OmDet: Language-Aware Object Detection with Large-scale Vision-Language Multi-dataset Pre-training

Anonymous ECCV submission

Paper ID 7

Abstract. Advancing object detection to open-vocabulary and few-shot transfer has long been a challenge for computer vision research. This work explores a continual learning approach that enables a detector to expand its zero/few-shot capabilities via multi-dataset vision-language pre-training. Using natural language as knowledge representation, we explore methods to accumulate "visual vocabulary" from different training datasets and unify the task as a language-conditioned detection framework. Specifically, we propose a novel language-aware detector OmDet and a novel training mechanism. The proposed multimodal detection network can resolve the technical challenges in multi-dataset joint training and it can generalize to arbitrary number of training datasets without the requirements for manual label taxonomy merging. We pre-train on more than 20 million images with 4 million unique object vocabulary, and the resulting model is evaluated on 35 downstream tasks of ODinW [9]. Results show that OmDet is able to achieve the state-of-the-art finetuned performance on ODinW. Moreover, analysis shows that by scaling up the proposed pre-training method, OmDet continues to improve its zero/few-shot tuning performance, suggesting a promising way for further scaling.

Keywords: Vision-Language Pretraining, Multimodal Machine Learning, Continual Learning

1 Introduction

Object detection (OD) is one of the monumental tasks in computer vision (CV). Classical OD research has been focusing on improving the detector network to achieve higher accuracy with lower latency [19, 18, 14, 26] with fixed output label set. Recently, an emerging line of research based on vision-language pretraining (VLP) has been striving to upgrade OD models to solve the more challenging open-vocabulary setting, where the detector can generalize to new visual concepts with zero/few-shot adaption [5, 8, 10, 16]. This paper explores a continual learning approach, i.e., *can a detector incrementally learn from many OD datasets with increasing total visual vocabulary, and eventually achieve the open-vocabulary detection capabilities?*. This approach is appealing for several reasons: (1) It opens the possibility of life-long learning since one can improve a detector's zero/few-shot performance by feeding it with new datasets. (2) It is cost-effective since creating many small domain-specific datasets is much cheaper than creating a single large-vocabulary large dataset [6].

We propose a novel VLP-based object detection framework: OmDet (Omni-dataset Detection). We first formulate *language-aware object detection* which is a generalized version of OD task, i.e. given an image and a task (a set of object n ames), detecting the object instances that appeared in the task. Secondly, a novel deep vision-language fusion network is introduced to enable both *localization* and *classification* to be language-aware. Lastly, a new multi-dataset training algorithm is developed to enable OmDet to learn from arbitrary number of OD datasets regardless of their label set, and we scale the pre-training to a large number of datasets with total vocabulary size large than 4 million unique text labels.

The proposed method is first validated in a small-scale study with four OD datasets to confirm its multi-dataset learning ability. Then, a larger scale of study is conducted to scale up OmDet to very large vocabulary pretraining. We pre-train using a mixture of OD datasets with 20 million images and 4 million unique text labels that include both human annotations and pseudo labels. The resulting model is evaluated on the recently proposed ODinW dataset [9] that cover 35 different OD tasks in various domains. Comprehensive evaluation suggests that the proposed continual learning paradigm is able to achieve a new state-of-the-art performance compared to GLIP [10] that is pre-trained on larger datasets. Also, we show that accumulating multiple datasets to expand to large vocabulary OD learning is an effective method to boost OmDet's zero/few shot ability as well as parameter-efficient training performance (e.g. prompt tuning). By generating pseudo labels and adjusting different sampling ratios, OmDet is able to achieve the SOTA results on ELEVATER challenge.

2 Related Work

Objection detection, one of the predominant tasks in computer vision, aims to detect bounding boxes and classes of object instances. It has significantly evolved through the contributions of massive research in recent years. R-CNN [4] formulates the two-stage detectors paradigm, which is composed of a region proposal detector and a region-wise classifier. Consequent R-CNN series such as Fast R-CNN [3] and Faster R-CNN [19] make enhancements on the network pipeline to improve performance. While one-stage detectors like SSD [14], YOLO [18], and RetinaNet [12] are also in a competitive position by skipping the region proposal stage to simplify and speed up the framework. Recently, DETR [1] has proposed a transformer-based end-to-end object detection framework by framing the object detection task to a set of predictions. Follow-up DETR variants have proposed this framework in different directions. However, objection detection is often formulated as a closed-set problem with fixed and predefined classes and is diverse from the real-world setting. To conquer the closed-set limitation, more realistic scenarios such as Open-Vocabulary Object Detection (OVOD) have attracted lots of attention.

OVOD refers to the capability of only training on annotated datasets and generalizing to unseen novel classes. Recently, OVOD has made such progress with the utilization of a multi-modal vision-language pre-train model. RegionCLIP [24] generates pseudo-labels for region-text pairs from caption datasets to perform regional vision-language pre-training and transfer to OVOD. VILD [5] proposed a two-stage open-vocabulary detector, which distil embeddings from teacher model CLIP [17] or ALIGN [7]. With inspiration from CoOp [25], DetPro [2] introduces a technique to learn continuous detection prompt which improves the performance of VILD. OWL-ViT [16] uses the pre-trained image-text model as the base, then transfers it to the object detection domain by adding downstream detection heads and fine-tuning on OD datasets.

Unlike previous multi-dataset object detections, the proposed method is not required to have any extra human cost and naturally learning objects with the fused task embeddings from multiple datasets. Additionally the proposed model has OVOD capabilities by simply expanding the visual concept vocabulary size with more datasets and pseudo labeling from image-caption datasets.

3 Proposed Method

OmDet is designed for task-conditioned detection. Let V be a large vocabulary of objects types that OmDet can potentially detect. A task $T = \{w_1, w_2, ...w_k\}$ is a set of k object types that the model should detect in this forward path, where $w \in V$. Note that the size of T can be dynamic ranging from 1 to K, where K is the maximum supported number of object types in a single inference run. Then given an input image x and a task T, the model is expected to detect all of objects that appeared in T from x. Since T is not fixed, an ideal model can dynamically adapt its detection targets conditioned on the task.

3.1 Model Architecture

Following the above design principle, OmDet is introduced, a task-conditioned detection network that can learn from infinite combinations of tasks. It is composed of a vision backbone, a task encoder, a label encoder, and a multimodal detection network. The overall structure is illustrated in Fig1. The following will describe each component in details.

Vision Backbone Starting from the initial image $x_{img} \in R^{3\times H_0 \times W_0}$ (with 3 color channels), let the vision encoder f_v be a conventional CNN backbone or Vision Transformer backbone (e.g. Swin Transformer) generates a lower-resolution visual feature map $f \in R^{C\times H\times W}$ at each output layer. Then Feature Pyramid Network (FPN) [11] is used to aggregate information from top to bottom and output a set of visual feature maps $\{P2, P3, P4, P5\}$.

Task Encoder and Label Encoder The task set $T = \{w_1, w_2, ...w_k\} \in R^{k\times V}$ is set of natural language words. Then a task encoder f_t or a label en-

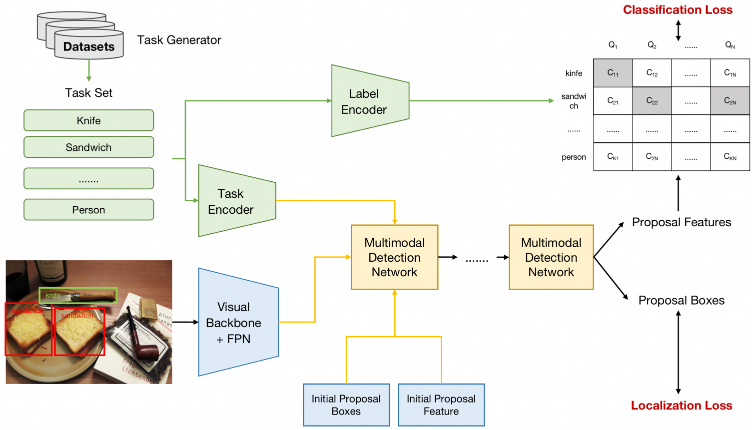

Fig. 1. Overview of the proposed OmDet Detector.

coder f_l is a transformer model that encode the task set T without order information, and output a set of contextual word embeddings, i.e. $\{t_1, t_2, ...t_k\} = f_t(w_1, w_2, ...w_k) \in R^{k \times d}$ and $\{l_1, l_2, ...l_k\} = f_l(w_1, w_2, ...w_k) \in R^{k \times d}$, where d is the contextual word embedding dimension size.

Multimodal Detection Network The Multimodal Detection Network (MDN) is a core component of OmDet. We deploy early fusion to combine information from the image and current task early on, in order to achieve strong performance. We are inspired by the Sparse-RCNN [22] network design, and developed an iterative query-based fusion mechanism.

Let $Q \in R^{N \times d}$ be a fixed small set of learnable proposal features. It is a set of high-dimensional (e.g., $d = 256$) latent features that capture the rich information of a potential instance, by combining information from the vision backbone and contextual task embedding from the task encoder. Also, let $B \in R^{N \times 4}$ be a set of learnable proposal boxes that is one-to-one assigned to each proposal feature. Then given the FPN output and task/label encoder output, the initial MDN operates as the following:

$$v_0 = \text{RoiPooler}(\{P2, P3, P4, P5\}, B_0) \quad (1)$$
$$[Q_1, T_1] = \text{MHSA}([Q_0, T_0]) \quad (2)$$
$$Q_2 = \text{DynamicCov}(Q_1, v_0) \quad (3)$$
$$B_1 = \text{RegHead}(Q_2) \quad (4)$$
$$C_1 = \gamma cosine(\text{ClsHead}(Q_2), L) \quad (5)$$

Note that MDN can be stacked to iterative refine its output the same as Sparse-RCNN, with the key difference that T is fused with the proposal feature before Dynamic Convolution layer and also T is also iterativly updated at each run of MDN block. This enables the network to learn to adjust the task embedding and the proposal embedding jointly and adapt both object localization and object classification head conditioned on the given task.

3.2 Model Training

Set Prediction Loss Given the above network, OmDet also uses set prediction loss [1] on the fixed-size set of predictions of classification and box coordinates. Set-based loss produces an optimal bipartite matching between predictions and ground truth objects using Hungarian algorithm. The matching cost is defined as follows:

$$L = \lambda_{cls} \cdot L_{cls} + \lambda_{L_1} \cdot L_{L_1} + \lambda_{giou} \cdot L_{giou} \tag{6}$$

Here L_{cls} is focal loss [12] of predicted classifications and ground truth category labels. L_{L_1} and L_{giou} are L1 loss and generalized IoU loss [1] between normalized center coordinates and height and width of predicted boxes and ground truth box, respectively. λ_{cls}, λ_{L_1} and λ_{giou} are coefficients of each component. The training loss is the same as the matching cost except that only performed on matched pairs. The final loss is the sum of all pairs normalized by the number of objects inside the training batch.

Task-Sampling Strategy In order to simulate the extreme multi-tasking setting at the training time and also enforce the model to condition its output on a given task, a novel task sampling strategy is used during training.

1. Let the max size of of a given task be K, for an image x from a dataset d in the mini-batch, we first sample $k \in [1, K]$ with a uniform distribution.
2. Let the number of unique object types in x be m, if $m > k$, then only a random subset of k object types are kept and the extra annotations are removed for this mini batch. If $m < k$, then additional negative object types are randomly selected from the vocabulary V of dataset d. If the vocabulary size of data d is less than K, then the reminder of missing negatives are filled with masking 0.

4 Pre-training and Transfer to ODinW

4.1 Experiment Setup

Large-scale Pre-training: COCO [13], Object365 [20], LVIS v1 [6], PhraseCut [23], and Google Conceptual Captions (GCC) [21] are used for large-scale pre-training. Specifically, GCC does not have bounding box annotations, so we utilize the phrase grounding ability of GLIP [10] to generate pseudo labels.

Downstream Tasks: ODinW is selected as the test data from ELEVATER benchmark [9] which is a new OD benchmark that consists of 35 diverse real-world tasks (Table 3 in Appendix). We select ODinW as the source of downstream tasks because of its diversity in terms of domain, training data size, number of categories, etc. Also, many of the 35 tasks have very limited (less than 100) training images, which makes it an extremely difficult task for standard detectors without any pre-training. We use the official train and test split for training and evaluation.

Training: For OmDet models, the initial learning rate is 5e-5 and it decays at 70% and 90% of total iteration steps by 0.1. ConvNeXt Tiny backbone and 6-layer detection head is used. For OmDet-Base, we use ConvNeXt Base as vision backbone. The batch size is 32 and the maximum number of detections per image is 300 and K is set to 80. All of the proposed models are pre-trained for 36 epochs using 16 NVIDIA A100 GPUs and then fine-tuned on the downstream data. All of the pre-training and fine-tuning experiments are conducted with the parameters of CLIP text encoder frozen.

Compared Models: The compared models including:

1. GLIP-Tiny: GLIP [10] is the state-of-the-art model used in ODinW dataset [9] that is pre-trained on a large set of visual grounding and OD data.
2. OmDet-C/CO/COL/COLP: we incrementally increase the number of pre-train datasets, including 4 intermediate variations. They are C (COCO) [13], CO (COCO + Object 365) [13, 20], COL (COCO + Object 365 + LVIS) [13, 20, 6] and COLP (COCO + Object 365 + LVIS + PhraseCut) [13, 20, 6, 23].
3. OmDet: OmDet is pre-trained on all of the pre-training data, and for GCC [21], pseudo labels generated on 3M images are used.
4. OmDet-Base: OmDet-Base is similar to OmDet, except switching to ConvNeXt [15] backbone and adding extra 3 million GCC images.

4.2 Results and Discussion

Overall, OmDet achieves the best detection performance compared to the other 4 variations (C/CO/COL/COLP) based on Table 1. Also, OmDet outperforms GLIP-Tiny [9] under full-model fine-tuning, which is pre-trained on a much larger dataset with a tunable text encoder. We then analyze the results from two aspects: (1) zero/few-shot performance and (2) parameter-efficient fine-tuning.

Models	Backbone	Pre-train Data	Zero-shot	Full-model FT	Head-only FT	Prompt FT
GLIP-Tiny [9]	Swin-T	O365,GOLDG	**19.7**	63.2	-	54.4
OmDet-C	ConvNeXt-T	COCO	9.8	61.7	54.7 (-11.3%)	21.0 (-65.9%)
OmDet-CO	ConvNeXt-T	COCO,O365	13.5	63.2	56.8 (-10.1%)	24.8 (-60.7%)
OmDet-COL	ConvNeXt-T	COCO,O365,LVIS	12.4	63.2	57.6 (-8.8%)	25.5 (-59.6%)
OmDet-COLP	ConvNeXt-T	COCO,O365,LVIS,PC	13.5	63.0	58.5 (-7.1%)	29.3 (-53.4%)
OmDet	ConvNeXt-T	COCO,O365,LVIS,PC,GCC3M	**16.0**	63.7	**59.8** (-6.1%)	**34.7** (-45.5%)
OmDet-Base	ConvNeXt-B	COCO,O365,LVIS,PC,GCC6M	-	**65.7**	-	-

Table 1. Average AP of zero-shot, full-model, head-only, and prompt fine-tuning (FT) on 35 downstream tasks in ODinW. The gray text shows the performance drop of parameter-efficient tuning compared to full-model tuning.

Zero/Few-Shot Object Detection As shown in Table 1, adding more pre-train datasets yields significant improvement in zero-shot settings. Specifically, adding object365 dataset gives an absolute gain of 3.7 points on the average mAP. Surprisingly, adding LVIS to the pre-train data hurts performance by 1.1 points. We speculate that the performance drop is due to the noisy and incomplete annotations of LVIS dataset. Adding GCC dataset to the pre-train corpora yields another huge gain, leading the zero-shot performance to 16.0 (compared to 9.8 for OmDet-C).

Meanwhile, the 35 downstream tasks in ODinW come with different training data sizes, varying from only 17 training images to more than 32K training images. Therefore, we divide the 35 tasks into three categories: (1) Few-shot (8 tasks): tasks with fewer than 200 training images (2) Medium-shot (13 tasks): tasks with between 200 to 2000 training data (3) Big-shot (14 tasks): tasks with more than 2000 training images. Results with full-model fine-tuning are summarized in Table 2. Results show that large-scale multi-dataset pre-training is particularly effective for few-shot and medium-shot tasks with limited in-domain training data. Especially for few-shot datasets, OmDet outperforms OmDet-C with 6.41 absolute AP points. Whereas for Big-shot tasks, we do not see consistent improvement when increasing the size of pre-training datasets. We suspect that big-shots tasks already contain enough information in the training set, which shadows the improvement from the pre-training stage.

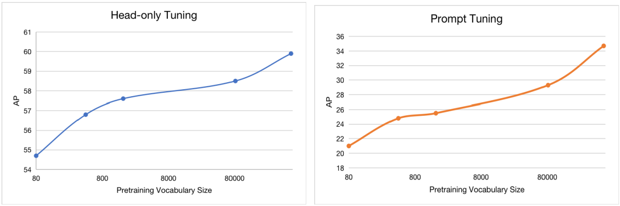

Fig. 2. Vocabulary size used in pre-training vs. the AP score of fine-tuning on ODinW with head-only and prompt tuning. X-axis is in log-scale.

Models	Few-Shot	Medium-Shot	Big-Shot
OmDet-C	49.48	57.09	70.16
OmDet-CO	54.37	58.89	70.98
OmDet-COL	55.07	57.99	**71.22**
OmDet-COLP	53.44	58.05	70.94
OmDet	**55.89**	**59.23**	70.54

Table 2. Average AP of full-model fine-tuning on 35 downstream tasks in ODinW for Few-shot, Medium-Shot, and Big-Shot tasks.

Parameter-efficient Fine-tuning As large-scale pretraining models get significantly larger, e.g., more than 1B parameters, the cost to fine-tune (FT) the entire model becomes prohibitive for low-end GPUs. Parameter-efficient fine-tuning is designed to alleviate this challenge by only tuning a very small proportion of the entire model. In this paper, we explore two options: Head-only Tuning and Prompt Tuning.

Experimental results show that large-scale multi-dataset pre-training is crucial for successful parameter-pretraining (Table 1). For Head-only FT, the performance drop is reduced from 11.3% for OmDet-C to only 6.1% for OmDet. The same trend is observed for Prompt FT, in which the performance drop compared to full-model tuning is reduced from 65.9% to 45.5% from OmDet-C to OmDet. Figure 2 also visualizes the trend of AP vs. the vocabulary size in pre-training (log-scale). The apparent up-going curve can be observed as more visual concepts are included during pre-training. This suggests that:

(1) Multi-dataset pre-training enables the accumulation of a large number of visual concepts, which leads to a stronger backbone that extracts general-purpose visual features (supported by head-only FT results).

(2) The diversity in language is crucial for successful prompt tuning such that the entire model output can be controlled by the task embedding only (less than 1% of the parameters of the entire model).

Also, we found that the prompt-tuning performance of OmDet is significantly lower than GLIP. We suspect the prompt tuning used in OmDet is too simple, i.e., initialize the task embedding with natural language and tune the task word embedding alone. We plan to improve the prompt-tuning strategy in a later version of this pre-print.

Training Strategy of SOTA In order to reach better performance on ELEVATER challenge, we pre-train a larger model, OmDet-Base, with ConvNext Base backbone. All pre-training data of OmDet are used, together with another 3 million images from GCC. After pre-training, we first jointly fine-tune the 35 datasets of ODinW for 3X schedule with a fair sampling strategy that assigns each dataset with the same probability. This first-stage fine-tuning already gives us better performance than experiments that we have done before. We further train another 1x schedule by increasing the sampling ratio to 2 for datasets that are not yet converged and keeping other datasets as 1. Using this sampling strategy, our full-shot result on ODinW increases to 65.7.

5 Conclusion

This work proposes to advance zero/few-shot OD via continual pre-training from a large number of OD datasets. OmDet is proposed to solve the taxonomy conflict and fore/background inconsistency problems during multi-datasets joint training. The proposed deep fusion mechanism, Multimodal Detection Network, is able to detect specified objects conditioned on users task input in the format of free-form natural language. Experiments show that enlarging the vocabulary size via multi-datasets pre-training effectively improves zero/few-shot learning and parameter-efficient fine-tuning. OmDet achieved the state-of-the-art performance on a diverse set of downstream tasks. Future research will focus on improving OmDet by efficient task-sampling strategy, utilizing more diverse multimodal datasets, and exploring diverse language and vision backbones with freezing particular parameters or fully updating them.

References

1. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
2. Du, Y., Wei, F., Zhang, Z., Shi, M., Gao, Y., Li, G.: Learning to prompt for open-vocabulary object detection with vision-language model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14084–14093 (2022)
3. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 1440–1448 (2015)
4. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 580–587 (2014)
5. Gu, X., Lin, T.Y., Kuo, W., Cui, Y.: Open-vocabulary object detection via vision and language knowledge distillation. arXiv preprint arXiv:2104.13921 (2021)
6. Gupta, A., Dollar, P., Girshick, R.: Lvis: A dataset for large vocabulary instance segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5356–5364 (2019)
7. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: International Conference on Machine Learning. pp. 4904–4916. PMLR (2021)
8. Kamath, A., Singh, M., LeCun, Y., Synnaeve, G., Misra, I., Carion, N.: Mdetr-modulated detection for end-to-end multi-modal understanding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1780–1790 (2021)
9. Li, C., Liu, H., Li, L.H., Zhang, P., Aneja, J., Yang, J., Jin, P., Lee, Y.J., Hu, H., Liu, Z., et al.: Elevater: A benchmark and toolkit for evaluating language-augmented visual models. arXiv preprint arXiv:2204.08790 (2022)
10. Li, L.H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., et al.: Grounded language-image pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10965–10975 (2022)
11. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
12. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
13. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
14. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision. pp. 21–37. Springer (2016)
15. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11976–11986 (2022)

16. Minderer, M., Gritsenko, A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., et al.: Simple open-vocabulary object detection with vision transformers. arXiv preprint arXiv:2205.06230 (2022)
17. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning. pp. 8748–8763. PMLR (2021)
18. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
19. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
20. Shao, S., Li, Z., Zhang, T., Peng, C., Yu, G., Zhang, X., Li, J., Sun, J.: Objects365: A large-scale, high-quality dataset for object detection. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 8430–8439 (2019)
21. Sharma, P., Ding, N., Goodman, S., Soricut, R.: Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 2556–2565 (2018)
22. Sun, P., Zhang, R., Jiang, Y., Kong, T., Xu, C., Zhan, W., Tomizuka, M., Li, L., Yuan, Z., Wang, C., et al.: Sparse r-cnn: End-to-end object detection with learnable proposals. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14454–14463 (2021)
23. Wu, C., Lin, Z., Cohen, S., Bui, T., Maji, S.: Phrasecut: Language-based image segmentation in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10216–10225 (2020)
24. Zhong, Y., Yang, J., Zhang, P., Li, C., Codella, N., Li, L.H., Zhou, L., Dai, X., Yuan, L., Li, Y., et al.: Regionclip: Region-based language-image pretraining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16793–16803 (2022)
25. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. International Journal of Computer Vision pp. 1–12 (2022)
26. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. arXiv preprint arXiv:1904.07850 (2019)

Dataset	Categories	# Train Image	#Test Image
CottontailRabbits	1	1980	10
EgoHands(generic)	1	3840	480
MountainDewCommercial	1	17	1
Packages	1	19	3
Raccoon	1	150	17
WildfireSmoke	1	516	74
Pistols	1	2377	297
Pothole	1	465	67
MaskWearing	2	105	15
NorthAmericaMushrooms	2	41	5
OxfordPets(species)	2	2523	358
PKLot640	2	8691	1242
ThermalCheetah	2	90	14
ThermalDogsAndPeople	2	142	20
BCCD	3	255	36
HardHatWorkers	3	5069	1766
ShellfishOpenImages	3	407	58
EgoHands(specific)	4	3840	480
AerialMaritimeDrone(large)	5	52	7
AerialMaritimeDrone(tiled)	5	371	32
VehiclesOpenImages	5	878	126
BrackishUnderwater	6	11739	1468
Dice	6	576	71
Aquarium	7	448	63
DroneControl	8	32688	4675
WebsiteScreenshots	8	1688	242
SelfDrivingCar	11	24000	3000
ChessPieces	13	202	29
UnoCards	15	6295	899
PascalVOC	20	13690	3422
AmericanSignLanguageLetters	26	1512	72
Plantdoc	30	2128	239
BoggleBoards	36	285	35
OxfordPets(breed)	37	2437	345
OpenPoetryVision	43	2798	402
Total	314	132314	20070

Table 3. Statistics of ElEVATER 35 object detection datasets