# Appendix for DetCLIP: Dictionary-Enriched Visual-Concept Paralleled Pre-training for Open-world Detection

## A Negative Impacts and Limitations

**Potential Negative Social Impact.** Our method has no ethical risk on dataset usage and privacy violation since all the benchmarks are publicly available and transparent.

**Limitations and Future Works.** The localization ability of the region proposals is still limited by the annotation of the bounding box. More weakly-supervision can be included to learn from image-text pairs. Furthermore, although we prove the effectiveness of our method on the web-collected dataset YFCC [12], we expect to extend our method to larger image-text pair datasets from the Internet.

## B Dataset Details

In this section, we provide more details of training datasets used in our experiments, which include (1) the approach of generating pseudo detection labels for image-text pair dataset; and (2) a dataset comparison with GLIP [9].

**Pseudo Labeling on Image-Text Pair Data.** We use image-text pair data from the web-collected dataset YFCC100m [12]. To generate pseudo detection labels for image-text pair data, we first use a Region Proposal Network (RPN) pre-trained on Objects365 to extract object proposals. To ensure the quality of proposals, we filter result bounding boxes with objectness scores below a threshold of 0.3 or region area smaller than 6000. This operation also helps significantly reduce the number of proposal candidates and accelerates the pseudo-labeling process.Then a powerful pre-trained CLIP [10] model (ViT-L) is used to predict pseudo class labels for each retained bounding box. To alleviate the partial-label problem, we use concept names from our proposed concept dictionary (Sec.3.2) instead of the raw caption as the text input. Following CLIP, the prompt "a photo of a category." is used to pad a category name into the sentence. Since the proposed dictionary consists of a large number of concepts (i.e., 14k), to accelerate the inference, we pre-compute the text embeddings of all concepts and store them for the later computation. For each proposal bounding box, we first crop it from the raw image, then resize it to $224 \times 224$ and feed it into the visual encoder to obtain the visual embedding. We use the cosine similarity as the classification score, which is computed as

$$s_i^j = f_\theta^I(R_i) g_\phi^T(c_j)^\top \tag{1}$$

where $f_\theta^I$ and $g_\phi^T$ stand for the image and text encoder of the CLIP model; $R_i$ and $c_j$ are $i$-th cropped proposal and $j$-th category, respectively. Both embeddings are L2-normalized before the similarity calculation. After category prediction, a second-stage filtering is adopted to drop proposals with a classification score below $0.24$. Finally, we sample 1M images from the results to form our final training image-text pair data.

**Training Data Comparison (with GLIP [9]).** Table 1 compares the training data used by DetCLIP and GLIP. The explanation of each dataset can be found in the table caption. Our DetCLIP-T uses less than half training data compared to GLIP-T, while DetCLIP-L uses **less than 10%** training data compared to GLIP-L.

Table 1: A training data comparison between DetCLIP and GLIP [9]. Numbers in parentheses indicate the volume of the corresponding dataset. O365-V1 and -V2 are 1st and 2nd version of Objects365 [11] dataset, respectively. 4ODs is a combination of 4 detection datasets, i.e., Objects365 [11], OpenImages [6], Visual Genome [7] and ImageNetBoxes [8]. GoldG is the grounding data introduced by MDETR[5]. Cap4M and Cap24M are web-crawled image-text pair dataset collected by GLIP. YFCC1M is our pseudo-labeled image-text pair data sampled from YFCC100M [12].

| | Detection | Grounding | Image-text | Total Volume |
|---|---|---|---|---|
| GLIP-T | O365 V1 (0.66M) | GoldG (0.77M) | Cap4M (4M) | ~5.43M |
| GLIP-L | 4ODs (2.66M) | GoldG (0.77M) | Cap24M (24M) | ~27.43 M |
| DetCLIP-T/L | Sampled O365 V2 (0.66M) | GoldG (0.77M) | YFCC1M (1M) | ~2.43M |

Table 2: Manually designed prompts for five downstream detection datasets.

| Dataset | Original Prompt | Manually Designed Prompt |
|---|---|---|
| Rabbits | Cottontail-Rabbits | rabbit, any of various burrowing animals of the family Leporidae having long ears and short tails. |
| Mushrooms | Cow | flat mushroom, a little cup shaped mushroom. |
| | Chanterelle | yellow mushroom, are orange, yellow or white, meaty and funnel-shaped. |
| Packages | package | there is a package on the porch. |
| Pothole | pothole | there are some holes on the road. |
| Pistols | pistol | handgun, Single shot, MAGNUM, USP, machine pistol, revolver, IMI Desert Eagle. |

## C More Results on LVIS and 13 Detection Datasets

**More results under GLIP-protocal.** We study the generalization ability of our models by zero-shot transferring them to LVIS [3] full validation set and 13 downstream detection datasets [9]. Following GLIP [9], we use manually designed prompts for some downstream datasets, as illustrated in Table 2. AP for LVIS full validation set and averaged AP over 13 datasets are reported in Table 3. Despite using much less training data, DetCLIP models can dominate their GLIP's [9] counterparts in most cases (except $AP_f$ on LVIS for DetCLIP-L). Notably, compared to GLIP, DetCLIP considerably boosts the performance for rare categories, which is an important indicator reflecting models' generalization ability for the open-world detection task. Detailed AP performances for 13 downstream detection datasets can refer to Table 4.

Table 3: Zero-shot transfer performance on LVIS [3] full validation dataset and the 13 downstream detection datasets [9]. $AP_r$/$AP_c$/$AP_f$ indicate the AP values for rare, common, frequent categories. 'DH' and 'F' in GLIP [9] baselines stand for the dynamic head [1] and cross-modal fusion.

| Model | Backbone | Pre-Train Data | LVIS val AP ($AP_r$ / $AP_c$ / $AP_f$) | 13 Data AP |
|---|---|---|---|---|
| GLIP-T(A)[9] | Swin-T+DH+F | O365 | 12.3 (6.00 / 8.00 / 19.4) | 28.8 |
| GLIP-T[9] | Swin-T+DH+F | O365,GoldG,Cap4M | 17.2 (10.1 / 12.5 / 25.2) | 46.5 |
| GLIP-L[9] | Swin-L+DH+F | 4ODs,O365,GoldG,Cap24M | 26.9 (17.1 / 23.3 / **36.4**) | 52.1 |
| DetCLIP-T(A) | Swin-T | O365 | 22.1 (18.4 / 20.1 / 26.0) | 31.2 |
| DetCLIP-T(B) | Swin-T | O365, GoldG | 27.2 (21.9 / 25.5 / 31.5) | 43.4 |
| DetCLIP-T | Swin-T | O365, GoldG, YFCC1M | 28.4 (25.0 / 27.0 / 31.6) | 45.2 |
| DetCLIP-L | Swin-L | O365, GoldG, YFCC1M | **31.2** (**27.6** / **29.6** / 34.5) | **52.9** |

Table 4: Detailed zero-shot transfer AP of DetCLIP on 13 detection datasets [9].

| Model | Thermal | Aquarium | Rabbits | Mushrooms | AerialDrone | PascalVOC | Vehicles |
|---|---|---|---|---|---|---|---|
| DetCLIP-T(A) | 38.5 | 14.6 | 70.3 | 11.2 | 14.4 | 53.6 | 57.6 |
| DetCLIP-T(B) | 56.8 | 15.2 | 75.8 | **59.9** | 15.9 | 54.2 | 58.0 |
| DetCLIP-T | 59.1 | 16.0 | 76.4 | 58.9 | 17.8 | 56.5 | 57.5 |
| DetCLIP-L | **68.3** | **24.5** | **78.2** | 58.0 | **25.4** | **64.0** | **65.5** |
| Model | EgoHands | Raccoon | Pothole | Pistols | Shellfish | Packages | Avg |
| DetCLIP-T(A) | 2.5 | 43.8 | 2.7 | 14.2 | 17.8 | 64.3 | 31.2 |
| DetCLIP-T(B) | 32.1 | 50.8 | 14.3 | 37.4 | 22.2 | 71.1 | 43.4 |
| DetCLIP-T | 42.5 | 52.2 | 15.7 | 39.0 | 23.2 | 73.1 | 45.2 |
| DetCLIP-L | **43.2** | **63.3** | **28.0** | **42.5** | **48.2** | **78.5** | **52.9** |

**More results under VILD-protocal.** To make a more comprehensive evaluation of our method, we also perform experiments under the VILD [2] protocol, i.e., the method is trained on base categories and then evaluated on novel categories using the original LVIS AP metric. We replace the Objects365 part in our training data with LVIS-base, and GoldG and YFCC1M are still included. Including additional data will lead to somehow unfair comparison with VILD but it is necessary since this is the core component in our method to enable zero-shot capability, which differs from VILD that distills knowledge from a pre-trained CLIP model. Note that we implement DetCLIP using the same training/testing setting as in the paper, and do not use techniques such as large-scale jittering and prompt ensemble which is adopted by VILD to boost the performance. The results are shown in the Table 5. Our method (27.3 mAP) outperforms VILD (22.5 mAP) by 4.8% mAP.

Table 5: An comparison with VILD model under VILD protocal.

| Model | Backbone | LVIS val AP ($AP_r$ / $AP_c$ / $AP_f$) |
|---|---|---|
| VILD [2] | ResNet50 | 22.5 (**16.1** / 20.0 / 28.3) |
| DetCLIP | ResNet50 | **27.3** (14.9 / **25.4** / **34.8**) |

## D Ablation Studies

**Sequential Formulation with Shuffled Grounding Data (Sec 3.1).** The sequential formulation (e.g., GLIP [9]) is not effective for modeling open-world object detection as a visual-language task since it leads to unnecessary interaction between category names in the attention module. To demonstrate the idea, we randomly shuffle the word order in the grounding training data and report the performance comparison in Table 6. It can be seen that randomly shuffling the word order in the grounding data can even bring a slight improvement (i.e., +1.4% on LVIS minival) on zero-shot transfer AP for the downstream detection task, indicating that the noun phrase is more critical for the detection task, compared to the context information. Therefore, DetCLIP drops the context information and treats each noun phrase as a paralleled text input, which avoids unnecessary attention among class names and achieves better training efficiency.

Table 6: Performance comparison of sequential formulation with different grounding data (shuffled word order/original caption). Zero-shot transfer performance on LVIS [3] dataset are reported. The model is trained on Objects365 [11] and GoldG datasets.

| Model | Grounding Data | LVIS minival AP ($AP_r$ / $AP_c$ / $AP_f$) | LVIS val AP ($AP_r$ / $AP_c$ / $AP_f$) |
|---|---|---|---|
| Sequential concept form | Original caption | 26.0 (18.0 / 22.8 / 30.3) | 18.9 (11.7 / 15.8 / 25.6) |
| | Shuffled Word Order | **27.4** (**18.6** / **23.8** / **32.3**) | **19.9** (**12.5** / **16.4** / **27.0**) |

**Impact of Concept Dictionary's Size.** To study the impact of the scale of the proposed concept dictionary, we build three concept dictionaries with different sizes by: (1) only using class names from Objects365 [11] dataset; (2) using class names from Objects365 + Things [4]; (3) using class names in (2) plus noun phrases extracted from YFCC100m [12] dataset. We equip our DetCLIP-T(B) with these three concept dictionaries and compare their performance in Table 7. It can be seen that using a small size dictionary, e.g., Objects365 + Things, can even bring a performance drop compared to without the dictionary, while scaling up the dictionary with nouns from YFCC can significantly improve the performance. We speculate this is because a large dictionary can provide rich negative concepts for grounding data, encouraging the model to learn more discriminative features.

Table 7: Performance comparison of using concept dictionaries with different sizes. Zero-shot transfer performance on LVIS [3] dataset is reported. The first row is the results without the concept dictionary. The numbers in parentheses indicate the size of the corresponding concept dictionary. Scaling up the dictionary benefits the learning.

| Model | Concept Dictionary | LVIS minival AP ($AP_r$ / $AP_c$ / $AP_f$) | LVIS val AP ($AP_r$ / $AP_c$ / $AP_f$) |
|---|---|---|---|
| DetCLIP-T(B) | / | 28.2 (21.6 / 25.0 / 32.2) | 20.9 (15.3 / 17.5 / 27.1) |
| | O365 (~0.36K) | 27.8 (22.3 / 23.7 / 32.4) | 21.6 (19.3 / 18.7 / 25.8) |
| | O365+Things (~1.9K) | 28.1 (20.8 / 24.8 / 32.4) | 20.5 (14.0 / 17.0 / 27.2) |
| | Detection + Image-text (~14K) | **34.4** (**26.9** / **33.9** / **36.3**) | **27.2** (**21.9** / **25.5** / **31.5**) |

**Important Role of Class Definition.** In DetCLIP, we augment the class names in the detection dataset with their definitions during both training and inference stage, which is termed as **concept**

**enrichment**. To verify that DetCLIP learns knowledge from class definitions, we compare the performances of including/excluding definitions in text input during the inference stage. Table 8 reports the results. It can be found that adding definitions to class names can significantly improve the zero-shot transfer performance.

Table 8: Effects of concept enrichment during the inference phrase. Text inputs with and without class definition are studied. Zero-shot transfer performance on LVIS [3] dataset is reported. The class definition helps detector better recognize objects.

| MODEL | TEXT INPUT | LVIS MINIVAL AP ($AP_r$ / $AP_c$ / $AP_f$) | LVIS VAL AP ($AP_r$ / $AP_c$ / $AP_f$) |
|---|---|---|---|
| DETCLIP-T(B) | CLASS NAMES | 30.4 (22.4 / 27.1 / 34.8) | 23.2 (14.3 / 20.7 / 29.9) |
| | CLASS NAMES + DEF. | **34.4 (26.9 / 33.9 / 36.3)** | **27.2 (21.9 / 25.5 / 31.5)** |

**Impact of pre-trained language models in Concept enrichment.** During training, we use a pre-trained language model to retrieve a definition in our dictionary for concepts without a direct match in WordNet. We conduct experiments to study how the pre-trained language model in the this process affects the final performance. Three different settings are considered: 1. do not use language model, i.e., directly adopt the category name as the input for the concepts not in WordNet; 2. use a pre-trained FILIP text encoder; and 3. use a pre-trained RoBERTa as in GLIP. The results are shown in the Table 9. We can observe that: 1) the concept enrichment procedure can bring significant improvements, (e.g., +3.6% on rare categories) even without using a pre-trained language model; 2) using FILIP can further boost the AP performance from 28.3 to 28.8, while using RoBERTa achieves similar performance with no language model is used.

Table 9: Performance comparison of using different pre-trained text encoders in concept enrichment procedure on LVIS minival dataset. The training dataset is Objects365.

| CONCEPTS ENRICHMENT | PRE-TRAINED TEXT ENCODER | LVIS MINIVAL |
|---|---|---|
| ✗ | / | 27.8 (22.2/26.8/29.7) |
| ✓ | NONE | 28.3 (25.8/27.0/29.9) |
| ✓ | ROBERTA-BASE | 28.2 (24.5/27.3/29.7) |
| ✓ | FILIP TEXT-ENCODER | **28.8 (26.0/28.0/30.0)** |

**Other Important Training Techniques.** Training a vision-language model that works for the open-world detection task is not easy. We highlight two important training techniques we found in our experiments: (1) using a small learning rate for the pre-trained language backbone, since it helps maintain the language model's knowledge learnt in the large-scale pretraining; and (2) removing the regression loss for non-detection data, since it helps alleviate the negative impact caused by inaccurate localization annotation of grounding/image-text pair data. Table 10 provides the ablation studies of these techniques.

Table 10: Important techniques for training DetCLIP. The learning rate of the image encoder is set to 2.8e-4. Models in this table use sequential formulation as in GLIP [9], since these experiments are conducted during our early-stage exploration.

| PRE-TRAIN DATA | LR (LANG. MODEL) | REG. LOSS | LVIS (MINIVAL) |
|---|---|---|---|
| O365 | 2.8E-4 | DET. | 15.9 (7.00/11.3/21.5) |
| O365 | 2.8E-5 | DET. | **23.7** (**16.6/20.5/27.7**) |
| O365, GOLDG | 2.8E-4 | DET | 22.3 (14.5/17.9/27.6) |
| O365, GOLDG | 2.8E-5 | DET. GOLDG. | 22.9 (15.3/21.5/25.6) |
| O365, GOLDG | 2.8E-5 | DET | **26.0** (**18.0/22.8/30.3**) |

## E Qualitative Results

**More visualizations of pseudo labels with concept dictionary.** Fig. 1 shows extra examples of YFCC data that pseudo labeled with the concept dictionary, as well as their comparisons with the results generated by using the original caption. Concept dictionary alleviates partial-label problem and helps CLIP model provide finer-grained and higher quality pseudo labels.

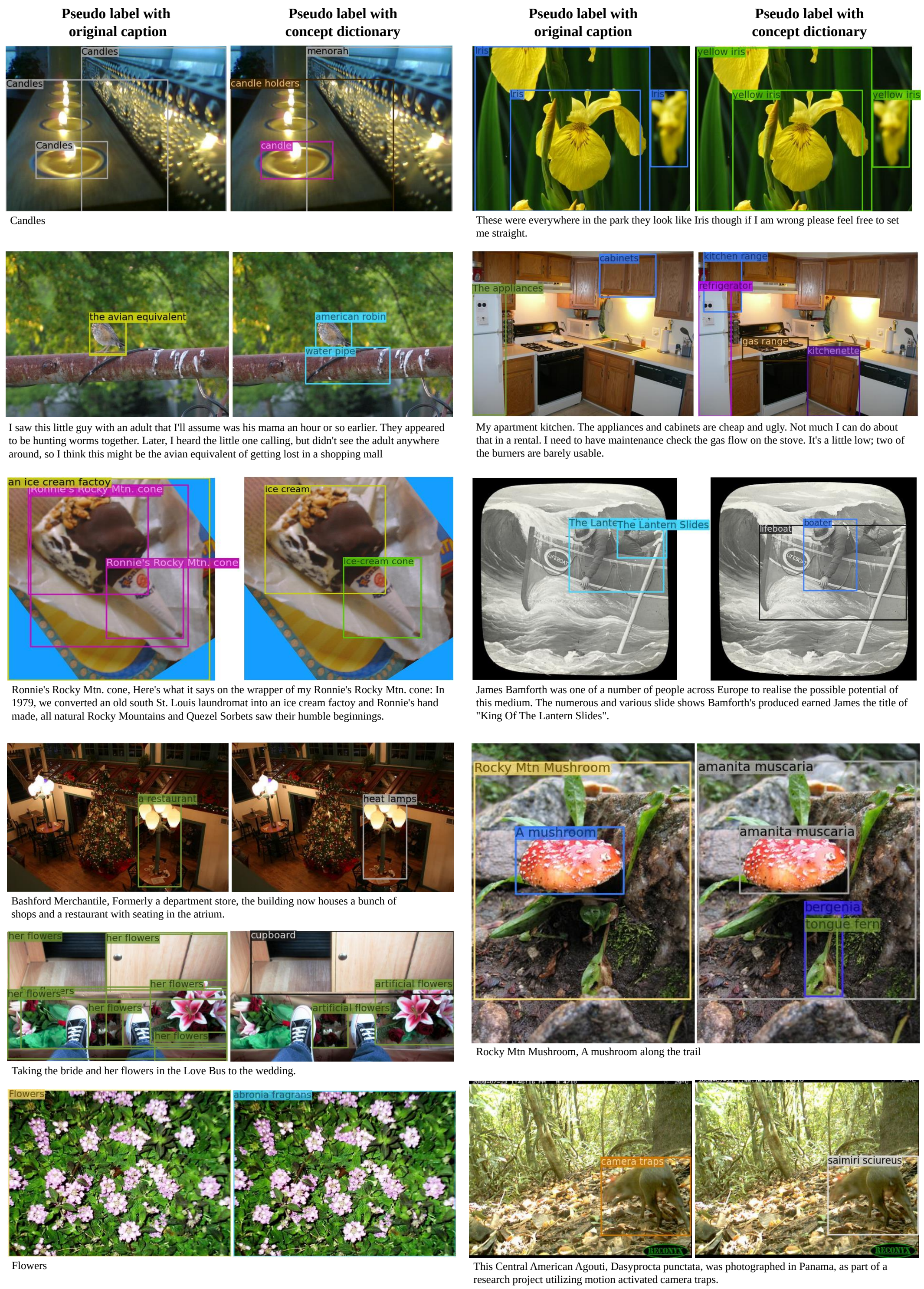


Figure 1: More visualizations of pseudo label results on YFCC [12] dataset. The texts below the images are the corresponding captions. Concept dictionary helps produce higher-quality pseudo labels.

**Retrieval with Concept Dictionary.** In our **concept enrichment**, to augment a given class name with its definition, we retrieve it in the constructed concept dictionary. If there is an exact match, we directly use the corresponding definition; otherwise, we use semantic similarity computed by a pre-trained language model to find the closest one. Table 11 illustrates some example retrieval results. For class names that are not contained in the dictionary, our method can find proper synonyms.

Table 11: Results of retrieval with the proposed concept dictionary. Class names of Objects365 are used as the queries. Our method can retrieve proper synonyms for class names not contained in the dictionary.

| Query Class Name | Retrieved Concept | Definition |
|---|---|---|
| Leather Shoes | Boot | Footwear that covers the whole foot and lower leg. |
| High Heels | Stiletto | A woman's shoe with a thin, high tapering heel. |
| Machinery Vehicle | Truck | An automotive vehicle suitable for hauling. |
| Cosmetics Mirror | Mirror | Polished surface that forms images by reflecting light. |
| Induction Cooker | Hotplate | A portable electric appliance for heating or cooking or keeping food warm. |
| Hoverboard | Rollerblade | Trademark an in line skate. |

**Illustrations of Concept Dictionary.** We illustrate some examples in our concept dictionary in Table 12. We observe that the concepts collected from the image-text pair data can cover more fine-grained categories (e.g., cotswold, cuniculus paca) and a wider range of classes (e.g., giant, cathedral).

# F Other issues

**Clarification.** The "ACLIP" in Fig.1 (in the main paper) should be "DetCLIP".

Table 12: Examples of our concept dictionary. The upper part of the table shows the concepts collected from the detection datasets, while the lower part shows the concepts collected from the image-text pair dataset, i.e., YFCC100M [12].

| Concept | Definition |
|---|---|
| Collected from Large-scale Detection Datasets. | |
| Cup | A small open container usually used for drinking; usually has a handle. |
| Chips | strips of potato fried in deep fat. |
| Chair | A seat for one person, with a support for the back. |
| Eraser | an implement used to erase something. |
| Gloves | The handwear used by fielders in playing baseball. |
| Recorder | equipment for making records. |
| Street Lights | A lamp supported on a lamppost; for illuminating a street. |
| Collected from Image-text Pair Data. | |
| Pet | A domesticated animal kept for companionship or amusement. |
| Pod | The vessel that contains the seeds of a plant (not the seeds themselves). |
| Taro | Edible starchy tuberous root of taro plants. |
| Shrub | A low woody perennial plant usually having several major stems. |
| Salmon | Any of various large food and game fishes of northern waters. |
| Brewery | A plant where beer is brewed by fermentation. |
| Pottery | Ceramic ware made from clay and baked in a kiln. |
| Giant | Any creature of exceptional size.giant and any creature of exceptional size. |
| Pagoda | An Asian temple; usually a pyramidal tower with an upward curving roof. |
| Fresco | A mural done with watercolors on wet plaster. |
| Wildflower | Wild or uncultivated flowering plant. |
| Water tower | A large reservoir for water. |
| Basin | A bowl-shaped vessel; usually used for holding food or liquids. |
| Cotswold | Sheep with long wool originating in the Cotswold Hills. |
| Insect | Small air-breathing arthropod. |
| Booth | A table (in a restaurant or bar) surrounded by two high-backed benches. |
| Office | Place of business where professional or clerical duties are performed. |
| Cab | A compartment at the front of a motor vehicle or locomotive where driver sits. |
| Gable | The vertical triangular wall between the sloping ends of gable roof. |
| Hotel | A building where travelers can pay for lodging and meals and other services. |
| Cathedral | Any large and important church. |
| Restaurant | A building where people go to eat. |
| Library | A room where books are kept. |
| Courtyard | An area wholly or partly surrounded by walls or buildings. |
| Footbridge | A bridge designed for pedestrians. |
| Cuniculus paca | Large burrowing rodent of South America and Central America. |

## References


[1] Xiyang Dai, Yinpeng Chen, Bin Xiao, Dongdong Chen, Mengchen Liu, Lu Yuan, and Lei Zhang. Dynamic head: Unifying object detection heads with attentions. In *CVPR*, pages 7373–7382, 2021. 3

[2] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2, 2021. C, 5

[3] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, pages 5356–5364, 2019. C, 3, 6, 7, 8

[4] Martin N Hebart, Adam H Dickter, Alexis Kidder, Wan Y Kwok, Anna Corriveau, Caitlin Van Wicklin, and Chris I Baker. Things: A database of 1,854 object concepts and more than 26,000 naturalistic object images. *PloS one*, 14(10):e0223792, 2019. D

[5] 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 1780–1790, 2021. 1

[6] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, et al. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github. com/openimages*, 2(3):18, 2017. 1

[7] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2016. 1

[8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, volume 25, pages 1097–1105, 2012. 1

[9] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. *arXiv preprint arXiv:2112.03857*, 2021. B, B, 1, C, 3, C, 4, D, 10

[10] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. B

[11] 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 8430–8439, 2019. 1, 6, D

[12] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016. A, B, 1, D, 1, 12