	42.5	28.9

(b) **Cumulative gains of Self-training strategies.** Self-training with our presented strategies consistently improves results.

Table 3: **Impact of label sources and training stages.** Crop refinement adds locally re-inferred, parent-contained child masks; self-training corrects and fills gaps of the initial pseudo-labels.

in our method vastly improves mAR compared to linear propagation. While values closer to $p = 1.2$ improve recall and isolate more objects, they also lower mAP and introduce noisy masks. For our experiments, we prioritize mask precision and thus select $p = 1.6$, where mAP peaks.

Training Supervision Crop-refined masks from the second round improve fine structures (Table 3a; +3.7 p.p. on SA-1B, +1.3 p.p. on PACO), whereas CutLER mixing helps most on "things" datasets (+3.6 p.p. on COCO) and, via coarse-instance coverage, on "stuff+things" datasets (*e.g.,* +2.5 p.p. AR_{1000} on EntitySeg). Table 3b shows that our initial model struggles with visual details, as its pseudo-labels stem from a low-resolution latent. A second self-training round following UnSAM, which merges only global predictions into the initial pseudo-labels, yields only marginal gains in our setup (stuff+things mAR $40.3 \rightarrow 40.7$), whereas our crop-based labels effectively mitigate the missing details and boost stuff+things, things, and parts by +2.2, +2.0, and +1.9 p.p., respectively.

5 Conclusion

We proposed Diffuse2Seg, the first method to extract dense multi-granular instance pseudo-labels from pre-trained diffusion models without retraining or supervision. Our extensive experiments show not only the generality of pseudo-label generation but also the effectiveness of training segmentation models unsupervised using our pseudo-labels for strong zero-shot performance. In addition, we show that our models can serve as a strong initialization for semi-supervised learning, even surpassing a fully supervised reference with significantly fewer labeled images. As a limitation, our pseudo-labels struggle with particularly small objects that cannot be recognized at the latent resolution. Additionally, the diffusion representation may encode not only semantics but also texture and therefore sometimes oversegments larger segments in the image, such as the sky. Adopting coarse-to-fine approaches and objectness ranking, as in the SSL literature, can be practical solutions to address these issues. Finally, our propagation framework is flexible in prompts and in-model propagation, enabling future extensions such as class-conditioned cross-attention prompting or affinity extraction from generative video models to segment objects over time.

6 AI USE STATEMENT

In this work, we used generative AI tools (GitHub Copilot) to implement methods. Additionally, we employed generative AI tools (GitHub Copilot) to create and edit software code and to edit the manuscript for readability and grammar (general-purpose LLMs). We have not used generative AI tools for any other task requiring disclosure. We reviewed all AI-assisted work: the authors verified and tested AI-assisted code for correctness, and checked all AI-assisted text edits to ensure no claim or citation was altered or introduced. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Shahaf Arica, Or Rubin, Sapir Gershov, and Shlomi Laufer. CuVLER: Enhanced unsupervised object discoveries through exhaustive self-supervised transformers. In *CVPR*, 2024.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT pre-training of image transformers. In *ICLR*, 2022.

Federico Bolelli, Stefano Allegretti, Lorenzo Baraldi, and Costantino Grana. Spaghetti labeling: Directed acyclic graphs for block-based connected components labeling. *IEEE Transactions on Image Processing*, 29:1999–2012, 2020.

Shengcao Cao, Dhiraj Joshi, Liangyan Gui, and Yu-Xiong Wang. HASSOD: Hierarchical adaptive self-supervised object detection. In *NeurIPS*, 2023.

Shengcao Cao, Jiuxiang Gu, Jason Kuen, Hao Tan, Ruiyi Zhang, Handong Zhao, Ani Nenkova, Liangyan Gui, Tong Sun, and Yu-Xiong Wang. SOHES: Self-supervised open-world hierarchical entity segmentation. In *ICLR*, 2024.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *NeurIPS*, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

Niccolò Cavagnero, Narges Norouzi, Gijs Dubbelman, and Daan de Geus. PMT: Plain mask transformer for image and video segmentation with frozen vision encoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2026.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 9620–9629. IEEE, 2021. doi: 10.1109/ICCV48922.2021.00950. URL https://doi.org/10.1109/ICCV48922.2021.00950.

Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022.

Ho Kei Cheng, Jihoon Chung, Yu-Wing Tai, and Chi-Keung Tang. CascadePSP: Toward class-agnostic and very high-resolution segmentation via global and local refinement. In *CVPR*, 2020.

Paul Couairon, Mustafa Shukor, Jean-Emmanuel Haugeard, Matthieu Cord, and Nicolas Thome. DiffCut: Catalyzing zero-shot semantic segmentation with diffusion features and recursive normalized cut. In *NeurIPS*, 2024.

Abderrahim Elmoataz, Olivier Lezoray, and Sébastien Bougleux. Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing. *IEEE Transactions on Image Processing*, 17(7):1047–1060, 2008. doi: 10.1109/TIP.2008.924284.

Yotam Erel, Olaf Dünkel, Rishabh Dabral, Vladislav Golyanik, Christian Theobalt, and Amit H Bermano. Attention (as discrete-time markov) chains. In *Advances in Neural Information Processing Systems*, 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *CVPR*, 2024.

Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. EVA-02: A visual representation for neon genesis. *Image and Vision Computing*, pp. 105171, 2024.

Valentin Gabeur, Shangbang Long, Songyou Peng, Paul Voigtlaender, Shuyang Sun, Yanan Bao, Karen Truong, Zhicheng Wang, Wenlei Zhou, Jonathan T Barron, et al. Image generators are generalist vision learners. *arXiv preprint arXiv:2604.20329*, 2026.

Golnaz Ghiasi, Yin Cui, A. Srinivas, Rui Qian, Tsung-Yi Lin, Ekin Dogus Cubuk, Quoc V. Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *CVPR*, 2021.

Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

Alec Helbling, Tuna Han Salih Meral, Benjamin Hoover, Pinar Yanardag, and Duen Horng Chau. ConceptAttention: Diffusion transformers learn highly interpretable features. In *ICML*, 2025.

Daniela Ivanova, Marco Aversa, Paul Henderson, and John Williamson. Unsupervised segmentation by diffusing, walking and cutting. In *WACV*, 2026.

Sanghyun Jo, Ziseok Lee, Wooyeol Lee, Jonghyun Choi, Jaesik Park, and Kyungsu Kim. TRACE: Your diffusion model is secretly an instance edge detector. In *ICLR*, 2026.

Markus Karmann and Onay Urfalioglu. Repurposing stable diffusion attention for training-free unsupervised interactive segmentation. In *CVPR*, June 2025.

Markus Karmann, Peng-Tao Jiang, Bo Li, and Onay Urfalioglu. M2n2v2: Multi-modal unsupervised and training-free interactive segmentation. *arXiv preprint arXiv:2503.16254*, 2025.

Tommie Kerssies, Niccolò Cavagnero, Alexander Hermans, Narges Norouzi, Giuseppe Averta, Bastian Leibe, Gijs Dubbelman, and Daan de Geus. Your ViT is secretly an image segmentation model. In *CVPR*, 2025.

Chaehyun Kim, Heeseong Shin, Eunbeen Hong, Heeji Yoon, Anurag Arnab, Paul Hongsuck Seo, Sunghwan Hong, and Seungryong Kim. Seg4Diff: Unveiling open-vocabulary segmentation in text-to-image diffusion transformers. In *NeurIPS*, 2025.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Dylan Li and Gyungin Shin. ProMerge: Prompt and merge for unsupervised instance segmentation. In *European Conference on Computer Vision (ECCV)*, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014.

Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. In *NeurIPS*, 2023.

Benyuan Meng, Qianqian Xu, Zitai Wang, Xiaochun Cao, and Qingming Huang. Not all diffusion model activations have been evaluated as discriminative features. In *NeurIPS*, 2024.

Benyuan Meng, Qianqian Xu, Zitai Wang, Xiaochun Cao, Longtao Huang, and Qingming Huang. Making training-free diffusion segmentors scale with the generative power. In *CVPR*, 2026.

Ramin Nakhli, Mahesh Ramachandran, and Luca Ballan. Unlocking diffusion hierarchies: Adaptive timestep selection for zero-shot segmentation, 2026. URL https://arxiv.org/abs/2606.15590.

Koichi Namekata, Amirmojtaba Sabour, Sanja Fidler, and Seung Wook Kim. EmerDiff: Emerging pixel-level semantic knowledge in diffusion models. In *ICLR*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024. URL https://openreview.net/forum?id=a68SUt6zFt.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024.

Lu Qi, Jason Kuen, Yi Wang, Jiuxiang Gu, Hengshuang Zhao, Philip Torr, Zhe Lin, and Jiaya Jia. Open world entity segmentation. *TPAMI*, 2022.

Lu Qi, Jason Kuen, Tiancheng Shen, Jiuxiang Gu, Weidong Guo, Jiaya Jia, Zhe Lin, and Ming-Hsuan Yang. High-quality entity segmentation. In *ICCV*, 2023.

Vignesh Ramanathan, Anmol Kalia, Vladan Petrovic, Yi Wen, Baixue Zheng, Baishan Guo, Rui Wang, Aaron Marquez, Rama Kovvuri, Abhishek Kadian, Amir Mousavi, Yiwen Song, Abhimanyu Dubey, and Dhruv Mahajan. PACO: Parts and attributes of common objects. In *CVPR*, 2023.

Sai Saketh Rambhatla, Ishan Misra, Rama Chellappa, and Abhinav Shrivastava. MOST: Multiple object localization with self-supervised transformers for object discovery. In *ICCV*, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022.

Leon Sick, Dominik Engel, Sebastian Hartwig, Pedro Hermosilla, and Timo Ropinski. CutS3D: Cutting semantics in 3D for 2D unsupervised instance segmentation. In *ICCV*, 2025.

Oriane Siméoni, Gilles Puy, Huy V. Vo, Simon Roburin, Spyros Gidaris, Andrei Bursuc, Patrick Pérez, Renaud Marlet, and Jean Ponce. Localizing objects with self-supervised transformers and no labels. In *BMVC*, 2021.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seung Eun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *Trans. Mach. Learn. Res.*, 2026. URL https://openreview.net/forum?id=2NlGyqNjns.

Oriane Siméoni, Chloé Sekkat, Gilles Puy, Antonin Vobecky, Éloi Zablocki, and Patrick Pérez. Unsupervised object localization: Observing the background to discover objects. In *CVPR*, 2023.

Nick Stracke, Stefan Andreas Baumann, Kolja Bauer, Frank Fundel, and Björn Ommer. CleanDIFT: Diffusion features without noise. In *CVPR*, 2025.

Lin Sun, Jiale Cao, Jin Xie, Fahad Shahbaz Khan, and Yanwei Pang. iSeg: An iterative refinement-based framework for training-free segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(8):9765–9781, 2026.

Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. In *NeurIPS*, 2023.

Junjiao Tian, Lavisha Aggarwal, Andrea Colaco, Zsolt Kira, and Mar Gonzalez-Franco. Diffuse, attend, and segment: Unsupervised zero-shot segmentation using stable diffusion. *CVPR*, 2024.

Letian Wang, Chuhan Zhang, Rishabh Kabra, Jasper Uijlings, Steven Waslander, Andrew Zisserman, Joao Carreira, Kaiming He, Misha Andriluka, Eduard Gabriel Bazavan, Andrei Zanfir, and Cristian Sminchisescu. Video generation models are general-purpose vision learners. In *ECCV*, 2026.

Weiyao Wang, Matt Feiszli, Heng Wang, and Du Tran. Unidentified video objects: A benchmark for dense, open-world segmentation. In *ICCV*, 2021.

Xudong Wang, Rohit Girdhar, Stella X Yu, and Ishan Misra. Cut and learn for unsupervised object detection and instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3124–3134, 2023.

Xudong Wang, Jingfeng Yang, and Trevor Darrell. Segment anything without supervision. In *NeurIPS*, 2024.

Yangtao Wang, Xi Shen, Shell Xu Hu, Yuan Yuan, James L. Crowley, and Dominique Vaufreydaz. Self-supervised transformers for unsupervised object discovery using normalized cut. In *CVPR*, 2022.

Yafei Yang, Zihui Zhang, and Bo Yang. unMORE: Unsupervised multi-object segmentation via center-boundary reasoning. *ICML*, 2025.

Jian Zhang and Zhijun Zhang. AnchorDiff: Training-free concept grounding for MM-DiTs via anchor-based graph propagation, 2026. URL `https://arxiv.org/abs/2605.26460`.

Xiao Zhang, David Yunis, and Michael Maire. Deciphering 'what' and 'where' visual pathways from spectral clustering of layer-distributed neural representations. In *CVPR*, pp. 4165–4175, June 2024.

Xiao Zhang, Xiangyu Han, Xiwen Lai, Yao Sun, Pei Zhang, Xia Liu, and Konrad Kording. Falcon: Fast proximal linearization of normalized cuts for unsupervised image segmentation. In *ICLR*, 2026.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ADE20K dataset. In *CVPR*, 2017.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. iBOT: Image BERT pre-training with online tokenizer. *International Conference on Learning Representations (ICLR)*, 2022.

A Further Background

Vision Foundation Models Self-supervised pre-training with contrastive learning (Caron et al., 2020; Chen et al., 2020; 2021), masked image modeling (He et al., 2022; Bao et al., 2022; Fang et al., 2024), and self-distillation (Caron et al., 2021; Zhou et al., 2022; Oquab et al., 2024; Siméoni et al., 2026) yields highly transferable features for dense perception (Oquab et al., 2024; Kerssies et al., 2025; Cavagnero et al., 2026). In parallel, generative models (Rombach et al., 2022; Esser et al., 2024; Podell et al., 2024) have appeared as a strong source of general features. Feature extraction by a single step of denoising yields semantically coherent representations that outperform strong SSL baselines (Tang et al., 2023; Meng et al., 2024; Stracke et al., 2025), with semantics carried by value-value and instance discrimination by query-key interactions (Zhang et al., 2024). Diffuse2Seg exploits this self-attention representation for spatial grouping. We build on the UNet-based SD2, since diffusion transformers downsample latents more aggressively in the denoising network ($2\times$ compared to SD2) and thus limit the discovery of small instances, though our method transfers to their affinities as well.

Segment Anything Kirillov et al. (2023) introduced a model and dataset based on the entity segmentation definition (Qi et al., 2022). Its main novelty is a flexible prompt interface that takes points or boxes as input to segment coherent instances. Consistent with this definition, the training dataset SA-1B contains all kinds of objects and object parts, allowing the model to generalize across different label definitions and granularities. SA-1B is built with a complex data engine in which the SAM model is repeatedly retrained across three stages: first, initial manual labeling, second semi-automatic labeling with human assistance to fill blind spots, and third automatic mask generation from a grid of points, followed by a deduplication and filtering stage to produce high-quality annotations. With our method Diffuse2Seg, we mirror the latter but in an purely unsupervised setting. We place an evenly spaced grid of point prompts on the input image, generate candidate masks from the diffusion model's self-attention, then aggregate, filter, and post-process them into a final set of pseudo-labels. Inspired by SAM, we then train a segmentation model in a two-round procedure on the collected pseudo-labels to fill gaps and improve performance.

Generative Diffusion Models These models learn complex data distributions through a two-directional process. Consider a data sample $\mathbf{x}_0 \sim p_{\text{data}}$. The forward process is a Markov chain, gradually injecting Gaussian noise at each step, $q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}\left(\sqrt{1-\beta_t}\,\mathbf{x}_{t-1},\, \beta_t \mathbf{I}\right)$, with $t = 1, \ldots, T$ and a chosen variance schedule β_t resulting in x_T being approximately Gaussian noise. A denoiser neural network $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$ then learns to generate new samples by inverting the process and predicting the reverse transitions, $p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t)$. The denoiser is shared across all timesteps. To scale this process to high-resolution data, variational autoencoders (VAEs) were introduced for efficiency. Such a VAE maps the input image I into a lower-dimensional latent representation $\mathbf{z} \in \mathbb{R}^{H_f \times W_f \times C}$ with an encoder, where $H_f \times W_f$ specifies the latent spatial resolution and C the number of channels and reconstructs the input image again from $\mathbf{z}$ using a decoder. In current high-resolution image generation models, denoising is conducted only on the latent, usually using either a UNet or a diffusion transformer. Text-to-image models (Rombach et al., 2022) additionally condition denoising on a text prompt y, allowing latent features to interact with text embeddings through cross-attention. As we target class-agnostic segmentation, we omit the text prompt and only pass null text embeddings. Consequently, we restrict ourselves to image-to-image attention and ignore text-to-image cross-attention in our experiments. We primarily adopt the UNet-based SD2 architecture, whose latent is downsampled less aggressively compared to diffusion transformers. This enables us to better handle small instances and preserve fine details. Since the denoising UNet is trained to denoise at different levels $t = 1, \ldots, T$, it can be repurposed as a feature extractor for images running just a single step (Luo et al., 2023; Meng et al., 2024; Zhang et al., 2024; Stracke et al., 2025). This involves encoding the input image into the latent representation $\mathbf{z}$ without applying the forward noise. During denoising, we can cache and leverage internal representations for downstream tasks. In our case, we are interested in high-resolution self-attention. Therefore, we cache multi-layer self-attention outputs from the last UNet block at the highest resolution. Depending on the timestep, different features are produced. Thus, the feature extraction can be optimized for specific target tasks. Therefore, we cache the self-attention matrices of the first two attention layers of the highest-resolution UNet decoder block Sec. B.1. Depending on the timestep, different features are produced. Thus, the feature extraction can be optimized for specific target tasks.

B Implementation Details

B.1 Pseudo-Label Generation

Attention extraction We extract the self-attention of the first two attention layers of the highest-resolution decoder block, *i.e.,* of the SD2 UNet modules `up_blocks.3.attentions.0.transformer_blocks.0.attn1` (weight $w_1 = 0.85$) and `up_blocks.3.attentions.1.transformer_blocks.0.attn1` (weight $w_2 = 0.15$). Row-wise softmax makes this affinity asymmetric. We therefore symmetrize the affinity and also apply temperature sharpening with $\tau_{\text{att}} = 0.55$ to amplify differences. The resulting affinity $\mathbf{A}$ is both non-negative and symmetric:

$$\widetilde{A}_{ij} = \frac{(A_{ij} + \delta)^{1/\tau_{\text{att}}}}{\sum_{j'} (A_{ij'} + \delta)^{1/\tau_{\text{att}}}} \ , \qquad \mathbf{A} \leftarrow \tfrac{1}{2}(\widetilde{\mathbf{A}} + \widetilde{\mathbf{A}}^{\top}) \ , \quad \delta = 10^{-12} \ . \tag{4}$$

Prompt propagation For our pseudo-label generation, we use an efficient solver following iterative Gauss-Jacobi introduced in (Elmoataz et al., 2008) which is described in Algorithm 1. We use a vectorized implementation and propagate all prompts in parallel. After producing our soft maps, we need to generate instance mask proposals from them. Our next steps consist of map merging and post-processing. Here, we show these two steps in detail in Algorithm 2.

Diffuse2Seg generates pseudo-labels without training, so we determine all label-generator hyperparameters once on a held-out SA-1B 1k split and keep them frozen across all datasets and downstream training runs. We show the hyperparameter behavior in Figure 6. For our productive configuration, we aim to balance mask precision and recall. In detail, we resize the images to 1120×1120 ($r = 140$), which is outside the native resolution of SD2 but still yields good segmentation results. We combine features from the top two decoder layers with $w_1 = 0.85$ and $w_2 = 0.15$, which lies on the precision plateau. Affinities are sharpened with $\tau_{\text{att}} = 0.55$, the optimum for recall and precision. Our prompt seeds are placed on an equidistant grid with spacing ($s = 6$), retaining 96.6% of the recall of a denser $s = 5$ grid. We set the propagation exponent to $p = 1.6$ to prioritize precision (0.018 vs. 0.015 at $p = 1.2$, avoiding noisy boundary fragments for downstream student training) with prompt anchoring $\lambda = 1 \times 10^{-5}$. Finally, multi-granularity masks are formed via average-link clustering over $L = 6$ log-spaced thresholds between $h_{\min} = 0.186$ and $h_{\max} = 2.99$, followed by NMS at $\tau_{\text{IoU}} = 0.9$ and noise filtering at $A_{\min} = 100$ px.

B.2 CascadePSP Pseudo-Label Post-Processing

Our post-processing with CascadePSP (Cheng et al., 2020) follows UnSAM and discards refined masks with area fraction > 0.9 or refinement overlap $\text{IoU}(m, \tilde{m}) < 0.5$. We group mask crops into spatial buckets and pad them to squares with zeros. We execute CascadePSP with FP16 autocast. In UnSAM, the refinement crop side length L scales up to $L_{\max} = 900$. As we generate more large masks that need less refinement, we cap post-processing at $L_{\max} = 480$ to accelerate refinement.

B.3 Training Hyperparameters

For training our segmentation models, we follow UnSAM and use Mask2Former (Cheng et al., 2022) with a ResNet-50 backbone. Moreover, we use three different training dataset budgets: 1%, 2%, and 4%. All experiments share the same optimizer and architecture: AdamW, learning rate 5×10^{-5}, weight decay 0.05, backbone multiplier 0.1, batch size 16 on 4 A100 GPUs, gradient clipping 0.01, input resolution 1024×1024, $2{,}000$ queries and loss weights $\lambda_{\text{cls}} = 2$, $\lambda_{\text{mask}} = \lambda_{\text{dice}} = 5$. The encoder in Round 1 is initialized from DINO ResNet-50, and decoder weights are initialized randomly. Round 2 is initialized from the matching Round-1 checkpoint.

Round 1 We train for approximately eight epochs at each budget, with learning-rate decays at 90% and 96% of the schedule. Warmup is $5{,}000$ iterations. The detector-free model is trained on Diffuse2Seg labels only. Both Round-1 configurations with and without detector use copy-paste with p_{cp}=0.5. For copy-paste a donor image is randomly sampled from the dataset. Small objects below 4096 pixels are favored with probability 0.8. At each iteration the number of donor masks is uniformly sampled between one and $\min(N, 30)$, where N is the number of valid donor masks.

Round 2 For the label enrichment, we re-infer the Round-1 model on the entire image and the padded parent crops of size $H_{\text{ref}} \times W_{\text{ref}} = 1024 \times 1024$ (with a padding ratio of 0.05). Parents are the top-k predictions for which the area fraction falls within the interval $[a_{\text{low}}, a_{\text{high}}] = [0.003, 0.30]$ and whose pairwise IoU is no more than 0.30. We retain the predictions that have a score of at least θ and those children for which the containment is $\tau_{\text{cont}} = 0.70$. We replace the original masks m with the new masks $\hat{m}$ when the intersection over union of the two masks exceeds τ_{merge}, which is set to 0.5. We set θ to 0.55, k to 12, and use a cropping NMS of 0.75 (with an intra-crop NMS of 0.40) when CutLER is not used. When augmenting with CutLER, we use $\theta = 0.65$, $k = 9$, crop NMS 0.85 (intra-crop NMS 0.50), and then append CutLER masks. Independent of the Round-1 training image budget, Round 2 fine-tunes only with 0.1M crop-refined labels to maintain efficiency.

Semi-Supervised Learning For semi-supervised fine-tuning, we keep the unsupervised checkpoint as initialization, freeze the ResNet-50 backbone, and fine-tune only the Mask2Former decoder on different image budgets of SAM ground-truth masks. We consider budgets of 100, 500, 1k, 2k, 5k, and 10k labeled images, plus a fully supervised 100k baseline trained from scratch on the same SAM-GT pool. We compare two initializations at every budget: Our final Round-2 0.4M checkpoint and the public UnSAM (4% SA-1B) checkpoint. The optimizer matches unsupervised training (AdamW, learning rate 5×10^{-5}, weight decay 0.05, batch size 16, copy-paste with $p_{\text{cp}} = 0.5$). The schedule consists of $\max(1000, \lfloor B/2 \rfloor)$ iterations, including a 10% warmup phase and then decaying at 90% and 96% of the run. We always use the latest checkpoint and assess it on the same zero-shot benchmark as our unsupervised models.

Algorithm 1 Non-linear prompt propagation: A grid of one-hot prompts is propagated into soft object maps

Input: image I; SD UNet; layer weights $\{w_l\}$; timestep t; constant ϵ; anchor λ; power p; threshold τ_{prop}; attention temperature τ_{att}
Output: soft object maps $\{f^{(1)}, \ldots, f^{(K)}\}$

// Self-attention extraction
$z \leftarrow \text{VAE}(I)$
$\epsilon_t \leftarrow \text{UNet}(z, t)$ ▷ Denoise at timestep t
$\mathbf{A} \leftarrow \sum_{l \in L} w_l \frac{1}{n_h} \sum_{h \in n_h} \mathbf{A}_{(l,h)}$ ▷ Collect $N \times N$ affinity
$\widetilde{\mathbf{A}} \leftarrow \text{softmax}_{\text{row}}\left(\log(\mathbf{A} + 10^{-12})/\tau_{\text{att}}\right)$
$\mathbf{A} \leftarrow \frac{1}{2}(\widetilde{\mathbf{A}} + \widetilde{\mathbf{A}}^{\top})$ ▷ Sharpen and Symmetrize
// Prompt propagation
$\{f_0^{(1)}, \ldots, f_0^{(K)}\} \leftarrow$ equidistant one-hot grid prompts on the latent
for $k = 1$ **to** K **do**
  $\mathbf{f} \leftarrow \mathbf{f}_k^{(0)}$
  repeat
    $g_i \leftarrow \sqrt{\sum_j A_{ij}(f_j - f_i)^2 + \epsilon^2} \quad \forall i$
    $\gamma_{ij} \leftarrow A_{ij}\left(g_i^{p-2} + g_j^{p-2}\right) \quad \forall i, j$
    $f_i \leftarrow \dfrac{\lambda f_{0,i}^{(k)} + \sum_j \gamma_{ij} f_j}{\lambda + \sum_j \gamma_{ij}} \quad \forall i$
  until $\|\mathbf{f}^{t+1} - \mathbf{f}^t\|_2^2 \le \tau_{\text{prop}}$
  $\mathbf{f}^{(k)} \leftarrow \mathbf{f}$
end for
return $\{f^{(1)}, \ldots, f^{(K)}\}$

Algorithm 2 Mask merging: Taking the initial propagated soft object maps as input, the mask merging step produces a final set of instance mask candidates

Input: soft object maps $\{\mathbf{f}_{(1)}, \ldots, \mathbf{f}_{(K)}\}$; thresholds $\mathcal{H} = \{h_1, \ldots, h_L\}$; min. area $a_{\min}$; IoU thr. τ_{IoU}; cap $M_{\max}$
Output: pseudo-mask set $\mathcal{M}$

// Mask merging
$\mathbf{p}_k = \mathbf{f}_k / \sum_i f_{k,i} \quad \forall k$ ▷ Normalize to distributions
$d_{\text{KL}} \leftarrow \frac{1}{2}\left(D_{\text{KL}}\left(\mathbf{p}_{(k)} \,\|\, \mathbf{p}_{(k')}\right) + D_{\text{KL}}\left(\mathbf{p}_{(k')} \,\|\, \mathbf{p}_{(k)}\right)\right)$ ▷ Compute pairwise prompt distance
$\mathcal{P} \leftarrow \emptyset$
for $h \in \mathcal{H}$ **do**
  $\{C_1^h, \ldots, C_{K_h}^h\} \leftarrow \text{AGGLOMCLUSTER}(\mathbf{D}, h)$
  $\bar{\mathbf{p}}_c^h \leftarrow \frac{1}{|C_c^h|} \sum_{k \in C_c^h} \mathbf{p}_{(k)} \quad \forall c$
  $\tilde{\mathbf{p}}_c^h \leftarrow \text{UPSAMPLE}\left(\bar{\mathbf{p}}_c^h, H \times W\right) \quad \forall c$
  $M^h(x) \leftarrow \arg\max_c \tilde{\mathbf{p}}_c^h(x)$
  for each label c in M^h **do**
    $r \leftarrow \text{CONNECTEDCOMPONENTS}(M^h)$
    if $\text{area}(r) \geq a_{\min}$ **then**
      $\mathcal{P} \leftarrow \mathcal{P} \cup \{r\}$
    end if
  end for
end for
// Area-descending non-maximum suppression
sort $\mathcal{P}$ by descending area; $\mathcal{M} \leftarrow \emptyset$
for $p \in \mathcal{P}$ **do**
  if $|\mathcal{M}| < N_{\max}$ **and** $\max_{q \in \mathcal{M}} \text{IoU}(p, q) \leq \tau_{\text{IoU}}$ **then**
    $\mathcal{M} \leftarrow \mathcal{M} \cup \{p\}$
  end if
end for
return $\mathcal{M}$

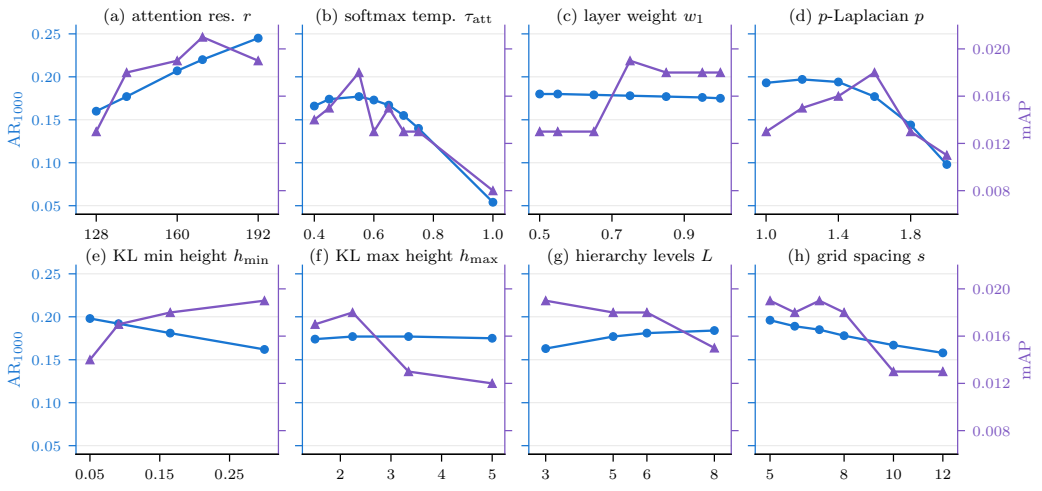


Figure 6: **Parameter sweeps of Diffuse2Seg label generation on a SA-1B holdout set.** Each panel varies a single parameter with all others held at a reference configuration. AR_{1000} (blue, left axis) and mAP (purple, right axis) are shown at their true values, using identical scales across all panels to enable direct comparison of parameter sensitivities. In some cases a shared optimum for both metrics (*e.g.,* τ_{att}) can be observed. However, often there is tradeoff, for instance recall being stable but precision decreasing (*e.g.,* w_1, L), and trade-offs of in-domain recall and precision (*e.g.,* p).

B.4 Semi-Supervised Results

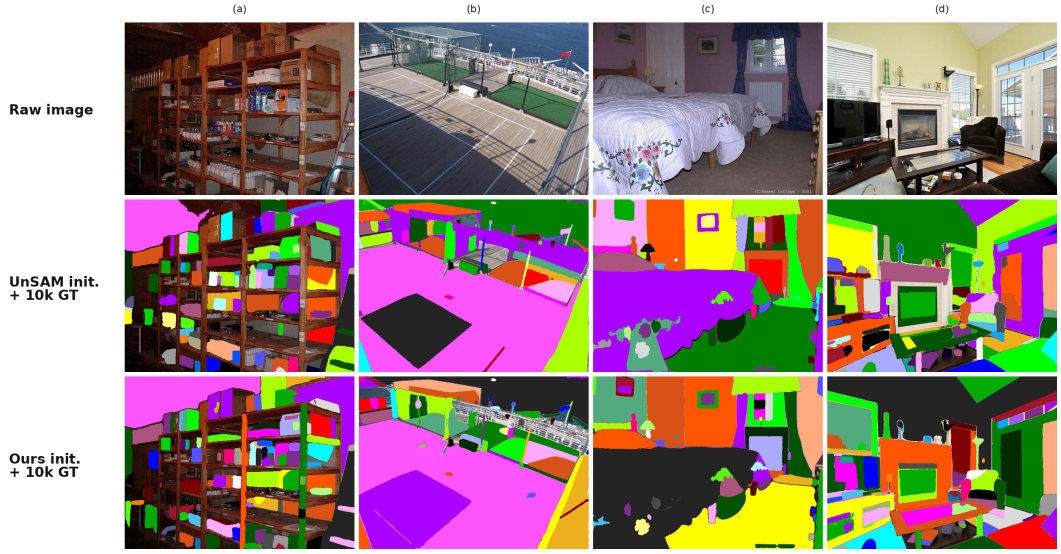


Figure 7: **Qualitative comparison of semi-supervised instance segmentation on ADE20K.** Each column (a–d) shows a different indoor/outdoor scene. Top: raw input image, middle: predictions of the UnSAM checkpoint fine-tuned on 10k labeled images, bottom: predictions of our Diffuse2Seg model fine-tuned on 10k labeled images. Under the same 10k-label budget, initializing from our diffusion-based unsupervised model yields higher mask coverage, crisper object boundaries, and better segmentation of fine structures (e.g., furniture legs, windows, and facade details).

C Evaluation Protocol

C.1 Baselines

We evaluate our pseudo-labeling mechanism against common baselines for unsupervised segmentation. For SSL-based unsupervised segmentation, DINO pre-training poses a widely used standard. Additionally, we consider entity-segmentation baselines, including SOHES (Cao et al., 2024), which performs global clustering and local re-clustering, and UnSAM, which relies on divide-and-conquer that first discovers coarse regions and then clusters them bottom-up. For diffusion-based pre-training, there is no prior work on open-world entity segmentation. That is why we repurpose several established baselines in unsupervised semantic segmentation. DiffSeg (Tian et al., 2024) uses a similar approach and merges attention maps into segmentation masks. We adapt the work by applying connected components at the end to isolate instances. Moreover, we add the recent method M2N2 (Karmann & Urfalioglu, 2025), which repurposes generative diffusion models for interactive segmentation. It propagates user-defined point prompts into segmentation masks. To make a fair comparison, we apply the same grid of point prompts and extract the resulting masks at image resolution. For our training experiments, we compare to UnSAM, the state-of-the-art for unsupervised entity segmentation, and, additionally, to established unsupervised instance segmentation methods that incorporate training, such as SOHES and CutLER (Wang et al., 2023).

Baseline Implementation Details For implementation we do the following adaptations: We reconstruct the two stages of the SOHES (Cao et al., 2024) pseudo-labeling pipeline using frozen DINO ViT-B/8 features and its default parameters. We follow UnSAM's divide-and-conquer procedure (Wang et al., 2024), which first obtains instance segmentation masks from a pretrained CutLER detector and subsequently discovers finer regions through bottom-up clustering. Local features are extracted using frozen DINO ViT-B/8 from crops resized to 256×256. We follow the paper and use the clustering thresholds $\{0.6, 0.5, 0.4, 0.3, 0.2, 0.1\}$. We use a lower confidence threshold of 0.1 instead of 0.3, as reported in the paper, to improve recall and ensure a fairer comparison to our method. For DiffSeg, we calibrate the thresholds on 10 SA-1B images, leading to the final thresholds $(1.4064, 0.4587, 1.8453, 3.1033, 4.3099)$. To extract the instance masks, we apply con-

nected components similar to our method. Moreover, we apply our duplicate-suppression NMS with IoU of 0.9 and filter masks below 100 pixels. For M2N2, we follow the feature extraction and propagation parameters from their paper. We do not apply JBU because it does not scale to a grid of prompts, instead, we apply bilinear upsampling. We also deduplicate the masks with NMS at IoU=0.9 and filter masks below 100 pixels. For CutLER, we use their pre-trained model and infer it on the target datasets.

C.2 Pseudo-Label and Zero-Shot Metrics

To evaluate both the quality of the diffusion-based pseudo-labels and the zero-shot performance of the derived entity-segmentation models, we mainly consider mean average recall (AR_{1000}), which is established in open-world entity segmentation (Wang et al., 2023; Cao et al., 2024; Wang et al., 2024). It averages recall over IoU matching thresholds from 0.5 to 0.95. We always consider up to 1000 predicted masks per image for metric calculation. In addition, we report average recall filtered by object size, denoted as AR_S, AR_M, and AR_L, following the COCO (Lin et al., 2014) definition. Because label definitions and taxonomies vary across datasets, mAP is less meaningful and is only considered for selected experiments.

D Additional Results and Ablations

D.1 Ablation of Merging Parameters

We next ablate the parameters of our post-processing step. Increasing the number of hierarchy levels consistently improves recall, since additional levels help to isolate a bigger variety of granularities. Going from 3 to 6 levels improves mAR from 17.9 to 20.1 (see Table 4a). To deduplicate across levels, we use a simple area-sorted non-maximum suppression (NMS). We analyze different thresholds in Table 4b and find that a conservative threshold of $\tau_{\mathrm{IoU}} = 0.9$ best retains mask candidates across granularities. Deduplication using a smaller overlap threshold, such as $\tau_{\mathrm{IoU}} = 0.5$, reduces recall significantly by 3.6 p.p. in mAR. We therefore keep 6 levels and $\tau_{\mathrm{IoU}} = 0.9$ for our experiments.

Levels	AR_{1000}	AR_S	AR_M	AR_L	mAP
3	17.9	2.0	12.5	34.9	**1.9**
5	19.7	2.4	14.1	37.9	1.8
6	**20.1**	**2.5**	**14.5**	**38.4**	1.7

(a) Number of levels in the map merging step

NMS τ_{IoU}	AR_{1000}	AR_S	AR_M	AR_L	mAP
0.5	16.5	2.3	12.1	31.2	1.6
0.7	18.4	2.5	13.4	34.8	1.7
0.8	19.2	**2.6**	14.0	36.5	**1.8**
0.9	**20.1**	2.5	**14.5**	**38.4**	1.7

(b) Cross-level deduplication

Table 4: **Diffuse2Seg mAR as a function of number of hierarchy levels and NMS threshold for deduplication.** Each panel varies one parameter around our final label-generation configuration ($L = 6$, $\tau_{\mathrm{IoU}} = 0.9$, $s = 6$, $r = 140$, $\tau_{\mathrm{att}} = 0.55$, $w_1 = 0.85$, $p = 1.6$, $\lambda = 10^{-5}$) on the 1k SA-1B holdout; the highlighted row is the same run in both panels. More hierarchy levels and a conservative τ_{IoU} increase recall.

D.2 Discussion of Pseudo-Labeling Baselines

To evaluate the quality and generalization of our diffusion-based pseudo-labels, we first analyze their performance on SA-1B in Table 1, comparing them against the baselines described in Sec. 4.1 without mask-refinement post-processing. Labeling the high-resolution SA-1B images is particularly challenging, as most methods operate on latent representations with substantially lower resolution. Nevertheless, despite operating on such a low-resolution representation, our method achieves higher average recall than all baselines: Diffuse2Seg's mAR is 4.7 p.p. higher than the strongest diffusion-based baseline DiffSeg (Tian et al., 2024) and 4.3 p.p. higher than the prior DINO-based state-of-the-art UnSAM (Wang et al., 2024).

Comparing the diffusion-based baselines, we find DiffSeg to significantly outperform M2N2 at all object scales on SA-1B, likely because its agglomerative clustering covers multiple object granularities. In contrast, M2N2 (Karmann & Urfalioglu, 2025) only produces a single granularity mask per prompt. For the DINO-based baselines, prior state-of-the-art UnSAM has an advantage on small objects: its divide-and-conquer strategy uses predictions from the pre-trained CutLER instance segmenter and

further splits the CutLER-based instances into smaller parts using a bottom-up approach. However, its dependency on the ImageNet pre-trained CutLER detector can cause the method to miss some coarse objects and their parts entirely if the object is not detected or assigned low objectness. This lowers UnSAM's recall on medium and large objects relative to our approach, which avoids this dependency for pseudo-label generation.

Next, we compare these segmentation methods on datasets from diverse domains, as shown in Table 1 in the main paper, keeping the previously used hyperparameter values. This cross-domain analysis covers a broad range of datasets, ranging from ADE20k that addresses Scene Parsing with a huge number of different classes, over EntitySeg that encompasses diverse entities from the open world, to UVO that provides densely annotated frames from unstructured web videos, and PACO which targets part segmentation. Among the training-free methods, our Diffuse2Seg technique clearly benefits from the generality of diffusion-model representations and consistently achieves the highest recall, reaching, for example, an AR_{1000} of 13.6 on PACO, a 2.9 p.p. margin over the strongest baseline CutLER, despite the latter being trained. This demonstrates Diffuse2Seg's ability to generate multi-granularity masks. In addition, we outperform the training-free baselines on entity-focused datasets such as ADE20k and EntitySeg, by a large margin gaining 6.5 p.p. and 4.5 p.p. of mAR compared to the prior state-of-the-art technique UnSAM. Generally, our diffusion-based approach outperforms DINO-based approaches in this setup, confirming our observation that diffusion priors are better suited for segmenting stuff+things datasets than DINO features. We further show improvements across domains compared to other diffusion-based baselines. M2N2 performs weakly on datasets with objects at different granularities and is roughly on par with Diffuse2Seg on UVO only due to the dataset's whole-object focus. DiffSeg, in contrast, includes a mechanism to cover different granularities. However, without a propagation step, it appears to be much less stable across domains, with a gap of up to 11.0 p.p. to our method. Notably, our method outperforms the CutLER-trained detector, which was trained on 1.3M ImageNet images for segmentation, on SA-1B, EntitySeg, and PACO and stays close on UVO and ADE20k.

Results after Refinement Similar to SOHES and UnSAM, we also apply CascadePSP (Cheng et al., 2020) to improve the quality of our generated masks. Our refinement hyperparameters are described in Sec. B.2. Table 5 shows that refinement lifts Diffuse2Seg's performance from 20.7 to 29.2 in AR_{1000}. Compared with refined UnSAM masks, our refined masks achieve 5.4 p.p. higher overall recall, with improvements of 2.5 and 14.8 p.p. on medium and large entities, respectively. Similar to the pre-refinement results, UnSAM achieves higher small-object recall (9.4 vs. 5.3). The resulting mask sets contain 366 masks per image for Diffuse2Seg and 387 for UnSAM on average. This demonstrates that our higher recall after all steps does not come with the cost of more proposals on average.

Method	AR_S	AR_M	AR_L	AR_{1000}	**Masks/Image**
UnSAM (CascadePSP)	**9.4**	22.1	33.9	23.8	387
Diffuse2Seg (CascadePSP)	5.3	**24.6**	**48.7**	**29.2**	366

Table 5: **Pseudo-mask quality after CascadePSP refinement** on the SA-1B (Kirillov et al., 2023) validation split (1000 images). Masks/image is the mean number of masks per image after refinement.

D.3 Trained Model Results

Table 2 reports aggregate AR_{1000} only. Because the unrefined pseudo-labels of Table 1 show a small-object deficit against UnSAM (AR_S of 2.6 vs. 5.7), we additionally report the size breakdown of the trained models on the SA-1B validation split, for our detector-augmented model at the 0.4M budget and the final UnSAM model at 4% of SA-1B in Table 6.

D.4 Additional EntitySeg Results

As EntitySeg contains a larger portion of LAION-5B (Schuhmann et al., 2022) images, we evaluate our method on 1k sampled EntitySeg training images excluding LAION-5B to make sure that our results are not biased by the LAION-5B pre-training of SD2. Table 7 shows a comparison of our best

Method	# Images	Average Recall				Average Precision			
		AR_{1000}	AR_S	AR_M	AR_L	AP	AP_S	AP_M	AP_L
UnSAM	0.4M	44.8	11.3	45.7	59.2	10.8	0.8	10.3	16.8
Diffuse2Seg (**ours**)	0.4M	**47.8**	**16.2**	**46.7**	**65.0**	**14.3**	**2.1**	**13.4**	**24.6**

Table 6: **Comparison of the trained students on SA-1B.** Both models are the detector-augmented variants trained on 0.4M images and evaluated with 1000 proposals per image, the same setting as the SA-1B column of Table 2. The small-object deficit of the raw pseudo-labels does not carry over to the students: Diffuse2Seg leads on every size bucket, most strongly on small ($+4.9$ p.p. AR_S) and large entities ($+5.8$ p.p. AR_L, $+7.8$ p.p. AP_L).

model vs. UnSAM's best model. Excluding LAION images maintains the same margin as on the validation set and similarly highlights our method's strong stuff+things segmentation performance.

Method	# Images	Average Recall				Average Precision			
		AR_{1000}	AR_S	AR_M	AR_L	AP	AP_S	AP_M	AP_L
UnSAM	0.4M	37.6	15.4	30.6	49.2	4.8	0.4	1.3	11.2
Diffuse2Seg (**ours**)	0.4M	**39.5**	**15.9**	**31.4**	**52.0**	**8.4**	**0.8**	**2.0**	**15.3**

Table 7: **Comparison of the trained students on 1k EntitySeg training images excluding LAION.** We compare the best public UnSAM model trained on 4% vs. our best final model with detector augmentation as in Table 2. Diffuse2Seg keeps its advantage on every size bucket. The margin is with $+1.9$ AR_{1000} similar to $+2.1$ on the full EntitySeg validation split (Table 2), and its AP is +3.6 p.p. higher than UnSAM.