SOS: Segment Object System for Open-World Instance Segmentation With Object Priors

Christian Wilms[1], Tim Rolff[1,2], Maris Hillemann[1], Robert Johanson[1], and Simone Frintrop[1]

[1] Computer Vision Group, University of Hamburg, Germany
[2] Human-Computer Interaction Group, University of Hamburg, Germany
{firstname.lastname}@uni-hamburg.de

Abstract. We propose an approach for Open-World Instance Segmentation (OWIS), a task that aims to segment arbitrary unknown objects in images by generalizing from a limited set of annotated object classes during training. Our Segment Object System (SOS) explicitly addresses the generalization ability and the low precision of state-of-the-art systems, which often generate background detections. To this end, we generate high-quality pseudo annotations based on the foundation model SAM [27]. We thoroughly study various object priors to generate prompts for SAM, explicitly focusing the foundation model on objects. The strongest object priors were obtained by self-attention maps from self-supervised Vision Transformers, which we utilize for prompting SAM. Finally, the post-processed segments from SAM are used as pseudo annotations to train a standard instance segmentation system. Our approach shows strong generalization capabilities on COCO, LVIS, and ADE20k datasets and improves on the precision by up to 81.6% compared to the state-of-the-art. Source code is available at: https://github.com/chwilms/SOS

Keywords: Open-world Instance Segmentation · Object Localization · Prompting

1 Introduction

Open-World Instance Segmentation (OWIS) is the task of segmenting all object instances in an image by learning from a limited set of known object classes [26, 49, 50]. In contrast to instance segmentation, OWIS is not limited by the closed-world assumption, which assumes that all object classes are known in advance. Since OWIS methods aim to detect not only learned but also unknown object classes, they return class-agnostic detections. This is of particular interest in real-world scenarios with previously unknown object classes (e.g., [55]) and challenges the systems' generalization capabilities. An example of this is shown in the first row of Fig. 1, where annotations for classes such as surfboard or tennis racket do not exist during training, but the objects should be detected during testing.

Previous OWIS methods mostly generate pseudo annotations for unannotated objects [24, 46, 49, 56] or replace the foreground-background classification

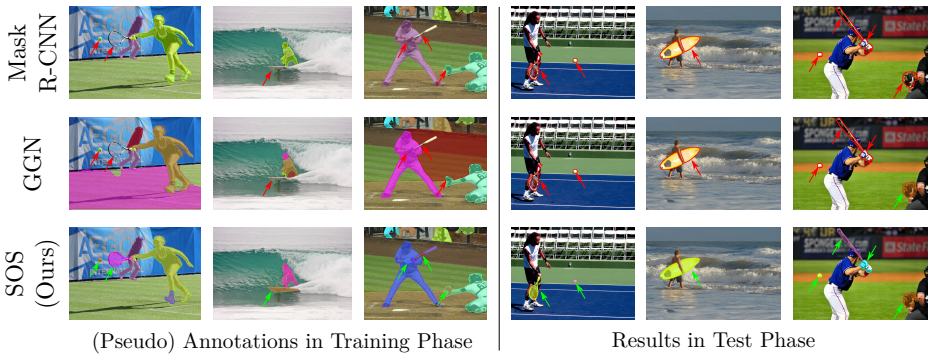


Fig. 1: Comparison of (pseudo) annotations (left) used by Mask R-CNN [19], GGN [49], and our SOS, when only original annotations of VOC object classes are given. While Mask R-CNN only uses original annotations without object classes such as tennis racket or surfboard (red arrows), SOS generates pseudo annotations covering those classes (green arrows). GGN generates noisy pseudo annotations including background areas. As a result, only SOS constantly detects these objects not annotated in training (green vs. red arrows on the right). Filled masks denote annotations (left) or detected objects (right), while red frames indicate missed objects.

of possible objects in instance segmentation systems with a learning target focusing on the localization quality (e.g., intersection over union) [26,48,54]. While OWIS methods generalize better to unseen object classes than standard instance segmentation systems, they still exhibit low precision [24,26,46,49,54]. One reason for this is the use of noisy pseudo annotations covering background areas as visible in the left part of the second row in Fig. 1.

Recently, foundation models [4] emerged as a promising technique [5,9,23,29, 42]. They are trained on large datasets with a surrogate task and applied to other tasks in a zero- or few-shot manner using prompting. The first foundation model for image segmentation, the Segment Anything Model (SAM), was proposed by [27]. Trained on a large automatically annotated dataset, SAM generates segments for object or stuff regions based on point, box, mask, or text prompts without further training. Despite not only segmenting objects but also stuff regions, given well-designed prompts, SAM is able to generate high-quality object segments. Note that vanilla SAM does not address object segmentation tasks like OWIS itself due to the aforementioned lacking focus on objects.

In this paper, we propose the *Segment Object System* (SOS), a novel OWIS method utilizing high-quality pseudo annotations to improve the generalization capability of a standard instance segmentation system (see lower row in Fig. 1). For generating pseudo annotations, we apply the foundation model SAM with prompts derived from an object prior to roughly localize objects of arbitrary classes and to limit segmentations of stuff regions by SAM. To ensure the high quality of our pseudo annotations based on SAM, improving the overall precision, we thoroughly study various hand-crafted and learned object priors for an

object-focused application of SAM. This is relevant beyond the scope of OWIS. Using our study findings, SOS utilizes self-attention maps from self-supervised Vision Transformers (ViTs) [11] as object prior for prompting SAM. Finally, SOS filters low-quality pseudo annotations, combines the pseudo annotations with the original annotations of the known classes, and trains a standard instance segmentation system with these mixed annotations. Our extensive evaluation shows the strong generalization abilities of SOS, considerably outperforming all previous state-of-the-art systems across COCO [31], LVIS [17], and ADE20k [61] datasets. Most notably, the results improve by up to 81.6% in terms of precision over the state-of-the-art due to the high-quality pseudo annotations. We also show that SOS is better suited for OWIS than directly applying SAM.

Overall, our contributions are threefold:

- We propose SOS, a novel OWIS method based on a learned object prior, prompt-based pseudo annotations, and an arbitrary instance segmentation system.
- We thoroughly study various object priors for focusing SAM on objects, leading to high-quality object segments for pseudo annotations and beyond.
- Our extensive evaluation shows that the high-quality pseudo annotations in SOS lead to high precision and strong overall results, clearly outperforming other methods.

2 Related Work

2.1 Open-world Instance Segmentation

To address OWIS, literature mainly offers two streams. First, systems replace the hard classification in instance segmentation systems with a localization score. This avoids classifying unseen objects as background and improves generalization ability [26]. In OLN [26], centerness and Intersection over Union (IoU) are learned as localization scores in a Mask R-CNN system [19]. SWORD [54] and OpenInst [48] follow the same idea for query-based systems.

The second line of works [24, 46, 49, 56] addresses OWIS by augmenting the annotations from known classes with pseudo annotations aiming to cover objects of unknown classes. To generate pseudo annotations, GGN [49] learns pixel affinities from known classes to generate segments, which are grouped. Similarly, UDOS [24] groups object proposals of object parts to create pseudo annotations, while LDET [46] applies copy-paste-augmentation. Recently, a new stream emerged [58, 62] explicitly utilizing class labels from known classes.

We follow the second stream and create pseudo annotations within SOS, since it offers the most flexibility w.r.t. the base instance segmentation system. Different from existing approaches that generate noisy pseudo annotations (see GGN in Fig. 1), we investigate and develop object priors to effectively use recent foundation models to create high-quality object-focused pseudo annotations, leading to high recall and precision.

2.2 Class-agnostic Object Localization

Localizing objects in images is related to several computer vision tasks. For instance, in traditional object proposal generation [2], cues like saliency [2, 37] or contour information [2, 10, 36, 43] were used to localize class-agnostic object candidates for object detection. Since the advent of deep learning, the models are learned from data [16, 22, 40, 41, 53].

Focusing on arbitrary salient objects, known as salient object detection, several approaches use the contrast of hand-crafted features such as color [8, 28, 34, 39] or edge information [28], as well as CNN-based approaches [21, 33, 59]. In a different task, several authors investigated the information stored in learned classifiers or self-supervised feature extractors based on CNNs or ViTs to locate objects. Notable avenues include Class Activation Maps (CAMs) [60] that highlight discriminative object or image parts in CNN-based classifiers and the self-attention maps in the self-supervised DINO feature extractor [6] based on ViTs, which contain information on the scene layout indicating object locations.

In our study on object priors to focus SAM on objects, we investigate several of these object localization cues. In contrast to these approaches, we follow a different goal, focusing prompt-based segmentations on objects for OWIS.

2.3 Applications of SAM

SAM [27] is used in several segmentation tasks [3,38,47]. Similar to us, [7] and [57] use SAM to generate pseudo annotation masks for weakly-supervised semantic segmentation based on image-level supervision. Similarly, [18] generate pseudo annotation masks for concealed object segmentation based on scribble annotations. Finally, [51] generate pseudo annotation masks for weakly-supervised instance segmentation in a multiple instance learning framework.

Different from the weakly-supervised approaches utilizing SAM, we do not rely on supervision. Hence, a key novelty of our work lies in investigating object priors to focus SAM on segmenting arbitrary objects in the absence of supervision for all object classes. Moreover, we are the first to apply SAM-based pseudo annotations in OWIS.

3 Segment Object System for OWIS

To address OWIS, we propose our *Segment Object System* (SOS) that generates high-quality pseudo annotations to train a standard instance segmentation system. As visible from Fig. 2, SOS consists of three main blocks. The first block, the Object Localization Module (OLM), described in Sec. 3.2 (yellow area in Fig. 2), aims to roughly localize objects by sampling locations from an object prior that reflects the object probability per image coordinate. The output of OLM is a set of likely object locations, that is used in the Pseudo Annotation Creator (PAC), described in Sec. 3.3 (green area in Fig. 2), to prompt the pre-trained SAM [27], which we briefly revisit in Sec. 3.1. From the segments

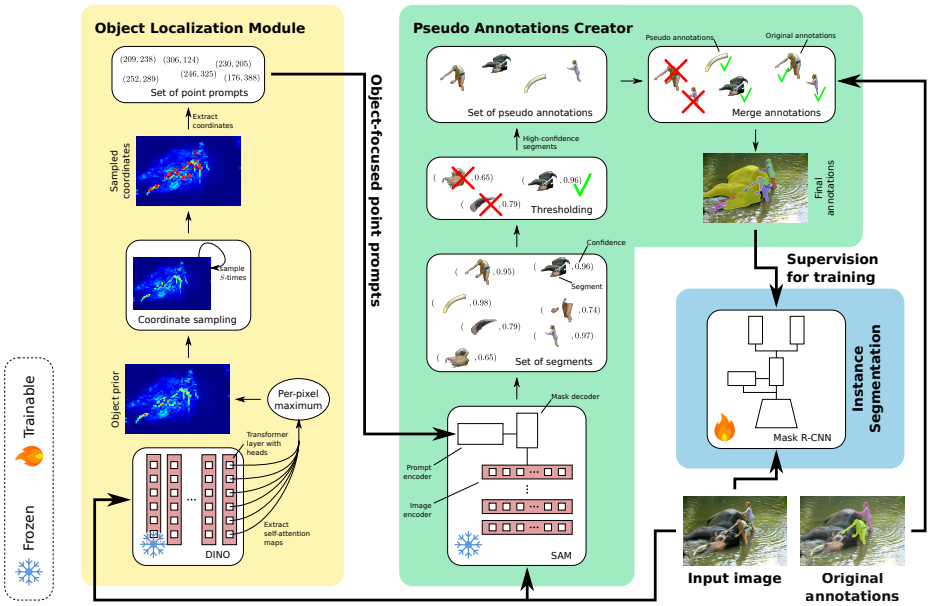


Fig. 2: Overview of our Segment Object System (SOS) for OWIS consisting of three blocks. First, the input image is processed in our Object Localization Module (OLM, yellow area) to create object-focused point prompts roughly localizing objects. Second, our Pseudo Annotations Creator (PAC, green area) generates segments based on the previously generated prompts using SAM [27], and further processes them, leading to a final set of merged original and pseudo annotations. Finally, the merged annotations are used to train an instance segmentation system (blue area).

generated from likely object locations, the PAC generates pseudo annotations by filtering low-quality segments and removing near-duplicates. Finally, the pseudo annotations are merged with the original annotations, and a standard instance segmentation system is trained with the merged annotations (see Sec. 3.4, blue area in Fig. 2). During testing, only this instance segmentation system is used.

3.1 Revisit Segment Anything Model

Since SAM [27] is integral to SOS, we will briefly review the model. The idea of SAM is to utilize prompts such as point coordinates to generate segments of image parts like objects, object parts, or stuff regions localized by these prompts. Hence, SAM does not exclusively segment objects unless explicitly instructed by object-focused prompts, as proposed in this paper. SAM is based on a ViT [11] image encoder creating an image embedding, while the embeddings for point prompts are generated based on positional encodings. Both embeddings are processed in a mask decoder utilizing a transformer decoder block and a mask prediction head to create the final segment including a confidence score. Note that SAM is class-agnostic and does not classify the segments.

For training SAM, [27] propose the new large-scale SA-1B dataset that is annotated with a data engine using SAM itself. In the pre-stage, SAM is trained with public segmentation datasets to aid the first stage of the data engine, where annotators use SAM as an interactive segmentation tool to create initial segmentations. Subsequently, SAM is retrained with the newly annotated data, fully avoiding data leakage w.r.t. public segmentation datasets. In subsequent stages, manual annotations were provided for segments missed by SAM. At last, annotations were created fully automatically by SAM. See [27] for more details.

3.2 Object Localization Module

The first block of SOS (yellow area in Fig. 2) is the Object Localization Module (OLM). OLM roughly localizes all objects in the input image, and creates object-focused point prompts. As the first step, OLM generates an object prior from the input image. The object prior is a probability mass function constrained to the image plane, although other formulations are possible (see Sec. 4), and indicates the probability of an image coordinate being part of any object. Hence, the object prior is class-agnostic and not limited to known object classes from the training set in OWIS. In our study in Sec. 4, we analyze various object priors. For final SOS, we utilize the six self-attention maps from the self-attention heads of a ViT's final layer, pre-trained in the self-supervised DINO framework [6] on ImageNet [45]. See Sec. 4 for justifications. OLM aggregates the six self-attention maps by elementwise max yielding one map highlighting all relevant scene elements. Subsequently, OLM converts the map to an object prior, by rescaling such that the minimum equals 0 and the sum over the object prior is 1.

Next, OLM randomly samples S (here: $S = 50$) image coordinates from the object prior, given the per-coordinate values of the object prior as probabilities. To diversify the samples across multiple objects, we prune parts of the object prior around the sampled coordinate. Specifically, given a coordinate (x_s, y_s), we set the object prior to zero for all coordinates $\{(x, y) \mid |x-x_s| \leq N \wedge |y-y_s| \leq N\}$, with $N = 20$. To transform the object prior back to a probability mass function, we again apply the rescaling as described above. Subsequently, OLM iteratively applies this sampling until a set of S coordinates are extracted, representing object-focused point prompts. This set is the output of the OLM and will be used for prompting in the Pseudo Annotation Creator.

3.3 Pseudo Annotations Creator

The Pseudo Annotations Creator (PAC, green area in Fig. 2) is the second block of SOS. It generates pseudo annotations utilizing SAM, given the object-focused point prompts provided by OLM. These object-focused point prompts lifts SAM from a system that segments anything to a system that segments objects. As the first step, PAC prompts a pre-trained SAM with the S point prompts from OLM. To handle ambiguous point prompts that may indicate multiple objects or object parts, we follow [27] and allow SAM to generate three segments per prompt. Each generated segment is a potential pseudo annotation. The resulting set of

class-agnostic segments is noisy, with segments covering objects and background, or covering the same image area. Therefore, PAC first utilizes SAM's confidence score per segment and removes segments with a confidence below τ_{conf}. Second, PAC removes near-duplicates by non-maximum suppression with an IoU threshold τ_{NMS}. We set $\tau_{\text{conf}} = 0.9$ and $\tau_{\text{NMS}} = 0.95$ for SOS.

Given this set of pseudo annotations, PAC merges the set with the original annotations representing known classes of the training dataset to combine the knowledge of the human annotators and the knowledge of our object-focused SAM. Since we only want to keep pseudo annotations for objects not covered by the original annotations, we suppress pseudo annotations that have a high IoU (τ_{NMS}) with at least one original annotation. Moreover, we limit the number of pseudo annotations to P in order to balance original and pseudo annotations. In our final SOS, we set $P = 10$. However, across the COCO training dataset, only 7.8 pseudo annotations are added on average. The other pseudo annotations were suppressed in previous steps. Overall, the output of PAC is a mixed set of annotations, containing all original annotations augmented with high-quality pseudo annotations covering objects of unknown classes.

3.4 Instance Segmentation

In the last block of SOS, we take the merged annotations generated by PAC and train a standard instance segmentation system. While this system can be arbitrary and is generally replaceable, we choose Mask R-CNN [19] with a ResNet-50+FPN backbone [30] trained in a class-agnostic setting, following recent OWIS methods [24, 26, 46, 49]. Overall, this leads to an OWIS method based on high-quality pseudo annotations.

4 Object Priors for SOS

As described in Sec. 3.2, SOS uses an object prior for rough object localization and subsequent point prompt generation for PAC. This section thoroughly investigates the performance of various object priors derived from previous class-agnostic object localization works in SOS. For details on the object priors beyond the subsequent descriptions, we refer to our supplementary.

4.1 Object Priors

Baselines: We introduce three baseline object priors that ignore the image content. First, *Grid* uses SAM with a regular 64×64 grid of points, following [27]. This directly leads to prompts without the sampling described in Sec. 3.2. Note that the *Grid* baseline will lead to segments for both object and stuff regions. Second, we use the spatial distribution of object centroids from the known classes across the training dataset as object prior (*Dist*). We assume that this generalizes well since it ignores class-specific object features. Finally, we utilize the centroids of the training set's objects from unknown classes as optimal prompts (GT), representing an upper bound.

Superpixels: Inspired by [15, 52], we utilize the centroids of superpixels generated with an adaptive superpixel segmentation method as object prior (*Spx*). The intuition is that large uniform areas not containing objects should be covered by few superpixels and vice versa. Hence, the density of superpixels is a surrogate for the density of objects. We use the well-known FH superpixels [13] since they adapt their density to the image content following our assumption. Note that every superpixel centroid is a point prompt.

Contour Density: Since objects are defined by their outer contours [63], we use contour density as an object prior (*Contour*). Similar to [52], we assume that in areas of high contour density, several objects are located and vice versa. As a surrogate for the contour density, we use edge density based on [10].

Saliency: Saliency is used in object proposal generation to localize objects [2, 37] as they stick out from their surrounding. Here, we evaluate two saliency methods as object priors. First, we apply the traditional approach VOCUS2 [14] based on color contrast and used by [37] (*VOCUS2*). Second, we use pre-trained DeepGaze IIE [32] saliency maps learned from eye fixation data (*DeepGaze*).

Class Activation Maps: CAMs [60] indicate discriminative image regions for CNN-based classifiers. Since they only need image-level supervision, they are frequently applied in weakly-supervised tasks to locate objects [1, 25, 35]. We use the CAM of the predicted class in a ResNet-50 classifier [20] pre-trained on ImageNet [45] as object prior (*CAM*).

Self-attention: Recently, [6] have shown that self-attention maps learned inside self-supervised ViTs encode the scene layout, including object locations. Therefore, we explore the self-attention maps from the final layer of a ViT-S backbone trained in the self-supervised DINO framework [6] as an object prior (*DINO*).

Learned Object Locations: Finally, we learn object locations based on the known classes of a dataset using a U-Net [44] in a binary segmentation task (object vs. no object) as object prior (*U-Net*). This assumes that, on the level of individual pixels, the model generalizes from known to unknown classes [49, 56].

4.2 Study Setup for Object Priors

To assess each object prior, we train SOS with the respective prior on the COCO training set using only annotations from the 20 PASCAL VOC [12] classes. We evaluate on COCO's validation set with annotations from the remaining 60 classes in COCO, following standard COCO (VOC) $\rightarrow$ COCO (non-VOC) cross-category evaluation in OWIS [26, 46, 49]. Note that we use the default class-agnostic Mask R-CNN in SOS but reduce the training schedule to only a quarter of steps for faster training. We evaluate the results of SOS given each object priors and report Average Recall (AR) for 100 detections and Average Precision (AP) as previous OWIS research [46, 49]. We also report F_1 score, the harmonic mean between AR and AP, yielding a single number for comparison.

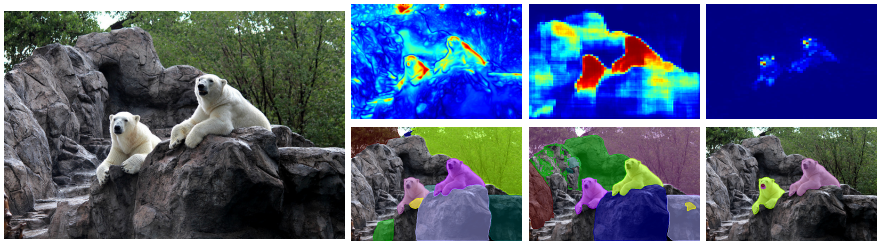


Fig. 3: Object priors *VOCUS2*, *U-Net*, and *DINO* with resulting pseudo annotations.

4.3 Results of Object Priors in SOS

Table 1 shows the results of our object prior study. First, all priors outperform the baselines *Grid* and *Dist* in terms of F_1. Both baselines exhibit a lower AP compared to all other object priors, reflecting the missing focus on objects. Hence, simply applying SAM (*Grid*) is not suitable to segment only objects, as several stuff regions are segmented as well. Second, most priors exhibit a similar performance in terms of all measures. For instance, learning-based *CAM* and *DeepGaze* do not outperform simple priors (*Spx*, *Contour*, *VOCUS2*). Only the priors *U-Net* and *DINO* perform substantially better, with *DINO* producing the best result. Comparing *DINO* to *GT* reveals that *DINO* is able to recall almost the same amount of objects, but exhibits a lower precision, as expected.

To illustrate the results, Fig. 3 depicts object priors for top-performing *VOCUS2*, *U-Net*, and *DINO*, as heatmaps (upper row) with the resulting pseudo annotations generated in SOS (lower row) for an example image. The heatmaps show that *DINO* is very focused on the discriminative object parts, here the polar bears' faces, while *U-Net* highlights, as learned, the entire objects. Moreover, *U-Net* also highlights several background areas. Similarly, *VOCUS2* highlights background areas and the polar bears since it is genrally focused on high-contrast areas. Overall, only *DINO* exclusively generates pseudo annotations for the polar bears and some sub-parts, while *U-Net* and *VOCUS2* produce noisy annotations including background patches, similar to previous approaches [49].

The results show that a well-designed object prior like *DINO* substantially outperforms the baseline *Grid* (+7.5 in F_1) in SOS. Therefore, we use *DINO* as the object prior in SOS. More generally, the study also reveals important insights on how to prompt SAM for class-agnostic, object-focused results for arbitrary tasks. For more study results, we refer to our supplementary.

5 Evaluation

In our evaluation, we compare SOS to recent OWIS methods OLN [26], LDET [46], GGN [49], SWORD [54], SOIS [56], OpenInst [48], UDOS [24], and against Mask R-CNN [19] and SAM [27] baselines, depending on the availability of public

Table 1: Results of our SOS with various object priors in the cross-category COCO (VOC) → COCO (non-VOC) setting. *: Uses ground truth of unknown classes.

Object Prior	AP	AR_{100}	F_1	Object Prior	AP	AR_{100}	F_1
SOS + *Grid*	3.8	36.5	6.9	SOS + *DeepGaze*	5.4	35.9	9.4
SOS + *Dist*	3.4	27.4	6.0	SOS + *CAM*	5.4	36.7	9.4
SOS + *Spx*	5.6	34.8	9.6	SOS + *DINO*	**8.9**	**38.1**	**14.4**
SOS + *Contour*	5.6	36.6	9.7	SOS + *U-Net*	7.3	37.3	12.2
SOS + *VOCUS2*	6.1	37.7	10.5	SOS + *GT*	18.1*	42.5*	25.4*

source code or reported results. To assess the quality of the approaches, we use four cross-category and cross-dataset settings, commonly used in OWIS. Specifically, we choose COCO (VOC) → COCO (non-VOC) on the COCO dataset [31], already discussed in Sec. 4.2. For cross-dataset evaluation, we use COCO → LVIS, COCO → ADE20k, and COCO → UVO that train on the entire COCO training set and evaluate on the validation sets of LVIS [17], ADE20k [61], and UVO [50] with exhaustive annotations. The LVIS dataset covers the same images as COCO but extends the annotated object classes from 80 to 1203. ADE20k includes 3169 classes of objects, object parts, and stuff regions. Finally, UVO features exhaustive annotations of all objects and, therefore, no system is penalized for detecting objects outside a datasets' taxonomy.

As in Sec. 4.2 and following common practice in OWIS [26,46,49], we report AR for up to 100 detections and AP. We again also report the F_1 score between AR and AP as a single, combined quantity, described in Sec. 4.2. Note that we only evaluate based on masks, not boxes.

5.1 Implementation Details and Data Usage

As discussed in Sec. 4.3, SOS uses the object prior *DINO* for best results. DINO [6] with a ViT-S backbone is used as an ImageNet [45] pre-trained model without fine-tuning. Similarly, SOS uses SAM [27] with the ViT-H backbone as a pre-trained model based on SA-1B [27] without fine-tuning. As an instance segmentation model, we apply Mask R-CNN [19] with an ImageNet pre-trained ResNet-50+FPN backbone [30], and the default configurations except for the class-agnostic training.

Overall, SOS is based on models pre-trained on standard ImageNet and SA-1B. Pre-training on SA-1B or similar large-scale datasets becomes common with the emergence of foundation models and is widely adopted by systems using SAM in similar contexts [7,51,57]. It is comparable to using models pre-trained on ImageNet that even have class information available during training and also allow a rough localization of objects [6,60]. Moreover, the final, noisy annotations of SA-1B representing objects, object parts, and stuff region were automatically generated by SAM [27], and no data leakage w.r.t. other datasets exists.

Table 2: Results of two baselines and various OWIS methods in the COCO (VOC) → COCO (non-VOC) setting. †: uses automatically annotated SA-1B dataset.

System	AP	AR_{100}	F_1
Mask R-CNN [19]	1.0	8.2	1.8
SAM† [27]	3.6	**48.1**	6.7
OLN [26]	4.2	28.4	7.3
LDET [46]	4.3	24.8	7.3
GGN [49]	4.9	28.3	8.4
SWORD [54]	4.8	30.2	8.3
UDOS [24]	2.9	34.3	5.3
SOS† (ours)	**8.9**	39.3	**14.5**

Table 3: Results of two baselines and various OWIS methods in the COCO → LVIS setting. †: uses automatically annotated SA-1B dataset.

System	AP	AR_{100}	F_1
Mask R-CNN [19]	7.5	23.6	11.4
SAM† [27]	6.8	**45.1**	11.8
LDET [46]	6.7	24.8	10.5
GGN [49]	6.5	27.0	10.5
SOIS [56]	-	25.2	-
OpenInst [48]	-	29.3	-
UDOS [24]	3.9	24.9	6.7
SOS† (ours)	**8.1**	33.3	**13.3**

5.2 Cross-category COCO (VOC) → COCO (non-VOC) Results

Table 2 presents the results of various OWIS methods on the cross-category setup COCO (VOC) → COCO (non-VOC). Our SOS clearly outperforms all OWIS methods in AR_{100}, AP, and F_1. Specifically, SOS outperforms the second-best system in terms of F_1, GGN [49], by 6.1. While some of the improvement comes from a better recall (+38.9% compared to GGN), the results are driven by a much-improved precision (+81.6% compared to GGN). This reflects the high quality of the pseudo annotations in SOS on this cross-category setting. Moreover, there is a clear improvement of SOS over Mask R-CNN (+12.7 in F_1). Compared to original SAM, the *DINO*-based SOS substantially improves the precision by focusing on objects, leading to an improved F_1 score (+7.8).

The qualitative results of various OWIS methods in Fig. 4 support the quantitative results. In the first example, only GGN [49] and SOS detect both giraffes, however, SOS generates more accurate segmentations. In the second example, only SOS detects all five small surfboards, while other systems detect at most three. Note that class-agnostic Mask R-CNN misses all non-VOC objects in both examples. For more qualitative results, see our supplementary.

5.3 Cross-dataset Results

COCO → LVIS: We evaluate the cross-dataset generalization capabilities of OWIS methods, starting with COCO → LVIS. The results in Tab. 3 show a trend similar to the cross-category results. SOS outperforms all OWIS methods across all measures with an improvement of 2.8 in F_1 compared to second-best methods GGN [49] and LDET [46]. Similar to the previous results, the improvement is based on gains in both recall and precision, however, the relative improvements in AP and AR_{100} are more similar to each other. Overall, our high-quality pseudo

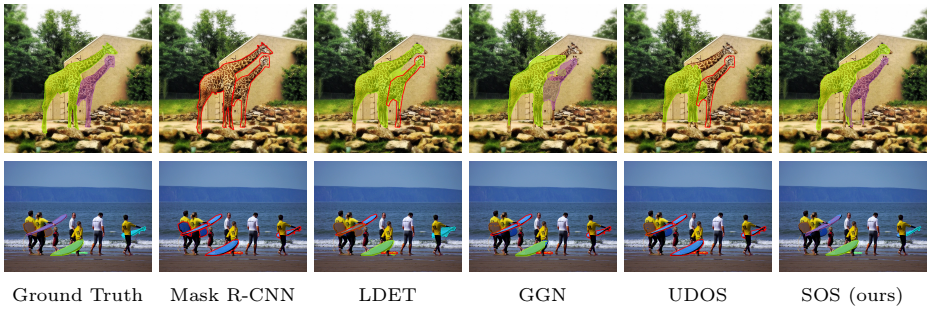


Fig. 4: Qualitative results of OWIS methods and baseline Mask R-CNN in the cross-category COCO (VOC) → COCO (non-VOC) setting. Filled masks denote detected objects, while red frames indicate missed objects.

Table 4: Results of Mask R-CNN baseline and various OWIS methods in the COCO → ADE20k setting. †: uses automatically annotated SA-1B dataset.

System	AP	AR_{100}	F_1
Mask R-CNN [19]	6.9	11.9	8.7
OLN [26]	-	20.4	-
LDET [46]	9.5	18.5	12.6
GGN [49]	9.7	21.0	13.3
UDOS [24]	7.6	22.9	11.4
SOS† (ours)	**12.5**	**26.5**	**17.0**

Table 5: Results of two baselines and various OWIS methods in the COCO → UVO setting. †: uses automatically annotated SA-1B dataset.

System	AP	AR_{100}	F_1
Mask R-CNN [19]	20.7	36.7	26.5
SAM† [27]	11.3	**50.1**	18.4
OLN [26]	-	41.4	-
LDET [46]	**22.0**	40.4	**28.5**
GGN [49]	20.3	43.4	27.7
UDOS [24]	10.6	43.1	17.0
SOS† (ours)	20.9	42.3	28.0

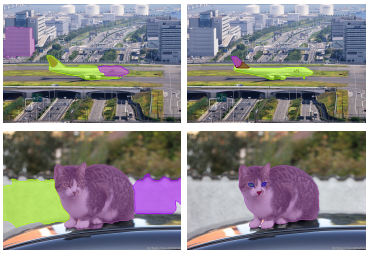
annotations in SOS lead to new state-of-the-art results, which indicate strong generalization to unknown object classes outside COCO.

COCO → ADE20k: The results for COCO → ADE20k in Tab. 4, which include labeled object parts of ADE20k, show that SOS again outperforms all other OWIS methods across all measures. The gain over the second-best method in F_1, GGN [49], is 3.7. Similar to COCO → LVIS, the relative gains in AR_{100} and AP are equally distributed. These results indicate that object parts do not degrade the results of SOS. This is in line with SOS's pseudo annotations in Fig. 3, covering entire objects and selected object parts.

COCO → UVO: Finally, Tab. 5 presents results for COCO → UVO. SOS still outperforms Mask R-CNN, SAM and recent UDOS [24] in F_1. Outperforming SAM implies that SOS does not exploit the limited taxonomy of annotations in COCO (only 60 object classes in test), but also improves the segmentation of objects outside the COCO object classes on the exhaustively labeled UVO

Table 6: Quality of pseudo annotation generated in GGN [49] and our SOS evaluated on the non-VOC annotations of the COCO training dataset. †: uses automatically annotated SA-1B dataset.

System	Prec.	Rec.	F_1
GGN_3 [49]	7.3	12.1	9.1
$SOS_3^\dagger$ (ours)	**19.0**	26.4	22.1
$SOS_{10}^\dagger$ (ours)	15.5	**41.7**	**22.6**



GGN SOS

Fig. 5: Pseudo annotations generated by GGN and our SOS.

dataset. However, the results of SOS are below LDET [46] in AP and F_1. We attribute this to several classes in UVO not available in ImageNet, which was used to learn *DINO* object prior. Hence, *DINO* will miss such objects in the COCO dataset during training, and SOS might miss them during testing. This could be mitigated using more diverse unlabeled images for training *DINO*.

5.4 Quality of Pseudo Annotations

We also investigate the quality of the pseudo annotations generated in SOS. To this end, we evaluate the pseudo annotations on the COCO training set against the annotations of non-VOC object classes, reflecting the COCO (VOC) $\rightarrow$ COCO non-VOC) setting. The results in terms of precision and recall for IoU = 0.5 as well as the F_1 score are presented in Tab. 6. We use up to three and 10 pseudo annotations generated per image in SOS (SOS_3 and SOS_{10}), and GGN_3 [49] that generates three pseudo annotations. The results show that with only three SOS pseudo annotations, more than 25% of the non-VOC objects are covered. This is substantially more than GGN_3 (12.1%). With up to 10 annotations, SOS's pseudo annotations cover more than 40% of the non-VOC objects. In our supplementary, we further provide class-specific results.

To complement these results, we visualize the annotations of GGN_3 and SOS_3 on two sample images in Fig. 5. It is visible that GGN_3's annotations cover background regions, leading to low precision in the overall results. Moreover, the pseudo annotations in the upper example do not adhere well to the airplane's boundaries. More qualitative examples are given in Fig. 1 and our supplementary.

5.5 Ablation Studies

All ablation studies follow the evaluation setup described in Sec. 4.2. More studies evaluating design choices in SOS are presented in our supplementary.

Influence of SOS's Components: We investigate the importance of each component in SOS and present the results in Tab. 7. The baseline for this

Table 7: Influence of added components in our SOS evaluated in the COCO (VOC) → COCO (non-VOC) setting.

Components	AP	AR_{100}	F_1
Mask R-CNN	1.2	10.8	2.2
+ *Grid*, w/o pp.	3.4	35.2	6.2
+ *DINO*, w/o pp.	**8.9**	37.1	14.3
+ *DINO*, w/ pp.	**8.9**	**38.1**	**14.4**

Table 8: SOS results with various numbers of pseudo annotations in the COCO (VOC) → COCO (non-VOC) setting.

#Pseudo Anns.	AP	AR_{100}	F_1
3	8.3	34.5	13.4
5	8.8	36.2	14.2
10	**8.9**	**38.1**	**14.4**
20	8.8	**38.1**	14.3

study is a class-agnostic Mask R-CNN trained without pseudo annotations (first row in Tab. 7). Subsequently, we add pseudo annotations based on SAM with the baseline *Grid* object prior as in [27] and without our post-processing in PAC (confidence-based thresholding and NMS, see Sec. 3.3). Next, we replace *Grid* with *DINO* (third row in Tab. 7), and finally add the post-processing leading to the complete SOS. The results in Tab. 7 show each step improves the F_1 results. Mainly, introducing SAM for generating pseudo annotations (first row vs. second row) substantially improves AR_{100}, while adding the *DINO* object prior removes background annotations leading to considerably improved precision. Hence, the object prior selection in SOS is crucial for strong results.

Number of Pseudo Annotations: In the second ablation study, we investigate the influence of the number of pseudo annotations per image on the results of SOS. To this end, we evaluate SOS with up to 3, 5, 10, and 20 pseudo annotations per image. The results in Tab. 8 show that 10 pseudo annotations are preferable. Hence, the ideal number of pseudo annotations in SOS is larger than in GGN [49], which only uses three. This indicates again the high quality of our pseudo annotations, since more than three annotations per image can cover relevant objects without adding too many noisy annotations.

6 Conclusion

In this paper, we addressed the challenging OWIS task and aimed to improve the low precision of previous methods in this open-world task. To improve the precision and the overall detection results, we generate high-quality pseudo annotations based on a prompt-guided foundation model, SAM. To focus the pseudo annotations on objects, we thoroughly investigated various object priors for prompt creation, which revealed important insights on how to prompt SAM for class-agnostic object-focused results in arbitrary tasks. As a result, our novel OWIS method SOS, which uses a self-attention-based object prior for generating pseudo annotations, outperforms recent state-of-the-art OWIS methods across challenging open-world setups in COCO, LVIS, and ADE20k datasets. SOS especially improves the precision based on the high-quality pseudo annotations. Moreover, SOS with its focus on objects also outperforms vanilla SAM that segments both object and stuff regions.

References

1. Ahn, J., Cho, S., Kwak, S.: Weakly supervised learning of instance segmentation with inter-pixel relations. In: Conference on Computer Vision and Pattern Recognition (2019)
2. Alexe, B., Deselaers, T., Ferrari, V.: What is an object? In: Conference on Computer Vision and Pattern Recognition (2010)
3. Baugh, M., Batten, J., Müller, J.P., Kainz, B.: Zero-shot anomaly detection with pre-trained segmentation models. arXiv preprint arXiv:2306.09269 (2023)
4. Bommasani, R., Hudson, D.A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M.S., Bohg, J., Bosselut, A., Brunskill, E., et al.: On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258 (2021)
5. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in Neural Information Processing Systems **33** (2020)
6. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: International Conference on Computer Vision (2021)
7. Chen, T., Mai, Z., Li, R., Chao, W.l.: Segment anything model (SAM) enhanced pseudo labels for weakly supervised semantic segmentation. arXiv preprint arXiv:2305.05803 (2023)
8. Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H., Hu, S.M.: Global contrast based salient region detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **37**(3) (2014)
9. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (2019)
10. Dollár, P., Zitnick, C.L.: Structured forests for fast edge detection. In: International Conference on Computer Vision (2013)
11. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2020)
12. Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The PASCAL visual object classes challenge: A retrospective. International Journal of Computer Vision **111** (2015)
13. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. International Journal of Computer Vision **59** (2004)
14. Frintrop, S., Werner, T., Martin Garcia, G.: Traditional saliency reloaded: A good old model in new shape. In: Conference on Computer Vision and Pattern Recognition (2015)
15. Gao, G., Lauri, M., Zhang, J., Frintrop, S.: Saliency-guided adaptive seeding for supervoxel segmentation. In: International Conference on Intelligent Robots and Systems (2017)
16. Girshick, R.: Fast R-CNN. In: International Conference on Computer Vision (2015)
17. Gupta, A., Dollár, P., Girshick, R.: LVIS: A dataset for large vocabulary instance segmentation. In: Conference on Computer Vision and Pattern Recognition (2019)
18. He, C., Li, K., Zhang, Y., Xu, G., Tang, L., Zhang, Y., Guo, Z., Li, X.: Weakly-supervised concealed object segmentation with SAM-based pseudo labeling and multi-scale feature grouping. arXiv preprint arXiv:2305.11003 (2023)

19. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: International Conference on Computer Vision (2017)
20. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Conference on Computer Vision and Pattern Recognition (2016)
21. Hou, Q., Cheng, M.M., Hu, X., Borji, A., Tu, Z., Torr, P.H.: Deeply supervised salient object detection with short connections. In: Conference on Computer Vision and Pattern Recognition (2017)
22. Hu, H., Lan, S., Jiang, Y., Cao, Z., Sha, F.: FastMask: Segment multi-scale object candidates in one shot. In: Conference on Computer Vision and Pattern Recognition (2017)
23. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: International Conference on Machine Learning (2021)
24. Kalluri, T., Wang, W., Wang, H., Chandraker, M., Torresani, L., Tran, D.: Open-world instance segmentation: Top-down learning with bottom-up supervision. arXiv preprint arXiv:2303.05503 (2023)
25. Kim, B., Yoo, Y., Rhee, C.E., Kim, J.: Beyond semantic to instance segmentation: Weakly-supervised instance segmentation via semantic knowledge transfer and self-refinement. In: Conference on Computer Vision and Pattern Recognition (2022)
26. Kim, D., Lin, T.Y., Angelova, A., Kweon, I.S., Kuo, W.: Learning open-world object proposals without learning to classify. IEEE Robotics and Automation Letters **7**(2) (2022)
27. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: International Conference on Computer Vision (2023)
28. Klein, D.A., Frintrop, S.: Center-surround divergence of feature statistics for salient object detection. In: International Conference on Computer Vision (2011)
29. Li, J., Li, D., Savarese, S., Hoi, S.: BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. arXiv preprint arXiv:2301.12597 (2023)
30. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Conference on Computer Vision and Pattern Recognition (2017)
31. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: European Conference on Computer Vision (2014)
32. Linardos, A., Kümmerer, M., Press, O., Bethge, M.: DeepGaze IIE: Calibrated prediction in and out-of-domain for state-of-the-art saliency modeling. In: International Conference on Computer Vision (2021)
33. Liu, N., Han, J., Yang, M.H.: PiCANet: Learning pixel-wise contextual attention for saliency detection. In: Conference on Computer Vision and Pattern Recognition (2018)
34. Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., Shum, H.Y.: Learning to detect a salient object. IEEE Transactions on Pattern analysis and machine intelligence **33**(2) (2010)
35. Liu, Y., Wu, Y.H., Wen, P., Shi, Y., Qiu, Y., Cheng, M.M.: Leveraging instance-, image-and dataset-level information for weakly supervised instance segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(3) (2020)
36. Ma, J., Ming, A., Huang, Z., Wang, X., Zhou, Y.: Object-level proposals. In: International Conference on Computer Vision (2017)

37. Martin García, G., Potapova, E., Werner, T., Zillich, M., Vincze, M., Frintrop, S.: Saliency-based object discovery on RGB-D data with a late-fusion approach. In: International Conference on Robotics and Automation (2015)
38. Mazurowski, M.A., Dong, H., Gu, H., Yang, J., Konz, N., Zhang, Y.: Segment anything model for medical image analysis: an experimental study. Medical Image Analysis **89** (2023)
39. Perazzi, F., Krähenbühl, P., Pritch, Y., Hornung, A.: Saliency filters: Contrast based filtering for salient region detection. In: Conference on Computer Vision and Pattern Recognition (2012)
40. Pinheiro, P.O., Collobert, R., Dollár, P.: Learning to segment object candidates. Advances in Neural Information Processing Systems **28** (2015)
41. Pinheiro, P.O., Lin, T.Y., Collobert, R., Dollár, P.: Learning to refine object segments. In: European Conference on Computer Vision (2016)
42. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning (2021)
43. Rahtu, E., Kannala, J., Blaschko, M.: Learning a category independent object detection cascade. In: International Conference on Computer Vision (2011)
44. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention (2015)
45. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision **115** (2015)
46. Saito, K., Hu, P., Darrell, T., Saenko, K.: Learning to detect every thing in an open world. In: European Conference on Computer Vision (2022)
47. Shi, Z., Sun, Y., Zhang, M.: Training-free object counting with prompts. In: Winter Conference on Applications of Computer Vision (2024)
48. Wang, C., Wang, G., Zhang, Q., Guo, P., Liu, W., Wang, X.: OpenInst: A simple query-based method for open-world instance segmentation. arXiv preprint arXiv:2303.15859 (2023)
49. Wang, W., Feiszli, M., Wang, H., Malik, J., Tran, D.: Open-world instance segmentation: Exploiting pseudo ground truth from learned pairwise affinity. In: Conference on Computer Vision and Pattern Recognition (2022)
50. Wang, W., Feiszli, M., Wang, H., Tran, D.: Unidentified video objects: A benchmark for dense, open-world segmentation. In: International Conference on Computer Vision (2021)
51. Wei, Z., Chen, P., Yu, X., Li, G., Jiao, J., Han, Z.: Semantic-aware SAM for point-prompted instance segmentation. arXiv preprint arXiv:2312.15895 (2023)
52. Wilms, C., Frintrop, S.: Edge adaptive seeding for superpixel segmentation. In: German Conference on Pattern Recognition (2017)
53. Wilms, C., Frintrop, S.: AttentionMask: Attentive, efficient object proposal generation focusing on small objects. In: Asian Conference on Computer Vision (2019)
54. Wu, J., Jiang, Y., Yan, B., Lu, H., Yuan, Z., Luo, P.: Exploring transformers for open-world instance segmentation. In: International Conference on Computer Vision (2023)
55. Xie, C., Xiang, Y., Mousavian, A., Fox, D.: Unseen object instance segmentation for robotic environments. IEEE Transactions on Robotics **37**(5) (2021)

56. Xue, X., Yu, D., Liu, L., Liu, Y., Li, Y., Yuan, Z., Song, P., Shou, M.Z.: Single-stage open-world instance segmentation with cross-task consistency regularization. arXiv preprint arXiv:2208.09023 (2022)
57. Yang, X., Gong, X.: Foundation model assisted weakly supervised semantic segmentation. In: Winter Conference on Applications of Computer Vision (2024)
58. Yang, Y., Zhou, Z., Wu, J., Wang, Y., Xiong, R.: Class semantics modulation for open-set instance segmentation. IEEE Robotics and Automation Letters (2024)
59. Zhao, J.X., Liu, J.J., Fan, D.P., Cao, Y., Yang, J., Cheng, M.M.: EGNet: Edge guidance network for salient object detection. In: International Conference on Computer Vision (2019)
60. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Conference on Computer Vision and Pattern Recognition (2016)
61. Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., Torralba, A.: Semantic understanding of scenes through the ADE20K dataset. International Journal of Computer Vision **127** (2019)
62. Zhu, M., Li, H., Chen, H., Fan, C., Mao, W., Jing, C., Liu, Y., Shen, C.: SegPrompt: Boosting open-world segmentation via category-level prompt learning. In: International Conference on Computer Vision (2023)
63. Zitnick, C.L., Dollár, P.: Edge boxes: Locating object proposals from edges. In: European Conference on Computer Vision (2014)