The Effect of Improving Annotation Quality on Object Detection Datasets: A Preliminary Study

Jiaxin Ma
OMRON SINIC X Corp.
jiaxin.ma@sinicx.com

Yoshitaka Ushiku
OMRON SINIC X Corp.
yoshitaka.ushiku@sinicx.com

Miori Sagara
Baobab Inc.
sagara@baobab-trees.com

Abstract

In this study, we partially reannotate conventional benchmark datasets for object detection and check whether there is performance improvement/drop compared with the original annotations. Recent studies on the annotation qualities of ImageNet for image classification revealed some issues of how to associate only a single label to each image accurately. Object detection, on the other hand, should have other nontrivial issues because there are multiple objects in a single image, and realizing consistency among bounding boxes is challenging. A team of professional annotators was formed for MS COCO and Google Open Images datasets. To realize highly-consistent annotations, we prepared carefully designed guidelines for each category and selected quality inspectors who checked the annotation quality of each annotator. Finally, we applied conventional object detection methods for reannotated parts of each dataset. We found mixed results: whether the performance dropped or improved depended on each category and dataset.

1. Introduction

Artificial intelligence (AI) systems are based on models and data. To date, most advancements in the AI field have been accomplished using standard benchmarks, fixed data, and improved models. In fact, most published works are about improving AI models and methods. On the other hand, the idea of "data-centric AI" focuses on the other element that makes up the AI system, which is the data. The concept of data-centric AI was first introduced by Andrew Ng in his open talk seminar in March 2021 [6]. Contrary to "model-centric AI" that fixes data and improves models, "data-centric AI" can be described as using a basic and fixed model and systematically improving the data quality.

According to the concept of data-centric AI, to improve the data means one or both of the following:

- increasing the training samples by applying data augmentation, generation, or collection (change input x).
- reducing the noise by fixing incorrect labels, or giving a more consistent definition for labels if they were ambiguous (change label y).

Although both strategies are useful, usually it takes more effort to collect new training samples than to clean up noise, especially for cases of small-scaled data such as medical data, or those long-tail classes in a big-data scenario.

In the computer vision field, the most prominent dataset is ImageNet [19]. Centered around ImageNet, advanced models, such as convolutional neural networks (CNNs), have rapidly developed, revealing the prosperity of model-centric AI. Recently, some scholars [3, 20, 31] have focused on the ImageNet dataset rather than the models. They revealed that ImageNet is "noisy" because a large portion of it contains multiple objects, and single-class labels implicitly assume there is only one object per image. Therefore, they revised the labels, expanding them to multiclass, and hence successfully improved the classification accuracy without modifying models.

Apart from ImageNet and the related image classification task, object detection is another fundamental and challenging problem in computer vision. Similar to the image classification task, new models have been continuously proposed to extend the boundaries in terms of object detection performance in fierce competitions. For example, the authors of [4] compared their model with up to 30 models. Meanwhile, studies focusing on improving the object detection dataset are inadequate.

Regarding the annotation quality of object detection tasks, giving bounding boxes to objects, correctly and consistently, requires skilled annotators, and the quality control of annotation is challenging. The authors of [21] stated that drawing a bounding box is significantly more difficult and time-consuming than the process of annotating classification labels (which is usually done by answering multiple-choice questions). Specifically, a bounding box needs to be correctly positioned to contain the target object while excluding nontarget objects and backgrounds as much as pos-

sible. In addition, in an image, if some objects of a category are given bounding boxes while others of the same category are not, the latter would be treated as negative samples by the learning algorithm and hence hinders the learning process. Moreover, for a typical case that has a mass of annotators working on a single dataset, it is critical and usually uneasy to guarantee that all annotators stick to the same standard to maintain the label consistency.

In this preliminary study, we reannotated 80k images of five categories (car, chair, cup, person, and traffic light) from the Microsoft Common Object in Context (MS COCO) dataset [12] and 5k images of five categories (building, car, dog, flower, and person) from the Google Open Images dataset [11]. To achieve high-quality annotation, our annotation process is performed by well-trained human annotators (from Baobab Inc.) under predetermined guidelines. We expect that our new labels improve correctness and consistency compared with the original labels and hence can benefit the machine learning process. Our reannotated datasets will be publicly available at https://baobab-trees.com/en/datasets/.

To verify the actual effect brought by our annotation, we performed object detection experiments using five well-known models (Faster RCNN [18], SSD [13], YOLOv3 [17], EfficientDet [23], and DETR [5]). Without modifying model architecture and intensively tuning hyperparameters, we trained the models with the original and reannotated datasets and tested their performances. The results are twofold: our annotation on MS COCO resulted in a performance drop, whereas our annotation on Google Open Images yielded an improvement. We concluded that although we thought its quality improved, our annotation does not necessarily benefit the learning process. It may either increase or decrease the difficulty of a task depending on the different annotation guidelines.

2. Related works

In this section, we describe related studies in two areas. The first one is data-centric AI, including some investigations of existing annotations for benchmark datasets. The second is a quick review of object detection and relations between object detection and other computer vision tasks, including the necessity of visiting annotation quality on object detection tasks.

The first data-centric AI competition was conducted by DeepLearning.AI and Landing AI. In the competition, participants were asked to perform a classification task, where they were not allowed to adjust the model architecture or hyperparameters. Instead, they were allowed to improve the dataset itself, e.g., fixing incorrect labels, adding data for side-case tuning [32], or applying data augmentation techniques. The provided training dataset has 3k labeled images, and the submission requires an improved dataset of up to 10k labeled images.

As described in 1, scholars have improved the annotation quality of standard benchmark datasets for computer vision. Studies that validate datasets in computer vision mainly include validation against image collection [24] and validation against their annotation. Recently, many studies have been conducted on the latter for ImageNet. Such label errors have been reported on multiple well-known datasets, resulting in performance drop, especially for deeper neural networks [14]. In [28], bird experts found around 4% error of annotations for bird images in CUB-200-2011 [30] and ImageNet. In [15], new test data for CIFAR-10 and ImageNet were collected, respectively, from Tiny Images [25] and Flickr. In [2], a large real-world test set was also collected for image classification such that backgrounds, rotations, and viewpoints were well-controlled. Rather than collecting new data, some studies [3, 26] kept the original test data and fixed their annotations. Some studies [20, 31] relabeled images with multiple categories rather than a single category as in the original ImageNet.

Notably, data-centric AI and dataset improvement are mutually related but not equivalent. ImageNetV2, provided in [15], showed a consistent accuracy drop for a wide range of classification models [15,20]. In [9], the VQA dataset [1] was modified to develop VQA v2.0 by balancing the frequency of answers, and its experimental results showed that the accuracy of answers on the original VQA dataset was better than that of VQA v2.0. The main difference between data-centric AI and dataset improvement is that the former changes the dataset aiming to increase accuracy, whereas the latter changes the dataset based on other motivations, such as redefining a problem.

Object detection is an intrinsic recognition task in computer vision and is incorporated as a module in numerous tasks. This task has a long history in computer vision, and there are numerous methods for object detection using neural networks. Region-based CNN (RCNN) [8] is a two-stage method for object detection. Whereas RCNN requires object proposals from selective search [27], Faster RCNN [18] enables a real-time object detection using a region proposal network. You Only Look Once (YOLO) [16] is the first single-stage method that reframes object detection as a regression problem by directly predicting object categories and bounding box attributes for each pixel. Single Shot Multibox Detector (SSD) [13] is another single-stage method that uses fully convolutional neural networks, achieving better accuracy and speed than YOLO. EfficientDet [23], which is also a single-stage method, further employs EfficientNet [22] and proposes bidirectional feature pyramid network. DEtection TRansformer (DETR) [5] uses a transformer [29] with a CNN backbone. Whereas most detection methods use non-maximum suppression for overlapped predictions, DETR uses the Hungarian algorithm for

a set-based prediction.

There are also multiple benchmark datasets for object detection. PASCAL VOC [7] is a well-known dataset having annotations for 20 categories. It was used for competitions for computer vision tasks, including object detection, from 2005 to 2012. MS COCO [12] is a larger dataset for 80 common objects. MS COCO has been used for several competitions of object detection, segmentation, captioning, and keypoint detection. Google Open Images [11] is another dataset having more images associated with bounding box annotations for 500 categories. This dataset is used for Open Images Challenge since 2018, including multiple tasks, such as object detection, visual relationship detection, and segmentation.

Rethinking annotations for object detection datasets is not a trivial extension of those for image classification. The existing datasets for image classification have a common issue of exclusively selecting a single category even if their images contain multiple objects. This issue does not exist in object detection because annotation for each image contains multiple objects associated with their bounding boxes. Moreover, labeling bounding boxes for every object is time-consuming, often skipped or wild, as shown later.

Notably, object detection is aimed at detecting objects on an instance-by-instance basis, in contrast to the presence of both semantic segmentation and instance segmentation in segmentation tasks. For semantic segmentation, annotations for a single category are not separated if multiple instances are mutually occluded. Instance segmentation tries separating them while ignoring some categories that cannot be counted, such as sky and road. There is also an interesting task called panoptic segmentation [10], a combination of semantic and instance segmentations. Annotations for object detection are mostly given in an instance-level manner, but sometimes a union of each object is associated with a single bounding box, as shown later.

3. Method

In the original annotation processes of MS COCO and Google Open Images, some efforts related to annotation quality have been made. In MS COCO, annotators need to pass a segmentation training task periodically, where their results need to match the ground truth [12]. In Google Open Images, annotators are shown some positive examples and common mistakes of bounding boxes before they start annotation sessions [11]. However, usually, such annotation processes on a large-scale dataset involve enormous annotators and they are likely to have different backgrounds and cultures, which may cast bias on their annotation outputs. A few positive and negative samples or simple training may be insufficient for those annotators to reach a consensus on every object category. In this study, to achieve high annotation quality, we formulated more detailed annotation guidelines and employed systematically well-trained annotators.

3.1. Annotation guidelines

We made the annotation guidelines after carefully studying the target datasets. For an object detection task, our guideline first includes some general standards related to how to annotate 1) crowding objects and 2) objects of limited visibility (occluded, cut off from the image, divided, poorly illuminated, and blurred). Notably, the above standards may vary among datasets (i.e., there is no gold standard), but the goal is to maintain the consistency between annotators. For example, in our COCO guideline, crowding objects are annotated individually as much as possible, and it is similar in our Open Images guideline, but 15 is given as a recommended upper limit number of bounding boxes per category per image. Meanwhile, crowding objects are allowed to be grouped into a single bounding box in the original COCO and Open Images datasets.

Second, our guideline provides instructions related to each target category, including 1) a definition that describes the category, 2) some common positive and negative examples with multiple illustrations, and 3) additional category-specific instructions with multiple illustrations. Here we use *chair* (a target category in MS COCO) as an example to show our category-related guideline. More complete guidelines are provided in the supplementary material.

1. Definition: a seat which has a back, for single person, and is a piece furniture.
2. Positive examples, see Fig. 1
   (a) a chair that satisfies the above definition
   (b) single-seater sofa
   (c) folding chair
   (d) reclining chair
   (e) umpire chair (a special case)
3. Negative examples, see Fig. 2
   (a) non-furniture seat (e.g., vehicle seat)
   (b) stool
   (c) a sofa or chair for more than two adults
   (d) a bench seat for more than two adults
   (e) auditorium seats
   (f) wheelchair
   (g) painting or illustration
4. Additional instructions, see Fig. 3
   (a) Annotate a chair that is blurred or partially cut off from the image, if it is identifiable.
   (b) Annotate a chair shown on TV, reflected in a mirror, or through a fence.
   (c) If there are multiple chairs stacked together, annotate them separately.
   (d) Do not enclose accessories, like a cushion or a footrest, if they are separable from the main body.
   (e) Do not annotate if you feel difficult to identify.

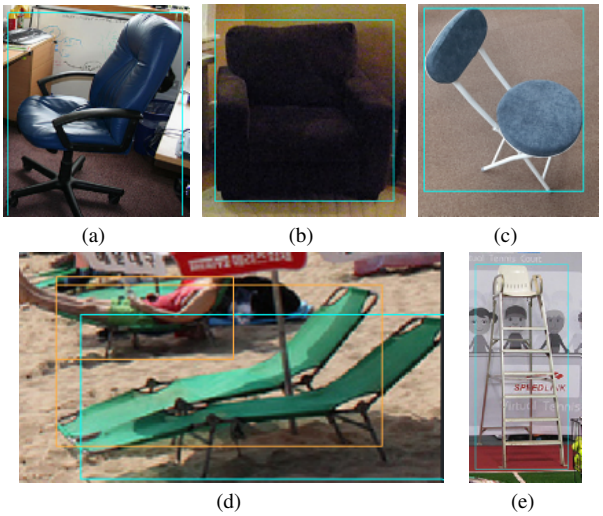


Figure 1. Positive examples of *chair*

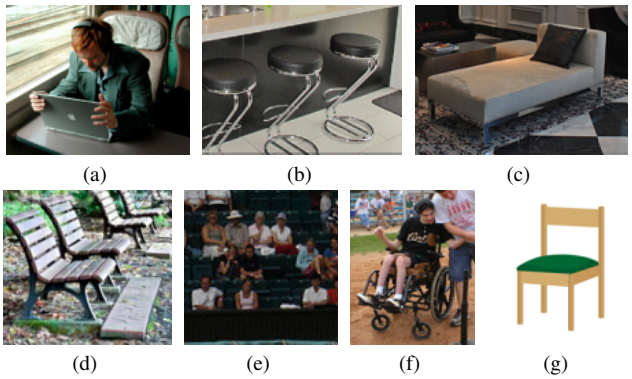


Figure 2. Negative examples of *chair*

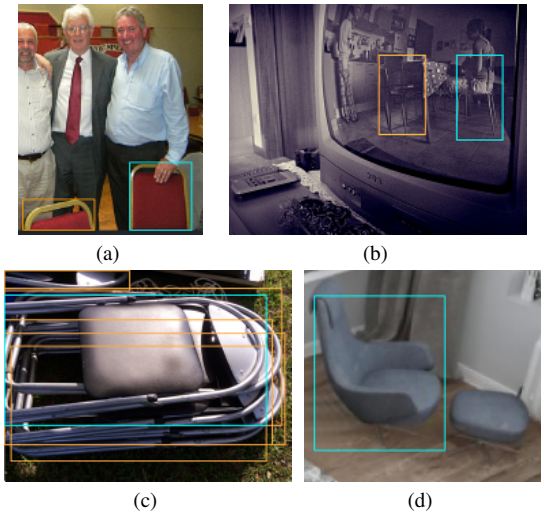


Figure 3. Additional instructions for *chair*

3.2. Annotation process

In this study, the annotation process was performed by Baobab Inc. A total of 135 annotators and 7 quality inspectors (QIs) were assigned to the MS COCO annotation, and 35 annotators and 5 QIs were assigned to the Google Open Images annotation. These workers were from Japan, Vietnam, Cambodia, and Thailand. Among all the annotators involved, 25% have developmental disability / autism. The working period (including annotation and verification) was 19 and 9 days for COCO and Open Images, respectively. The annotation time per image or per bounding box was not recorded because doing so might add to the stress of annotators. In this study, the annotation software used by annotators were Pose Annotator and V7 (v7labs.com), where the former is an in-house software of Baobab Inc., and the latter is commercial software.

To become a qualified annotator, the candidate needs to first participate in an image annotation training program. The program includes three steps, which are tutorials on how to use software to annotate with i) bounding boxes, ii) keypoints, and iii) polygons. Before annotators started working on a new dataset, they would perform a trial of annotating 10 image samples with relatively high difficulties chosen from that dataset. The results of the trial would be verified by QIs, and only those who successfully passed the trial could proceed to the main work.

During the annotation, annotators performed self-checking, and QIs also randomly checked their results and give them feedback. In general, QIs check 30%–60% of the annotation results. Any question raised by annotators will be answered through a Q&A sheet shared among all annotators, QIs, and project leaders.

3.3. Dataset statistics

From both MS COCO and Google Open Images, we chose five target object categories for this preliminary study: *car, chair, cup, person, and traffic light* from COCO and *building, car, dog, flower, and person* from Open Images. The reason for these choices is that they are larger in the quantity of bounding boxes and have less ambiguous definitions than other categories.

For COCO, we collected all images (from the *train2017* and *val2017* datasets) that contain one or more labels that match any of the five categories. The total number of such images was 80,067. For Open Images, because the dataset is too large (about 15 times larger than COCO), we randomly collected 5,000 images (1,000 images per category).

During our annotation, we found that some images in the original dataset did not contain any object that meets the standard of our annotation guidelines (e.g., an object originally annotated as a "chair" may rather be a "bench" according to our guideline). Consequently, these images were not given any bounding box in our annotation; they

	COCO		Open Images	
	original	ours	original	ours
Categories		5		
Images	80,067	78,145	5,000	4,520
Boxes	394,620	569,309	16,961	24,995
per image	4.9	7.3	3.4	5.5
small[*]	14.1%	15.9%	34.3%	48.0%
medium[*]	29.4%	34.7%	34.5%	30.6%
large[*]	56.4%	49.4%	31.2%	21.4%

[*] box area $< 32^2$ for *small*, $32^2 <$ area $< 96^2$ for *medium*, and $> 96^2$ for *large*, according to MS COCO evaluation standard

Table 1. General statistics of the COCO and Open Images datasets used in this study

were removed from our reannotated dataset. Although the number of images decreased, the number of bounding boxes largely increased for both datasets. This is because 1) our guidelines do not group crowding objects and 2) the original datasets have some missing bounding boxes. See Table 1 for a summary of the original and the reannotated datasets.

4. Experiments

4.1. Experiment design

This study aims to verify the proposal of the data-centric AI that improving the data itself can efficiently improve the machine learning performance. Specifically, we want to find whether a dataset annotated in high quality can benefit the machine learning process compared with its original counterpart. Because our annotation process involved well-trained annotators, detailed guidelines, and strict quality control, we expect that the label correctness and consistency in our reannotated dataset are much higher than that of the original one.

We evaluated the original and reannotated datasets with five models: Faster RCNN, SSD, YOLOv3, EfficientDet, and DETR. We only performed minimal tuning on epoch, batch size, and learning rate to ensure model convergence. Most hyperparameters remain untouched as the default settings in their GitHub repositories. The results are shown in mean average precision (mAP) and other related indexes.

In our experiments, we show the following four types of results to provide a comprehensive comparison: 1) trained and tested by the original dataset (which is the baseline), 2) trained by the reannotated and tested by the original, 3) trained by the original and tested by the reannotated, and 4) trained and tested by the reannotated dataset. For the sake of simplicity, in the tables and figures they are noted as *old/old*, *new/old*, *old/new*, and *new/new*, respectively.

4.2. Experiments on COCO

The original COCO dataset of five categories used in this study has 80,067 images, where 76,813 images from *train2017* constituted the training and validation sets (9:1 split), and 3,254 images from *val2017* constituted the test set. The reannotated dataset that has 78,145 images also employed the same split for training, validation, and testing as the original. To accelerate the training process, all models used ImageNet pretrained backbones.

Table 2 and Fig. 4 show the experimental results on COCO. The results indicate that our new annotation negatively affects mAP and most other indexes. Specifically, using the original training and test dataset achieved the highest mAP score. Reannotating the training set slightly decreased the score, whereas reannotating the test set decreased the score largely. It is likely that our reannotation made the detection problem more challenging in both learning and evaluation, as discussed later.

4.3. Experiments on Open Images

We split the 5,000 images of the original Open Images dataset into training, validation, and test sets in an 8:1:1 ratio. The reannotated dataset that has 4,520 images also took the same split. To accelerate the training process, the employed models use COCO pretrained model weights, which are publicly available.

Table 3 and Fig. 5 show the experimental results on Open Images. The results indicate that our new annotation positively affects mAP and most other indexes. Similar to COCO, compared to the baseline, reannotating only the training set decreased the mAP score. But differently, reannotating the test set increased the score, and reannotating both training and test set achieved the highest score. The results can be interpreted that our reannotation on the test set significantly benefited the evaluation process.

5. Discussion

This section discusses the quantitative results on COCO and Open Image. Some qualitative results with some detection examples are provided in the supplementary material.

5.1. Comparison of the original and reannotated Open Images

When we compared our annotation results with the original one, we found the original dataset had enormous incorrectness and ambiguities. Figures 6a–6c show some typical examples of the incorrect cases that occurred in the original Open Images dataset. Particularly, in Fig. 6a, our annotation shows there were 10 *persons* and 11 *dogs* in this image, whereas the original dataset annotated only 11 *dogs*, and in Figs. 6b and 6c, which show a cat and a fox, respectively

Method	train/test	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	car	chair	cup	person	traffic light
Faster RCNN [18]	old/old	**0.352**	**0.591**	**0.365**	**0.207**	**0.431**	**0.501**	**0.399**	**0.216**	**0.377**	**0.516**	0.251
	new/old	0.338	0.573	0.347	0.198	0.414	0.497	0.390	0.180	0.366	0.515	0.239
	old/new	0.309	0.514	0.321	0.163	0.414	**0.501**	0.335	0.203	0.363	0.376	0.268
	new/new	0.327	0.543	0.338	0.188	0.424	0.5	0.346	0.206	0.363	0.424	**0.295**
SSD [13]	old/old	**0.16**	**0.335**	**0.136**	**0.033**	**0.193**	0.356	**0.179**	0.091	**0.145**	**0.316**	0.068
	new/old	0.148	0.315	0.125	0.029	0.177	0.34	0.168	0.073	0.143	0.301	0.055
	old/new	0.132	0.279	0.113	0.024	0.184	**0.384**	0.149	**0.098**	0.134	0.208	**0.071**
	new/new	0.126	0.275	0.103	0.024	0.172	0.34	0.141	0.092	0.133	0.206	0.062
YOLOv3 [17]	old/old	**0.249**	**0.5**	**0.22**	**0.117**	**0.322**	**0.395**	**0.261**	**0.157**	**0.261**	**0.422**	0.146
	new/old	0.229	0.467	0.197	0.108	0.292	0.371	0.258	0.124	0.234	0.409	0.117
	old/new	0.208	0.425	0.186	0.081	0.305	0.386	0.217	0.137	0.239	0.290	**0.157**
	new/new	0.206	0.431	0.175	0.094	0.29	0.371	0.214	0.142	0.213	0.318	0.141
EfficientDet [23]	old/old	**0.14**	**0.266**	**0.132**	**0.043**	**0.186**	0.261	0.148	0.046	**0.132**	0.325	0.051
	new/old	0.138	0.26	0.13	0.04	0.18	**0.267**	0.128	**0.056**	0.102	0.23	0.054
	old/new	0.114	0.219	0.102	0.031	0.168	0.261	0.122	0.052	0.121	0.219	0.053
	new/new	0.114	0.219	0.104	0.032	0.171	0.265	**0.151**	0.048	0.109	**0.327**	**0.056**

* All values are related to mAP except for AP_{50} and AP_{75}. The results of DETR [5] were very low (mAP<0.1) thus not included.

Table 2. The experimental results on COCO

Method	train/test	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	building	car	dog	flower	person
Faster RCNN [18]	old/old	0.387	0.548	0.429	0.062	0.099	0.443	**0.311**	0.542	0.669	0.238	0.176
	new/old	0.349	0.494	0.387	**0.091**	0.086	0.393	0.219	0.507	0.666	0.223	0.131
	old/new	0.426	0.625	0.465	0.062	**0.187**	0.501	0.163	0.604	0.689	0.276	0.398
	new/new	**0.465**	**0.664**	**0.501**	0.072	0.179	**0.54**	0.226	**0.665**	**0.692**	**0.282**	**0.459**
SSD [13]	old/old	0.366	0.527	0.404	**0.005**	0.047	0.413	**0.306**	0.513	0.669	**0.242**	0.101
	new/old	0.332	0.485	0.371	0.002	0.045	0.372	0.225	0.483	0.666	0.209	0.078
	old/new	0.390	0.599	0.427	0.001	0.081	0.461	0.139	0.575	0.678	**0.242**	0.313
	new/new	**0.403**	**0.618**	**0.442**	0.002	**0.085**	**0.478**	0.172	**0.602**	**0.684**	0.228	**0.328**
YOLOv3 [17]	old/old	0.344	0.53	0.385	**0.093**	0.079	0.39	**0.251**	0.502	0.609	0.209	0.15
	new/old	0.312	0.484	0.351	0.09	0.076	0.351	0.154	0.491	0.588	0.191	0.139
	old/new	0.390	0.606	0.435	0.058	**0.21**	0.456	0.110	0.567	**0.652**	**0.285**	0.342
	new/new	**0.417**	**0.64**	**0.46**	0.074	0.194	**0.482**	0.172	**0.627**	0.631	0.269	**0.402**
EfficientDet [23]	old/old	0.413	0.553	0.452	0.057	0.082	0.464	**0.334**	0.533	0.691	0.286	0.125
	new/old	0.381	0.503	0.403	0.015	0.082	0.43	0.185	0.533	0.757	**0.323**	0.107
	old/new	0.439	0.608	0.478	**0.091**	0.147	0.509	0.145	0.603	0.754	0.274	0.416
	new/new	**0.465**	**0.618**	**0.503**	0.025	**0.162**	**0.534**	0.141	**0.667**	**0.787**	0.271	**0.461**
DETR [5]	old/old	0.379	0.515	0.397	**0.038**	0.045	0.436	**0.367**	0.463	0.751	0.205	0.108
	new/old	0.335	0.45	0.356	0.012	0.049	0.385	0.205	0.428	0.769	**0.262**	0.101
	old/new	0.394	0.592	0.407	0.012	**0.142**	0.479	0.133	0.525	0.772	0.167	0.352
	new/new	**0.429**	**0.629**	**0.44**	0.007	0.135	**0.514**	0.152	**0.574**	**0.799**	0.220	**0.399**

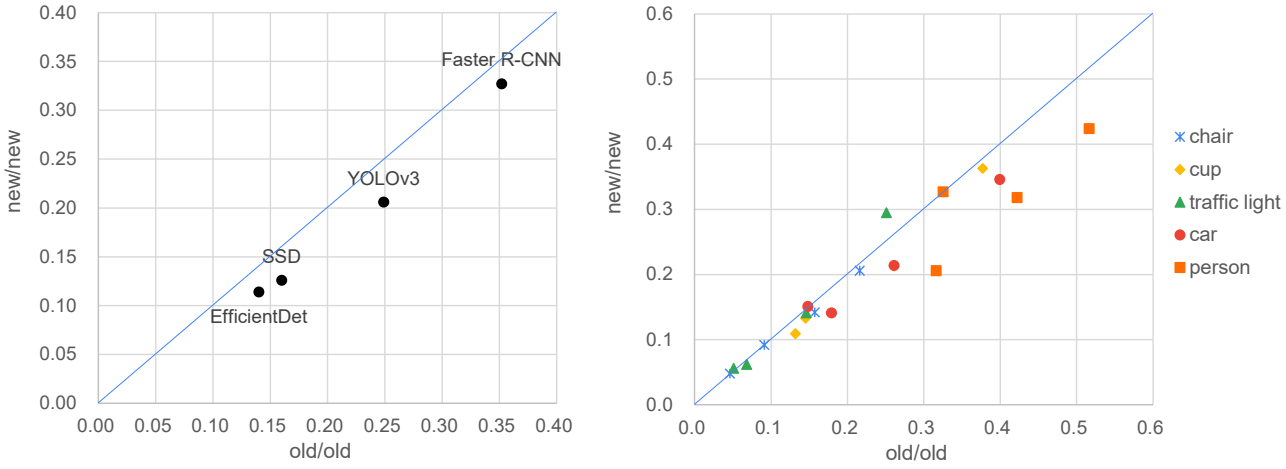
* All values are related to mAP except for AP_{50} and AP_{75}.

Table 3. The experimental results on Open Images

(and thus not annotated in our case), the original dataset incorrectly annotated both as *dogs*.

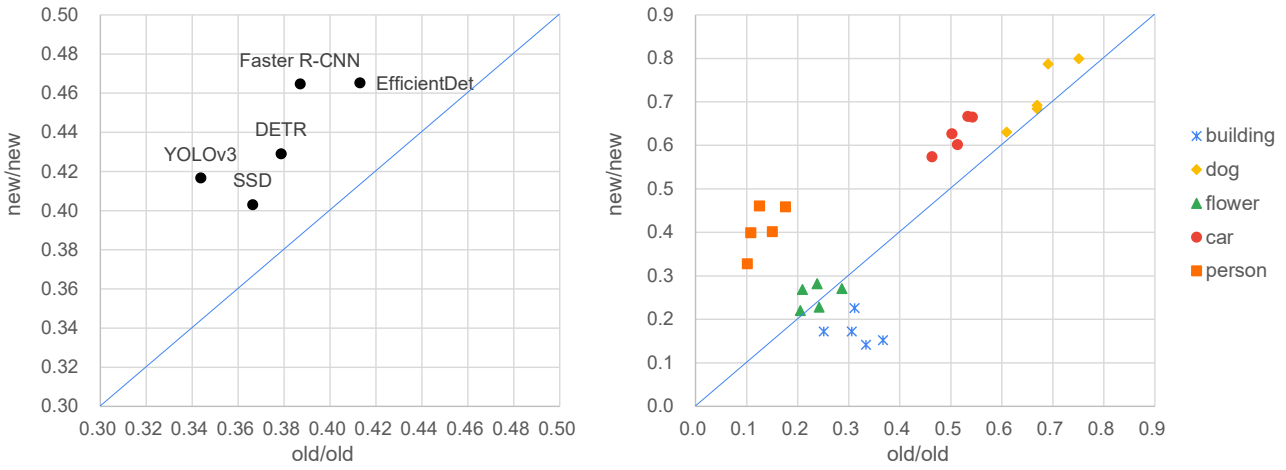
Figures 6d–6h show some typical ambiguous cases that occurred in the original Open Images dataset. Particularly, Fig. 6d shows a part of *dog*, Fig. 6e shows some blurred and unidentifiable *flowers*, Fig. 6f shows non-real *flowers*, Fig. 6g shows parts of *person*, and Fig. 6h shows non-real *persons*. All these examples were annotated in the original dataset but not annotated in our case.

As shown in Fig. 5, the overall performance of our rean-



(a) COCO mAP (b) COCO category-wise mAP

Figure 4. The experimental results on COCO



(a) Open Images mAP (b) Open Images category-wise mAP

Figure 5. The experimental results on Open Images

notated Open Images dataset improved, and the category-wise results were organized in an intuitively clear pattern. One may think that our new annotation removed ambiguity from the original dataset, which may apparently result in a simpler task. However, our goal is not to simplify the task (i.e., to boost the model performance on purpose) but purely to improve the label correctness and consistency. Consequently, the task difficulty may either increase or decrease.

5.2. Comparison of our Open Images with our COCO

In fact, our reannotated COCO may just be the example where the task difficulty increased, where the overall performance decreased. By comparing Figs. 4b and 5b, it is especially interesting to find that the *person* category decreased the most in our COCO, whereas it improved the most in our Open Images. About the reason that the same category performs oppositely on the two datasets, we believe that it is mainly due to the differences between our annotation guidelines, which are summarized below.

1. One should annotate as many targets as possible in our COCO guideline, whereas 15 bounding boxes per category per image is a recommended upper limit in our Open Images guideline. See Fig. 7a.
2. Human body parts should be annotated in our COCO guideline, whereas only the parts larger than 20% of the whole body should be annotated in our Open Images guideline. See Fig. 7b.
3. A person that is a reflection in a glass or a photo in a newspaper/poster should be annotated in our COCO guideline, whereas it should not be annotated in our Open Images guideline. See Fig. 7c.

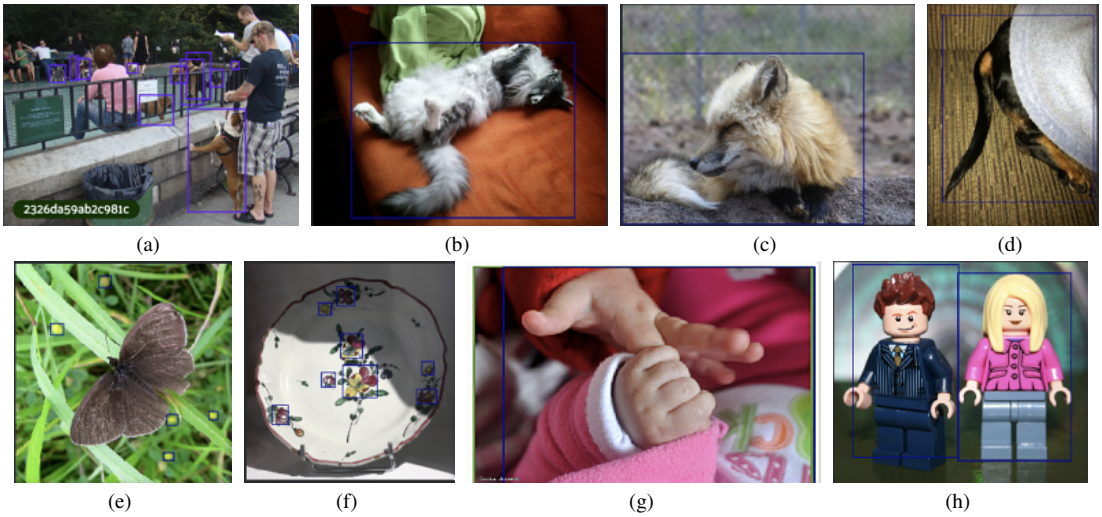


Figure 6. Some incorrect or ambiguous annotation examples in the original Open Images dataset

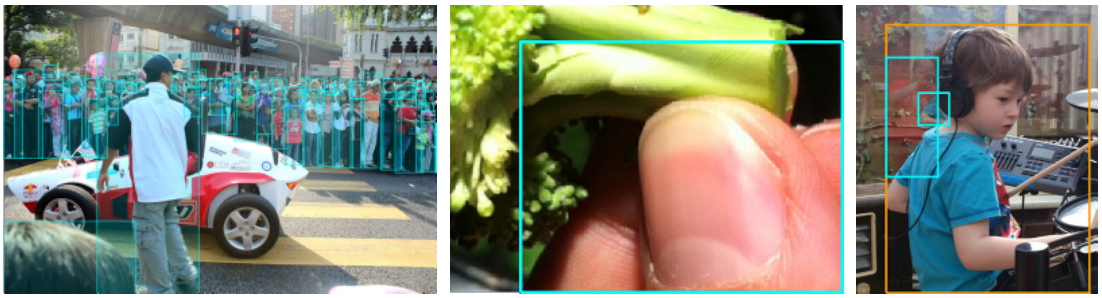


Figure 7. The reannotation guideline differences between COCO and Open Images related to the *person* category: (a) as many targets as possible are annotated in COCO, (b) human-body parts (smaller than 20% of the body) are annotated in COCO, (c) human reflections (the blue bounding boxes) or photos are annotated in COCO

These differences significantly increased the number of small objects, human-body parts, and persons in reflections or photos being annotated. Consequently, we can reasonably infer that the task difficulty of *person* category increased considerably in the reannotated COCO dataset. In addition, we can infer that, for other categories of COCO, it is also the similar guideline-related reason that caused the decrease in the overall performance.

6. Conclusion

In this preliminary study, we performed high-quality reannotation on 80k and 5k images from MS COCO and Google Open Images datasets, respectively, and verified how these annotations affect the performance of object detection tasks using five models. Our experimental results showed an increase in mAP on the reannotated Open Images, but a decrease in mAP on the reannotated COCO. Although, in data-centric AI, improving label correctness and consistency is an efficient means to improve machine learning task performance, our results indicated a remarkable fact that the process of improving labels may increase or decrease the task difficulty. Consequently, the final performance is unnecessarily improved.

Regarding our future works, first, we need to propose criteria to quantitatively measure the change on label correctness and consistency and possibly the change on task difficulty brought by reannotation. Based on that, we would like to re-examine our annotation results on COCO and Open Images. Finally, we plan to provide a fully reannotated COCO or Open Images dataset that can become a new benchmark.

Acknowledgement

The authors thank Shiau Chou Jen and Kazuhiro Koide for their contributions on programming and experimenting. This work was supported by JST-Mirai Program Grant Number JPMJMI21G2, Japan.

References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433, 2015. 2

[2] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. ObjectNet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019. 2

[3] Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with ImageNet? *arXiv preprint arXiv:2006.07159*, 2020. 1, 2

[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

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 2, 6

[6] DeepLearningAI. A chat with Andrew on MLOps: From Model-centric to Data-centric AI, 2021. https://www.youtube.com/watch?v=06-AZXmwHjo. 1

[7] Mark Everingham, SM Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015. 3

[8] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014. 2

[9] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in VQA matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913, 2017. 2

[10] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollar. Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9404–9413, 2019. 3

[11] 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. *International Journal of Computer Vision*, 128(7):1956–1981, 2020. 2, 3

[12] 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 *European conference on computer vision*, pages 740–755. Springer, 2014. 2, 3

[13] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 2, 6

[14] Curtis G Northcutt, Anish Athalye, and Jonas Mueller. Pervasive label errors in test sets destabilize machine learning benchmarks. In *Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. 2

[15] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In *International Conference on Machine Learning*, pages 5389–5400, 2019. 2

[16] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016. 2

[17] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 2, 6

[18] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 2, 6

[19] 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. *International journal of computer vision*, 115(3):211–252, 2015. 1

[20] Vaishaal Shankar, Rebecca Roelofs, Horia Mania, Alex Fang, Benjamin Recht, and Ludwig Schmidt. Evaluating machine accuracy on ImageNet. In *International Conference on Machine Learning*, pages 8634–8644. PMLR, 2020. 1, 2

[21] Hao Su, Jia Deng, and Li Fei-Fei. Crowdsourcing annotations for visual object detection. In *Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012. 1

[22] Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114, 2019. 2

[23] Mingxing Tan, Ruoming Pang, and Quoc V Le. EfficientDet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020. 2, 6

[24] Antonio Torralba and Alexei A. Efros. Unbiased look at dataset bias. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1521–1528, 2011. 2

[25] Antonio Torralba, Rob Fergus, and William T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1958–1970, 2008. 2

[26] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. From ImageNet to image classification: Contextualizing progress on benchmarks. In *International Conference on Machine Learning*, pages 9625–9635, 2020. 2

[27] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for ob-

ject recognition. *International journal of computer vision*, 104(2):154–171, 2013. 2

[28] Grant Van Horn, Steve Branson, Ryan Farrell, Scott Haber, Jessie Barry, Panos Ipeirotis, Pietro Perona, and Serge Belongie. Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 595–604, 2015. 2

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2

[30] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 2

[31] Sangdoo Yun, Seong Joon Oh, Byeongho Heo, Dongyoon Han, Junsuk Choe, and Sanghyuk Chun. Re-labeling ImageNet: from single to multi-labels, from global to localized labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2340–2350, 2021. 1, 2

[32] Jeffrey O Zhang, Alexander Sax, Amir Zamir, Leonidas Guibas, and Jitendra Malik. Side-tuning: a baseline for network adaptation via additive side networks. In *European Conference on Computer Vision*, pages 698–714. Springer, 2020. 2