MosaicOS: A Simple and Effective Use of Object-Centric Images for Long-Tailed Object Detection

Cheng Zhang[1*] Tai-Yu Pan[1*] Yandong Li[2] Hexiang Hu[3]
Dong Xuan[1] Soravit Changpinyo[2] Boqing Gong[2] Wei-Lun Chao[1]

[1]The Ohio State University [2]Google Research [3]University of Southern California

Abstract

*Many objects do not appear frequently enough in complex scenes (*e.g.*, certain handbags in living rooms) for training an accurate object detector, but are often found frequently by themselves (*e.g.*, in product images). Yet, these* object-centric *images are not effectively leveraged for improving object detection in* scene-centric *images. In this paper, we propose **Mosaic** of **O**bject-centric images as **S**cene-centric images (*MosaicOS*), a simple and novel framework that is surprisingly effective at tackling the challenges of long-tailed object detection. Keys to our approach are three-fold: (i) pseudo scene-centric image construction from object-centric images for mitigating domain differences, (ii) high-quality bounding box imputation using the object-centric images' class labels, and (iii) a multi-stage training procedure. On LVIS object detection (and instance segmentation),* MosaicOS *leads to a massive* 60% *(and* 23%*) relative improvement in average precision for rare object categories. We also show that our framework can be compatibly used with other existing approaches to achieve even further gains. Our pre-trained models are publicly available at* https://github.com/czhang0528/MosaicOS/.

1. Introduction

Detecting objects in complex daily scenes is a long-standing task in computer vision [15, 19, 52, 70]. With rapid advances in deep neural networks [28, 33, 39, 60, 62] and the emergence of large-scale datasets [41, 48, 59, 67, 77], there has been remarkable progress in detecting *common* objects (*e.g.*, cars, humans, *etc.*) [4, 27, 46, 47, 49, 54, 57, 83]. However, detecting *rare* objects (*e.g.*, unicycles, bird feeders, *etc.*) proves much more challenging due to the inherent limitation of training data. In particular, complex scenes in which an object appears pose another variation factor that is too diverse to capture from a small amount of data [22, 69, 82]. Developing algorithms to overcome such a "long-tailed" distribution of object instances in *scene-centric* images (SCI) [22, 41, 48, 59] has thus attracted a flurry of research interests [45, 63, 72].

*Equal contributions

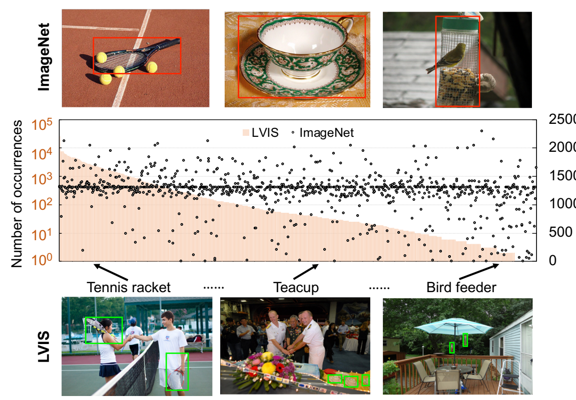

Figure 1. **Object frequencies in scene-centric and object-centric images.** Orange bars show the number of instances per class in the scene-centric LVIS v0.5 dataset [22]. Class indices are sorted by the instance numbers. Black dots show the number of images in the object-centric ImageNet datasets [11] for each corresponding class. *The two types of images have very different trends of object frequencies.* We also show three examples of both datasets, corresponding to frequent, common, and rare classes in LVIS. **Red** and **green** boxes indicate the objects. *These two types of images have different focuses and object sizes.*

Fortunately, the uncommon objects in scene-centric images often appear more frequently in *object-centric* images (OCI) in which the objects of interest occupy the main and most salient regions (*e.g.*, product images). For example, given "bird feeder" as query, a popular image search engine (*e.g.*, Google Images) mostly retrieves object-centric "bird feeder" results. Similarly, curated object recognition datasets such as ImageNet [11] contain more than a thousand object-centric "bird feeder" images, nearly a hundred times more than scene-centric images from LVIS v0.5 [22]. We further illustrate this point in Figure 1, in which a discrepancy in frequencies of the same objects in ImageNet and LVIS can generally be observed (see § 3 for details).

Can we leverage such abundant object-centric images to improve long-tailed object detection? The most common approach to this is to leverage these images for pre-training the object detector's backbone [26, 27, 83]. While this general approach may benefit various tasks beyond object detection, it is highly data-intensive and does not take care of the domain gap between the pre-training and downstream tasks (see § 6.3 for analysis). As a result, they do not always improve the object detection accuracy [26, 83].

In this paper, we propose MosaicOS (**Mosaic** of **O**bject-centric images as **S**cene-centric images), a simple and effective framework to leverage object-centric images for object detection. MosaicOS directly uses object-centric images during the *training* of object detectors. There are three key ingredients. The first one is the construction of *pseudo* scene-centric images from object-centric images using mosaic[1]. The second one is the imputation of bounding box annotations using image class labels. The third ingredient is a multi-stage training procedure for learning from both gold scene-centric and synthesized pseudo scene-centric annotations. Figure 2 illustrates our framework.

Our use of mosaic and bounding box imputation to construct *pseudo* scene-centric images from object-centric images tackles two key challenges in leveraging object-centric images for object detection. The first challenge is a "domain" gap between object-centric and scene-centric images: an object-centric image usually contains fewer (but bigger) object instances and a less complex background and this discrepancy is believed to unfavorably hinder knowledge transfer [73, 78]. The second challenge is missing detection annotations: object-centric images, either from the Internet or object recognition datasets (*e.g*., ImageNet), are not *perfectly object-centric*, usually provided *without* accurate object localization in the form of bounding boxes.

Our proposed framework leads to significant accuracy gains in long-tailed object detection and instance segmentation on LVIS [22], using object-centric images from ImageNet [11] and the Internet. In particular, for the task of object detection for rare objects, we observe a significant boost from 13% to over 20% in average precision. For the task of instance segmentation, our approach even improves the accuracy on common objects. More importantly, unlike the baseline approaches, we do so without sacrificing the accuracy on the frequent classes. Finally, we also explore combining our approach with existing techniques [56] that results in even better performance.

Our main contributions are summarized as follows:

- Bringing the best of object-centric images to the long-tailed object detection on scene-centric images.
- Algorithms for mosaicking and pseudo-labeling to mitigate the domain discrepancy between two image types.
- A multi-stage training framework that leverages the pseudo scene-centric images (from object-centric images) to improve the detector on scene-centric images.
- Extensive evaluation and analysis of the proposed approach on the challenging LVIS benchmark [22].

2. Related Work

Long-tailed object detection has attracted increasing attentions recently. The challenge is the drastically low accuracy for detecting rare objects. Most existing works develop training strategies or objectives to address this [22, 32, 45, 53, 56, 63, 64, 71, 72, 74]. Wang et al. [71] found that the major performance drop is by mis-classification, suggesting the applicability of class-imbalanced classification methods (*e.g*., re-weighting, re-sampling) [6–8, 10, 10, 21, 24, 24, 36, 38, 48, 67, 69, 80]. Different from them, we study an alternative and orthogonal solution to the problem (*i.e*., exploiting abundant object-centric images).

Weakly-supervised or semi-supervised object detection learns or improves object detectors using images with weak supervision (*e.g*., image-level labels) [3, 12, 17, 43, 68] or even without supervision [17, 35, 44, 61, 83]. They either leverage scene-centric images or detect only a small number of common classes (*e.g*., classes in Pascal VOC [13], MSCOCO [48], or ILSVRC [58]). Our work can be seen as weakly supervised object detection, but we focus on the challenging long-tailed detection with more than $1,000$ objects. Meanwhile, we leverage object-centric images, which is different from scene-centric images in both appearances and layouts. The most related work is [53], which uses the YFCC-100M dataset [67] (Flickr images) to improve the detection on LVIS [22]. However, YFCC-100M contains both object-centric images and scene-centric images and a non-negligible label noises. Thus, [53] employs more sophisticated data pre-processing and pseudo-labeling steps, yet our approach achieves higher accuracy (see Table 5).

Other works use object-centric images to expand the label space of the object detector [29–31, 40, 55, 65, 66]. Such approaches mostly only use object-centric images to learn the last fully-connected classification layer, instead of improving the features extractor. In contrast, our approach can improve the feature extractor, and successfully transfer knowledge to long-tailed instance segmentation.

3. Scene-Centric vs. Object-Centric Images

Images taken by humans can roughly be categorized into object-centric and scene-centric images. The former captures objects of interest (*e.g*., cats) and usually contains just one salient class whose name is used as the image label. The later captures a scene and usually contains multiple object

[1]Mosaic was exploited in [4, 9, 81], but mainly to combine multiple *scene-centric* images to simulate smaller object sizes or to increase the scene complexity, not to turn object-centric images into scene-centric ones.

instances of different classes in a complex background.

Recent object detection methods mainly focus on scene-centric images [22, 48, 59]. Since scene-centric images are not intended to capture specific objects, *object frequencies in our daily lives will likely be reflected in the images.* As such, the learned detector will have a hard time detecting rare objects: it just has not seen sufficient instances to understand the objects' appearances, shapes, variations, etc.

In contrast, humans tend to take object-centric pictures that capture interesting (and likely uncommon, rare) objects, especially during events or activities (*e.g.*, bird watching, *etc.*). Thus, a rare object in our daily lives may occur more often in the online object-centric images.

Discrepancy w.r.t. object frequencies. We compare object frequencies of the ImageNet [11] and LVIS (v0.5) [22] datasets. The former retrieved images from the Internet by querying search engines using the object class names (thus object-centric). Whereas the later used MSCOCO [48] dataset, which collects daily scene images with many common objects in a natural context (thus scene-centric).

The full ImageNet has $21,841$ classes, whereas LVIS has around $1,230$ classes. Using the WordNet synsets [51], we can match $1,025$ classes ($1,016$ classes are downloadable) between them. Figure 1 shows the number of object instances per class in LVIS and the number of images per corresponding class in ImageNet. It presents *a huge difference between object frequencies of these two datasets.* For example, ImageNet has a balanced distribution across classes and LVIS is extremely long-tailed. Even for rare classes in LVIS (those with < 10 training images), ImageNet usually contains more than $1,000$ images. Such a difference offers an opportunity to resolve the long-tailed object detection in scene-centric images via the help of object-centric images.

Discrepancy w.r.t. visual appearances and contents. Beyond frequencies, these two types of images also have other, less favorable discrepancies. The obvious one is the number of object instances per image. LVIS on average has **12.1** labeled object instances per image (the median number of instances per image is **6**). While most of the ImageNet images are not annotated with object bounding boxes, according to a subset of images used in the ILSVRC detection challenge [58], each image has **2.8** object instances. The larger number of object instances, together with the intention behinds the images, implies that scene-centric images also have *smaller* objects in size and more complex backgrounds. This type of discrepancies, contrast to that in object frequencies, is not favorable for leveraging the object-centric images, and may lead to negative transfer [73, 78].

4. Overall Framework

To better leverage object-centric images for object detection, we present a novel learning framework, which includes **three simple[2] yet effective components** to handle

(a) the domain gap between two image sources;
(b) the missing bounding box labels;
(c) the integration of both image sources for training.

Figure 2 gives an illustration of our framework. Concretely, the framework begins with pre-training an object detector using the accurately labeled scene-centric images. Any object detector can be applied. Without loss of generality, we focus on Faster R-CNN [57], one of the most popular object detectors in the literature. The pre-trained object detector serves for two purposes: it can help impute the missing boxes in object-centric images; it will be used as the initialization for training with the object-centric images.

To turn object-centric images into training examples for object detection, we must handle both (a) and (b). *We postpone the details of these two components to § 5.* For now, let us assume that we have processed and labeled object-centric images with pseudo ground-truth boxes and class labels like the labeled scene-centric images. To differentiate from the original object-centric images, we call the new images *pseudo scene-centric images* (see Figure 2).

The pseudo scene-centric images may still have domain gaps from real scene-centric images, to which the learned detector will finally be applied. Besides, the pseudo ground-truth boxes may contain noises (*e.g.*, wrong locations). To effectively learn from these images (especially for rare objects) while not sacrificing the detector's ultimate accuracy on identifying and locating objects, we propose to *fine-tune the pre-trained object detector via two stages.* In what follows, we first give a brief review of object detection.

Backgrounds on object detection. An object detector has to identify objects with their class names and locate each of them by a bounding box. Taking Faster R-CNN [57] as an example, it first generates a set of object proposals (usually around 512) that likely contain objects of interest. This is done by the region proposal network (RPN) [57]. Faster R-CNN then goes through each proposal to identify its class (can be "background") and refine the box location and size.

The entire Faster R-CNN is learned with three loss terms

$$\mathcal{L} = \mathcal{L}_{\text{rpn}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{reg}}, \tag{1}$$

where $\mathcal{L}_{\text{rpn}}$ is for RPN training, $\mathcal{L}_{\text{cls}}$ is for multi-class classification, and $\mathcal{L}_{\text{reg}}$ is for box refinement.

Two-stage fine-tuning. Given the pre-trained detector, we first fine-tune it using the pseudo scene-centric images that are generated from object-centric images (see § 5). We then fine-tune it using the labeled scene-centric images. We separate the two image sources since they are still different

[2]We claim our approach to be "simple" as it employs simple methods to address the fundamental challenges. Pseudo-labeling is an essential step to use weakly-supervised data, and we apply simple fixed locations. We apply mosaicking and multi-stage training to bridge the domain gap, instead of applying sophisticated methods like domain adversarial training [16].

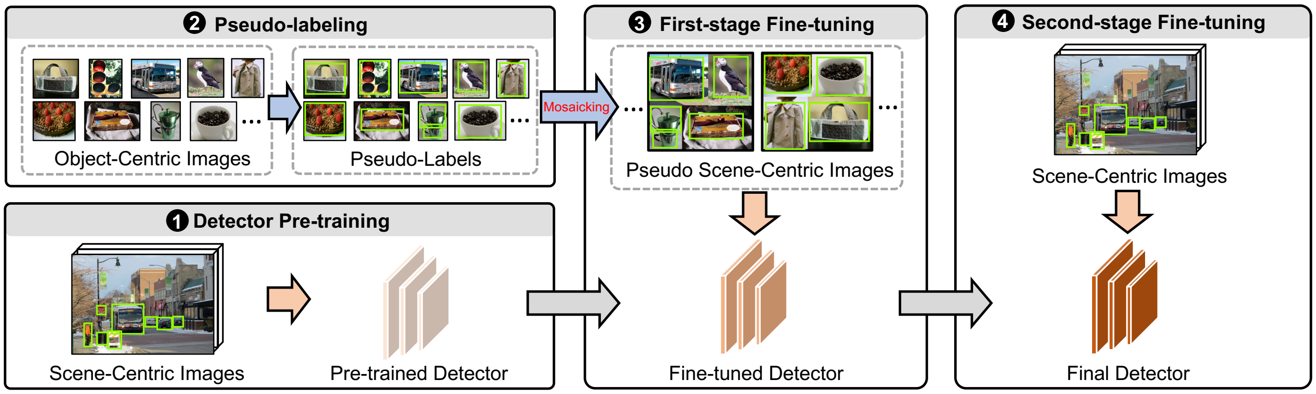

Figure 2. **Our MosaicOS framework for leveraging object-centric images for long-tailed object detection.** It consists of four stages. ❶ **Detector Pretraining**: we pre-train an object detector using scene-centric images with gold-labeled box annotations. ❷ **Pseudo-labeling**: we construct pseudo scene-centric images from object-centric images using box annotation imputation (possibly using the pre-trained detector in stage 1) as well as mosaicking (stitching multiple images together). ❸ **First-stage Fine-tuning**: we fine-tune the pre-trained detector from stage 1 with pseudo scene-centric images from stage 2. ❹ **Second-stage Fine-tuning**: we further fine-tune the object detector from stage 3 using scene-centric images with gold-labeled box annotations again, similar to stage 1. Orange arrows indicate data feeding for training. Gray arrows indicate model cloning. Green boxes indicate the (pseudo & gold-labeled) box annotations.

in appearances and label qualities. The second stage helps adapt the detector back to real scene-centric images.

In both stages, all the three loss terms in Equation 1 are optimized. We do not freeze any parameters except the batch-norm layers [34] in the backbone feature network which are kept frozen by default. We will compare to single-stage fine-tuning with both images and fine-tuning using only $\mathcal{L}_{\text{cls}}$ for pseudo scene-centric images in § 6.

5. Creating Pseudo Scene-Centric Images

We now focus on the missing components of our framework: generating pseudo scene-centric images from object-centric images. Our goal is to create images that are more *scene-centric-like* and label them with pseudo ground truths. We collect object-centric images from two sources: ImageNet [11] and Google Images. See § 6.1 for details.

5.1. Assigning Pseudo-Labels

Each object-centric image has one object class label, but no bounding box annotations. Some images may contain multiple object instances and classes, in which the class label only indicates the most salient object. Our goal here is to create a set of pseudo ground-truth bounding boxes that likely contain the object instances for each image, and assign each of them a class label, such that we can use the image to directly fine-tune an object detector.

There are indeed many works on doing so, especially those for weakly-supervised and semi-supervised object detection [3, 12, 17, 43, 44, 53, 68]. The purpose of this subsection is therefore not to propose a new way to compare with them, but to investigate approaches that are more effective and efficient in a large-scale long-tailed setting. Specifically, we investigate five methods that do not require an extra detector or proposal network beyond the pre-trained one. *As will be seen in § 6.2, imputing the box class labels using the image class label is the key to success.* Figure 3 illustrates the difference of these methods. Please see the supplementary material for other possibilities.

Fixed locations (F). We simply assign some fixed locations of an object-centric image to be the pseudo ground-truth boxes, regardless of the image content. The hypothesis is that many of the object-centric images may just focus on one object instance whose location is likely in the centre of the image (*i.e.*, just like those in [14, 20]). Specifically, we investigate the combination of the whole image, the center crop, and the four corner crops: in total **six** boxes per image. The height and width of the crops are 80% of the original image. We assign each box the image class label.

Trust the pre-trained detector (D). We apply the pre-trained detector learned with the scene-centric images to the object-centric images, and treat the detected boxes and predicted class labels as the pseudo-labels. Specifically, we keep all the detection of confidence scores > 0.5[3]. We apply non-maximum suppression (NMS) among the detected boxes of each class using an IoU (intersection-over-union) threshold 0.5. By doing so, every image will have boxes of different sizes and locations, labeled with different classes.

Trust the pre-trained detector & image class labels (D†). One drawback of the above method is its tendency to assign high-frequency labels, a notorious problem in class-imbalanced learning [36, 63, 76]. For instance, if "bird" is a frequent class and "eagle" is a rare class, the detector may correctly locate an "eagle" in the image but assign the label "bird" to it. To resolve this issue, we choose to trust the box locations generated by the above method but assign

[3] 0.5 is the default threshold for visualizing the detection results.

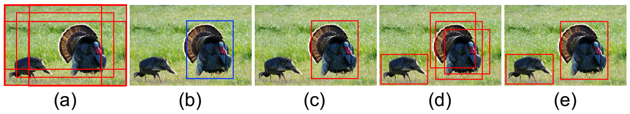

Figure 3. **A comparison of pseudo-label generation:** (a) fixed locations, (b) trust the detector, (c) trust the detector + image labels, (d) trust the calibrated detector + image labels, and (e) localization by region removal. The image label is "turkey," a rare class in LVIS. Red/blue boxes are labeled as "turkey"/other classes.

each box the image class label instead of the predicted class labels. In other words, a box initially labeled as "bird" is replaced by the label "eagle" if "eagle" is the image label. The rationale is that in an object-centric image, most of the object instances belong to the image's class.

Trust the calibrated detector & image class labels (D‡). Another way to resolve the above issue is to set for each class a different confidence threshold[4]. The rational is that a classifier trained with long-tailed data tends to assign lower probabilities to minor classes of scarce training data [6, 36, 37, 76]. We thus reduce the threshold for each class according to its number of training images. Let N_{max} be the size of the most major class and let N_{c} be the size of class c, we apply a threshold $0.5 \times (N_{\text{c}}/N_{\text{max}})^{\gamma}$ for class c, inspired by [50]. We set $\gamma = 0.5$ according to validation. Compared to the vanilla "trust the detector," this method will generate more boxes for common and rare classes. We again replace their detected labels by the image class label[5].

Localization by region removal (LORE). We investigate yet another way for pseudo-labeling, taking the following intuition: an image classifier should fail to predict the correct label if the true object regions are removed. To this end, we first train a ResNet-50 [25] image classifier using our object-centric image pool[6]. We then collect the pre-trained detector's predicted boxes on these images, trusting the box locations but not the class labels. We sort these boxes by how much removing each boxed region alone reduces the image classifier's confidence on the image label. We then remove these boxed regions *in turn* until the classifier fails to predict the true label. The bounding boxes of the removed regions are then collected as the pseudo ground truths for the image. We assign each box the image class label. Please see the supplementary material for details.

5.2. Synthesizing Pseudo Scene-centric Images

We apply a simple technique, *i.e.*, *image mosaic*, to make object-centric images more scene-centric, in terms of appearances, contents, and layouts. Concretely, we stitch multiple object-centric images together to obtain a new image that contains more object instances and classes, smaller object sizes, and more complex background. Specifically, we stitch 2×2 images together, which are sampled either randomly within a class, or randomly from the entire image pool. We do not apply sophisticated stitching tools like [5, 23, 79] but simply concatenate these images one-by-one. The resulting images are thus more like *mosaics*, having artifacts along the stitched boundaries (see Figure 2).

6. Experiments

We conduct experiments and analysis for MosaicOS, on the tasks of long-tailed Object Detection (OD) and Instance Segmentation (IS). We begin by introducing the experimental setup (§ 6.1), then present the main object detection results as well as detailed ablation studies (§ 6.2, § 6.3), and finally show additional results that evaluate our model on instance segmentation and the other dataset (§ 6.4). *We include qualitative results in the supplementary material.*

6.1. Setup

Long-tailed OD & IS datasets and metrics. We evaluate our approach on LVIS instance segmentation benchmark [22]. We focus on v0.5 as most existing works, and report additional key results on v1.0 (more in the supplementary). LVIS v0.5 contains $1,230$ entry-level object categories with around 2 million high-quality annotations. The training set contains all the classes with a total of $57,623$ image; the validation set contains 830 classes with a total of $5,000$ images. The categories are naturally long-tailed distributed and are divided into three groups based on the number of training images per class: rare (1-10 images), common (11-100 images), and frequent ($>$100 images). *All results are reported on the validation set.* We adopt the standard mean average precision (AP) metric in LVIS [22]. *We specifically focus on object detection using the standard bounding box evaluation, AP^b.* The AP on the rare, common, and frequent classes (AP^b_r, AP^b_c, AP^b_f) are also reported separately.

Object-centric data sources. We mainly use images from two sources: ImageNet [11] and Google Images [2]. ImageNet is a classification benchmark. Most people use its $1,000$ categories version in ILSVRC [58] and treat it as the standard dataset for backbone pre-training in various computer vision tasks. The full version of ImageNet has $21,842$ classes. In LVIS and ImageNet, each category has a unique WordNet synset ID [51], and we are able to match $1,016$ LVIS classes and retrieve the corresponding images from ImageNet (in total, $769,238$ images). Besides, we retrieve images via Google by querying with class names and descriptions provided by LVIS. Such a text-to-image search returns hundreds of iconic images and we take the top 100 for each of the $1,230$ classes.

[4] For Faster R-CNN, each RPN proposal can lead to multiple detected boxes, one for each class whose probability is larger than the threshold.

[5] Without doing so, the approach can hardly improve "trust the detector" due to more noisy boxes being included. See the supplementary for details.

[6] That is, we train the classifier with these images, and then apply this classifier back to these images (after some regions are removed).

Implementation. We use Faster R-CNN [57][7] as our base detector and ResNet-50 [28] with a Feature Pyramid Network (FPN) [46] as the backbone, which is pre-trained on ImageNet (ILSVRC) [58]. Our base detector is trained on the LVIS training set with *repeated factor sampling*, following the standard training procedure in [22] (1x schedule). To fairly compare with our fine-tuning results, we further extend the training process with another 90K iterations and select the checkpoint with the best AP^b as **Faster R-CNN$\star$**. The following experiments are initialized by Faster R-CNN$\star$. See the supplementary for more details.

Baselines. We compare to the following baselines:

- **Self-training** is a strong baseline for semi-supervised learning [42]. We follow the state-of-the-art self-training method for detection [83] and use Faster R-CNN$\star$ to create pseudo-labels on the object-centric images, same as "trust the pre-trained detector". We then fine-tune Faster R-CNN$\star$ using both pseudo scene-centric and LVIS images for 90K iterations, with the normalized loss [83].
- **Single-stage fine-tuning** fine-tunes Faster R-CNN$\star$ with both pseudo scene-centric and LVIS images in one stage. In each mini-batch, we have 50% of data from each source. Different ratios do not lead to notable differences.
- **DLWL** [53] is the state-of-the-art method that uses extra unlabeled images from the YFCC-100M [67].

For self-training and single-stage training, we perform 2×2 mosaicking to create pseudo scene-centric images.

Variants of MosaicOS. (a) We compare different object-centric image sources and their combinations. (b) We compare with or without mosaicking. (c) For mosaicking, we compare stitching images from the same classes (so the artifacts can be reduced) or from randomly selected images. (d) We compare different ways to generate pseudo-labels (see § 5.1). (e) We study fine-tuning with pseudo scene-centric images using only the classification loss $\mathcal{L}_{\text{cls}}$.

6.2. Results on Object Detection

Main results. Table 1 summarizes the results. It shows that the model trained with pseudo-labels generated by six fixed location (F) has a very competitive performance comparing to other strategies. We therefore consider it as the default pseudo-labeling method given its simplicity and effectiveness. Meanwhile, our two-stage approach with object-centric images outperforms Faster R-CNN$\star$ (and Faster R-CNN) notably. On AP_r^b for rare classes, our best result of 20.25% is $\sim$ 7.2% higher than Faster R-CNN$\star$, justifying our motivation: object-centric images that are resistant to the long-tailed nature of object frequencies can improve object detection in scene-centric images.

Mosaicking is useful (red in Table 1). A simple 2×2 stitching leads to a notable gain: $\sim 1.8\%$ at AP_r^b, supporting our claim that making object-centric images similar to scene-centric images is important. Indeed, according to § 3, a 2×2 stitched image will have around 12 objects, very close to that in LVIS images. Stitching images from different classes further leads to a $\sim 1.0\%$ gain (green in Table 1).

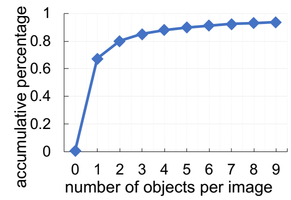

Figure 4. **# of objects per object-centric image found by LORE.** We use ImageNet images. Y-axis is the accumulative percentage of images whose objects are no more than the X-axis number.

Fixed-location boxes are effective (blue in Table 1). By comparing different ways for pseudo-labels, we found that both localization by region removal (L) and the simple six fixed locations (F) lead to strong results without querying the pre-trained detector. Using six fixed locations slightly outperforms using one location (S) (*i.e.*, the image boundary), probably due to the effect of data augmentation. All the three methods significantly outperform "trust the pre-trained detector" (D), and we attribute this to the poor pre-trained detector's accuracy on rare classes: it either cannot identify rare classes from object-centric images or is biased to detect frequent classes. **By replacing the detected classes with the image labels** and/or further calibrating the detector for more detected boxes, *i.e.*, "trust the (calibrated) pre-trained detector and image label" (D†, D‡), we see a notable boost, which supports our claim. Nevertheless, they are either on par with or worse than fixed locations (F), especially on AP_r^b for rare classes, again showing the surprising efficacy of the simple method. We note that, both D† and D‡ are specifically designed in this work for long-tailed problems and should not be seen as existing baselines.

To further analyze why fixed locations work well, we check the numbers of boxes LORE finds per image. LORE keeps removing regions until the classifier fails to classify the image. The number of regions it found is thus an estimation of the number of target objects (those of the image label) in the image. Figure 4 shows the accumulative number of images whose object numbers are no more than a threshold: $\sim$ 70% of ImageNet images have one target object instance, suggesting that it may not be necessary to locate and separate object instances in pseudo-labeling.

Self-training and single-stage fine-tuning. As shown in Table 1, self-training (with D and loss normalization [83]) outperforms Faster R-CNN$\star$ on AP_r^b. As self-training is sensitive to the pseudo-label quality, we also apply the fixed locations (F) to it and achieve improvement. By comparing it to single-stage fine-tuning (with F), we see the benefit of loss normalization between the two image sources.

By comparing self-training to its counterparts in two-stage fine-tuning (with D and F), we however find that two-

[7] Our implementation is based on [75], which uses RoIAlign [27] instead of RoIPool [18] to extract region features for R-CNN training.

Table 1. **Comparison of object detection on LVIS v0.5 validation set. OCIs:** object-centric images sources (IN: ImageNet, G: Google). **Mosaic:** ✓means 2×2 image mosaicking. **Hybrid:** ✓means stitching images from different classes. **P-GT:** ways to generate pseudo-labels (**F**: six fixed locations, **D**: trust the detector, **D**†: trust the detector and image label, **D**‡: trust the calibrated detector and image class label, **L** (LORE): localization by region removal, and **S**: a single box of the whole image). The best result per column is in bold.

	OCIs	Mosaic	Hybrid	P-GT	AP^b	AP^b_{50}	AP^b_{75}	AP^b_r	AP^b_c	AP^b_f
Faster R-CNN	-	-	-	-	23.17	38.94	24.06	12.64	22.40	28.33
Faster R-CNN⋆	-	-	-	-	23.35	39.15	24.15	12.98	22.60	28.42
Self-training [83]	IN	✓	✓	D	22.71	38.22	23.79	14.52	21.41	27.61
	IN	✓	✓	F	23.46	39.03	24.82	16.20	22.19	27.94
Single-stage	IN	✓	✓	F	20.09	35.34	20.27	12.96	19.08	24.20
MosaicOS (Two-stage)	IN	✗	✗	F	24.27	40.30	25.61	16.97	23.29	28.42
	IN	✓	✗	F	24.48	40.12	25.65	18.76	23.26	28.29
	IN	✓	✓	D	23.04	38.97	23.72	13.93	21.51	28.14
	IN	✓	✓	D†	24.66	40.31	25.99	17.45	23.62	28.83
	IN	✓	✓	D‡	24.93	40.48	26.71	19.31	23.51	**28.95**
	IN	✓	✓	S	24.59	40.20	25.78	19.13	23.35	28.32
	IN	✓	✓	L	24.83	40.58	26.27	20.06	23.25	28.71
	IN	✓	✓	F	24.75	40.44	26.09	19.73	23.44	28.39
	IN+G	✓	✓	F	**25.01**	**40.76**	**26.46**	**20.25**	**23.89**	28.32

Table 2. **Losses in the first fine-tuning stage.**

Losses	AP^b	AP^b_r	AP^b_c	AP^b_f
$\mathcal{L}_{\text{cls}}$	24.53	18.87	23.07	28.61
$\mathcal{L}_{\text{rpn}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{reg}}$	24.75	19.73	23.44	28.39

stage fine-tuning leads to higher accuracy in most cases. This demonstrates the benefit of separating image sources in fine-tuning, in which the second stage adapts the detector back to accurately labeled true scene-centric images.

The amount of object-centric data (brown in Table 1). As ImageNet only covers $1,016$ classes of LVIS, we augment it with 100 Google images per class for all the $1,230$ LVIS classes. We see another 0.5% gain at rare classes (AP^b_r).

For the following analyses besides Table 1, we will focus on our approach with two-stage fine-tuning, ImageNet object-centric images, 2×2 *mosaic with images from multiple classes, and fixed locations (F) as the pseudo-labels.*

Losses in fine-tuning. We compare using all three losses of Faster R-CNN (*i.e.*, $\mathcal{L}_{\text{rpn}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{reg}}$) or just the classification loss (*i.e.*, $\mathcal{L}_{\text{cls}}$) in the first-stage fine-tuning with pseudo scene-centric images. Table 2 shows the results. The former outperforms the latter on three out of four metrics. This tells that, while the pseudo boxes do not accurately bound the objects, learning the RPN and box refinement with them (*e.g.*, to predict a high objectness score) is still beneficial.

Other baselines. We compare to state-of-the-art methods that use no extra object-centric images in Table 3. We obtain comparable or better results, especially on rare classes.

Compatibility with existing efforts. Our approach is compatible with recent efforts in better backbone pre-training [45] and advanced training objectives (*e.g.*, [56]). For instance, following BaGS [45] to pre-train the backbone using MSCOCO images, we achieve an improved 26.28 AP^b (see Table 3). Further incorporating the balanced loss [56] into the second-stage fine-tuning boosts AP^b to 28.06.

Table 3. **Object detection on LVIS v0.5.** We use ImageNet + Google Images. MSCOCO: for pre-training. [56]: balanced loss.

	MSCOCO	[56]	AP^b	AP^b_r	AP^b_c	AP^b_f
BaGS [45]	✓		25.96	17.65	25.75	29.54
TFA [72]			24.40	16.90	24.30	27.70
MosaicOS			25.01	20.25	23.89	28.32
	✓		26.28	17.37	26.13	30.02
		✓	26.83	**21.00**	26.31	29.81
	✓	✓	**28.06**	19.11	**28.23**	**31.41**

Table 4. **The importance of mosaicking object-centric images.** SCI: object-centric images in the original LVIS training set.

	AP^b	AP^b_r	AP^b_c	AP^b_f
Faster R-CNN⋆	23.35	12.98	22.60	28.42
Stitching SCI [4]	23.83	13.99	23.02	**28.76**
Stitching SCI [9]	23.58	14.00	22.58	28.66
Stitching cropped SCI [81]	23.55	13.40	23.04	28.26
MosaicOS	**24.75**	**19.73**	**23.44**	28.39

6.3. Detailed Analysis of MosaicOS

The importance of object-centric images. In Table 1, we show that even without mosaic, the use of object-centric images already notably improves the baseline (AP^b: 24.27 vs. 23.35). Here, we further investigate the importance of mosaic of object-centric images: our use of mosaic is different from [4, 9, 81], which stitch scene-centric images in the training set to simulate smaller objects or increase the scene complexity. We apply their methods to stitch LVIS images and study two variants: stitching scene-centric images [4, 9] or the cropped objects [81] from them. Table 4 shows that MosaicOS surpasses both variants on rare and common objects, justifying the importance of incorporating ample object-centric images to capture the diverse appearances of objects, especially for rare objects in scene-centric images.

Does the quality of data sources matter? DLWL [53] uses YFCC-100M [67], a much larger data source than Im-

Figure 5. **A comparison of object-centric image sources.** We show images of a rare class ("ax") in LVIS. ImageNet [11] (right) and Google Images [2] (middle) give images with more salient "ax" inside, while Flickr [1] (left) gives more noisy images, either with very small axes, cluttered backgrounds, or even no axes.

Table 5. **Comparison of object detection on LVIS v0.5 using different extra data sources.** G: Google Images. IN: ImageNet.

	Data	AP^b	AP^b_r	AP^b_c	AP^b_f
Faster R-CNN⋆	–	23.35	12.98	22.60	28.42
DLWL [53]	YFCC-100M	22.14	14.21	-	-
MosaicOS	Flickr	24.05	16.17	23.06	28.43
	G	24.45	19.09	23.27	28.08
	IN	24.75	19.73	23.44	28.39

Table 6. **Object detection on the 176 overlapped classes** between ImageNet-1K (ILSVRC) and LVIS v0.5.

	AP^b	AP^b_r	AP^b_c	AP^b_f
# Category	176	21	84	71
Faster R-CNN⋆	26.05	14.78	23.92	31.16
MosaicOS	**27.50**	**21.16**	**25.45**	**31.80**

ageNet. YFCC-100M images are mainly collected from Flickr, which mixes object-centric and scene-centric images and contains higher label noises. DLWL [53] thus develops sophisticated pre-processing and pseudo-labeling steps. In contrast, we specifically leverage *object-centric* images that have higher object frequencies and usually contain only single object classes (the image labels), leading to a much simpler approach. As shown in Table 5, our method (with IN) outperforms DLWL by a large margin: $> 5.5\%$ at AP^b_r. We attribute this to our ways of strategically collecting object-centric images and stitching them to make them scene-centric-like. The fact that we identify a better data source should not lead to an impression that we merely solve a simpler problem, but an evidence that selecting the right data source is crucial to simplify a problem. Figure 5 illustrates the difference among these sources.

For a fair comparison to [53] in terms of the algorithms, we also investigate Flickr images. Since [53] does not provide their processed data, we directly crawl Flickr images (100 per class) and re-train our algorithm. We achieve $24.05/16.17$ AP^b/AP^b_r, better than DLWL. Using pure Google images beyond ImageNet can achieve $24.45/19.09$. Our novelties and contributions thus lie in both the algorithm and the direction we investigate. The latter specifically leads to simpler solutions but higher accuracy.

The importance of learning for the downstream tasks. We found 176 classes of LVIS validation set in ILSVRC. That is, the corresponding ImageNet images used by MosaicOS are already seen by the pre-trained detector's backbone. Surprisingly, as shown in Table 6, MosaicOS still leads to a notable gain for these classes, which not only justifies its efficacy, but also suggests the importance of learning the downstream tasks directly with those images.

Table 7. **Instance segmentation on LVIS v0.5.** We use images from IN + G as Table 1. + [56]: include the balanced loss.

	AP	AP_r	AP_c	AP_f
Mask R-CNN [22]	24.38	15.98	23.96	28.27
BaGS [45]	26.25	17.97	26.91	28.74
BALMS [56]	27.00	19.60	**28.90**	27.50
MosaicOS	26.26	19.63	26.60	28.49
MosaicOS + [56]	**27.86**	**20.44**	28.82	**29.62**

Table 8. **LVIS v1.0 Results.** We report both object detection and instance segmentation performances of our method.

OD Results	AP^b	AP^b_{50}	AP^b_{75}	AP^b_r	AP^b_c	AP^b_f
Faster R-CNN⋆	22.01	36.36	23.14	10.57	20.09	29.18
MosaicOS	**23.90**	**38.61**	**25.32**	**15.45**	**22.39**	**29.30**
IS Results	AP	AP_{50}	AP_{75}	AP_r	AP_c	AP_f
Mask R-CNN	22.59	35.44	23.87	12.31	21.30	28.55
MosaicOS	**24.49**	**38.02**	**25.87**	**18.30**	**23.00**	**28.87**

6.4. Results on Instance Segmentation & LVIS v1.0

Instance segmentation. We also validate our approach on instance segmentation, in a similar manner: we prepare pseudo scene-centric images *with box labels* and use them in the first fine-tuning stage by optimizing the losses in Equation 1. That is, we do not optimize segmentation losses in this stage. We apply Mask R-CNN [27] with ResNet-50 as the backbone. Table 7 shows the results: *the baselines do not use extra object-centric images*. We see a notable gain against vanilla Mask R-CNN for rare and common classes, even if we have no segmentation labels on object-centric images. This supports the claim in [71]: even for detection and segmentation, the long-tailed problem is mainly in the classification sub-network. We perform on par with the state-of-the-art methods. Details are in the supplementary.

LVIS v1.0 results. We highlight consistent empirical results on LVIS v1.0, where our approach wins in both object detection and instance segmentation, using ResNet-50-FPN (see Table 8). More comparisons are in the supplementary.

7. Discussion and Conclusion

We investigate the use of object-centric images to facilitate long-tailed object detection on scene-centric images. We propose a concrete framework for this idea that is both simple and effective. Our results are encouraging, improving the baseline by a large margin on not only detecting but also segmenting rare objects. We hope that our study can attract more attention in using these already available but less explored object-centric images to overcome the long-tailed problem. Please see the supplementary for more discussion.

Acknowledgments. We are thankful for the generous support by Ohio Supercomputer Center and AWS Cloud Credits for Research.

References

[1] Flickr images. https://flickr.com/. 8

[2] Google images. https://www.google.com/imghp?hl=EN. 5, 8

[3] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *CVPR*, 2016. 2, 4

[4] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020. 1, 2, 7

[5] Matthew Brown and David G Lowe. Automatic panoramic image stitching using invariant features. *IJCV*, 74(1):59–73, 2007. 5

[6] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018. 2, 5

[7] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *NeurIPS*, 2019.

[8] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002. 2

[9] Yukang Chen, Peizhen Zhang, Zeming Li, Yanwei Li, Xiangyu Zhang, Gaofeng Meng, Shiming Xiang, Jian Sun, and Jiaya Jia. Stitcher: Feedback-driven data provider for object detection. *arXiv preprint arXiv:2004.12432*, 2020. 2, 7

[10] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, 2019. 2

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1, 2, 3, 4, 5, 8

[12] Santosh K Divvala, Ali Farhadi, and Carlos Guestrin. Learning everything about anything: Webly-supervised visual concept learning. In *CVPR*, 2014. 2, 4

[13] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 2

[14] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR*, 2004. 4

[15] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 32(9):1627–1645, 2009. 1

[16] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *JMLR*, 17(1):2096–2030, 2016. 3

[17] Jiyang Gao, Jiang Wang, Shengyang Dai, Li-Jia Li, and Ram Nevatia. Note-rcnn: Noise tolerant ensemble rcnn for semi-supervised object detection. In *ICCV*, 2019. 2, 4

[18] Ross Girshick. Fast R-CNN. In *ICCV*, 2015. 6

[19] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 1

[20] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007. 4

[21] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *ECCV*, 2016. 2

[22] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 1, 2, 3, 5, 6, 8

[23] James Hays and Alexei A Efros. Scene completion using millions of photographs. *ACM Transactions on Graphics (TOG)*, 26(3):4–es, 2007. 5

[24] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009. 2

[25] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009. 5

[26] Kaiming He, Ross Girshick, and Piotr Dollár. Rethinking imagenet pre-training. In *ICCV*, 2019. 2

[27] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017. 1, 2, 6, 8

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 6

[29] Judy Hoffman, Sergio Guadarrama, Eric S Tzeng, Ronghang Hu, Jeff Donahue, Ross Girshick, Trevor Darrell, and Kate Saenko. Lsda: Large scale detection through adaptation. In *NIPS*, 2014. 2

[30] Judy Hoffman, Deepak Pathak, Trevor Darrell, and Kate Saenko. Detector discovery in the wild: Joint multiple instance and representation learning. In *CVPR*, 2015.

[31] Judy Hoffman, Deepak Pathak, Eric Tzeng, Jonathan Long, Sergio Guadarrama, Trevor Darrell, and Kate Saenko. Large scale visual recognition through adaptation using joint representation and multiple instance learning. *JMLR*, 17(1):4954–4984, 2016. 2

[32] Xinting Hu, Yi Jiang, Kaihua Tang, Jingyuan Chen, Chunyan Miao, and Hanwang Zhang. Learning to segment the tail. In *CVPR*, 2020. 2

[33] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 1

[34] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 4

[35] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *NeurIPS*, 2019. 2

[36] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *ICLR*, 2020. 2, 4, 5

[37] Byungju Kim and Junmo Kim. Adjusting decision boundary for class imbalanced learning. *IEEE Access*, 8:81674–81685, 2020. 5

[38] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123(1):32–73, 2017. 2

[39] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1
[40] Jason Kuen, Federico Perazzi, Zhe Lin, Jianming Zhang, and Yap-Peng Tan. Scaling object detection by transferring classification weights. In *ICCV*, 2019. 2
[41] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4. *IJCV*, pages 1–26, 2020. 1
[42] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML Workshop on challenges in representation learning*, 2013. 6
[43] Dong Li, Jia-Bin Huang, Yali Li, Shengjin Wang, and Ming-Hsuan Yang. Weakly supervised object localization with progressive domain adaptation. In *CVPR*, 2016. 2, 4
[44] Yandong Li, Di Huang, Danfeng Qin, Liqiang Wang, and Boqing Gong. Improving object detection with selective self-supervised self-training. In *ECCV*, 2020. 2, 4
[45] Yu Li, Tao Wang, Bingyi Kang, Sheng Tang, Chunfeng Wang, Jintao Li, and Jiashi Feng. Overcoming classifier imbalance for long-tail object detection with balanced group softmax. In *CVPR*, 2020. 1, 2, 7, 8
[46] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 1, 6
[47] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 1
[48] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1, 2, 3
[49] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016. 1
[50] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. In *ECCV*, 2018. 5
[51] George A Miller. WordNet: A lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 3, 5
[52] Constantine P Papageorgiou, Michael Oren, and Tomaso Poggio. A general framework for object detection. In *ICCV*, 1998. 1
[53] Vignesh Ramanathan, Rui Wang, and Dhruv Mahajan. DLWL: Improving detection for lowshot classes with weakly labelled data. In *CVPR*, 2020. 2, 4, 6, 7, 8
[54] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016. 1
[55] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *CVPR*, 2017. 2
[56] Jiawei Ren, Cunjun Yu, Shunan Sheng, Xiao Ma, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Balanced meta-softmax for long-tailed visual recognition. In *NeurIPS*, 2020. 2, 7, 8
[57] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 1, 3, 6
[58] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 2, 3, 5, 6
[59] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *ICCV*, 2019. 1, 3
[60] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 1
[61] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020. 2
[62] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 1
[63] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *CVPR*, 2020. 1, 2, 4
[64] Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. Long-tailed classification by keeping the good and removing the bad momentum causal effect. In *NeurIPS*, 2020. 2
[65] Yuxing Tang, Josiah Wang, Boyang Gao, Emmanuel Dellandréa, Robert Gaizauskas, and Liming Chen. Large scale semi-supervised object detection using visual and semantic knowledge transfer. In *CVPR*, 2016. 2
[66] Yuxing Tang, Josiah Wang, Xiaofang Wang, Boyang Gao, Emmanuel Dellandréa, Robert Gaizauskas, and Liming Chen. Visual and semantic knowledge transfer for large scale semi-supervised object detection. *TPAMI*, 40(12):3045–3058, 2017. 2
[67] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *arXiv preprint arXiv:1503.01817*, 2015. 1, 2, 6, 7
[68] Jasper Uijlings, Stefan Popov, and Vittorio Ferrari. Revisiting knowledge transfer for training object class detectors. In *CVPR*, 2018. 2, 4
[69] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *CVPR*, pages 8769–8778, 2018. 1, 2
[70] Paul Viola, Michael Jones, et al. Robust real-time object detection. *IJCV*, 4(34-47):4, 2001. 1
[71] Tao Wang, Yu Li, Bingyi Kang, Junnan Li, Junhao Liew, Sheng Tang, Steven Hoi, and Jiashi Feng. The devil is in classification: A simple framework for long-tail instance segmentation. In *ECCV*, 2020. 2, 8
[72] Xin Wang, Thomas E Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. In *ICML*, 2020. 1, 2, 7
[73] Zirui Wang, Zihang Dai, Barnabás Póczos, and Jaime Carbonell. Characterizing and avoiding negative transfer. In *CVPR*, 2019. 2, 3
[74] Jialian Wu, Liangchen Song, Tiancai Wang, Qian Zhang, and

Junsong Yuan. Forest R-CNN: Large-vocabulary long-tailed object detection and instance segmentation. In *ACM MM*, 2020. 2

[75] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 6

[76] Han-Jia Ye, Hong-You Chen, De-Chuan Zhan, and Wei-Lun Chao. Identifying and compensating for feature deviation in imbalanced deep learning. *arXiv preprint arXiv:2001.01385*, 2020. 4, 5

[77] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. BDD100K: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, 2020. 1

[78] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *CVPR*, 2018. 2, 3

[79] Fan Zhang and Feng Liu. Parallax-tolerant image stitching. In *CVPR*, 2014. 5

[80] Yubo Zhang, Pavel Tokmakov, Martial Hebert, and Cordelia Schmid. A study on action detection in the wild. *arXiv preprint arXiv:1904.12993*, 2019. 2

[81] Dongzhan Zhou, Xinchi Zhou, Hongwen Zhang, Shuai Yi, and Wanli Ouyang. Cheaper pre-training lunch: An efficient paradigm for object detection. In *ECCV*. Springer. 2, 7

[82] Xiangxin Zhu, Dragomir Anguelov, and Deva Ramanan. Capturing long-tail distributions of object subcategories. In *CVPR*, pages 915–922, 2014. 1

[83] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. Rethinking pre-training and self-training. In *NeurIPS*, 2020. 1, 2, 6, 7