Learning with Free Object Segments for Long-Tailed Instance Segmentation

Cheng Zhang, Tai-Yu Pan, Tianle Chen,
Jike Zhong, Wenjin Fu, and Wei-Lun Chao

The Ohio State University, Columbus OH 43210, USA

Abstract. One fundamental challenge in building an instance segmentation model for a large number of classes in complex scenes is the lack of training examples, especially for rare objects. In this paper, we explore the possibility to increase the training examples without laborious data collection and annotation. We find that an abundance of instance segments can potentially be obtained freely from object-centric images, according to two insights: (i) an object-centric image usually contains one salient object in a simple background; (ii) objects from the same class often share similar appearances or similar contrasts to the background. Motivated by these insights, we propose a simple and scalable framework FreeSeg for extracting and leveraging these "free" object foreground segments to facilitate model training in long-tailed instance segmentation. Concretely, we investigate the similarity among object-centric images of the same class to propose candidate segments of foreground instances, followed by a novel ranking of segment quality. The resulting high-quality object segments can then be used to augment the existing long-tailed datasets, *e.g.*, by copying and pasting the segments onto the original training images. Extensive experiments show that FreeSeg yields substantial improvements on top of strong baselines and achieves state-of-the-art accuracy for segmenting rare object categories.

1 Introduction

Object detection and instance segmentation are fundamental building blocks for many high-impact real-world applications (*e.g.*, autonomous driving). Recent years have witnessed an unprecedented breakthrough in both of them, thanks to deep neural networks [33,16,14] and large-scale datasets for common objects (*e.g.*, persons and cars) [26,12,64]. Yet, when it comes to rare, less commonly seen objects (*e.g.*, an unusual traffic sign), there is a drastic performance drop due to insufficient training examples [15,39,59]. This challenge has attracted significant attention lately in how to learn an object detection or instance segmentation model given labeled data of a "long-tailed" distribution across classes [62]. Specifically, a number of works have been dedicated to developing new training algorithms, objectives, or model architectures [28,19,24,51,45,52,15,48,44,49].

In this paper, we explore a drastically different approach. *We investigate the possibility of obtaining more labeled instances (*i.e.*, instance segments of objects) at a minimal cost, especially for rare objects.* We build upon the recent

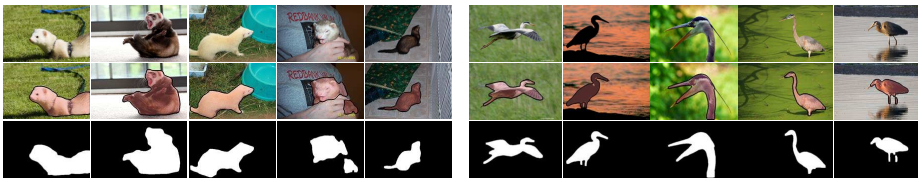

Fig. 1. Illustration of our approach FreeSeg**.** We sample two rare classes, *ferret* (left) and *heron* (right) from LVIS v1 [15], and retrieve object-centric images (the upper row of each class) from the ImageNet dataset [37]. We then show the discovered object segments (middle) and binary masks (bottom) by FreeSeg. The abundant object segments have diverse appearances and poses and can be effectively used to improve the instance segmentation.

observation in [56] — many objects do not appear frequently enough in complex scenes but are found frequently alone in object-centric images — to acquire an abundance of object-centric images (*e.g.*, ImageNet [11] or Google images) for rare classes. Zhang *et al.* [56] have shown that, even with only pseudo bounding boxes for these images, they can already improve the detector effectively.

We take one key step further to leverage the underlying properties of object-centric images to create high-quality instance labels that can facilitate both detection and segmentation model training. In general, object-centric images usually contain one salient object in a relatively simple background than scene-centric images like those in MSCOCO [26]. Moreover, objects of the same class usually share similar appearances, shapes, contrasts, or more abstractly, common parts to the background [35] (see Figure 1 for an example). These properties open up the opportunity to discover object segments almost *freely* from object-centric images of the same class — by exploring their common salient regions.

To this end, we propose a framework named FreeSeg (Free Object Segments) to take advantage of these properties. We first extract the common foreground regions from object-centric images of the same class. This can be done, for example, by off-the-shelf co-segmentation models [57]. While not perfect, sometimes missing the true objects or including backgrounds, these extracted regions have surprisingly captured a decent portion of objects with tight segmentation masks. Nevertheless, directly using all of these regions, mixed with false positive and noisy segments, would inevitably introduce a great amount of noise to the downstream tasks. To address this, we propose a novel segment ranking approach to mine the most reliable and high-confident object masks. After all, we aim for a set of high-quality instance segments from object-centric images, not to segment all the object-centric images well.

How can we leverage these high-quality instance segments from object-centric images? One naive way is to directly train the instance segmentation model on the object-centric images, using these segments as supervision. Nevertheless, the fact that these objects mostly show up alone in simple backgrounds makes them somewhat too simple for the model. We, therefore, choose to place these object segments in the context of complex scene-centric images, via simple copy-paste augmentation [13]. Unlike [13], which merely pastes human-annotated segments

from one image to another to increase the *context diversity*, our FreeSeg approach brings the best of abundant free object segments to increase the *appearance diversity*, especially for rare object categories.

We evaluate FreeSeg on the long-tailed LVIS benchmark [15]. FreeSeg leads to a massive improvement in segmenting rare object instances by effectively increasing the labeled training data for them. Moreover, FreeSeg is detached from the model training phase and is thus model-agnostic. Namely, it can potentially benefit all kinds of instance segmentation model architectures. FreeSeg is also compatible with existing efforts on long-tailed object detection and segmentation to achieve further gains.

In summary, our FreeSeg framework opens up a novel direction that brings the best of discovering pixel-level supervision in object-centric images to facilitate long-tailed instance segmentation. Our **main contributions** are:

- We demonstrate the possibility to increase the number of training examples for instance segmentation without laborious pixel-level data collection and annotation.
- We propose a simple and scalable pipeline for discovering, extracting, and leveraging free object foreground segments to facilitate long-tailed instance segmentation.
- Our FreeSeg framework shows promising gains on the challenging LVIS dataset and demonstrates a strong compatibility with existing works.

2 Related Work

Long-tailed object detection and instance segmentation. Most existing works tackle the problem of "long-tailed" distributions in the model training phase, by developing training objectives or algorithms [51,49,21,19,32,24,58,50]. They usually first pre-train the models in a conventional way, using data from all or just the head classes, and then fine-tunes them on the entire long-tailed dataset using either re-sampling [15,5,40] or cost-sensitive learning [45,44,18,48]. Instead of directly learning a model from long-tailed data, another thread of works investigate data augmentation techniques to improve the performance of long-tailed object detection and instance segmentation [13,30,56,55]. For example, Simple Copy-Paste [13] augments the training data in the image space using the original long-tailed dataset. FASA [55] enhances class-wise features using a Gaussian prior. DLWL [30] and MosaicOS [56] extensively leverage extra data sources from YFCC-100M [46], ImageNet [11] or Internet to augment the long-tailed LVIS dataset [15].

Our work follows the second thread on learning with additional weakly-supervised or unsupervised data, similar to the recently proposed MosaicOS framework [56]. We, however, further develop an effective way to obtain high-quality instance segments from object-centric images, while MosaicOS merely learns with pseudo bounding box annotations. Since collecting pixel-level annotations is more challenging and prone to error, we develop a novel ranking mech-

anism such that only the high-quality segments will be used for model training. Moreover, by copying and pasting the segments into the context of scene-centric images, our method can further bridge the domain gap between different data sources. Overall, we view our approach as a critical leap upon [56,13] that can significantly improve long-tailed instance segmentation by largely increasing the training segments of rare objects.

Image-based foreground object segmentation. There are a variety of techniques that we can potentially leverage to extract the foreground object segments from object-centric images without laborious annotations. Representative methods include image (co-)saliency detection [20,9], unsupervised/weakly-supervised object segmentation [34], attention [3], instance localization [60,38], and image co-segmentation [35,7,57]. The purpose of our work is thus not to propose a new way or compare to those methods, but to investigate approaches that are more effective and efficient for the large-scale long-tailed setting. In this paper, we mainly focus on one potential solution for segmenting foreground objects: image co-segmentation. Aiming at jointly segmenting the common foreground regions from a group of images, co-segmentation is very useful in many semantic labeling tasks [4,7] and is a direct fit to the object-centric images we collect. Existing image co-segmentation models are usually trained and evaluated on relatively small-scale benchmarks such as MSRC [41], Internet [36], iCoseg [2], PASCAL-VOC [12], etc. Our work is almost the first attempt to test the generalizability of existing, pre-trained co-segmentation models on a much larger-scale setting that contains more than $1,000$ categories; each category consists of hundreds or thousands of object-centric images with various appearances and poses. As will be shown in the experimental results and analyses, our framework can effectively utilize the off-the-shelf image co-segmentation models.

3 Approach

Our FreeSeg (Free Object Segments) framework for data augmentation is fairly simple and scalable for large-vocabulary and long-tailed instance segmentation. Figure 2 illustrates the overall pipeline, which consists of three major steps: (i) segment generation and refinement, (ii) segment ranking, and (iii) data synthesis for model training.

3.1 Generating Object Segments

We assume that we can obtain a sufficient amount of object-centric images for each class of interest. As discussed in [56], this is mostly doable. We can take advantage of existing image classification datasets like ImageNet [11] or leverage image search engine (*e.g.*, Google Images).

Raw segments generation. Given object-centric images of the same class, *which usually share similar appearances or contrasts to the background*, we apply image co-segmentation techniques [35,57] to extract their common foreground regions. Without loss of generality, we use the state-of-the-art image

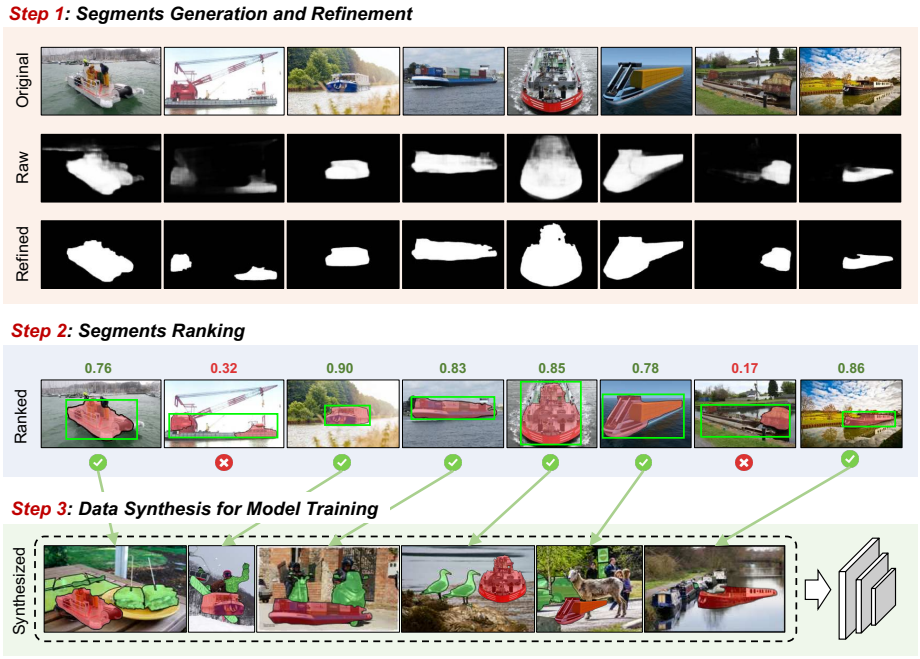

Fig. 2. Illustration of the FreeSeg pipeline. We show a rare class *Barge* in LVIS v1 [15] as the example. We first perform image co-segmentation on top of the object-centric images of *Barge* (outside LVIS v1) to obtain raw object segments, followed by segments refinement. The segments are then scored by a learned ranker (the green boxes in step 2) such that only the high-quality ones would be used for augmenting data for model training. Finally, we randomly paste the selected object segments (red) onto the original scene-centric images of LVIS v1 to improve the long-tailed instance segmentation. Green segments indicate the original objects in scene-centric images.

co-segmentation algorithms, Spatial and Semantic Modulation (SSM) [57]. The outputs of SSM are raw segments in gray scales for each image, as shown in Figure 2 (see Section 4.1 for more details). Please be referred to Section 2 for other potential algorithms for this stage.

Post-processing for segment refinement. To turn the raw, grayscale segmentation map into a binary one that can be used to train a segmentation model, we threshold the map. As the suitable threshold value may vary across images and classes, we apply a Gaussian filter followed by dynamic thresholding, *i.e.*, Li thresholding [23,22], which minimizes the cross-entropy between the foreground and the background to find the optimal threshold to distinguish them. To further improve the resulting binary map, we apply erosion and dilation to smooth the boundary. Finally, we then remove small, likely false positive segments by only keeping the largest connected component in the binary map. Figure 2 (Step 1) gives an illustration. Please also see supplementary materials for more details.

3.2 Learning to Rank the Segments

While the post-processing step has greatly improved the binary masks and made them look more like the true object masks, they may occasionally miss the target objects (*i.e.*, the objects of the image labels) or include background pixels. This is not surprising: we apply image co-segmentation class-by-class to only explore the within-class similarity. Some co-occurring objects (*e.g.*, persons for unicycles) thus may be miss-identified as the target objects; some target objects that are too small may be dominated by other objects.

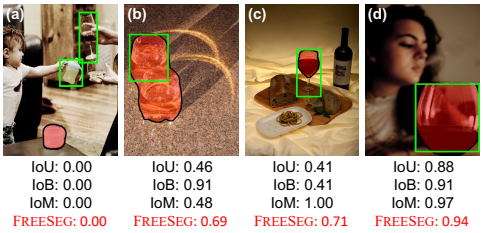

Fig. 3. Comparison of metrics for ranking segments. We show four examples of the class *wine glasses*. The red masks are by our method; green boxes are by LORE. In (a) and (d), IoU ranks the segments well, when the box locations are precise. However, in (b), the poor box location leads to a small IoU, even if the segment is precise. In (c), IoU fails due to the specific shape of *wine glasses*, even if the segment is precise. FREESEG score is able to take all the above into account to faithfully rank segments.

At first glance, this seems to paint a grim picture. However, as mentioned in Section 1, our ultimate goal is to *obtain a set of high-quality instance segments from object-centric images, not to segment all the object-centric images well*. Therefore, in the second stage, we develop a novel approach to rank the object segments for each class. Specifically, we aim to select images whose masks truly cover the target objects and are as tight as possible to them.

Ranking by learning a classifier. Given an object segment obtained from co-segmentation, how can we determine if the segment truly covers the target object? Here, we take one intuition: *if a segment covers the target object, then by removing it from the image, an image classifier*[1] *will unlikely classify the manipulated image correctly.* This idea has indeed been used in [56] to discover pseudo bounding boxes given only image labels. More specifically, the authors developed "localization by region removal (LORE)", which sequentially removes bounding box regions from an image till the image classifier fails to predict the right class. Those removed bounding boxes are then treated as pseudo bounding boxes for the target object class.

We thus adopt the idea of LORE to rank our object segments. But instead of removing the discovered segments and checking the classifier's failure, we directly compare our object segments to the bounding boxes selected by LORE. *In essence, if the LORE boxes and our segments are highly overlapped, then the segments are considered high-quality.*

Ranking metrics. Arguably the most common way to characterize the overlap/agreement between two masks/boxes is intersection over union (IoU), which

[1] We have image labels for object-centric images, and thus we can train an image classifier upon them.

simply treats all contents in a box or mask as foreground. However, this metric is not suitable in our case for the following reasons: (i) both boxes and segments may be noisy, and simply measuring the IoU between them fails to rank good segments when the boxes are poor; (ii) object shapes are not always convex, and thus IoU may underestimate the agreement. As shown in Figure 3, IoU fails to recall true positives.

We therefore propose the FreeSeg score to rank the segments. We make one mild assumption: either the object box or the segment is trustable, and introduce two metrics: intersection over bounding box (IoB) and intersection over mask (IoM). While they share the same numerator with intersection over union (IoU), they have different denominators. IoM implies that the bounding box is precise and measures how much portion of the mask is inside the box, and vice versa for IoB. We take both into account by averaging them as our FreeSeg score. As shown in Figure 4, it effectively keeps the good segments in the pool.

Drop rate by the classifier. We introduce another metric for ranking the segment or, more precisely, its corresponding object-centric image. The rationale is, if an object does not clearly show up in an image (*e.g.*, occluded or of small sizes), then the obtained segment is unlikely accurate. To this end, we leverage the image classifier trained for LORE, and compute the drop rate — the classifier's relative confidence drop for the target class, before and after LORE box removal. Let $s(c)$ and $s'(c)$ denote the classifier's confidence of the target class c before and after LORE box removal from the image, the drop rate is $\frac{s(c)-s'(c)}{s(c)}$. The drop rate indicates how easily, by removing LORE's localized target objects, would the classifier's confidence reduce. The larger the drop is, the easier the localization of target objects is, and thus the higher-quality the object-centric image is. See Figure 4 for an illustration on the drop rate.

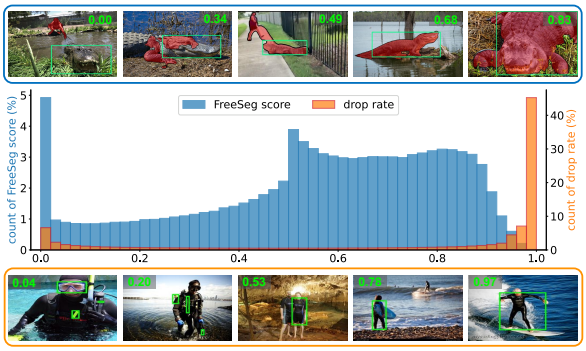

Fig. 4. Ranking the object segments. We apply the FreeSeg score and the drop rate to select high-quality segments/images. We show the LORE boxes in green and discovered segments in red. In the top images of the class *alligator*, FreeSeg scores on the upper right corner of images imply the alignment between the segments and the box locations. In the bottom images of the class *wet suit*, drop rates on the upper left corner of images indicate the quality of object-centric images (the larger, the better). These metrics are shown to be effective to rank the segments/images. For both metrics, we simply set a threshold 0.5 to discard low-confidence segments and images.

Ranking the segments. We use both the FreeSeg score and the drop rate to rank the object-centric images and their co-segmentation segments. We keep those with both scores larger than 0.5 as the high-quality segments.

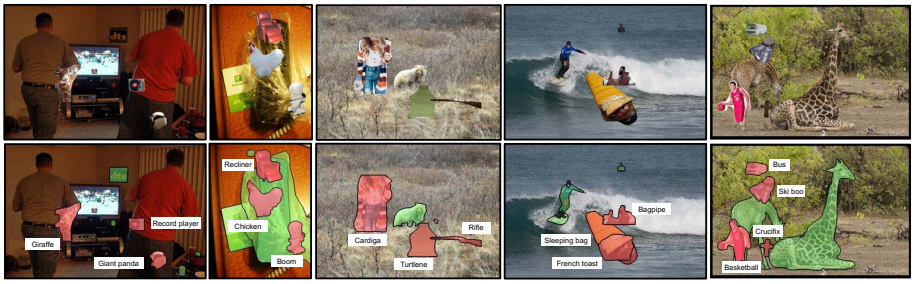

Fig. 5. Synthesized examples via FreeSeg**.** We generate object segments from object-centric images and randomly paste them onto scene-centric images. Red masks indicate pasted segments by FreeSeg; green masks indicate original objects in scene-centric images. Please see supplementary materials for more examples.

3.3 Putting the Segments in the Context

We now describe how we leverage the discovered high-quality instance segments to facilitate segmentation model training. As discussed in Section 1, instead of directly training the model with the segments on top of object-centric images, we choose to synthesize more scene-centric alike examples by pasting the segments into labeled scene-centric images (*e.g.*, those in LVIS v1 [15]). We adopt the idea of simple copy-paste augmentation [13] for this purpose. Specifically, we randomly (i) sample several object-centric images, (ii) re-scale and horizontally flip the object segments, and (iii) paste them onto the scene-centric images from the original training set (see Section 4.1 for details). The resulting synthesized images (see Figure 5) then can be used to improve model training.

4 Experiments

4.1 Setup

Dataset and evaluation metrics. We validate our approach on the LVIS v1 instance segmentation benchmark [15]. (See the supplementary materials for the results on COCO-LT [49].) The dataset contains 1,203 entry-level object categories with around 2 million high-quality instance annotations. The training set contains 100,170 images for all the classes; the validation set contains 19,809 images for 1,035 classes. The categories follow a long-tailed distribution and are divided into three groups based on the number of training images: rare (1-10 images), common (11-100 images), and frequent (>100 images). *We report our results on the validation set by convention.* We adopt the standard mean average precision (AP) metric [15], which sets the cap of detected objects per image to 300. We denote the AP for rare, common, and frequent classes as AP_r, AP_c, and AP_f, respectively. We also report AP for bounding boxes (*i.e.*, AP^b), predicted by the same instance segmentation models. Following [53,15], we set the score threshold to 1×10^{-4} during testing. No test-time augmentation is used.

Table 1. Statistics of LVIS training data and the augmented data by FreeSeg. Collected: # of all images collected from ImageNet-22K and Google Images. Selected via FreeSeg: # of remaining images selected by segments ranking. Note that our data curation process is quite straightforward and fully automated.

# of samples	ImageNet	Google Images	Total
Original instance	–	–	1,270K
Original image	–	–	100K
Collected	1,242K	588K	1,830K
Selected via FreeSeg	662K	304K	966K

Object-centric data sources. We follow [56] to search images in ImageNet-22K [11] and Google Images [1]. Specifically, we use the unique WordNet synset ID [27] to match the categories between ImageNet-22K and LVIS v1. We are able to match 997 LVIS classes and retrieve $1,242,180$ images from ImageNet. Because ImageNet images are nearly balanced by design, with around 1K images/class, the imbalance situation in LVIS can be largely reduced. In addition, we retrieve images via Google by querying with class names provided by LVIS. Such a search returns hundreds of iconic images and we take top 500 for each of the $1,203$ classes. Overall, for the rarest class (one image in LVIS), the increase factor is larger than 500 times. Please see the supplementary material for more details.

Image co-segmentation algorithm. We adopt the state-of-the-art image co-segmentation algorithm Spatial and Semantic Modulation (SSM) [57] to discover raw segments of objects from the object-centric images. SSM designs a spatial and semantic modulated deep network to jointly learn the structural and semantic information from the objects in the same class. The checkpoint of released SSM model is pre-trained on COCO-SEG dataset [47] with a VGG16 backbone [42]. We directly apply the model on all the object-centric images for each category without bells and whistles.

Learning an object segments ranker. As mentioned in Section 3.2, we train a 1203-way classifier with a ResNet-50 backbone [17], using all the object-centric images, to rank the candidate segments within each class. We use a batch size 256 and follow the standard training schedule. The classifier achieves 85% Top-1 accuracy on the training images. We use the idea of "localization by region removal" (LORE) [56] to detect the bounding boxes of objects. Table 1 shows the statistics of the augmented data before and after the ranking via FreeSeg.

Base models for instance segmentation. We mainly evaluate the performance of FreeSeg using two base models for instance segmentation, *i.e.*, Mask R-CNN [16] and MosaicOS [56], implemented with [53]. Both models use ResNet [17], which is pre-trained on ImageNet [37], with a Feature Pyramid Network (FPN) [25] as the backbone. The base Mask R-CNN model is trained with the LVIS v1 training set with *repeated factor sampling* and follows the standard training procedure in [15] (1× scheduler).

MosaicOS [56] is one of the state-of-the-art models[2], which is further pre-trained with balanced object-centric images from ImageNet-22K and Google Images. However, MosaicOS mainly focuses on improving long-tailed object detection with pseudo-labeled bounding boxes. As will be shown in the experimental results, FreeSeg can notably boost the performance upon MosaicOS with the same image resources. Furthermore, such an improvement can not be achieved by the vanilla simple copy-paste [13] using the training data from LVIS alone, especially for rare object categories.

Details of object segments pasting. We follow the pasting mechanism in [13] to randomly pick examples from LVIS training set as the background images. We then paste segments from N random object-cetric images at different locations of each background image, where N is in [1, 6][3]. For LVIS images, we follow the standard data augmentation policy in [15] and [53]. For binary masks originally on LVIS images, we remove pixel annotations if the objects are occluded by the pasted ones in the front. Please see the supplementary material for more details.

Training and optimization. Given the base instance segmentation model, we first fine-tune the model for 90K iterations with FreeSeg segments, using all the loss terms in Mask R-CNN. We fine-tune all the parameters except the batch-norm layers in the backbone. We then fine-tune the model again for another 90K iterations using the original LVIS training images. The rationale of training with multiple stages is to prevent the augmented instances from dominating the training process (see Table 1 for statistics) and it is shown to be effective in [56]. Both fine-tuning steps are trained with stochastic gradient descent with a batch size of 8, momentum of 0.9, weight decay of 10^{-4}, and learning rate of 2×10^{-4}. All models are trained with four NVIDIA A6000 GPUs.

4.2 Main Results on Instance Segmentation

State-of-the-art comparison. We compare to the state-of-the-art methods for long-tailed instance segmentation in Table 2. The proposed FreeSeg method achieves comparable or even better results, especially for rare object categories. For example, FreeSeg outperforms all the other methods except Seesaw loss [48], which is implemented with a different framework [6] and trained with a stronger scheduler. (We provide further comparisons in this aspect in the supplementary.)

Backbone agnostic. Beyond ResNet-50, we further evaluate FreeSeg with stronger backbone model architectures: ResNet-101 [17] and ResNeXt-101 [54], following the same training pipeline as ResNet-50. FreeSeg achieves notably gains over MosaicOS [56], justifying that FreeSeg can benefit different instance segmentation models and architectures.

Compatibility with existing methods. We further apply post-processing calibration [29] on top of the model trained with FreeSeg. Results are shown

[2] We note that FreeSeg is detector-agnostic and is thus complementary to and compatible with other models [31,43,61] that incorporate external images like [56].

[3] The median number of instances per image in LVIS dataset is 6.

Table 2. State-of-the-art comparison on LVIS v1 instance segmentation. FreeSeg are initialized with MosaicOS [56] as the base model. 2×: Seesaw applies a stronger 2× training schedule while other methods are with 1× schedule. ⋆: with post-processing calibration introduced by [29].

Backbone	Method	AP	AP_r	AP_c	AP_f	AP^b
ResNet-50 FPN	RFS [15]	22.58	12.30	21.28	28.55	23.25
	BaGS [24]	23.10	13.10	22.50	28.20	25.76
	Forest R-CNN [52]	23.20	14.20	22.70	27.70	24.60
	RIO [5]	23.70	15.20	22.50	28.80	24.10
	EQL v2 [44]	23.70	14.90	22.80	28.60	24.20
	FASA [55]	24.10	17.30	22.90	28.50	–
	DisAlign [58]	24.30	8.50	26.30	28.10	23.90
	Seesaw [48]$^{2\times}$	26.40	19.60	26.10	29.80	27.40
	MosaicOS [56]	24.45	18.17	23.00	28.83	25.05
	w/ FreeSeg	25.19	***20.23***	23.80	28.92	25.98
	MosaicOS [56] ⋆	26.76	23.86	25.82	29.10	27.77
	w/ FreeSeg ⋆	27.34	***25.11***	26.29	29.49	28.47
ResNet-101 FPN	RFS [15]	24.82	15.18	23.71	30.31	25.45
	FASA [55]	26.30	19.10	25.40	30.60	–
	Seesaw [48]$^{2\times}$	28.10	20.00	28.00	31.90	28.90
	MosaicOS [56]	26.73	20.52	25.78	30.53	27.41
	w/ FreeSeg	27.54	***23.00***	26.48	30.72	28.63
	MosaicOS [56] ⋆	29.03	26.38	28.15	31.19	29.96
	w/ FreeSeg ⋆	29.72	***28.69***	28.67	31.34	31.11
ResNeXt-101 FPN	RFS [15]	26.67	17.60	25.58	31.89	27.35
	MosaicOS [56]	28.29	21.75	27.22	32.35	28.85
	w/ FreeSeg	28.86	***23.34***	27.77	32.49	29.98
	MosaicOS [56] ⋆	29.81	25.73	28.92	32.59	30.56
	w/ FreeSeg ⋆	30.37	***26.43***	29.63	32.92	31.81

in Table 2 (FreeSeg ⋆) and the improvements are consistent. More surprisingly, FreeSeg can boost the performance of rare classes to be similar to common classes. This indicates that by introducing more while not so perfect training instances, FreeSeg dramatically overcomes the long-tailed problem.

4.3 Detailed Analyses and Ablation Studies

Does segment ranking help? The quality of the segments is important because inferior pixel-level annotations for instance segmentation may contain certain noise (cf. Section 1). Such an issue will be amplified for rare categories when the training examples are long-tailed. Here we conduct experiments with and without ranking object segments. As shown in Table 1, we are able to collect 1,830K segments from ImageNet-22k and Google Images, while only half of them are left after filtering with FreeSeg. Table 3 shows the results. While both versions outperform the baseline models, segment ranking does help more

Table 3. Ablation study on object segments ranking. We evaluate the performance of the model trained with and without the segments ranking mechanism by FreeSeg. Results demonstrate the importance of ranking the object segments.

Method	Random	Ranking	#Image	AP	AP_r	AP_c	AP_f
MosaicOS [56]				24.45	18.17	23.00	28.83
			1,830K	24.87	19.13	23.55	28.86
w/ FreeSeg	✓		966K	24.50	18.68	23.18	28.52
		✓	966K	**25.19**	**20.23**	**23.80**	**28.92**

Table 4. Analysis on different object segments ranking metrics. The proposed FreeSeg score can take different scenarios into account thus achieves better results.

Method	Ranking Metrics	AP	AP_r	AP_c	AP_f
MosaicOS [56]	–	24.45	18.17	23.00	28.83
w/ FreeSeg	IoU	24.74	19.04	23.58	28.53
	IoB	24.69	18.41	23.58	28.70
	IoM	24.56	18.62	23.14	28.74
	FreeSeg	**25.19**	**20.23**	**23.80**	**28.92**

(row 4 *vs.* row 2 in Table 3), suggesting that the quality of pixel labels is more important than the quantity for instance segmentation.

We notice that filtering by ranking gives higher quality but fewer masks. To further understand the effect of quality and quantity of object segments on the accuracy of FreeSeg, we randomly sample the original co-segmentation masks such that the remaining ones are of the same quantity as those *selected by our ranking method*. We see a bigger gain by our ranking method (row 4 *vs.* row 3 in Table 3), justifying its effectiveness in selecting high-quality masks.

Ranking metrics. We show both quantitative and qualitative comparisons of different ranking metrics for filtering noisy segments in Table 4 and Figure 3. FreeSeg score can take different scenarios into account and successfully select confident segments from noisy ones. This verifies that the quality of the segments is the key and that the proposed FreeSeg pipeline effectively does the job.

Effect of segments filtering by drop rate. Table 5 reports results with and without segments filtering with drop rate (cf. Section 3.2). By jointly using drop rate and FreeSeg score, our method achieves better results by using fewer and cleaner object segments for training.

Importance of the context. We investigate training the instance segmentation model directly with the object-centric images without pasting FreeSeg segments to LVIS images. The results are shown in Table 6. As expected, we see that the performance is worse than the proposed FreeSeg framework, in which we apply copy-paste augmentation to put the object instances into the context of original training images. This demonstrates the fact that there exists

Table 5. Ablation study on segments filtering by drop rate.

Method	Drop Rate	#Image	AP	AP_r	AP_c	AP_f
MosaicOS [56]			24.45	18.17	23.00	28.83
w/ FreeSeg	✗	1,134K	24.81	19.37	23.62	28.54
	✓	966K	**25.19**	**20.23**	**23.80**	**28.92**

Table 6. Importance of the context. Segments Pasting [13]: ✗ indicates directly training the instance segmentation model on the object-centric images, using FreeSeg segments as supervision.

Method	Segments Pasting	AP	AP_r	AP_c	AP_f
MosaicOS [56]		24.45	18.17	23.00	28.83
w/ FreeSeg	✗	24.78	18.85	23.40	28.90
	✓	**25.19**	**20.23**	**23.80**	**28.92**

a gap between the contexts of two different image resources, which could limit the improvement on the main task.

Additional results. Please see supplementary materials, including the results with other evaluation metrics and datasets, the analysis of multi-stage training, effects of different data sources, **qualitative results**, etc.

4.4 Comparison to Pasting Ground-truth Segments

Ghiasi *et al.* [13] show that copying and pasting human-annotated segments from one image to another as augmentation can improve instance segmentation with richer *context diversity*. They employ a much larger batch size and longer scheduler with another strong augmentation, large-scale jittering [13]. However, in this work, we focus on enriching *appearance diversity* for objects with abundant free segments from object-centric images. We, therefore, conduct a detailed comparison between pasting ground truth and FreeSeg object segments. We follow the pasting mechanism in Section 4.1 but use ground truths segments instead. That is, we randomly pick two images from the LVIS training set, apply the same data augmentation policy following the standard instance segmentation model (*i.e.*, resizing shortest edge and random horizontal flip), and then *paste random numbers of instances from one image onto the other image*.

We show results in Table 7. We validate FreeSeg on two base models and compare to the results of using ground truth segments for augmentation. FreeSeg achieves consistent gains against the baseline models and, more importantly, outperforms those with copy-paste augmentation using only ground truth segments. This demonstrates that with ample images that can be acquired easily online, even noisy labels without any human efforts could significantly improve long-tailed instance segmentation. We also note that FreeSeg is more effective when the baseline is already re-balanced (*e.g.*, MosaicOS in Table 7

Table 7. Comparison of pasting ground truth (GT) object segments and FreeSeg. The base models are trained with ResNet-50 and FPN. †: models from [56].

Method	GT	FreeSeg	AP	AP_r	AP_c	AP_f	AP^b
Mask R-CNN [15]†			22.58	12.30	21.28	28.55	23.25
	✓		24.06	17.00	22.62	28.77	24.91
		✓	24.28	17.68	22.79	28.83	25.13
	✓	✓	24.74	***18.80***	23.38	28.86	25.51
MosaicOS [56]†			24.45	18.17	23.00	28.83	25.05
	✓		24.57	18.63	23.31	28.59	25.52
		✓	25.19	20.23	23.80	28.92	25.98
	✓	✓	25.36	***20.72***	24.00	28.92	26.00

bottom and Table 2), while GT-only can hardly improve upon it due to the lack of training examples. Furthermore, by learning with copy-paste from both sources, the gain can be even larger on both base models. These observations demonstrate that, besides context diversity, the appearance diversity of objects is also the key to improve segmentation.

5 Conclusion

Our main contribution and novelty are the insight that object segments emerge freely from object-centric images, and they effectively benefit the challenging long-tailed instance segmentation problem. We propose a scalable framework FreeSeg to realize this idea. We show that, with the underlying properties of object-centric images, simple co-segmentation with proper ranking can result in high-quality instance segments to largely increase the labeled training instances.

We believe that the prospect of leveraging ample data without human labeling has enormous future potential. We note that there are several ways to realize this insight, and [57] is just an instantiation but turns out to be very useful: it is worth mentioning that co-segmentation has never been used to enhance instance segmentation. Further, our pipeline is clean and conceptually simple, clearly indicating where future improvement can be made (*e.g.*, segment discovery, extraction, leveraging). We expect our approach to serve as a strong baseline for this direction: for future work to build upon and take advantage of.

Acknowledgments This research is supported in part by grants from the National Science Foundation (IIS-2107077, OAC-2118240, OAC-2112606), the OSU CCTS pilot grant, and Cisco Systems, Inc. We are thankful for the generous support of the computational resources by the Ohio Supercomputer Center and AWS Cloud Credits for Research.

References

1. Google images. https://www.google.com/imghp?hl=EN 9
2. Batra, D., Kowdle, A., Parikh, D., Luo, J., Chen, T.: icoseg: Interactive co-segmentation with intelligent scribble guidance. In: CVPR (2010) 4
3. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: ICCV (2021) 4
4. Cech, J., Matas, J., Perdoch, M.: Efficient sequential correspondence selection by cosegmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **32**(9), 1568–1581 (2010) 4
5. Chang, N., Yu, Z., Wang, Y.X., Anandkumar, A., Fidler, S., Alvarez, J.M.: Image-level or object-level? a tale of two resampling strategies for long-tailed detection. In: ICML (2021) 3, 11
6. Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., Zhang, Z., Cheng, D., Zhu, C., Cheng, T., Zhao, Q., Li, B., Lu, X., Zhu, R., Wu, Y., Dai, J., Wang, J., Shi, J., Ouyang, W., Loy, C.C., Lin, D.: MMDetection: Open mmlab detection toolbox and benchmark. arXiv preprint arXiv:1906.07155 (2019) 10
7. Chen, Y.C., Lin, Y.Y., Yang, M.H., Huang, J.B.: Show, match and segment: joint weakly supervised learning of semantic matching and object co-segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2020) 4
8. Cheng, B., Girshick, R., Dollár, P., Berg, A.C., Kirillov, A.: Boundary IoU: Improving object-centric image segmentation evaluation. In: CVPR (2021) 20, 21
9. 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 (TPAMI) **37**(3), 569–582 (2014) 4
10. Dave, A., Dollár, P., Ramanan, D., Kirillov, A., Girshick, R.: Evaluating large-vocabulary object detectors: The devil is in the details. arXiv preprint arXiv:2102.01066 (2021) 20, 21
11. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009) 2, 3, 4, 9
12. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International Journal of Computer Vision (IJCV) **88**(2), 303–338 (2010) 1, 4
13. Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.Y., Cubuk, E.D., Le, Q.V., Zoph, B.: Simple copy-paste is a strong data augmentation method for instance segmentation. In: CVPR (2021) 2, 3, 4, 8, 10, 13, 20
14. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014) 1
15. Gupta, A., Dollar, P., Girshick, R.: LVIS: A dataset for large vocabulary instance segmentation. In: CVPR (2019) 1, 2, 3, 5, 8, 9, 10, 11, 14, 20, 21, 22
16. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: ICCV (2017) 1, 9
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) 9, 10
18. Hsieh, T.I., Robb, E., Chen, H.T., Huang, J.B.: Droploss for long-tail instance segmentation. In: AAAI (2021) 3
19. Hu, X., Jiang, Y., Tang, K., Chen, J., Miao, C., Zhang, H.: Learning to segment the tail. In: CVPR (2020) 1, 3

20. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE Transactions on pattern analysis and machine intelligence (TPAMI) **20**(11), 1254–1259 (1998) 4
21. Kang, B., Xie, S., Rohrbach, M., Yan, Z., Gordo, A., Feng, J., Kalantidis, Y.: Decoupling representation and classifier for long-tailed recognition. In: ICLR (2020) 3
22. Li, C., Tam, P.K.S.: An iterative algorithm for minimum cross entropy thresholding. Pattern Recognition Letters (PRL) **19**(8), 771–776 (1998) 5, 20
23. Li, C.H., Lee, C.: Minimum cross entropy thresholding. Pattern recognition **26**(4), 617–625 (1993) 5, 20
24. Li, Y., Wang, T., Kang, B., Tang, S., Wang, C., Li, J., Feng, J.: Overcoming classifier imbalance for long-tail object detection with balanced group softmax. In: CVPR (2020) 1, 3, 11
25. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR (2017) 9
26. 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: ECCV (2014) 1, 2
27. Miller, G.A.: WordNet: A lexical database for english. Communications of the ACM **38**(11), 39–41 (1995) 9
28. Oksuz, K., Cam, B.C., Kalkan, S., Akbas, E.: Imbalance problems in object detection: A review. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2020) 1
29. Pan, T.Y., Zhang, C., Li, Y., Hu, H., Xuan, D., Changpinyo, S., Gong, B., Chao, W.L.: On model calibration for long-tailed object detection and instance segmentation. In: NeurIPS (2021) 10, 11
30. Ramanathan, V., Wang, R., Mahajan, D.: DLWL: Improving detection for lowshot classes with weakly labelled data. In: CVPR (2020) 3
31. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. In: CVPR (2017) 10
32. Ren, J., Yu, C., Sheng, S., Ma, X., Zhao, H., Yi, S., Li, H.: Balanced meta-softmax for long-tailed visual recognition. In: NeurIPS (2020) 3
33. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **39**(6), 1137–1149 (2016) 1
34. Rother, C., Kolmogorov, V., Blake, A.: "grabcut" interactive foreground extraction using iterated graph cuts. ACM transactions on graphics (TOG) **23**(3), 309–314 (2004) 4
35. Rother, C., Minka, T., Blake, A., Kolmogorov, V.: Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. In: CVPR (2006) 2, 4
36. Rubinstein, M., Joulin, A., Kopf, J., Liu, C.: Unsupervised joint object discovery and segmentation in internet images. In: CVPR (2013) 4
37. 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. IJCV **115**(3), 211–252 (2015) 2, 9
38. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: ICCV (2017) 4
39. Shao, S., Li, Z., Zhang, T., Peng, C., Yu, G., Zhang, X., Li, J., Sun, J.: Objects365: A large-scale, high-quality dataset for object detection. In: ICCV (2019) 1
40. Shen, L., Lin, Z., Huang, Q.: Relay backpropagation for effective learning of deep convolutional neural networks. In: ECCV (2016) 3

41. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: ECCV (2006) 4
42. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014) 9
43. Sohn, K., Zhang, Z., Li, C.L., Zhang, H., Lee, C.Y., Pfister, T.: A simple semi-supervised learning framework for object detection. arXiv preprint arXiv:2005.04757 (2020) 10
44. Tan, J., Lu, X., Zhang, G., Yin, C., Li, Q.: Equalization loss v2: A new gradient balance approach for long-tailed object detection. In: CVPR (2021) 1, 3, 11
45. Tan, J., Wang, C., Li, B., Li, Q., Ouyang, W., Yin, C., Yan, J.: Equalization loss for long-tailed object recognition. In: CVPR (2020) 1, 3
46. Thomee, B., Shamma, D.A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., Li, L.J.: YFCC100M: The new data in multimedia research. Communications of the ACM **59**(2), 64–73 (2016) 3
47. Wang, C., Zha, Z.J., Liu, D., Xie, H.: Robust deep co-saliency detection with group semantic. In: AAAI (2019) 9
48. Wang, J., Zhang, W., Zang, Y., Cao, Y., Pang, J., Gong, T., Chen, K., Liu, Z., Loy, C.C., Lin, D.: Seesaw loss for long-tailed instance segmentation. In: CVPR (2021) 1, 3, 10, 11, 23
49. Wang, T., Li, Y., Kang, B., Li, J., Liew, J., Tang, S., Hoi, S., Feng, J.: The devil is in classification: A simple framework for long-tail instance segmentation. In: ECCV (2020) 1, 3, 8, 21, 22
50. Wang, T., Zhu, Y., Zhao, C., Zeng, W., Wang, J., Tang, M.: Adaptive class suppression loss for long-tail object detection. In: CVPR (2021) 3
51. Wang, X., Huang, T.E., Darrell, T., Gonzalez, J.E., Yu, F.: Frustratingly simple few-shot object detection. In: ICML (2020) 1, 3
52. Wu, J., Song, L., Wang, T., Zhang, Q., Yuan, J.: Forest R-CNN: Large-vocabulary long-tailed object detection and instance segmentation. In: ACM MM (2020) 1, 11
53. Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., Girshick, R.: Detectron2. https://github.com/facebookresearch/detectron2 (2019) 8, 9, 10, 20
54. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: CVPR (2017) 10
55. Zang, Y., Huang, C., Loy, C.C.: FASA: Feature augmentation and sampling adaptation for long-tailed instance segmentation. In: ICCV (2021) 3, 11, 21, 22
56. Zhang, C., Pan, T.Y., Li, Y., Hu, H., Xuan, D., Changpinyo, S., Gong, B., Chao, W.L.: MosaicOS: a simple and effective use of object-centric images for long-tailed object detection. In: ICCV (2021) 2, 3, 4, 6, 9, 10, 11, 12, 13, 14, 19, 21, 22, 23
57. Zhang, K., Chen, J., Liu, B., Liu, Q.: Deep object co-segmentation via spatial-semantic network modulation. In: AAAI (2020) 2, 4, 5, 9, 14, 19
58. Zhang, S., Li, Z., Yan, S., He, X., Sun, J.: Distribution alignment: A unified framework for long-tail visual recognition. In: CVPR (2021) 3, 11
59. Zhang, Y., Kang, B., Hooi, B., Yan, S., Feng, J.: Deep long-tailed learning: A survey. arXiv preprint arXiv:2110.04596 (2021) 1
60. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR (2016) 4
61. Zhou, X., Girdhar, R., Joulin, A., Krähenbühl, P., Misra, I.: Detecting twenty-thousand classes using image-level supervision. In: ECCV (2022) 10
62. Zhu, X., Anguelov, D., Ramanan, D.: Capturing long-tail distributions of object subcategories. In: CVPR (2014) 1

63. Zoph, B., Ghiasi, G., Lin, T.Y., Cui, Y., Liu, H., Cubuk, E.D., Le, Q.: Rethinking pre-training and self-training. In: NeurIPS (2020) 22
64. Zou, Z., Shi, Z., Guo, Y., Ye, J.: Object detection in 20 years: A survey. arXiv preprint arXiv:1905.05055 (2019) 1

Supplementary Materials

In this supplementary material, we provide details and results omitted in the main text.

- Section A: implementation details.
- Section B: results on other metrics: AP fixed and boundary IoU.
- Section C: results on COCO-LT dataset.
- Section D: additional ablation studies.
- Section E: qualitative results.

A Implementation Details

A.1 Data Curation

As mentioned in Section 4.1 of the main paper, we followed [56] to collect Google images. We use class names as the keywords and take top images from the respective search engine without extra user interventions. Thus, the curation process is quite straightforward. The collected Google data are balanced (500/class). ImageNet images are nearly balanced by design, with around 1K images/class, including rare objects in LVIS. The imbalance situation in LVIS is largely reduced. For the rarest class (one image in LVIS), the increase factor is larger than 500 times.

A.2 Generating Object Segments

As mentioned in Section 3.1 and Section 4.1 of the main paper, we apply spatial and semantic modulation (SSM) co-segmentation method [57] to the object-centric images for each class, followed by segment refinement. We show more examples of object segments by FreeSeg in Figure 7, Figure 8, and Figure 9. With the proper ranking algorithm, our approach can identify the most reliable instance segments to improve long-tailed instance segmentation.

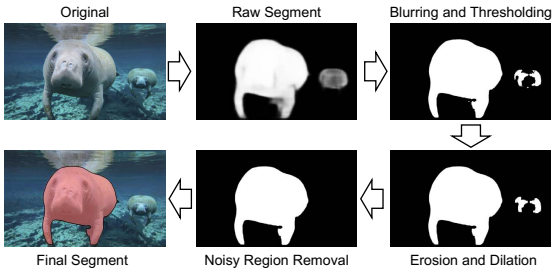

Fig. 6. Procedure of generating object segments. We generate the final segments (*e.g.*, *manatee*) by post-processing the raw segments obtained from the image co-segmentation methods.

A.3 Post-Processing for Segment Refinement

To turn the raw, gray scale segmentation map into a binary one that can be used to train a segmentation model, we threshold the map. As the suitable threshold value may vary across images and classes, we apply Gaussian filter followed by dynamic thresholding, *i.e.*, Li thresholding [23,22], which minimizes the cross-entropy between the foreground and the background to find the optimal threshold to distinguish them.

To further improve the resulting binary map, we apply erosion and dilation to smooth the boundary. We then remove small, likely false positive segments by only keeping the largest connected component in the binary map. Figure 6 shows the entire post-processing procedure for refinement, which greatly improves the quality of the segmentation masks, as illustrated in Figure 2 of the main paper.

A.4 Putting Segments in Context

As introduced in Section 3.3 and Section 4.1 of the main paper, we follow the mechanism in [13] to paste our ranked segments. More specifically, we randomly pick an example from LVIS training set as a background image, followed by pasting segments from 1 to 6 object-centric images on it at different locations. For LVIS images, we follow the standard data augmentation policy in [15] and [53]. That is, we randomly resize the shortest edge of the image into [640, 672, 704, 736, 768, 800] with a limit of max size of width or height to 1333, followed by a random horizontal flip with $p = 0.5$. For the selected object-centric images, we apply random horizontal flip ($p = 0.5$) followed by random resize with a scale of [0.1, 2.0]. We then randomly crop (or pad) the object-centric images to match the size of the background image. Note that, this step ensures that the object segments will be randomly pasted at different locations on each of the LVIS images. For binary masks on LVIS images used for supervision, we remove pixel annotations if the objects are occluded by the pasted ones in the front.

The examples of synthesized data via vanilla copy-paste (*i.e.*pasting ground truths) can be found in Figure 10. We also provide examples generated by FreeSeg framework in Figure 11. We can see that FreeSeg can increase the appearance diversity of foreground instances, especially for rare object categories. We will leave a better way to leverage the object segments as our future work.

A.5 Model Training

We apply a two-stage strategy to fine-tune the pre-trained instance segmentation model (cf. Section 4.1 of the main paper). Both stages follow the same training and optimization setting, which is summarized in Table 8.

B Results on AP Fixed with Boundary IoU

FreeSeg is effective in AP Fixed with Boundary IoU. Besides standard Mask AP, we also report the results in AP Fixed [10] with Boundary IoU [8],

Table 8. Optimization configuration for the two-stage fine-tuning.

Config	Value
Optimizer	SGD
Learning rate	2e-4
Weight decay	0.0001
Optimizer momentum	0.9
Batch size	8 (larger batch size, *e.g.*, 16, does not lead to notable differences)
Warm up epoch	0
Training iteration	90,000
Aug. for background image	ResizeShortestEdge [640, 672, 704, 736, 768, 800], RandomFlip
Aug. for pasted image	RandomFlip, ResizeScale [0.1, 2], FixedSizeCrop

following the official evaluation metrics in LVIS challenge 2021. AP Fixed replaces the cap (*i.e.*, 300) of number of detected objects per image by a cap (*i.e.*, 10,000) per class for the entire validation set. Table 9 reports the results. We see that the improvement is consistent, demonstrating FreeSeg is metric-agnostic.

Table 9. Results on AP Fixed [10] with Boundary IoU [8]. All models are based on ResNet-50 FPN backbone architecture.

Method	AP	AP_r	AP_c	AP_f
Mask R-CNN [15]	19.88	14.76	19.32	22.76
w/ FreeSeg	**21.25**	**18.33**	**20.85**	**23.00**
MosaicOS [56]	21.20	18.79	20.63	22.90
w/ FreeSeg	**21.86**	**20.12**	**21.50**	**23.03**

C Results on COCO-LT dataset

To further validate the generalizability of our framework, we conduct experiments on another popular long-tailed dataset, *i.e.*, COCO-LT [49]. We match class names to find object-centric images from ImageNet-22K and Google for each class in COCO-LT. We follow the same evaluation protocol in [49,55] and show results in Table 10. FreeSeg (with Mask R-CNN as baseline) outperforms SimCal [49] and FASA [55], justifying the generalizability.

D Additional Ablation Studies

Effect of data sources We first study the effect of data sources. As ImageNet only covers 997 classes of LVIS, we augment it with Google images for all

Table 10. Results on COCO-LT dataset.

Method	AP	AP_1	AP_2	AP_3	AP_4
Mask R-CNN [56]	18.70	0.00	8.20	24.40	26.00
SimCal [49]	21.80	15.00	16.20	24.30	26.00
FASA [55]	23.40	13.50	19.00	25.20	27.50
FreeSeg	**25.10**	**15.80**	**20.60**	**27.60**	**28.80**

the 1,203 LVIS classes (Section 4.1 of the main paper). Table 11 shows results with different data sources, we compare the performance of using different data sources. We see that both Google images and ImageNet are useful. We achieve the best result by combing them.

Table 11. Results on different object-centric image sources. G: Google Images. IN: ImageNet.

Method	G	IN	AP	AP_r	AP_c	AP_f
Mask R-CNN [15]			22.58	12.30	21.28	28.55
w/ FreeSeg	✓		24.08	17.08	22.68	28.72
		✓	24.12	17.23	22.67	28.75
	✓	✓	**24.28**	**17.68**	**22.79**	**28.83**

Importance of multi-stage training Table 12 reports results after the first and second stage training (cf. Section 4.1 of the main paper). As introduced in [56], the first stage learns better features with diverse and balanced data, but noisy labels; the second stage trained with accurate labels helps correct the prediction. We note that both stages use repeat factor sampling [15] to further balance data.

Table 12. Importance of multi-stage training.

Method	Stage	AP	AP_r	AP_c	AP_f
Mask R-CNN [15]	–	22.58	12.30	21.28	28.55
w/ FreeSeg	First	23.35	16.67	21.78	28.04
	Second	**24.28**	**17.68**	**22.79**	**28.83**

Comparison with self-training. We also study different ways to generate pseudo-masks for training instance segmentation models. We replace FreeSeg segments with those generated by Mask R-CNN (pre-trained on LVIS) — treating it as the teacher model to generate pseudo-labels for self-training [63]. We

only keep masks whose class labels matched the object-centric image labels to filter out noises. Table 13 shows the results. All methods use the same training pipeline. FreeSeg outperforms this baseline. We attribute this to the benefit of co-segmentation which explores the similarity across images.

Table 13. Comparison with self-training.

Method	AP	AP_r	AP_c	AP_f
MosaicOS [56]	24.45	18.17	23.00	28.83
w/ LVIS Network	24.70	18.96	23.42	28.64
w/ FreeSeg	**25.19**	**20.23**	**23.80**	**28.92**

Extended training of FreeSeg. Finally, to further compare to Seesaw [48], which applies 2× training scheduling (cf. Section 4.2 of the main paper), we double the training epochs of FreeSeg. Table 14 summarizes the results. FreeSeg (2×) achieves further gains and outperforms Seesaw (2×) on all metrics except AP_f for frequent classes. The improvement on AP_r/AP_c (*i.e.*, rare/common) is significant, justifying the effectiveness of our approach.

Table 14. FreeSeg with a stronger training schedule.

Method	Schedule	AP	AP_r	AP_c	AP_f
Seesaw [48]	2×	26.40	19.60	26.10	**29.80**
MosaicOS [56]	1×	24.45	18.17	23.00	28.83
w/ FreeSeg	1×	25.19	20.23	23.80	28.92
	2×	**26.80**	**21.70**	**26.90**	28.60

E Qualitative Results

One common problem for long-tailed instance segmentation is the trained detector will be overconfident on the frequent objects and suppress the rare objects. Figure 12 shows qualitative results. The baseline model tends to predict many false positives which their classes appear more frequently in the training data. FreeSeg uses augmented training data with high-quality segments to improve the features, especially for rare objects.

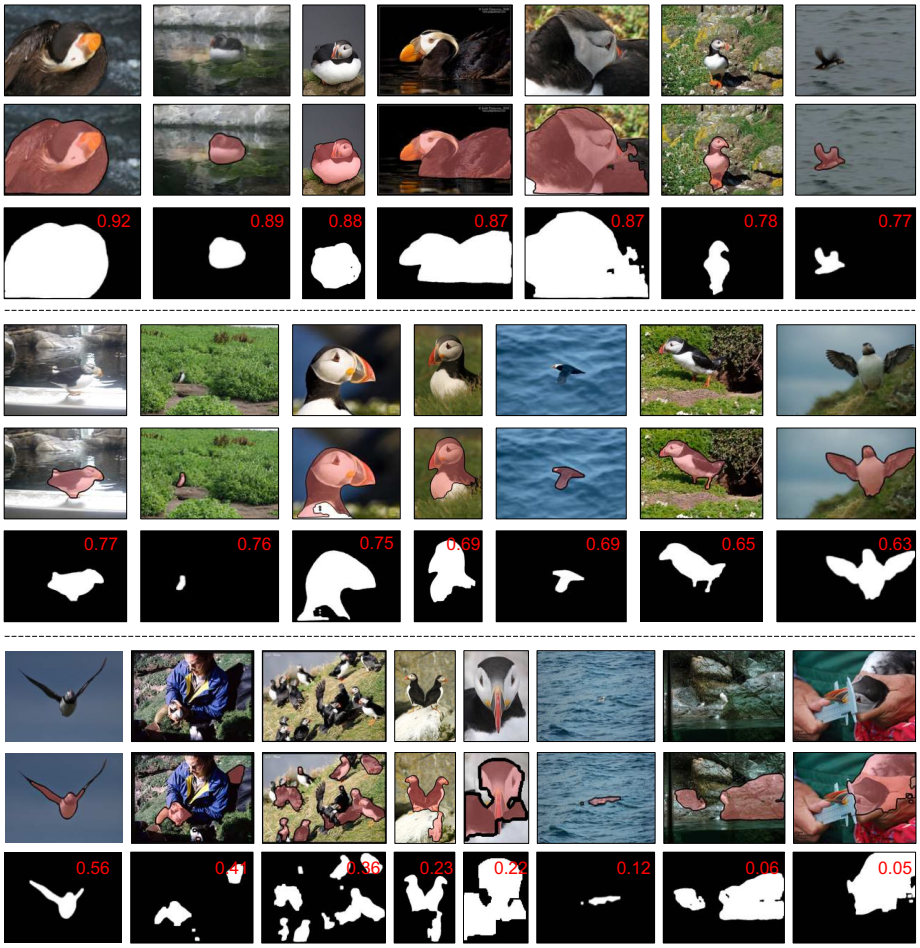

Fig. 7. Randomly sampled examples of object segments on ImageNet images by FreeSeg. We show a rare class *puffin* in LVIS v1. For each triplet, we show the original image, the object segment, and the binary mask. FreeSeg scores are on the upper right corner of the images. We keep the segments with FreeSeg scores larger than 0.5 (cf. Section 3.2 of the main paper).

Fig. 8. Randomly sampled examples of object segments on ImageNet images by FreeSeg. We show a rare class *bulldoze* in LVIS v1. For each triplet, we show the original image, the object segment, and the binary mask. FreeSeg scores are on the upper right corner of the images. We keep the segments with FreeSeg scores larger than 0.5 (cf. Section 3.2 of the main paper).

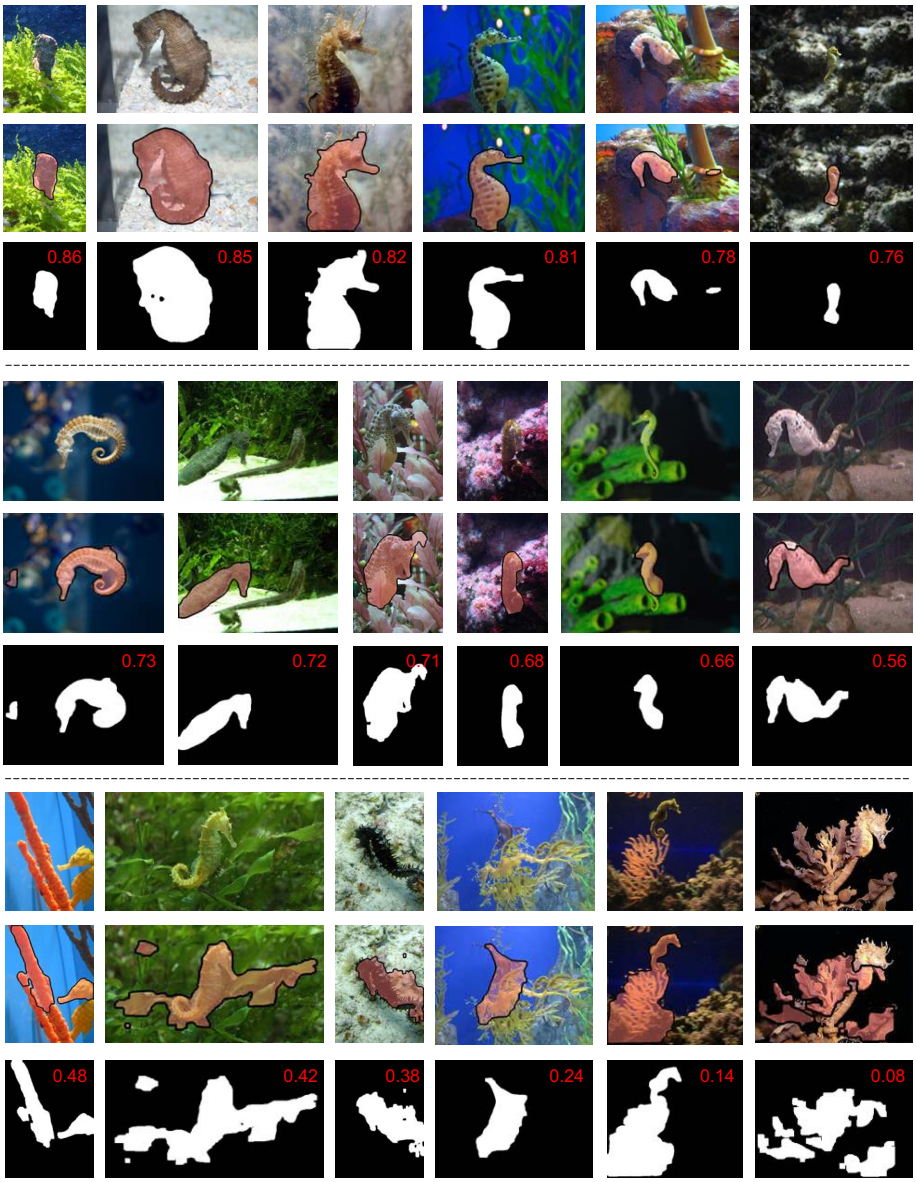

Fig. 9. Randomly sampled examples of object segments on ImageNet images by FreeSeg. We show a rare class *seehorse* in LVIS v1. For each triplet, we show the original image, the object segment, and the binary mask. FreeSeg scores are on the upper right corner of the images. We keep the segments with FreeSeg scores larger than 0.5 (cf. Section 3.2 of the main paper).

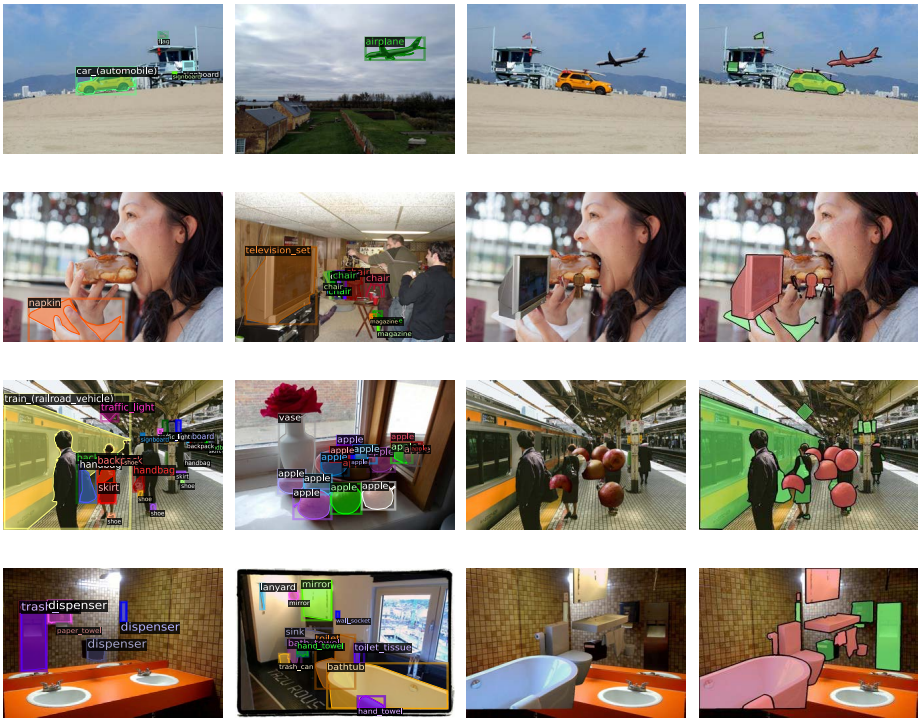

Fig. 10. Four examples of vanilla copy-paste augmentation using original training images. For each example, we show the background image with ground-truths, the pasted image with ground-truths, the synthesized image, and the synthesized image with ground-truths. We first randomly pick the background and pasted images from LVIS training set, followed by random shortest edge resize and horizontal flip (cf. Section 4.1 of the main paper). We then select a random number of objects from the pasted image and paste them onto the background image. In the last column, red masks indicate pasted segments; green masks indicate the objects in background images.

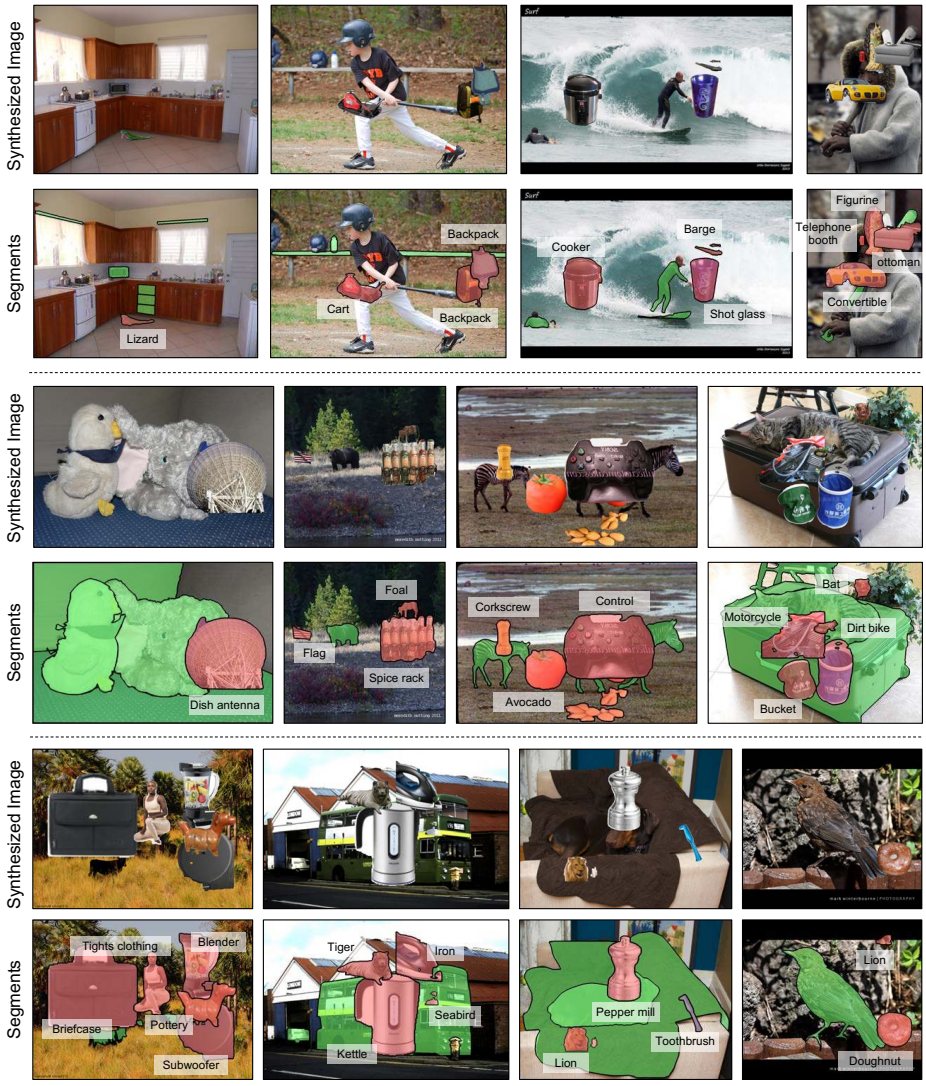

Fig. 11. Examples of copy-paste augmentation with FreeSeg segments. We generate object segments from object-centric images and randomly paste them onto scene-centric images. Red masks indicate pasted segments by FreeSeg; green masks indicate original objects in scene-centric images.

Fig. 12. Qualitative results. Green arrows are used to indicate the improvement. FREESEG successfully detects *school bus*, *martini*, *parasail*, *ram*, *rhinoceros*, *bullet train*, *postbox*, *lion*, and *goat*.