Language-conditioned Detection Transformer

Supplementary Material

7. Comparing to OV-DETR [66].

Conditioning mechanism. Figure 5 illustrates the difference in conditioning mechanism and multi-class inference between DECOLA and OV-DETR. During training Phase 1, DECOLA learns to use text embedding of each present object class in order to locate proposals. DECOLA learns **dense language-vision alignment** by modeling the objecness function as the similarity score between text embedding and proposal features defined as equation 3 in the Section 4 of the main paper. DECOLA transforms equal number of proposals into query embedding to sufficiently cover all classes. On the other hand, OV-DETR trains with the same DETR object queries and add CLIP features of randomly sampled object classes. This difference results in a significant improvement in *conditioned* AP and AR (**+9.1** c-AP$_{@20}$, **+16.5** c-AR), as shown in Figure 4a and 4c of the main paper.

Multi-class detection. DECOLA finetunes for multi-class object detection during Phase 2 whereas OV-DETR maintains the original conditioning mechanism for finetuning. Finetuning with multi-class detection objective is critical for the final detection task: Detector needs to **calibrate the multi-class scores** over the dataset-level vocabulary to maximize mAP. OV-DETR trains with randomly sampled set of classes every iteration, which makes it unable to properly rank objects over all classes. This leads to a severe degradation in frequent classes as shown in Table 1 of the main paper. Moreover, the conditioning mechanism of OV-DETR requires *splitting* the text vocabulary over multiple chunks. For LVIS dataset, OV-DETR needs about 40 forward passes for every image at inference, leading to a substantial difference in speed at run-time (0.07 vs 6.4 sec / img) as shown in Table 6 of the main paper. The final models, DECOLA Phase 2 and OV-DETR† under identical training and architectural settings, exhibit large difference of **5.8** AP$_{\text{novel}}$ and **7.7** mAP, as shown in Table 1 of the main paper.

Training setup. Both models are trained on LVIS-base for $4\times$ (DECOLA Phase 1 and OV-DETR). OV-DETR undergoes extra $4\times$ with the original self-training using CLIP labeling [66]. This model is the same as the original OV-DETR reported in the original paper. We further finetune DECOLA and OV-DETR on ImageNet-21K for $4\times$ for fair comparison, which result DECOLA Phase 2 and OV-DETR†.

8. Experimental Details

Training configuration. We closely follow [74] to train DECOLA as well as *baseline* for both Deformable DETR and CenterNet2 results. Table 7a and 7b highlight important hyper-parameters in all experiments with Deformable DETR. For experiments with CenterNet2, we follow the same training and model configuration as Detic [74]. For all experiments, we used 8 V100 GPUs with 32G memory. All models are trained on `float16` using Automatic Mixed Precision from PyTorch [46]. With this computing environment, training DECOLA for Deformable DETR with ResNet-50 backbone takes about 50 hours and the baseline takes about 45 hours for $4\times$ training schedule. For ImageNet-21K pre-trained ResNet-50, we used the model from Ridnik *et al.* [50] consistent with [74]. Our codebase uses Detectron2 [64] based on PyTorch [46]. For *direct zero-shot transfer to LVIS* experiments, we use Swin-T and L [39] pretrained on ImageNet-21K. For both methods, we train Phase 1 on Object365 same number of iterations as GLIP [34]. We finetune Phase 2 on the entire ImageNet-21K for Swin-T, and ImageNet-21K and OpenImages [32] for the same number of iterations as Phase 1. Please note that the model may continue to improve as training longer. Swin-L model is trained with 2 nodes of 8 V100 machines, with 32 images per global batch. All our experiments are conducted under academic-scale compute and open-sourced datasets.

9. Additional Experimental Results

***Conditioned* AP.** Table 8 and 9 compare *conditioned* mAP and AP$_{\text{novel}}$ of baseline and DECOLA Phase 1. We show AP with different per image detection limit, reported with $@k$. *Conditioned* AP is defined in Section 5.1. Results at low detection limit follows more closely to the labeling quality; pseudo-labels are sampled based on the confidence score and typically only save the top-1 prediction. DECOLA consistently improve baseline not only for novel classes but for overall. This difference is the core reason for DECOLA's scalability by self-training.

Box-efficient detector. In this section, we highlight an interesting property of DECOLA. Object detectors for large-vocabulary dataset often tend to **over-shoot** predictions with a high number of boxes in order to increase recall for rare object classes. This behavior may be undesirable since lots of spammed boxes makes it difficult to interpret for downstream tasks. Therefore, Table 11 and 12 report c-AP of baseline and DECOLA Phase 1 *with a limited number of query (prediction) per class*. $n = 1$ means the detector only gets to predict a single box for each class present in image. Please recall that c-AP provides a set of present classes during inference. We show that DECOLA show highly accurate predictions with low per-image detection limit.

Impact of different pre-training. Table 13 shows how backbone pretraining impact the final result on DECOLA as

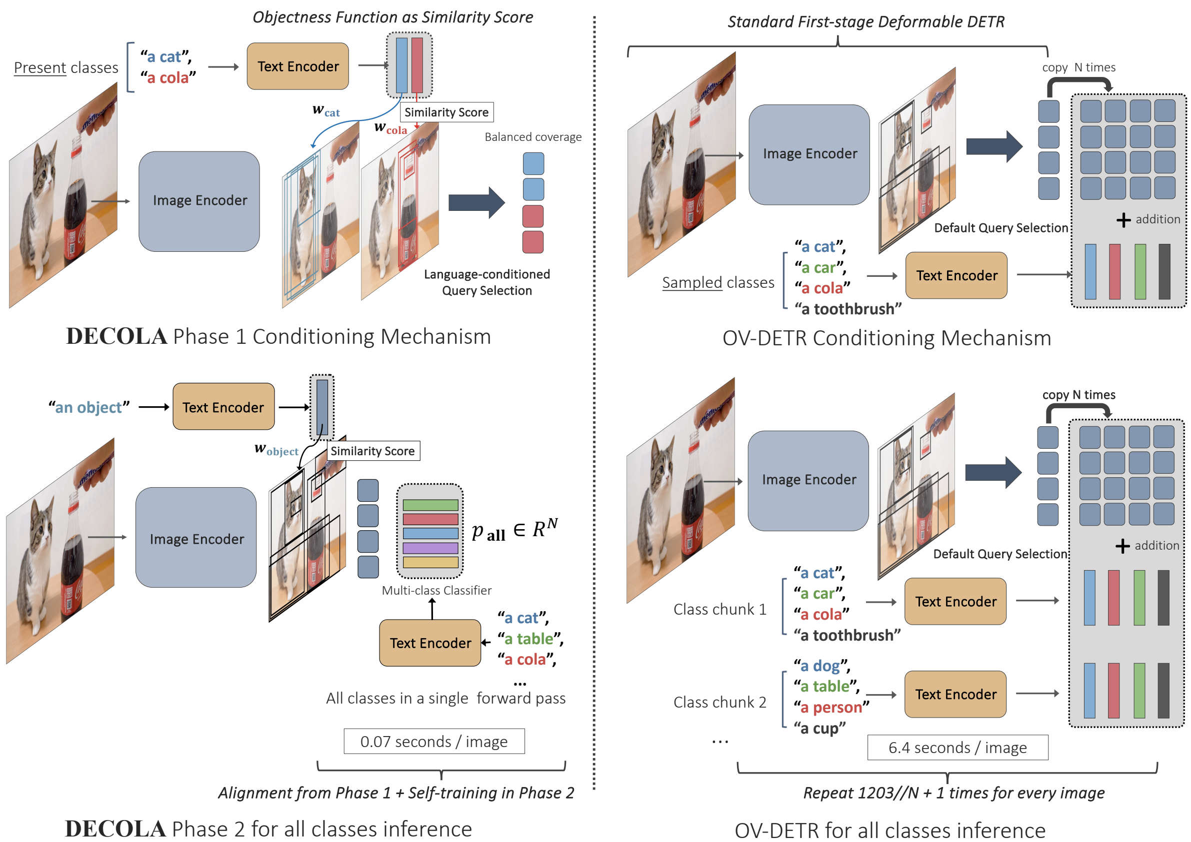

Figure 5. **Difference in conditioning mechanism and multi-class inference between DECOLA and OV-DETR.** DECOLA produces different *proposals* for each present object class in image by modeling the objectness function with the similarity score between text embedding and proposal feature. OV-DETR copies object queries from the first-stage DETR and add CLIP features. Bottom row illustrates how DECOLA and OV-DETR performs multi-class detection.

well as the *baseline*. In Table 13, ImageNet-21K worked the best overall, but surprisingly there was no substantial difference in $\text{AP}^{\text{box}}_{\text{novel}}$ from Deformable DETR framework, contrary to the finding in [74] with CenterNet2 detector. Here all models are trained on LVIS-base. Table 13b shows that pretraining on Object365 substantially improve LVIS result. Since both Object365 and LVIS are large-scale detection datasets of natural objects, we expect some degree of semantic overlap between the datasets.

Co-training. DECOLA trains language-conditioning and multi-class prediction in two separate phases. Here, we explore if we can co-train both conditioning and multi-class prediction. Specifically, we set a probability p to train a detector by language-condition (conditioning the first-stage with class name) and multi-class (conditioning the first-stage with "`an object`") and using multi-class classifier with text embedding same as *baseline*. Table 10e reports the conditioned AP after training $4\times$ on LVIS-base with different p. We observe that c-AP is maximized with $p = 0.0$, but mAP can match with the standard detection training with $p = 0.5$. Table 10f extends co-training to finetuning for Phase 2 on weakly labeled data. Here $p_1 \to p_2$ denotes the sampling probability of "`a object`" conditioning for LVIS-base (p_1) and LVIS-base and ImageNet-21K (p_2). We confirm that the quality of pseudo-labels is the most important for finetuning with weakly-labeled data.

Other ablations. In Table 10a, we show that DECOLA label improves using box regression loss. Detic [74] only trains for classification loss since max-size loss samples pseudo-label that does not localize object accurately. This improvement shows that DECOLA label provides a significant supervisory signal for localization as well as classification. In DECOLA Phase 1, each query is conditioned to an object class and predicts a *single* score after decoding layers ("single"). Table 10c explores different second-stage formulation. After the first stage, we ignore the conditioned classes and predict multi-class scores after decoding layers, denoted as "multi".

config	baseline training	baseline + self-train
shared configuration		
optimizer	AdamW [40]	AdamW [40]
optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$	$\beta_1, \beta_2 = 0.9, 0.999$
weight decay	0.0001	0.0001
total iterations	360000	360000
base learning rate	0.0002	0.0002
learning rate schedule	step decay	step decay
learning rate decay factor	0.1	0.1
learning rate decay step	300000	300000
gradient clip value	0.01	0.01
gradient clip norm	2.0	2.0
different configuration		
batch size	16	(16, 64)
dataset ratio	n/a	1 : 4 [74]
image min-size range	(480, 800)	((480, 800), (240, 400)) [74]
image max-size	1333	(1333, 667) [74]
input augmentation	DETR-style [3]	resize shortest edge [74]
input sampling	repeated factor sampling [20]	(repeated factor [20], random)

(a) Training configuration. The values inside parentheses are for LVIS and ImageNet-21K, respectively.

config	baseline	DECOLA Phase 1	DECOLA Phase 2
shared configuration			
`cls` weight	2.0	2.0	2.0
`giou` weight	2.0	2.0	2.0
`l1` weight	5.0	5.0	5.0
two-stage	`True`	`True`	`True`
box refinement	`True`	`True`	`True`
feed-forward dim.	1024	1024	1024
look-forward-twice	`True`	`True`	`True`
drop-out rate	0.0	0.0	0.0
different configuration			
number of queries	300	300 per class	300
classification loss type	federated loss [73]	biniary cross-entropy	federated loss [73]
1st-stage classifier type	learnable	"`a [class name].`"	"`an object.`"
1st-stage classifier norm	n/a	L2	L2
1st-stage classifier temp.	n/a	50	50
1st-stage top-k per class†	n/a	10000	n/a
2nd-stage classifier type	"`a [class name].`"	"`a [class name].`"	"`a [class name].`"
2nd-stage classifier norm	L2	L2	L2
2nd-stage classifier temp.	50	50	50
classifier # classes‡	1203	1	1203
classifier bias init. value	$-\log(0.99/0.01)$	$-\log(0.99/0.01)$	$-\log(0.99/0.01)$

(b) Model configuration.

Table 7. **Configurations.** Training and model details for experiments with Deformable DETR. † is the top-k only for Hungarian matching and loss computation to reduce computation, as explained in the main paper. ‡ is for LVIS experiments. Here DECOLA for language-condition training has # classes as 1, since the second-stage with language-conditioned query is binary classification as opposed to multi-class classification in baseline and open-vocabulary detection.

10. Qualitative Results

We show more visualization. Figure 6 and 7 show randomly sampled images and the pseudo-labels of DECOLA and baseline. Images are from the ImageNet-21K from *unseen* categories, which none of the models are trained on. Boxes are the most confident prediction from DECOLA and *baseline* and maximum size box ([74]). **Green**: the most confident prediction (max-score) DECOLA trained on LVIS-base. **Red**: the most confident prediction (max-score) *baseline* trained on LVIS-base. **Purple**: the largest box prediction (max-size, Detic loss [74]) *baseline* trained on LVIS-base. All models use a Deformable DETR detector with a ResNet-50 backbone. We show randomly sampled images.

model	data	c-AP$^{box}_{novel}$@10	c-AP$^{box}_{novel}$@20	c-AP$^{box}_{novel}$@50	c-AP$^{box}_{novel}$@100	c-AP$^{box}_{novel}$@300
ResNet-50						
baseline	LVIS-base	6.0	11.3	19.2	26.8	31.9
DECOLA Phase 1	LVIS-base	19.4 (+13.4)	28.5 (+17.2)	34.1 (+14.9)	38.7 (+11.9)	40.0 (+8.1)
Swin-B						
baseline	LVIS-base	7.4	16.1	27.5	33.1	41.9
DECOLA Phase 1	LVIS-base	21.9 (+14.5)	32.0 (+15.9)	40.0 (+12.5)	44.0 (+6.9)	47.7 (+5.8)

(a) c-AP of unseen categories at different k.

model	data	c-AP$^{box}_{rare}$@10	c-AP$^{box}_{rare}$@20	c-AP$^{box}_{rare}$@50	c-AP$^{box}_{rare}$@100	c-AP$^{box}_{rare}$@300
ResNet-50						
baseline	LVIS	21.3	29.4	36.9	41.1	44.6
DECOLA Phase 1	LVIS	26.6 (+5.3)	39.1 (+9.7)	45.2 (+8.3)	47.1 (+6.0)	48.8 (+4.2)
Swin-B						
baseline	LVIS	30.1	38.2	45.5	49.3	53.2
DECOLA Phase 1	LVIS	33.5 (+3.4)	43.9 (+5.7)	51.4 (+5.9)	53.8 (+4.5)	55.8 (+2.6)

(b) c-AP of rare categories at different k.

Table 8. ***Conditioned* AP$_{\text{rare/novel}}$** result of DECOLA Phase 1 and baseline pre-trained on LVIS-base (*top*) and LVIS (*bottom*). *Conditioned* AP measures detection performance when the set of object categories present in each image is given. baseline adapts its classification layer to the classes and DECOLA conditions itself to the classes, as described in Section 4 of the main paper.

model	data	c-mAP^{box}@10	c-mAP^{box}@20	c-mAP^{box}@50	c-mAP^{box}@100	c-mAP^{box}@300
ResNet-50						
baseline	LVIS-base	24.4	29.8	35.0	37.9	40.2
DECOLA Phase 1	LVIS-base	30.0 (+5.6)	36.8 (+7.0)	41.9 (+6.9)	44.2 (+6.3)	45.6 (+5.4)
Swin-B						
baseline	LVIS-base	29.6	36.9	43.4	46.0	48.8
DECOLA Phase 1	LVIS-base	33.5 (+3.9)	41.3 (+4.4)	47.4 (+4.0)	49.7 (+3.7)	51.5 (+2.7)

(a) c-mAP of all categories at different k.

model	data	c-mAP^{box}@10	c-mAP^{box}@20	c-mAP^{box}@50	c-mAP^{box}@100	c-mAP^{box}@300
ResNet-50						
baseline	LVIS	27.3	33.4	38.8	41.2	43.1
DECOLA Phase 1	LVIS	31.1 (+3.8)	38.5 (+5.1)	43.7 (+4.9)	45.6 (+4.4)	47.1 (+4.0)
Swin-B						
baseline	LVIS	33.3	40.4	46.2	48.6	50.5
DECOLA Phase 1	LVIS	35.7 (+2.4)	43.6 (+3.2)	49.4 (+3.2)	51.6 (+3.0)	53.2 (+2.7)

(b) c-mAP of all categories at different k.

Table 9. ***Conditioned* mAP** result of DECOLA in phase 1 and baseline pre-trained on LVIS-base (*top*) and LVIS (*bottom*). *Conditioned* mAP measures detection performance when the set of object categories present in each image is known. baseline adapts its classification layer to the classes and DECOLA condition itself to the classes, as described in Section 4 of the main paper.

model	reg. loss	AP$^{box}_{novel}$	mAP^{box}
DECOLA label		27.6	36.6
DECOLA label	✓	29.5	37.7

(a) **Box regression loss** for weak data.

model	AP$^{box}_{novel}$	mAP^{box}
baseline + DECOLA label	25.1	36.9
DECOLA Phase 2	27.6	38.3

(b) **DECOLA Phase 2 vs. baseline + DECOLA label**.

type	c-AP$^{box}_{novel}$	c-mAP^{box}
multi	14.2	20.7
single	28.5	40.0

(c) **Second-stage type** for Phase 1 ($k = 20$).

type	c-AP$^{box}_{novel}$	c-mAP^{box}
base	20.9	30.4
text	21.2	31.6
image	22.3	35.1

(d) **Query types** ($k = 20$).

p	c-AP$^{box}_{novel}$	mAP^{box}
1.0	10.7	30.2
0.5	19.1	30.2
0.0	22.3	n/a

(e) **Co-training**: Phase 1 ($k = 20$).

p	AP$^{box}_{novel}$	AP$^{box}_{c}$	AP$^{box}_{f}$	mAP^{box}
$0.5 \rightarrow 0.5$	21.0	31.9	37.0	32.0
$0.5 \rightarrow 1.0$	20.8	33.2	37.8	32.9
$0.0 \rightarrow 1.0$	23.8	34.4	38.3	34.1

(f) **Co-training**: Phase 2.

Table 10. **Additional results.** open-vocabulary LVIS results for various ablation study. We used Deformable DETR with ResNet-50 for all models here. For all results with c-AP, we use Phase 1. k represents the detection limit per image. Note that the bottom row tables (d), (e), (f) are trained with DECOLA and *baseline* trained using ResNet-50 pretrained with ImageNet-1K.

model	data	$n=1$	$n=2$	$n=5$	$n=10$	$n=20$
ResNet-50						
baseline	LVIS-base	14.7	22.4	27.6	30.9	32.2
DECOLA Phase 1	LVIS-base	25.2 (+10.5)	31.4 (+9.0)	36.0 (+8.4)	37.9 (+7.0)	39.9 (+7.7)
Swin-B						
baseline	LVIS-base	17.8	26.0	33.7	37.6	40.9
DECOLA Phase 1	LVIS-base	31.0 (+13.2)	37.3 (+11.3)	44.1 (+10.4)	46.2 (+8.6)	47.2 (+6.3)

(a) *Conditioned* AP of unseen categories with different number of queries *per class*.

model	data	$n=1$	$n=2$	$n=5$	$n=10$	$n=20$
ResNet-50						
baseline	LVIS	17.8	24.8	33.0	38.7	42.3
DECOLA Phase 1	LVIS	29.7 (+11.9)	36.7 (+11.9)	41.8 (+8.8)	45.9 (+7.2)	48.3 (+6.0)
Swin-B						
baseline	LVIS	20.7	29.9	42.4	48.4	51.6
DECOLA Phase 1	LVIS	34.5 (+13.8)	42.3 (+12.4)	49.0 (+6.6)	50.8 (+2.4)	52.7 (+1.1)

(b) *Conditioned* AP of rare categories with different number of queries *per class*.

Table 11. **DECOLA is more box-efficient (c-AP$_{\text{rare/novel}}$).** We measure *conditioned* AP of rare/unseen classes (c-AP$_{\text{rare/novel}}$) of DECOLA Phase 1 and baseline pre-trained on LVIS-base (*top*) and LVIS (*bottom*) with different *per-class* number of query. DECOLA uses $n = |Q_y|$ language-conditioned queries for each class in image. Baseline uses $n \cdot |C_x|$ object queries where C_x is the set of object classes in image x. Two models use the same total number of object queries.

model	data	$n=1$	$n=2$	$n=5$	$n=10$	$n=20$
ResNet-50						
baseline	LVIS-base	14.8	22.2	29.6	33.6	35.2
DECOLA Phase 1	LVIS-base	24.5 (+9.7)	31.5 (+9.3)	37.9 (+8.3)	41.1 (+7.5)	43.4 (+8.2)
Swin-B						
baseline	LVIS-base	18.0	26.9	36.6	41.0	43.4
DECOLA Phase 1	LVIS-base	28.0 (+10.0)	35.2 (+8.3)	42.7 (+6.1)	46.5 (+5.5)	48.9 (+5.5)

(a) *Conditioned* mAP of all categories with different number of queries *per class*.

model	data	$n=1$	$n=2$	$n=5$	$n=10$	$n=20$
ResNet-50						
baseline	LVIS	15.0	22.6	31.1	35.7	38.3
DECOLA Phase 1	LVIS	25.5 (+10.5)	32.2 (+9.6)	38.9 (+7.8)	42.6 (+6.9)	44.8 (+6.5)
Swin-B						
baseline	LVIS	18.3	27.2	37.6	42.4	44.4
DECOLA Phase 1	LVIS	29.3 (+11.0)	36.8 (+9.6)	44.0 (+6.4)	47.7 (+5.3)	50.1 (+5.7)

(b) *Conditioned* mAP of all categories with different number of queries *per class*.

Table 12. **DECOLA is more box-efficient (c-mAP).** We measure *Conditioned* mAP of DECOLA Phase 1 and baseline pre-trained on LVIS-base (*top*) and LVIS (*bottom*) with different *per-class* number of query. DECOLA uses $n = |Q_y|$ language-conditioned queries for each class in image. Baseline uses $n \cdot |C_x|$ object queries where C_x is the set of object classes in image x. Two models use the same total number of object queries. *Conditioned* mAP measures with $k = 300$ per-image detection limit.

method	pretrain	AP$^{\text{box}}_{\text{novel}}$	AP$^{\text{box}}_{\text{c}}$	AP$^{\text{box}}_{\text{f}}$	mAP$^{\text{box}}$
baseline	IN-1K	10.2	30.9	38.0	30.1
	RegionCLIP [72]	9.1	32.6	39.9	31.4
	IN-21K	9.4 (-0.8)	33.8 (+2.9)	40.4 (+2.4)	32.2 (+2.1)
baseline + self-train	IN-1K	19.2	31.7	37.1	31.7
	IN-21K	23.2 (+4.0)	36.5 (+4.8)	41.6 (+4.5)	36.2 (+4.5)
DECOLA Phase 2	IN-1K	23.8	34.4	38.3	34.1
	IN-21K	27.6 (+3.8)	38.3 (+3.9)	42.9 (+4.6)	38.3 (+4.2)

(a) Impact of the backbone pretrain.

method	O365	c-AP$^{\text{box}}_{\text{rare}}$@300	c-AP$^{\text{box}}_{\text{c}}$@300	c-AP$^{\text{box}}_{\text{f}}$@300	c-mAP$^{\text{box}}$@300
DECOLA Phase 1		54.6	52.7	52.3	52.9
	✓	62.0 (+7.4)	62.0 (+9.3)	61.6 (+9.3)	61.8 (+8.9)

(b) Impact of Object365 [53] pretrain on DECOLA Phase 1.

Table 13. **Impact of different pretraining.** Evaluated on open-vocabulary LVIS with ResNet-50 Deformable DETR (top) and large-vocabulary LVIS with Swinl-L Deformable DETR (bottom). We explore the impact of different pretraining on the final mAP (top) and conditioned AP (bottom).

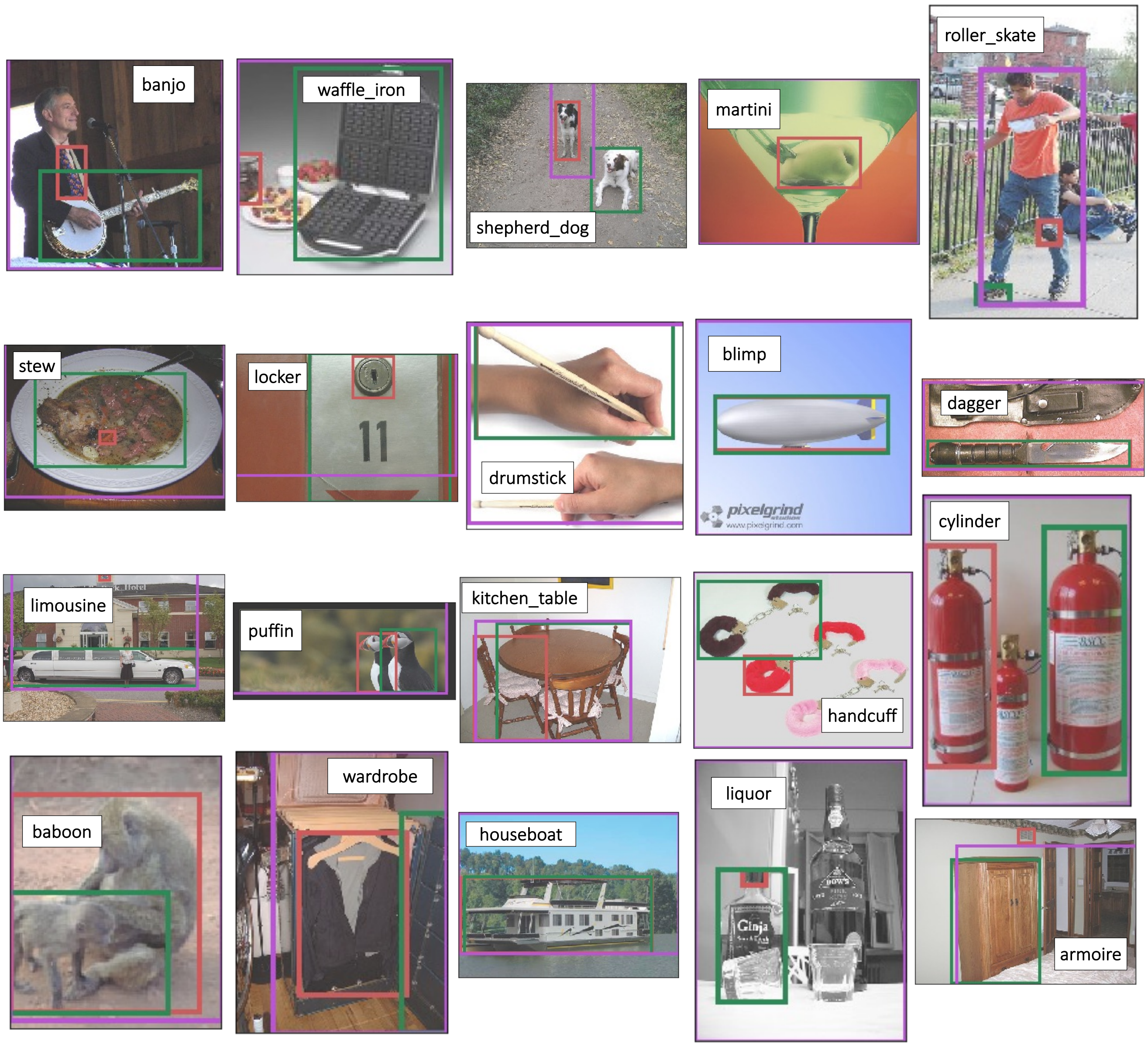

Figure 6. **Random samples of prediction on unseen categories.**

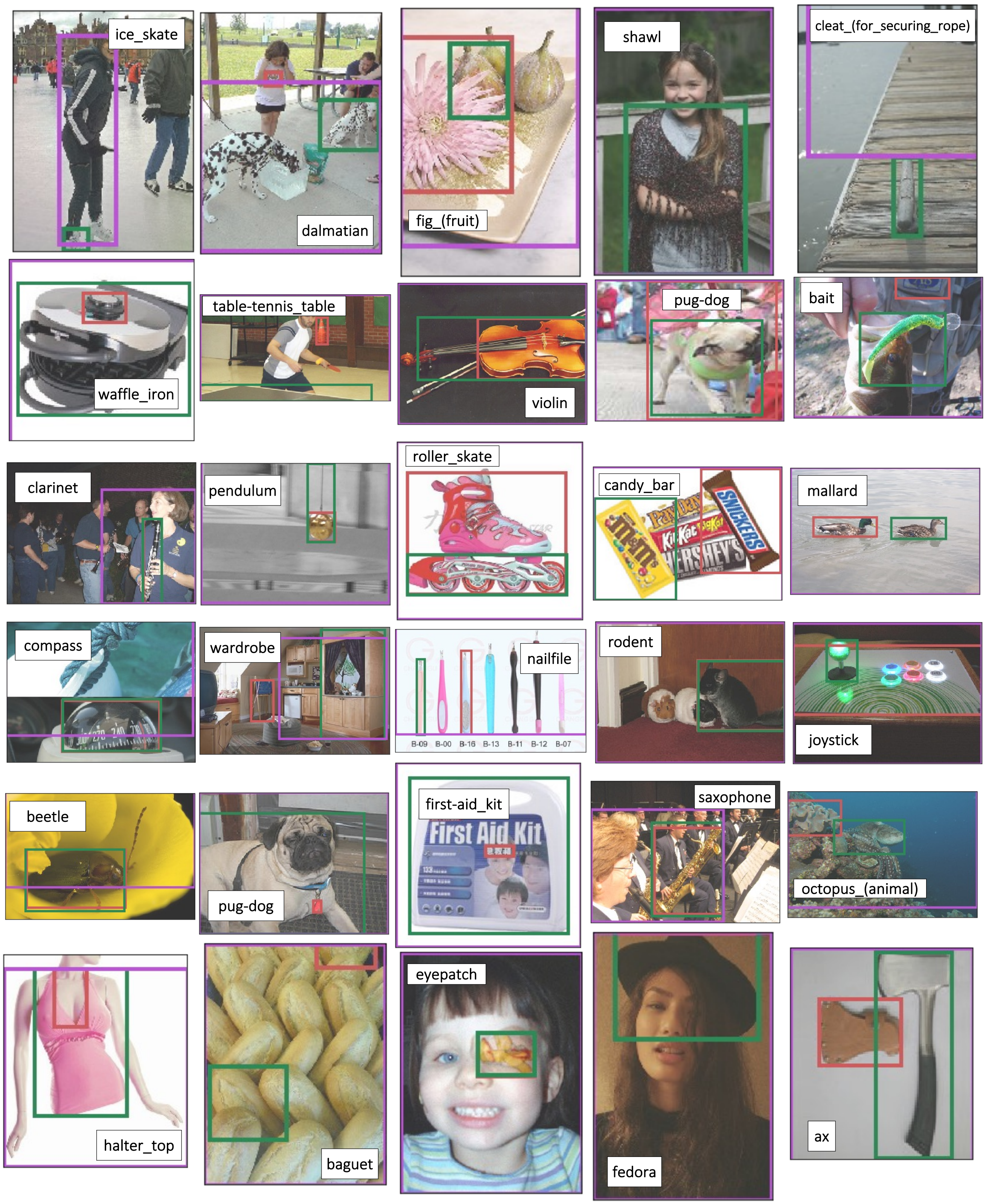

Figure 7. **Random samples of prediction on unseen categories.**