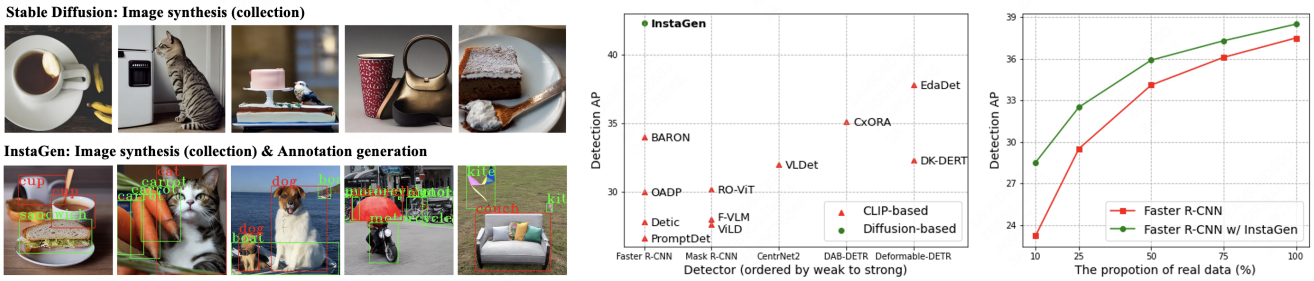
InstaGen: Enhancing Object Detection by Training on Synthetic Dataset

Chengjian Feng[1] Yujie Zhong[1] Zequn Jie[1,†] Weidi Xie[2,†] Lin Ma[1]
[1] Meituan Inc. [2] CMIC, Shanghai Jiao Tong University
https://fcjian.github.io/InstaGen



(a) Demonstration of the synthetic images generated from Stable Diffusion and InstaGen with text prompt: 'a photograph of [base category] and [novel category]'.

(b) The detection AP of novel categories of different paradigms on COCO-OVD benchmarks.

(c) The detection AP with respect to the portion (%) of real data (COCO) usage.

Figure 1. (a) The synthetic images generated from **Stable Diffusion** and our proposed **InstaGen**, which can serve as a *dataset synthesizer* for sourcing photo-realistic images and instance bounding boxes at scale. (b) On open-vocabulary detection, training on synthetic images demonstrates significant improvement over CLIP-based methods on novel categories. (c) Training on the synthetic images generated from InstaGen also enhances the detection performance in close-set scenario, particularly in data-sparse circumstances.

Abstract

In this paper, we present a novel paradigm to enhance the ability of object detector, e.g., *expanding categories or improving detection performance, by training on* ***synthetic dataset*** *generated from diffusion models. Specifically, we integrate an instance-level grounding head into a pre-trained, generative diffusion model, to augment it with the ability of localising instances in the generated images. The grounding head is trained to align the text embedding of category names with the regional visual feature of the diffusion model, using supervision from an off-the-shelf object detector, and a novel self-training scheme on (novel) categories not covered by the detector. We conduct thorough experiments to show that, this enhanced version of diffusion model, termed as* **InstaGen**, *can serve as a data synthesizer, to enhance object detectors by training on its generated samples, demonstrating superior performance over existing state-of-the-art methods in open-vocabulary (*$+4.5$ *AP) and data-sparse (*$+1.2 \sim 5.2$ *AP) scenarios.*

1. Introduction

Object detection has been extensively studied in the field of computer vision, focusing on the localization and categorization of objects within images [3, 5, 12, 26, 27]. The common practise is to train the detectors on large-scale image datasets, such as MS-COCO [20] and Object365 [30], where objects are exhaustively annotated with bounding boxes and corresponding category labels. However, the procedure for collecting images and annotations is often laborious and time-consuming, limiting the datasets' scalability.

In the recent literature, text-to-image diffusion models have demonstrated remarkable success in generating high-quality images [28, 29], that unlocks the possibility of training vision systems with synthetic images. In general, existing text-to-image diffusion models are capable of synthesizing images based on some free-form text prompt, as shown in the first row of Figure 1a. Despite being photo-realistic, such synthesized images *can not* support training sophisticated systems, that normally requires the inclusion of instance-level annotations, *e.g.*, bounding boxes for object detection in our case. In this paper, we investigate a novel paradigm of *dataset synthesis* for training object detector, *i.e.*, augmenting the text-to-image diffusion model to generate instance-level bounding boxes along with images.

To begin with, we build an image synthesizer by fine-tuning the diffusion model on existing detection dataset. This is driven by the observation that off-the-shelf diffusion models often generate images with only one or two objects on simplistic background, training detectors on such images may thus lead to reduced robustness in complex real-world scenarios. Specifically, we exploit the existing detection dataset, and subsequently fine-tune the diffusion model with the image-caption pairs, constructed by taking

†: corresponding author.

random image crops, and composing the category name of the objects in the crop. As illustrated in the second row of the Figure 1a, once finetuned, the image synthesizer now enables to produce images with multiple objects and intricate contexts, thereby providing a more accurate simulation of real-world detection scenarios.

To generate bounding boxes for objects within synthetic images, we propose an instance grounding module that establishes the correlation between the regional visual features from diffusion model and the text embedding of category names, and infers the coordinates for the objects' bounding boxes. Specifically, we adopt a two-step training strategies, *firstly*, we train the grounding module on synthetic images, with the supervision from an off-the-shelf object detector, which has been trained on a set of base categories; *secondly*, we utilize the trained grounding head to generate pseudo labels for a larger set of categories, including those not seen in existing detection dataset, and self-train the grounding module. Once finished training, the grounding module will be able to identify the objects of arbitrary category and their bounding boxes in the synthetic image, by simply providing the name in free-form language.

To summarize, we explore a novel approach to enhance object detection capabilities, such as expanding detectable categories and improving overall detection performance, by training on *synthetic dataset* generated from diffusion model. We make the following contribution: (i) We develop an image synthesizer by fine-tuning the diffusion model, with image-caption pairs derived from existing object detection datasets, our synthesizer can generate images with multiple objects and complex contexts, offering a more realistic simulation for real-world detection scenarios. (ii) We introduce a data synthesis framework for detection, termed as **InstaGen**. This is achieved through a novel grounding module that enables to generate labels and bounding boxes for objects in synthetic images. (iii) We train standard object detectors on the combination of *real and synthetic* dataset, and demonstrate superior performance over existing state-of-the-art detectors across various benchmarks, including open-vocabulary detection (increasing Average Precision [AP] by +4.5), data-sparse detection (enhancing AP by +1.2 to +5.2), and cross-dataset transfer (boosting AP by +0.5 to +1.1).

2. Related Work

Object Detection. Object detection aims to simultaneously predict the category and corresponding bounding box for the objects in the images. Generally, object detectors [3, 4, 6, 26, 27] are trained on a substantial amount of training data with bounding box annotations and can only recognize a predetermined set of categories present in the training data. In the recent literature, to further expand the ability of object detector, open-vocabulary object detection (OVD) has been widely researched, for example, OVR-CNN [37] introduces the concept of OVD and pre-trains a vision-language model with image-caption pairs. The subsequent works make use of the robust multi-modal representation of CLIP [24], and transfer its knowledge to object detectors through knowledge distillation [9, 36], exploiting extra data [5, 41] and text prompt tuning [2, 5]. In this paper, we propose to expand the ability of object detectors, *e.g.*, expanding categories or improving detection performance, by training on synthetic dataset.

Generative Models. Image generation has been considered as a task of interest in computer vision for decades. In the recent literature, significant progress has been made, for example, the generative adversarial networks (GANs) [8], variational autoencoders (VAEs) [15], flow-based models [14], and autoregressive models (ARMs) [32]. More recently, there has been a growing research interest in diffusion probabilistic models (DPMs), which have shown great promise in generating high-quality images across diverse datasets. For examples, GLIDE [23] utilizes a pre-trained language model and a cascaded diffusion structure for text-to-image generation. DALL-E 2 [25] is trained to generate images by inverting the CLIP image space, while Imagen [29] explores the advantages of using pre-trained language models. Stable Diffusion [28] proposes the diffusion process in VAE latent spaces rather than pixel spaces, effectively reducing resource consumption. In general, the rapid development of generative models opens the possibility for training large models with synthetic dataset.

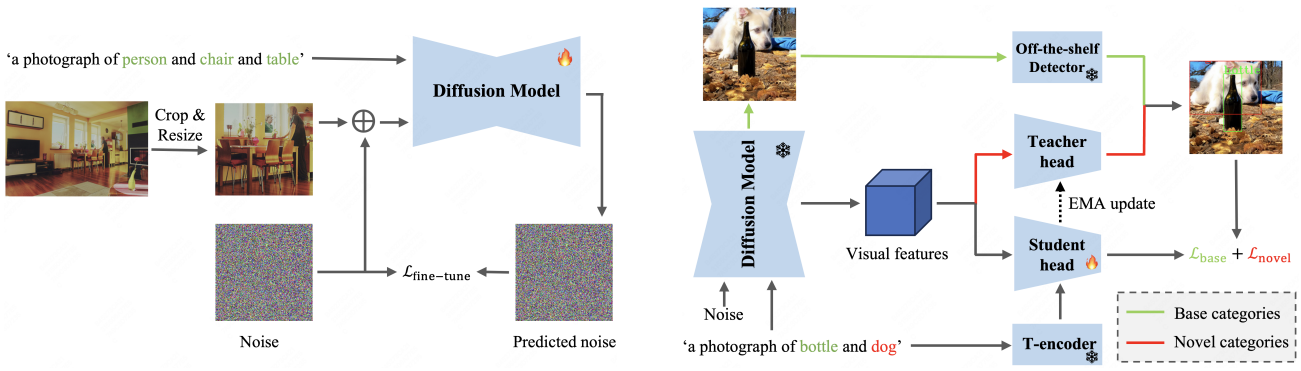
3. Methodology

In this section, we present details for constructing a *dataset synthesizer*, that enables to generate photo-realistic images with bounding boxes for each object instance, and train an object detector on the combined real and synthetic datasets.

3.1. Problem Formulation

Given a detection dataset of real images with manual annotations, *i.e.*, $\mathcal{D}_{\text{real}} = \{(x_1, \mathcal{B}_1, \mathcal{Y}_1), \ldots, (x_N, \mathcal{B}_N, \mathcal{Y}_N)\}$, where $\mathcal{B}_i = \{b_1, \ldots, b_m | b_j \in \mathbb{R}^{2\times 2}\}$ denotes the set of box coordinates for the annotated instances in one image, and $\mathcal{Y}_i = \{y_1, \ldots, y_m | y_j \in \mathcal{R}^{\mathcal{C}_{\text{base}}}\}$ refers to the categories of the instances. Our goal is thus to exploit the given real dataset ($\mathcal{D}_{\text{real}}$), to steer a generative diffusion model into *dataset synthesizer*, that enables to augment the existing detection dataset, *i.e.*, $\mathcal{D}_{\text{final}} = \mathcal{D}_{\text{real}} + \mathcal{D}_{\text{syn}}$. As a result, detectors trained on the combined dataset demonstrate enhanced ability, *i.e.*, extending the detection categories or improving the detection performance.

In the following sections, we first describe the procedure for constructing an *image synthesizer*, that can generate images suitable for training object detector (Section 3.2). To



(a) Fine-tuning diffusion model on detection dataset.

(b) Supervised training and self-training for grounding head (*i.e.* student).

Figure 2. Illustration of the process for finetuning diffusion model and training the grounding head: (a) stable diffusion model is fine-tuned on the detection dataset on base categories. (b) The grounding head is trained on synthetic images, with supervised learning on base categories and self-training on novel categories.

simultaneously generate the images and object bounding boxes, we propose a novel instance-level grounding module, which aligns the text embedding of category name with the regional visual features from *image synthesizer*, and infers the coordinates for the objects in synthetic images. To further improve the alignment towards objects of arbitrary category, we adopt self-training to tune the grounding module on object categories not existing in $\mathcal{D}_{\text{real}}$ (Section 3.3). As a result, the proposed model, termed as **InstaGen**, can automatically generate images along with bounding boxes for object instances, and construct *synthetic dataset* ($\mathcal{D}_{\text{syn}}$) at scale, leading to improved ability when training detectors on it (Section 3.4).

3.2. Image Synthesizer for Object Detection

Here, we build our *image synthesizer* based on an off-the-shelf stable diffusion model (SDM [28]). Despite of its impressive ability in generating photo-realistic images, it often outputs images with only one or two objects on simplistic background with the text prompts, for example, 'a photograph of a [category1 name] and a [category2 name]', as demonstrated in Figure 4b. As a result, object detectors trained on such images may exhibit reduced robustness when dealing with complex real-world scenarios. To bridge such domain gap, we propose to construct the *image synthesizer* by fine-tuning the SDM with an existing real-world detection dataset ($\mathcal{D}_{\text{real}}$).

Fine-tuning procedure. To fine-tune the stable diffusion model (SDM), one approach is to naïvely use the sample from detection dataset, for example, randomly pick an image and construct the text prompt with all categories in the image. However, as the image often contains multiple objects, such approach renders significant difficulty for fine-tuning the SDM, especially for small or occluded objects. We adopt a mild strategy by taking random crops from the images, and construct the text prompt with categories in the image crops, as shown in Figure 2a. If an image crop contains multiple objects of the same category, we only use this category name once in the text prompt.

Fine-tuning loss. We use the sampled image crop and constructed text prompt to fine-tune SDM with a squared error loss on the predicted noise term as follows:

$$\mathcal{L}_{\text{fine-tune}} = \mathbb{E}_{z,\epsilon\sim\mathcal{N}(0,1),t,y}\left[||\epsilon - \epsilon_\theta(z^t, t, y)||_2^2\right], \quad (1)$$

where z denotes a latent vector mapped from the input image with VAE, t denotes the denoising step, uniformly sampled from $\{1, \ldots, T\}$, T refers to the length of the diffusion Markov chain, and ϵ_θ refers to the estimated noise from SDM with parameters θ being updated. We have experimentally verified the necessity of this fine-tuning step, as shown in Table 4.

3.3. Dataset Synthesizer for Object Detection

In this section, we present details for steering the *image synthesizer* into *dataset synthesizer* for object detection, which enables to simultaneously generate images and object bounding boxes. Specifically, we propose an instance-level grounding module that aligns the text embedding of object category, with the regional visual feature of the diffusion model, and infers the coordinates for bounding boxes, effectively augmenting the *image synthesizer* with instance grounding, as shown in Figure 3. To further improve the alignment in large visual diversity, we propose a self-training scheme that enables the grounding module to generalise towards arbitrary categories, including those not exist in real detection dataset ($\mathcal{D}_{\text{real}}$). As a result, our *data synthesizer*, termed as **InstaGen**, can be used to construct synthetic dataset for training object detectors.

3.3.1 Instance Grounding on Base Categories

To localise the object instances in synthetic images, we introduce an open-vocabulary grounding module, that aims to simultaneously generate image (x) and the corresponding instance-level bounding boxes ($\mathcal{B}$) based on a set of

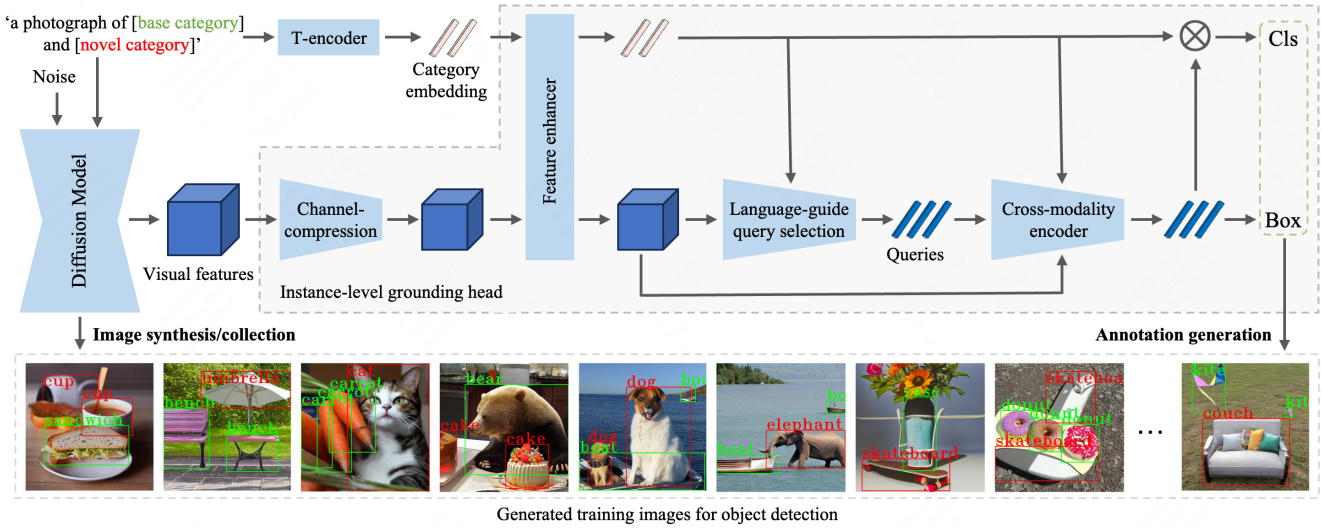


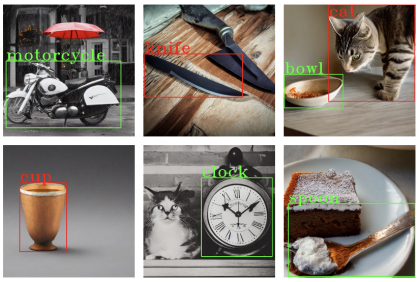
Figure 3. Illustration of the dataset generation process in InstaGen. The data generation process consists of two steps: (i) **Image collection**: given a text prompt, SDM generates images with the objects described in the text prompt; (ii) **Annotation generation**: the instance-level grounding head aligns the category embedding with the visual feature region of SDM, generating the corresponding object bounding-boxes.

categories ($\mathcal{Y}$), *i.e.*, $\{x, \mathcal{B}, \mathcal{Y}\} = \Phi_{\text{InstaGen}}(\epsilon, \mathcal{Y})$, where $\epsilon \sim \mathcal{N}(0, I)$ denotes the sampled noise.

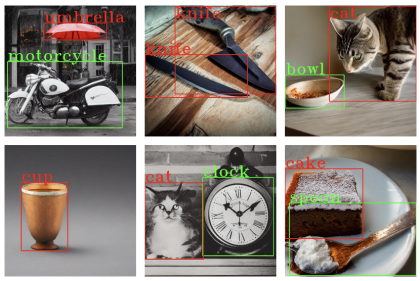
To this end, we propose an instance grounding head, as shown in Figure 3, it takes the intermediate representation from *image synthesizer* and the text embedding of category as inputs, then predicts the corresponding object bounding boxes, *i.e.*, $\{\mathcal{B}_i, \mathcal{Y}_i\} = \Phi_{\text{g-head}}(\mathcal{F}_i, \Phi_{\text{t-enc}}(g(\mathcal{Y}_i)))$, where $\mathcal{F}_i = \{f_i^1, \ldots, f_i^n\}$ refers to the multi-scale dense features from the *image synthesizer* at time step $t = 1$, $g(\cdot)$ denotes a template that decorates each of the visual categories in the text prompt, *e.g.*, 'a photograph of [category1 name] and [category2 name]', $\Phi_{\text{t-enc}}(\cdot)$ denotes the text encoder.

Inspired by GroundingDINO [22], our grounding head $\Phi_{\text{g-head}}(\cdot)$ mainly contains four components: (i) a channel-compression layer, implemented with a 3×3 convolution, for reducing the dimensionality of the visual features; (ii) a feature enhancer, consisting of six feature enhancer layers, to fuse the visual and text features. Each layer employs a deformable self-attention to enhance image features, a vanilla self-attention for text feature enhancers, an image-to-text cross-attention and a text-to-image cross-attention for feature fusion; (iii) a language-guided query selection module for query initialization. This module predicts top-N anchor boxes based on the similarity between text features and image features. Following DINO [38], it adopts a mixed query selection where the positional queries are initialized with the anchor boxes and the content queries remain learnable; (iv) a cross-modality decoder for classification and box refinement. It comprises six decoder layers, with each layer utilizing a self-attention mechanism for query interaction, an image cross-attention layer for combining image features, and a text cross-attention layer for combining text features. Finally, we apply the dot product between each query and the text features, followed by a Sigmoid function to predict the classification score $\hat{s}$ for each category. Additionally, the object queries are passed through a Multi-Layer Perceptron (MLP) to predict the object bounding boxes $\hat{b}$, as shown in Figure 3. We train the grounding head by aligning the category embedding with the regional visual features from diffusion model, as detailed below. *Once trained, the grounding head is open-vocabulary*, *i.e.*, given any categories (even beyond the training categories), the grounding head can generate the corresponding bounding-boxes for the object instances.

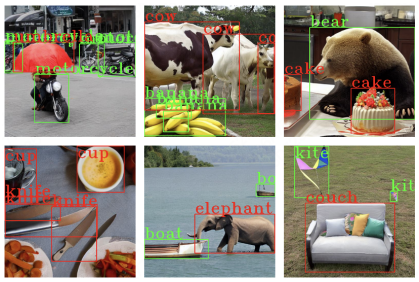
Training triplets of base categories. Following [18], we apply an automatic pipeline to construct the {visual feature, bounding-box, text prompt} triplets, with an object detector trained on base categories from a given dataset ($\mathcal{D}_{\text{real}}$). In specific, assuming there exists a set of base categories $\{c_{\text{base}}^1, \ldots, c_{\text{base}}^N\}$, *e.g.*, the classes in MS-COCO [20]. We first select a random number of base categories to construct a text prompt, *e.g.*, 'a photograph of [base category1] and [base category2]', and generate both the visual features and images with our *image synthesizer*. Then we take an off-the-shelf object detector, for example, pre-trained Mask R-CNN [12], to run the inference procedure on the synthetic images, and infer the bounding boxes of the selected categories. To acquire the confident bounding-boxes for training, we use a score threshold α to filter out the bounding-boxes with low confidence (an ablation study on the selection of the score threshold has been conducted in Section 4.5). As a result, an infinite number of training triplets for the given base categories can be constructed by repeating the above operation.



(a) Stable Diffusion + Grounding head w/ **Supervised training**.



(b) Stable Diffusion + Grounding head w/ Supervised- and **Self-training**.



(c) Stable Diffusion w/ **Fine-tuning** + Grounding head w/ Supervised- and Self-training.

Figure 4. Visualization of the synthetic images and bounding-boxes generated from different models. The bounding-boxes with green denote the objects from base categories, while the ones with red denote the objects from novel categories.

Training loss. We use the constructed training triplets to train the grounding head:

$$\mathcal{L}_{\text{base}} = \sum_{i=1}^{N} [\mathcal{L}_{\text{cls}}(\hat{s}_i, c_i) + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}}(\hat{b}_i, b_i)], \quad (2)$$

where the ith prediction $(\hat{s}_i, \hat{b}_i)$ from the N object queries is assigned to a ground-truth (c_i, b_i) or $\varnothing$ (no object) with bipartite matching. $\mathcal{L}_{\text{cls}}$ and $\mathcal{L}_{\text{box}}$ denote the classification loss (*e.g.* Focal loss) and box regression loss (*e.g.* L1 loss and GIoU loss), respectively.

3.3.2 Instance Grounding on Novel Categories

Till here, we have obtained a diffusion model with open-vocabulary grounding, which has been only trained with base categories. In this section, we propose to further leverage the synthetic training triplets from a wider range of categories to enhance the alignment for novel/unseen categories. Specifically, as shown in Figure 2b, we describe a framework that generates the training triplets for novel categories using the grounded diffusion model, and then self-train the grounding head.

Training triplets of novel categories. We design the text prompts of novel categories, *e.g.*, 'a photograph of [novel category1] and [novel category2]', and pass them through our proposed *image synthesizer*, to generate the visual features. To acquire the corresponding bounding-boxes for novel categories, we propose a self-training scheme that takes the above grounding head as the student, and apply a mean teacher (an exponential moving average (EMA) of the student model) to create pseudo labels for update. In contrast to the widely adopted self-training scheme that takes the image as input, the student and teacher in our case only take the visual features as input, thus *cannot* apply data augmentation as for images. Instead, we insert dropout module within each feature enhancer layer and decoder layer in the student. During training, we run inference (without dropout module) with teacher model on the visual features to produce bounding boxes, and then use a score threshold β to filter out those with low confidence, and use the remaining training triplets $(\mathcal{F}_i, \hat{b}_i, y_i^{\text{novel}})$ to train the student, *i.e.*, grounding head.

Training loss. Now, we can also train the grounding head on the mined triplets of novel categories (that are unseen in the existing real dataset) with the training loss $\mathcal{L}_{\text{novel}}$ defined similar to Eq. 2. Thus, the total training loss for training the grounding head can be: $\mathcal{L}_{\text{grounding}} = \mathcal{L}_{\text{base}} + \mathcal{L}_{\text{novel}}$.

3.4. Training Detector with Synthetic Dataset

In this section, we augment the real dataset ($\mathcal{D}_{\text{real}}$), with synthetic dataset ($\mathcal{D}_{\text{syn}}$), and train popular object detectors, for example, Faster R-CNN [27] with the standard training loss:

$$\mathcal{L}_{\text{det}} = \mathcal{L}_{\text{rpn\_cls}} + \mathcal{L}_{\text{rpn\_box}} + \mathcal{L}_{\text{det\_cls}} + \mathcal{L}_{\text{det\_box}}, \quad (3)$$

where $\mathcal{L}_{\text{rpn\_cls}}$, $\mathcal{L}_{\text{rpn\_box}}$ are the classification and box regression losses of region proposal network, and $\mathcal{L}_{\text{det\_cls}}$, $\mathcal{L}_{\text{det\_box}}$ are the classification and box regression losses of the detection head. Generally speaking, the synthetic dataset enables to improve the detector's ability from two aspects: (i) expanding the original data with more categories, (ii) improve the detection performance by increasing data diversity.

Expanding detection categories. The grounding head is designed to be open-vocabulary, that enables to generate object bounding boxes for novel categories, even though it is trained with a specific set of base categories. This feature enables **InstaGen** to construct a detection dataset for any category. Figure 4 demonstrates several synthetic images and object bounding boxes for novel categories, *i.e.*, the object with red bounding box. We evaluate the effectiveness of training on synthetic dataset through experiments on open-vocabulary detection benchmark. For more details, please refer to Figure 1b and Section 4.2.

Increasing data diversity. The base diffusion model is trained on a large corpus of image-caption pairs, that enables to generate diverse images. Taking advantage of such capabilities, **InstaGen** is capable of generating dataset with diverse images and box annotations, which can expand the

Method	Supervision	Detector	Backbone	$AP50^{box}_{all}$	$AP50^{box}_{base}$	$AP50^{box}_{novel}$
Detic [41]	CLIP	Faster R-CNN	R50	45.0	47.1	27.8
PromptDet [5]	CLIP	Faster R-CNN	R50	-	50.6	26.6
BARON [34]	CLIP	Faster R-CNN	R50	53.5	60.4	34.0
OADP [33]	CLIP	Faster R-CNN	R50	47.2	53.3	30.0
ViLD [9]	CLIP	Mask R-CNN	R50	51.3	59.5	27.6
F-VLM [16]	CLIP	Mask R-CNN	R50	39.6	-	28.0
RO-ViT [13]	CLIP	Mask R-CNN	ViT-B [1]	41.5	-	30.2
VLDet [19]	CLIP	CenterNet2 [40]	R50	45.8	50.6	32.0
CxORA [35]	CLIP	DAB-DETR [21]	R50	35.4	35.5	35.1
DK-DETR [17]	CLIP	Deformable DETR [42]	R50	-	61.1	32.3
EdaDet [31]	CLIP	Deformable DETR [42]	R50	52.5	57.7	37.8
InstaGen	Stable Diffusion	Faster R-CNN	R50	52.3	55.8	**42.3**

Table 1. Results on open-vocabulary COCO benchmark. $AP50^{box}_{novel}$ is the main metric for evaluation. Our detector, trained on synthetic dataset from **InstaGen**, significantly outperforms state-of-the-art CLIP-based approaches on novel categories.

original dataset, *i.e.*, increase the data diversity and improve detection performance, particularly in data-sparse scenarios. We conducted experiments with varying proportions of COCO [20] images as available real data, and show the effectiveness of training on synthetic dataset when the number of real-world images is limited. We refer the readers for more details in Section 4.3, and results in Figure 1c.

4. Experiment

In this section, we use the proposed **InstaGen** to construct synthetic dataset for training object detectors, *i.e.*, generating images with the corresponding bounding boxes. Specifically, we present the implementation details in Section 4.1. To evaluate the effectiveness of the synthetic dataset for training object detector, we consider three protocols: open-vocabulary object detection (Section 4.2), data-sparse object detection (Section 4.3) and cross-dataset object detection (Section 4.4). Lastly, we conduct ablation studies on the effectiveness of the proposed components and the selection of hyper-parameters (Section 4.5).

4.1. Implementation details

Network architecture. We build *image synthesizer* from the pre-trained Stable Diffusion v1.4 [28], and use the CLIP text encoder [24] to get text embedding for the category name. The channel compression layer maps the dimension of visual features to 256, which is implemented with a 3×3 convolution. For simplicity, the feature enhancer, language-guided query selection module and cross-modality decoder are designed to the same structure as the ones in [22]. The number of the object queries is set to 900.

Constructing image synthesizer. In our experiments, we first fine-tune the stable diffusion model on a real detection dataset, *e.g.*, the images of base categories. During training, the text encoder of CLIP is kept frozen, while the remaining components are trained for 6 epochs with a batch size of 16 and a learning rate of 1e-4.

Instance grounding module. We start by constructing the training triplets using base categories *i.e.*, the categories present in the existing dataset. The text prompt for each triplet is constructed by randomly selecting one or two categories. The regional visual features are taken from the *image synthesizer* time step $t = 1$, and the oracle ground-truth bounding boxes are obtained using a Mask R-CNN model trained on base categories, as explained in Section 3.3.1.

Subsequently, we train the instance grounding module with these training triplets for 6 epochs, with a batch size of 32. In the 6th epoch, we transfer the weights from the student model to the teacher model, and proceed to train the student for an additional 6 epochs. During this training, the student receives supervised training on the base categories and engages in self-training on novel categories, and the teacher model is updated using exponential moving average (EMA) with a momentum of 0.999. The initial learning rate is set to 1e-4 and is subsequently reduced by a factor of 10 at the 11th epoch, and the score thresholds α and β are set to 0.8 and 0.4, respectively.

Training object detector on combined dataset. In our experiment, we train an object detector (Faster R-CNN [27]) with ResNet-50 [11] as backbone, on a combination of the existing real dataset and the synthetic dataset. Specifically, for synthetic dataset, we randomly select one or two categories at each iteration, construct the text prompts, and feed them as input to generates images along with the corresponding bounding boxes with β of 0.4. Following the standard implementation [27], the detector is trained for 12 epochs ($1\times$ learning schedule) unless specified. The initial learning rate is set to 0.01 and then reduced by a factor of 10 at the 8th and the 11th epochs.

4.2. Open-vocabulary object detection

Experimental setup. Following the previous works [5, 39], we conduct experiments on the open-vocabulary COCO benchmark, where 48 classes are treated as base categories,

InstaGen	10%	25%	50%	75%	100%
✗	23.3	29.5	34.1	36.1	37.5
✓	28.5	32.6	35.8	37.3	38.5

Table 2. Results on data-sparse object detection. We employ Faster R-CNN with the ResNet-50 backbone as the default object detector and evaluate its performance using the AP metric on MS COCO benchmark. Please refer to the text for more details.

Method	Supervision	Detector	Extra Data	Object365	LVIS
Gao *et al.* [7]	CLIP	CenterNet2	✓	6.9	8.0
VL-PLM [39]	CLIP	Mask R-CNN	✓	10.9	22.2
InstaGen	Stable Diffusion	Faster R-CNN	✗	**11.4**	**23.3**

Table 3. Results on generalizing COCO-base to Object365 and LVIS. All detectors utilize the ResNet-50 backbone. The evaluation protocol follows [7] and reports AP50. Extra data refers to an additional dataset that encompasses objects from the categories within the target dataset. In both experiments, the extra data consists of all the images from COCO, which has covered the majority of categories in Object365 and LVIS.

G-head	ST	FT	$AP50^{box}_{all}$	$AP50^{box}_{base}$	$AP50^{box}_{novel}$
✓			50.6	55.3	37.1
✓	✓		51.1	55.0	40.3
✓	✓	✓	52.3	55.8	42.3

Table 4. The effectiveness of the proposed components. G-head, ST and FT refer to the grounding head, self-training the grounding head and fine-tuning SDM, respectively.

and 17 classes as the novel categories. More results for LVIS can be found in the **supplementary material**. To train the grounding head, we employ 1250 synthetic images per category per training epoch. While for training the object detector, we use 3000 synthetic images per category, along with the original real dataset for base categories. The object detector is trained with input size of 800×800 and scale jitter. The performance is measured by COCO Average Precision at an Intersection over Union of 0.5 (AP50).

Comparison to SOTA. As shown in Table 1, we evaluate the performance by comparing with existing CLIP-based open-vocabulary object detectors. It is clear that our detector trained on synthetic dataset from **InstaGen** outperforms existing state-of-the-art approaches significantly, *i.e.*, around +5AP improvement over the second best. In essence, through the utilization of our proposed open-vocabulary grounding head, **InstaGen** is able to generate detection data for novel categories, enabling the detector to attain exceptional performance. To the best of our knowledge, this is the first work that applies generative diffusion model for dataset synthesis, to tackle open-vocabulary object detection, and showcase its superiority in this task.

4.3. Data-sparse object detection

Experimental setup. Here, we evaluate the effectiveness of synthetic dataset in data-spare scenario, by varying the amount of real data. We randomly select subsets comprising 10%, 25%, 50%, 75% and 100% of the COCO training set, this covers all COCO categories. These subsets are used to fine-tune stable diffusion model for constructing *image synthesizer*, and train a Mask R-CNN for generating oracle ground-truth bounding boxes in synthetic images. We employ 1250 synthetic images per category to train a Faster R-CNN in conjunction with the corresponding COCO subset. The performance is measured by Average Precision [20].

Comparison to baseline. As shown in Table 2, the Faster R-CNN trained with synthetic images achieves consistent improvement across various real training data budgets. Notably, as the availability of real data becomes sparse, synthetic dataset plays even more important role for performance improvement, for instance, it improves the detector by +5.2 AP (23.3→28.5 AP) when only 10% real COCO training subset is available.

4.4. Cross-dataset object detection

Experimental setup. In this section, we assess the effectiveness of synthetic data on a more challenging task, namely cross-dataset object detection. Following [39], we evaluate the COCO-trained model on two unseen datasets: Object365 [30] and LVIS [10]. Specifically, we consider the 48 classes in the open-vocabulary COCO benchmark as the source dataset, while Object365 (with 365 classes) and LVIS (with 1203 classes) serve as the target dataset. When training the instance grounding module, we acquire 1250 synthetic images for base categories from the source dataset, and 100 synthetic images for the category from the target dataset at each training iteration. In the case of training the object detector, we employ 500 synthetic images per category from the target dataset for each training iteration. The detector is trained with input size of 1024×1024 and scale jitter [39].

Comparison to SOTA. The results presented in Table 3 demonstrate that the proposed **InstaGen** achieves superior performance in generalization from COCO-base to Object365 and LVIS, when compared to CLIP-based methods such as [7, 39]. It is worth noting that CLIP-based methods require the generation of pseudo-labels for the categories from the target dataset on COCO images, and subsequently train the detector using these images. These methods necessitate a dataset that includes objects belonging to the categories of the target dataset. In contrast, **InstaGen** possesses the ability to generate images featuring objects of any category without the need for additional datasets, thereby enhancing its versatility across various scenarios.

#Images	$AP50^{box}_{all}$	$AP50^{box}_{base}$	$AP50^{box}_{novel}$
1000	51.6	55.9	39.7
2000	51.7	55.4	41.1
3000	52.3	55.8	42.3

Table 5. Number of generated images.

α	$AP50^{box}_{all}$	$AP50^{box}_{base}$	$AP50^{box}_{novel}$
0.7	51.3	55.1	40.6
0.8	52.3	55.8	42.3
0.9	51.8	55.6	41.1

Table 6. α for bounding-box filtration.

β	$AP50^{box}_{all}$	$AP50^{box}_{base}$	$AP50^{box}_{novel}$
0.3	46.4	53.3	26.9
0.4	52.3	55.8	42.3
0.5	51.2	55.4	39.2

Table 7. β for bounding-box filtration.

4.5. Ablation study

To understand the effectiveness of the proposed components, we perform thorough ablation studies on the open-vocabulary COCO benchmark [20], investigating the effect of fine-tuning stable diffusion model, training instance grounding module, self-training on novel categories. Additionally, we investigate other hyper-parameters by comparing the effectiveness of synthetic images and different score thresholds for base and novel categories.

Fine-tuning diffusion model. We assess the effectiveness of fine-tuning stable diffusion model, and its impact for synthesizing images for training object detector. Figure 4c illustrates that **InstaGen** is capable of generating images with more intricate contexts, featuring multiple objects, small objects, and occluded objects. Subsequently, we employed these generated images to train Faster R-CNN for object detection. The results are presented in Table 4, showing that *image synthesizer* from fine-tuning stable diffusion model delivers improvement detection performance by 2.0 AP (from 40.3 to 42.3 AP).

Instance grounding module. To demonstrate the effectiveness of the grounding head in open-vocabulary scenario, we exclusively train it on base categories. Visualization examples of the generated images are presented in Figure 4a. These examples demonstrate that the trained grounding head is also capable of predicting bounding boxes for instances from novel categories. Leveraging these generated images to train the object detector leads to a 37.1 AP on novel categories, surpassing or rivaling all existing state-of-the-art methods, as shown in Table 1 and Table 4.

Self-training scheme. We evaluate the performance after self-training the grounding head with novel categories. As shown in Table 4, training Faster R-CNN with the generated images of novel categories, leads to a noticeable enhancement in detection performance, increasing from 37.1 to 40.3 AP. Qualitatively, it also demonstrates enhanced recall for novel objects after self-training, as shown in Figure 4b.

Number of synthetic images. We investigate the performance variation while increasing the number of the generated images per category for detector training. As shown in Table 5, when increasing the number of generated images from 1000 to 3000, the detector's performance tends to be increasing monotonically, from 39.7 to 42.3 AP on novel categories, showing the scalability of the proposed training mechanism.

Score thresholds for bounding box filtration. We compare the performance with different score thresholds α and β for filtering bounding boxes on base categories and novel categories, respectively. From the experiment results in Table 6, we observe that the performance is not sensitive to the value of α, and $\alpha = 0.8$ yields the best performance. The experimental results using different β are presented in Table 7. With a low score threshold ($\alpha = 0.3$), there are still numerous inaccurate bounding boxes remaining, resulting in an AP of 26.9 for novel categories. by increasing β to 0.4, numerous inaccurate bounding boxes are filtered out, resulting in optimal performance. Hence, we set $\alpha = 0.8$ and $\beta = 0.4$ in our experiments.

5. Limitation

Using synthetic or artificially generated data in training AI algorithms is a burgeoning practice with significant potential. It can address data scarcity, privacy, and bias issues. However, there remains two limitations for training object detectors with synthetic data, (i) synthetic datasets commonly focus on clean, isolated object instances, which limits the exposure of the detector to the complexities and contextual diversity of real-world scenes, such as occlusions, clutter, varied environmental factors, deformation, therefore, models trained on synthetic data struggle to adapt to real-world conditions, affecting their overall robustness and accuracy, (ii) existing diffusion-based generative model also suffers from long-tail issue, that means the generative model struggles to generate images for objects of rare categories, resulting in imbalanced class representation during training and reduced detector performance for less common objects.

6. Conclusion

This paper proposes a dataset synthesis pipeline, termed as **InstaGen**, that enables to generate images with object bounding boxes for arbitrary categories, acting as a annotation-free approach for constructing large-scale synthetic dataset to train object detector. We have conducted thorough experiments to show the effectiveness of training on synthetic data, on improving detection performance, or expanding the number of detection categories. Significant improvements have been shown in various detection scenarios, including open-vocabulary (+4.5 AP) and data-sparse (+1.2 $\sim$ 5.2 AP) detection.

Acknowledgement. WX is supported by National Key R&D Program of China (No. 2022ZD0161400).

References

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 6

[2] Yu Du, Fangyun Wei, Zihe Zhang, Miaojing Shi, Yue Gao, and Guoqi Li. Learning to prompt for open-vocabulary object detection with vision-language model. In *CVPR*, pages 14084–14093, 2022. 2

[3] Chengjian Feng, Yujie Zhong, Yu Gao, Matthew R Scott, and Weilin Huang. Tood: Task-aligned one-stage object detection. In *ICCV*, pages 3490–3499. IEEE Computer Society, 2021. 1, 2

[4] Chengjian Feng, Yujie Zhong, and Weilin Huang. Exploring classification equilibrium in long-tailed object detection. In *ICCV*, pages 3417–3426, 2021. 2

[5] Chengjian Feng, Yujie Zhong, Zequn Jie, Xiangxiang Chu, Haibing Ren, Xiaolin Wei, Weidi Xie, and Lin Ma. Promptdet: Towards open-vocabulary detection using uncurated images. In *ECCV*, pages 701–717. Springer, 2022. 1, 2, 6

[6] Chengjian Feng, Zequn Jie, Yujie Zhong, Xiangxiang Chu, and Lin Ma. Aedet: Azimuth-invariant multi-view 3d object detection. In *CVPR*, pages 21580–21588, 2023. 2

[7] Mingfei Gao, Chen Xing, Juan Carlos Niebles, Junnan Li, Ran Xu, Wenhao Liu, and Caiming Xiong. Open vocabulary object detection with pseudo bounding-box labels. In *ECCV*, pages 266–282. Springer, 2022. 7

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[9] 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*, 2021. 2, 6

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

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 6

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 1, 4

[13] Dahun Kim, Anelia Angelova, and Weicheng Kuo. Region-aware pretraining for open-vocabulary object detection with vision transformers. In *CVPR*, pages 11144–11154, 2023. 6

[14] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *NeurIPS*, 31, 2018. 2

[15] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2

[16] Weicheng Kuo, Yin Cui, Xiuye Gu, AJ Piergiovanni, and Anelia Angelova. F-vlm: Open-vocabulary object detection upon frozen vision and language models. 2022. 6

[17] Liangqi Li, Jiaxu Miao, Dahu Shi, Wenming Tan, Ye Ren, Yi Yang, and Shiliang Pu. Distilling detr with visual-linguistic knowledge for open-vocabulary object detection. In *ICCV*, pages 6501–6510, 2023. 6

[18] Ziyi Li, Qinye Zhou, Xiaoyun Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. Open-vocabulary object segmentation with diffusion models. In *CVPR*, pages 7667–7676, 2023. 4

[19] Chuang Lin, Peize Sun, Yi Jiang, Ping Luo, Lizhen Qu, Gholamreza Haffari, Zehuan Yuan, and Jianfei Cai. Learning object-language alignments for open-vocabulary object detection. 2022. 6

[20] 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 *ECCV*, pages 740–755. Springer, 2014. 1, 4, 6, 7, 8

[21] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. Dab-detr: Dynamic anchor boxes are better queries for detr. *arXiv preprint arXiv:2201.12329*, 2022. 6

[22] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 4, 6

[23] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 2

[24] 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. pages 8748–8763. PMLR, 2021. 2, 6

[25] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 2

[26] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, pages 779–788, 2016. 1, 2

[27] 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. 1, 2, 5, 6

[28] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 1, 2, 3, 6

[29] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *NeurIPS*, 35:36479–36494, 2022. 1, 2

[30] 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, 7

[31] Cheng Shi and Sibei Yang. Edadet: Open-vocabulary object detection using early dense alignment. In *ICCV*, pages 15724–15734, 2023. 6

[32] Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. *NeurIPS*, 29, 2016. 2

[33] Luting Wang, Yi Liu, Penghui Du, Zihan Ding, Yue Liao, Qiaosong Qi, Biaolong Chen, and Si Liu. Object-aware distillation pyramid for open-vocabulary object detection. In *CVPR*, pages 11186–11196, 2023. 6

[34] Size Wu, Wenwei Zhang, Sheng Jin, Wentao Liu, and Chen Change Loy. Aligning bag of regions for open-vocabulary object detection. In *CVPR*, pages 15254–15264, 2023. 6

[35] Xiaoshi Wu, Feng Zhu, Rui Zhao, and Hongsheng Li. Cora: Adapting clip for open-vocabulary detection with region prompting and anchor pre-matching. In *CVPR*, pages 7031–7040, 2023. 6

[36] Johnathan Xie and Shuai Zheng. Zsd-yolo: Zero-shot yolo detection using vision-language knowledge distillation. *arXiv preprint arXiv:2109.12066*, 2(3):4, 2021. 2

[37] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *CVPR*, pages 14393–14402, 2021. 2

[38] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022. 4

[39] Shiyu Zhao, Zhixing Zhang, Samuel Schulter, Long Zhao, BG Vijay Kumar, Anastasis Stathopoulos, Manmohan Chandraker, and Dimitris N Metaxas. Exploiting unlabeled data with vision and language models for object detection. In *ECCV*, pages 159–175. Springer, 2022. 6, 7

[40] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Probabilistic two-stage detection. *arXiv preprint arXiv:2103.07461*, 2021. 6

[41] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, pages 350–368. Springer, 2022. 2, 6

[42] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 6