YOLO-World: Real-Time Open-Vocabulary Object Detection
Supplementary Materials

YOLO-World Team
https://github.com/AILab-CVC/YOLO-World

This appendix aims to supplement the main text of YOLO-World in the following three aspects, *i.e.*, additional experimental results (refer to Sec. A), implementation details (refer to Sec. B), and automatic labeling on image-text data (refer to Sec. D). We hope this appendix can help readers better comprehend our work, and provide sufficient details for researchers in further study and development.

A. Additional Experiments

A.1. Main Results

In Tab. A1, we further provide the performance of largest YOLO-World-XL, which contains 74M parameters and is pre-trained on O365 [6], GoldG [2], and the labeled CC3M [7].

Method	Params	Pre-trained Data	AP	AP_r	AP_c	AP_f
YOLO-World-S	13M	O365,GoldG	26.2	19.1	23.6	29.8
YOLO-World-M	29M	O365,GoldG	31.0	23.8	29.2	33.9
YOLO-World-L	48M	O365,GoldG	35.0	27.1	32.8	38.3
YOLO-World-L	48M	O365,GoldG,CC3M†(245K)	35.4	27.6	34.1	38.0
YOLO-World-XL	74M	O365,GoldG,CC3M†(245K)	36.6	28.7	35.0	39.5

Table A1. **Zero-shot Evaluation on LVIS.** We evaluate YOLO-World on LVIS `minival` [2] in a zero-shot manner and report *Fixed AP* [1]. † denotes the pseudo-labeled CC3M in our setting, which contains 246k samples.

A.2. Additional Visualizations

In this section, we further extend the visualizations (Sec. 4.5 in the main text) from the following three aspects. The visualization results provide strong evidence of the superiority of our pre-training strategy for open-vocabulary object detection.

Zero-shot Inference on LVIS. Fig. A1 shows the visualization results based on the LVIS categories which are generated by the pre-trained YOLO-World-L in a zero-shot manner. The pre-trained YOLO-World exhibits strong zero-shot transfer capabilities and is able to detect as many objects as possible within the image.

Inference with User's Vocabulary. In Fig. A2, we explore the detection capabilities of YOLO-World with our defined categories. The visualization results demonstrate that the pre-trained YOLO-World-L also exhibits the capability for (1) fine-grained detection (*i.e.*, detect the parts of one object) and (2) fine-grained classification (*i.e.*, distinguish different sub-categories of objects.).

Referring Object Detection. In Fig. A3, we leverage some descriptive (discriminative) noun phrases as input, *e.g.*, the standing person, to explore whether the model can locate regions or objects in the image that match our given input. The visualization results display the phrases and their corresponding bounding boxes, demonstrating that the pre-trained YOLO-World has the referring or grounding capability. This ability can be attributed to the proposed pre-training strategy with large-scale training data.

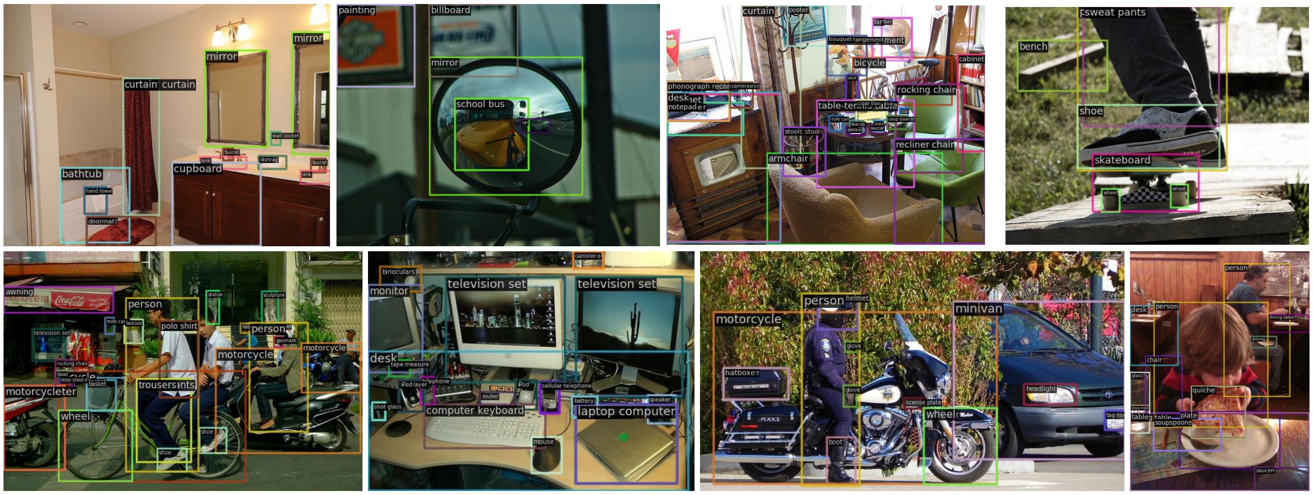
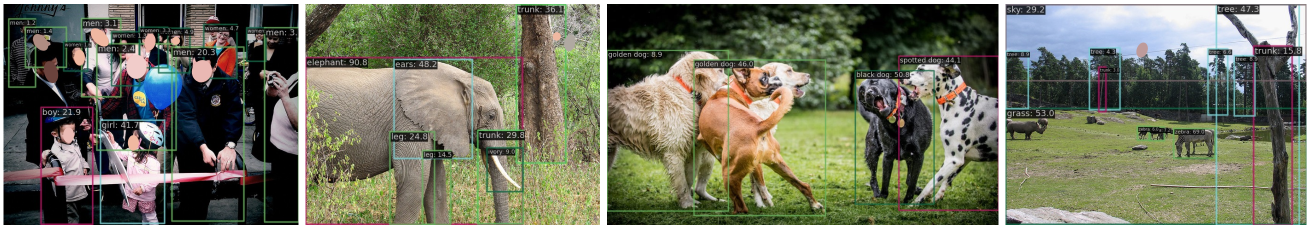


Figure A1. **Additional Visualization Results on Zero-shot Inference on LVIS.** We adopt the pre-trained YOLO-World-L and infer with the LVIS vocabulary (containing 1203 categories) on the COCO `val2017`.



{men, women, boy, girl} {elephant, ear, leg, trunk, ivory} {golden dog, black dog, spotted dog} {grass, sky, zebra, trunk, tree}

Figure A2. **Additional Visualization Results on User's Vocabulary.** We define the custom vocabulary for each input image and YOLO-World can detect the accurate regions according to the vocabulary. Images are obtained from COCO `val2017`.

B. Additional Implementation Details for YOLO-World

B.1. Re-parameterization for RepVL-PAN

During inference on an offline vocabulary, we adopt re-parameterization for RepVL-PAN for faster inference speed and deployment. Firstly, we pre-compute the text embeddings $W \in \mathbb{R}^{C\times D}$ through the text encoder.

Re-parameterize T-CSPLayer. For each T-CSPLayer in RepVL-PAN, we can re-parameterize and simplify the process of adding text guidance by reshaping the text embeddings $W \in \mathbb{R}^{C\times D\times 1\times 1}$ into the weights of a 1×1 convolution layer (or a linear layer), as follows:

$$X' = X \odot \texttt{Sigmoid}(\texttt{max}(\texttt{Conv}(X, W), \texttt{dim=1})), \tag{1}$$

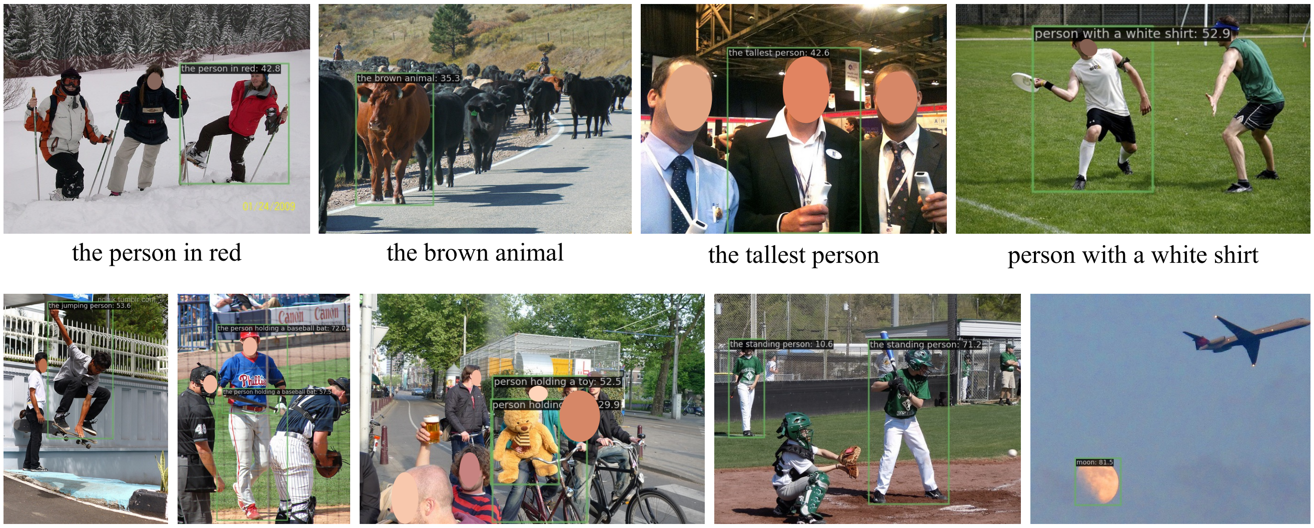
where $X \times \in \mathbb{R}^{B\times D\times H\times W}$ and $X' \in \mathbb{R}^{B\times D\times H\times W}$ are the input and output image features. $\odot$ is the matrix multiplication with `reshape` or `transpose`.

Re-parameterize I-Pooling Attention. The I-Pooling Attention can be re-parameterize or simplified by:

$$\tilde{X} = \texttt{cat}(\texttt{MaxPool}(X_3, 3), \texttt{MaxPool}(X_4, 3), \texttt{MaxPool}(X_5, 3)), \tag{2}$$

where `cat` is the concentration and $\texttt{MaxPool}(\cdot, 3)$ denotes the max pooling for 3×3 output features. $\{X_3, X_4, X_5\}$ are the multi-scale features in RepVL-PAN. $\tilde{X}$ is flattened and has the shape of $B \times D \times 27$. Then we can update the text embeddings by:

$$W' = W + \texttt{Softmax}(W \odot \tilde{X}), \texttt{dim=-1}) \odot W, \tag{3}$$


the person in red: 42.8
the brown animal: 35.3
the tallest person: 42.6
person with a white shirt: 52.9
the jumping person: 53.6
the person holding a baseball bat: 72.0
person holding a toy: 52.5
the standing person: 10.6
the standing person: 71.2
moon: 81.5

the person in red | the brown animal | the tallest person | person with a white shirt

the jumping person | person holding a baseball bat | person holding a toy | the standing person | moon

Figure A3. **Additional Visualization Results on Referring Object Detection.** We explore the capability of the pre-trained YOLO-World to detect objects with descriptive noun phrases. Images are obtained from COCO `val2017`.

B.2. Fine-tuning Details.

We remove all T-CSPLayers and Image-Pooling Attention in RepVL-PAN when transferring YOLO-World to COCO [4] object detection, which only contains 80 categories and has a relatively low dependency on visual-language interaction. During fine-tuning, we initialize YOLO-World using pre-trained weights. The learning rate of fine-tuning is set to 0.0002 with the weight decay set to 0.05. After fine-tuning, we pre-compute the class text embeddings with given COCO categories and store the embeddings into the weights of the classification layers.

C. Pre-training YOLO-World at Scale

When pre-training small models, *e.g.*, YOLO-World-S, a natural question we have is: how much capacity does a small model have, and how much training data or what kind of data does a small model need? To answer this question, we leverage different amounts of pseudo-labeled region-text pairs to pre-train YOLO-World. As shown in Tab. A2, adding more image-text samples can increase the zero-shot performance of YOLO-World-S. Tab. A2 indicates: (1) adding image-text data can improve the overall zero-shot performance of YOLO-World-S; (2) using an excessive amount of pseudo-labeled data may have some negative effects for small models (YOLO-World-S), though it can improve the on rare categories (AP$_r$). However, using fine-grained annotations (GoldG) for small models can provide significant improvements, which indicates that large-scale high-quality annotated data can significantly enhance the capabilities of small models. And Tab. 3 in the main text has shown that pre-training with the combination of fine-annotated data and pseudo-annotated data can perform better. We will explore more about the data for pre-training small models or YOLO detectors in future work.

D. Automatic Labeling on Large-scale Image-Text Data

In this section, we add details procedures for labeling region-text pairs with large-scale image-text data, *e.g.*, CC3M [7]. The overall labeling pipeline is illustrated in Fig. A4, which mainly consists of three procedures, *i.e.*, (1) extract object nouns, (2) pseudo labeling, and (3) filtering. As discussed in Sec. 3.4, we adopt the simple n-gram algorithm to extract nouns from captions.

Region-Text Proposals. After obtaining the set of object nouns $T = \{t_k\}^K$ from the first step, we leverage a pre-trained open-vocabulary detector, *i.e.*, GLIP-L [3], to generate pseudo boxes $\{B_i\}$ along with confidence scores $\{c_i\}$:

$$\{B_i, t_i, c_i\}_{i=1}^N = \texttt{GLIP-Labeler}(I, T), \quad (4)$$

Method	Pre-trained Data	Samples	AP	AP_r	AP_c	AP_f
YOLO-World-S	O365	0.61M	16.3	9.2	14.1	20.1
YOLO-World-S	O365+GoldG	1.38M	24.2	16.4	21.7	27.8
YOLO-World-S	O365+CC3M-245k	0.85M	16.5	10.8	14.8	19.1
YOLO-World-S	O365+CC3M-520k	1.13M	19.2	10.7	17.4	22.4
YOLO-World-S	O365+CC3M-750k	1.36M	18.2	11.2	16.0	21.1

Table A2. **Zero-shot Evaluation on LVIS.** We evaluate the performance of pre-training YOLO-World-S with different amounts of data, the image-text data.

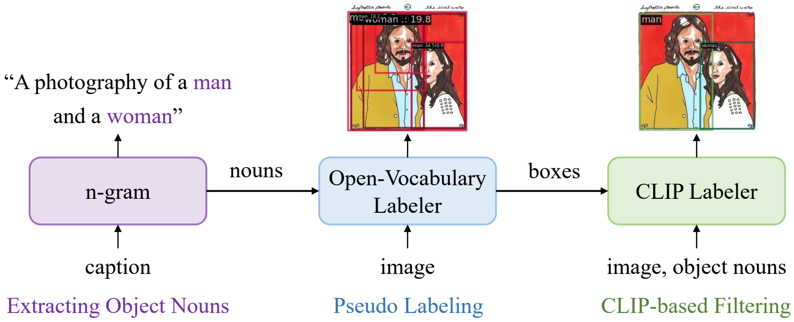


Figure A4. **Labeling Pipeline for Image-Text Data** We first leverage the simple n-gram to extract object nouns from the captions. We adopt a pre-trained open-vocabulary detector to generate pseudo boxes given the object nouns, which forms the coarse region-text proposals. Then we use a pre-trained CLIP to rescore or relabel the boxes along with filtering.

where $\{B_i, t_i, c_i\}_{i=1}^N$ are the coarse region-text proposals.

CLIP-based Re-scoring & Filtering. Considering the region-text proposals containing much noise, we present a restoring and filtering pipeline with the pre-trained CLIP [5]. Given the input image I, caption T, and the coarse region-text proposals $\{B_i, t_i, c_i\}_{i=1}^N$, the specific pipeline is listed as follows:

- (1) Compute Image-Text Score: we forward the image I with its caption T into CLIP and obtain the image-text similarity score s^{img}.
- (2) Compute Region-Text Score: we crop the region images from the input image according to the region boxes $\{B_i\}$. Then we forward the cropped images along with their texts $\{t_i\}$ into CLIP and obtain the region-text similarity $S^r = \{s_i^r\}_{i=1}^N$.
- (3) `[Optional]` Re-Labeling: we can forward each cropped image with all nouns and assign the noun with maximum similarity, which can help correct the texts wrongly labeled by GLIP.
- (4) Rescoring: we adopt the region-text similarity S^r to rescore the confidence scores $\tilde{c_i} = \sqrt{c_i * s_i^r}$.
- (5) Region-level Filtering: we first divide the region-text proposals into different groups according to the texts and then perform non-maximum suppression (NMS) to filter the duplicate predictions (the NMS threshold is set to 0.5). Then we filter out the proposals with low confidence scores (the threshold is set to 0.3).
- (6) Image-level Filtering: we compute the image-level region-text scores s^{region} by averaging the kept region-text scores. Then we obtain the image-level confidence score by $s = \sqrt{s^{img} * s^{region}}$ and we keep the images with scores larger than 0.3.

The thresholds mentioned above are empirically set according to the part of labeled results and the whole pipeline is automatic without human verification. Finally, the labeled samples are used for pre-training YOLO-World. We will provide the pseudo annotations of CC3M for further research.

References

[1] Achal Dave, Piotr Dollár, Deva Ramanan, Alexander Kirillov, and Ross B. Girshick. Evaluating large-vocabulary object detectors: The devil is in the details. *CoRR*, abs/2102.01066, 2021. 1

[2] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. MDETR - modulated detection for end-to-end multi-modal understanding. In *ICCV*, pages 1760–1770, 2021. 1

[3] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, Kai-Wei Chang, and Jianfeng Gao. Grounded language-image pre-training. In *CVPR*, pages 10955–10965, 2022. 3

[4] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 740–755, 2014. 3

[5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 4

[6] 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*, pages 8429–8438, 2019. 1

[7] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, pages 2556–2565, 2018. 1, 3