YOLOE: Real-Time Seeing Anything

Ao Wang[1*] Lihao Liu[1*] Hui Chen[2] Zijia Lin[1] Jungong Han[3] Guiguang Ding[1,†]
[1]School of Software, Tsinghua University [2]BNRist, Tsinghua University
[3]Department of Automation, Tsinghua University

Abstract

Object detection and segmentation are widely employed in computer vision applications, yet conventional models like YOLO series, while efficient and accurate, are limited by predefined categories, hindering adaptability in open scenarios. Recent open-set methods leverage text prompts, visual cues, or prompt-free paradigm to overcome this, but often compromise between performance and efficiency due to high computational demands or deployment complexity. In this work, we introduce YOLOE, which integrates detection and segmentation across diverse open prompt mechanisms within a single highly efficient model, achieving real-time seeing anything. For text prompts, we propose Re-parameterizable Region-Text Alignment (RepRTA) strategy. It refines pretrained textual embeddings via a re-parameterizable lightweight auxiliary network and enhances visual-textual alignment with zero inference and transferring overhead. For visual prompts, we present Semantic-Activated Visual Prompt Encoder (SAVPE). It employs decoupled semantic and activation branches to bring improved visual embedding and accuracy with minimal complexity. For prompt-free scenario, we introduce Lazy Region-Prompt Contrast (LRPC) strategy. It utilizes a built-in large vocabulary and specialized embedding to identify all objects, avoiding costly language model dependency. Extensive experiments show YOLOE's exceptional zero-shot performance and transferability with high inference efficiency and low training cost. Notably, on LVIS, with 3× less training cost and 1.4× inference speedup, YOLOE-v8-S surpasses YOLO-Worldv2-S by 3.5 AP. When transferring to COCO, YOLOE-v8-L achieves 0.6 AP^b and 0.4 AP^m gains over closed-set YOLOv8-L with nearly 4× less training time. Code and models are available at here.

1. Introduction

Object detection and segmentation are foundational tasks in computer vision [15, 48], with widespread applications spanning autonomous driving [2], medical analyses [55], and robotics [8], *etc*. Traditional approaches like YOLO series [1, 3, 21, 47], have leveraged convolutional neural networks to achieve real-time remarkable performance. However, their dependence on predefined object categories constrains flexibility in practical open scenarios. Such scenarios increasingly demand models capable of detecting and segmenting arbitrary objects guided by diverse prompt mechanisms, such as texts, visual cues, or without prompt.

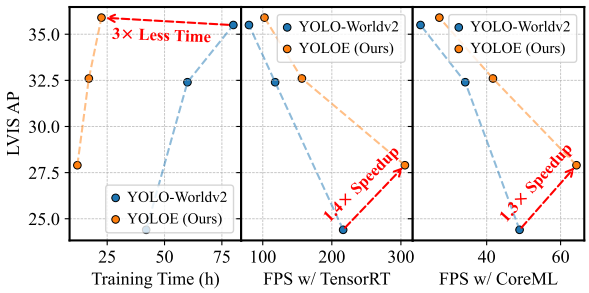


Figure 1. Comparison of performance, training cost, and inference efficiency between YOLOE (Ours) and advanced YOLO-Worldv2 in terms of open text prompts. LVIS AP is evaluated on `minival` set and FPS w/ TensorRT and w/ CoreML is measured on T4 GPU and iPhone 12, respectively. The results highlight our superiority.

Given this, recent efforts have shifted towards enabling models to generalize for open prompts [5, 20, 49, 80]. They target single prompt type, *e.g*., GLIP [32], or multiple prompt types in a unified way, *e.g*., DINO-X [49]. Specifically, with region-level vision-language pretraining [32, 37, 65], text prompts are usually processed by text encoder to serve as contrastive objectives for region features [20, 49], achieving recognition for arbitrary categories, *e.g*., YOLO-World [5]. For visual prompts, they are often encoded as class embeddings tied to specified regions for identifying similar objects, by the interaction with image features or language-aligned visual encoder [5, 19, 30, 49], *e.g*., T-Rex2 [20]. In prompt-free scenario, existing methods typically integrate language models, finding all objects and generating the corresponding category names conditioned on region features sequentially [49, 62], *e.g*., GenerateU [33].

Despite notable advancements, a single model that sup-

*Equal contributions. † Corresponding author.

ports diverse open prompts for arbitrary objects with high efficiency and accuracy is still lacking. For example, DINO-X [49] features a unified architecture, which, however, incurs resource-intensive training and inference overhead. Additionally, individual designs for different prompts in separate works exhibit suboptimal trade-offs between performance and efficiency, making it difficult to directly combine them into one model. For example, text-prompted approaches often incur substantial computational overhead when incorporating large vocabularies, due to complexity of cross-modality fusion [5, 32, 37, 49]. Visual-prompted methods usually compromise deployability on edge devices owing to the transformer-heavy design or reliance on additional visual encoder [20, 30, 67]. Prompt-free ways, meanwhile, depend on large language models, introducing considerable memory and latency costs [33, 49].

In light of these, in this paper, we introduce YOLOE(ye), a highly **efficient**, **unified**, and **open** object detection and segmentation model, like human eye, under different prompt mechanisms, like texts, visual inputs, and prompt-free paradigm. We begin with YOLO models with widely proven efficacy. For text prompts, we propose a Re-parameterizable Region-Text Alignment (RepRTA) strategy, which employs a lightweight auxiliary network to improve pretrained textual embeddings for better visual-semantic alignment. During training, pre-cached textual embeddings require only the auxiliary network to process text prompts, incurring low additional cost compared with closed-set training. At inference and transferring, auxiliary network is seamlessly re-parameterized into the classification head, yielding an architecture identical to YOLOs with zero overhead. For visual prompts, we design a Semantic-Activated Visual Prompt Encoder (SAVPE). By formalizing regions of interest as masks, SAVPE fuses them with multi-scale features from PAN to produce grouped prompt-aware weights in low dimension in an activation branch and extract prompt-agnostic semantic features in a semantic branch. Prompt embeddings are derived through aggregation of them, resulting in favorable performance with minimal complexity. For prompt-free scenario, we introduce Lazy Region-Prompt Contrast (LRPC) strategy. Without relying on costly language models, LRPC leverages a specialized prompt embedding to find all objects and a built-in large vocabulary for category retrieval. By matching only anchor points with identified objects against the vocabulary, LRPC ensures high performance with low overhead.

Thanks to them, YOLOE excels in detection and segmentation across diverse open prompt mechanisms within one model, enjoying high inference efficiency and low training cost. Notably, as shown in Fig. 1, under 3× less training cost, YOLOE-v8-S significantly outperforms YOLO-Worldv2-S [5] by 3.5 AP on LVIS [14], with 1.4× and 1.3× inference speedups on T4 and iPhone 12, respectively. In visual-prompted and prompt-free settings, YOLOE-v8-L outperforms T-Rex2 by 3.3 AP_r and GenerateU by 0.4 AP with 2× less training data and 6.3× fewer parameters, respectively. For transferring to COCO [34], YOLOE-v8-M / L outperforms YOLOv8-M / L by 0.4 / 0.6 AP^b and 0.4 / 0.4 AP^m with nearly 4× less training time. We hope that YOLOE can establish a strong baseline and inspire further advancements in real-time open prompt-driven vision tasks.

2. Related Work

Traditional detection and segmentation. Traditional approaches for object detection and segmentation primarily operate under closed-set paradigms. Early two-stage frameworks [4, 12, 15, 48], exemplified by Faster R-CNN [48], introduce region proposal networks (RPNs) followed by region-of-interest (ROI) classification and regression. Meanwhile, single-stage detectors [10, 35, 38, 56, 72] prioritizes speed through grid-based predictions within a single network. The YOLO series [1, 21, 27, 47, 59, 60] plays a significant role in this paradigm and are widely used in real world. Moreover, DETR [28] and its variants [28, 69, 77] mark a major shift by removing heuristic-driven components with transformer-based architectures. To achieve finer-grained results, existing instance segmentation methods predict pixel-level masks rather than bounding box coordinates [15]. For this, YOLACT [3] facilitates real-time instance segmentation through integration of prototype masks and mask coefficients. Based on DINO [69], MaskDINO [29] utilizes query embeddings and a high-resolution pixel embedding map to produce binary masks.

Text-prompted detection and segmentation. Recent advancements in open-vocabulary object detection [13, 25, 61, 65, 68, 74–76] have focused on detecting novel categories by aligning visual features with textual embeddings. Specifically, GLIP [32] unifies object detection and phrase grounding through grounded pre-training on large-scale image-text pairs, demonstrating robust zero-shot performance. Grounding DINO [37] enhances this by integrating cross-modality fusion into DINO, improving alignment between text prompts and visual representations. YOLO-World [5] further shows the potential of pretraining small detectors with open recognition capabilities based on the YOLO architecture. YOLO-UniOW [36] builds upon YOLO-World by leveraging the adaptive decision-learning strategy. Similarly, several open-vocabulary instance segmentation models [11, 18, 26, 45, 63] learn rich visual-semantic knowledge from advanced foundation models to perform segmentation on novel object categories. For example, X-Decoder [79] and OpenSeeD [71] explore both the open-vocabulary detection and segmentation tasks. APE [54] introduces universal visual perception model that aligns and prompts all objects using various text prompts.

Visual-prompted detection and segmentation. While

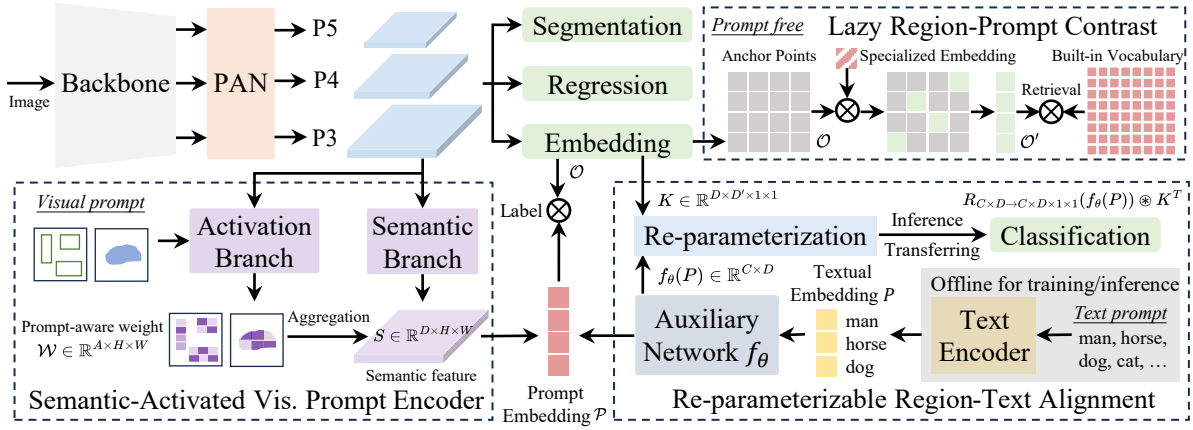


Figure 2. The overview of YOLOE, which supports detection and segmentation for diverse open prompt mechanisms. For text prompts, We design a re-parameterizable region-text alignment strategy to improve performance with zero inference and transferring overhead. For visual prompts, SAVPE is employed to encode visual cues with enhanced prompt embedding under minimal cost. For prompt-free setting, we introduce lazy region-prompt contrast strategy to provide category names for all identified objects efficiently by retrieval.

text prompts offer a generic description, certain objects can be challenging to describe with language alone, such as those requiring specialized domain knowledge. In such cases, visual prompts can guide detection and segmentation more flexibly and specifically, complementing text prompts [19, 20, 23]. OV-DETR [67] and OWL-ViT [41] leverage CLIP encoders to process text and image prompts. DINOv [30] explores visual prompts as in-context examples for generic and referring vision tasks. T-Rex2 [20] integrates visual and text prompts by region-level contrastive alignment. For segmentation, SEEM [80] explores segmenting objects with various prompt types. Semantic-SAM [31] excels in semantic comprehension and granularity detection, handling both panoptic and part segmentation.

Prompt-free detection and segmentation. Existing approaches still depend on explicit prompts during inference for open-set detection and segmentation. To address this limitation, several works [33, 40, 49, 62, 66] explore integrating with generative language models to produce object descriptions for all found objects. For instance, GRiT [62] employs a text decoder for both dense captioning and object detection tasks. DetCLIPv3 [66] trains an object captioner on large-scale data, enabling model to generate rich label information. GenerateU [33] leverages the language model to generate object names in a free-form way.

Closing remarks. To the best of our knowledge, aside from DINO-X [49], few efforts have achieved object detection and segmentation across various open prompt mechanisms within a single architecture. However, DINO-X entails extensive training cost and notable inference overhead, severely constraining the practicality for real-world edge deployments. In contrast, our YOLOE aims to deliver an efficient and unified model that enjoys real-time performance and efficiency with easy deployability.

3. Methodology

In this section, we detail designs of YOLOE. Building upon YOLOs (Sec. 3.1), YOLOE supports text prompts through RepRTA (Sec. 3.2), visual prompts via SAVPE (Sec. 3.3), and prompt-free scenario with LRPC (Sec. 3.4).

3.1. Model architecture

As shown in Fig. 2, YOLOE adopts the typical YOLOs' architecture [1, 21, 47], consisting of backbone, PAN, regression head, segmentation head, and object embedding head. The backbone and PAN extracts multi-scale features for the image. For each anchor point, the regression head predicts the bounding box for detection, and the segmentation head produces the prototype and mask coefficients for segmentation [3]. The object embedding head follows the structure of classification head in YOLOs, except that the output channel number of last $1\times$ convolution layer is changed from the class number in closed-set scenario to the embedding dimension. Meanwhile, given text and visual prompts, we employ RepRTA and SAVPE to encode them as normalized prompt embeddings $\mathcal{P}$, respectively. They serve as the classification weights and contrast with the anchor points' object embeddings $\mathcal{O}$ to obtain category labels. The process can be formalized as

$$\text{Label} = \mathcal{O} \cdot \mathcal{P}^T : \mathbb{R}^{N\times D} \times \mathbb{R}^{D\times C} \to \mathbb{R}^{N\times C}, \quad (1)$$

where N denotes the number of anchor points, C indicates the number of prompts, and D means the feature dimension of embeddings, respectively.

3.2. Re-parameterizable region-text alignment

In open-set scenarios, the alignment between textual and object embeddings determines the accuracy of identified categories. Prior works usually introduce complex cross-

modality fusion to improve the visual-textual representation for better alignment [5, 37]. However, these ways incur notable computational overhead, especially with large number of texts. Given this, we present Re-parameterizable Region-Text Alignment (RepRTA) strategy, which improves pretrained textual embeddings during training through the re-parameterizable lightweight auxiliary network. The alignment between textual and anchor points' object embeddings can be enhanced with zero inference and transferring cost.

Specifically, with the text prompts of T with length of C, we first employ the CLIP text encoder [44, 57] to obtain pretrained textual embedding $P = \text{TextEncoder}(T)$. Before training, we cache all embeddings of texts in datasets in advance and the text encoder can be removed with no extra training cost. Meanwhile, as shown in Fig. 3.(a), we introduce a lightweight auxiliary network f_θ with only one feed forward block [53, 58], where θ indicates the trainable parameters and introduces low overhead compared with closed-set training. It derives the enhanced textual embedding $\mathcal{P} = f_\theta(P) \in \mathbb{R}^{C\times D}$ for contrasting with the anchor points' object embedding during training, leading to improved visual-semantic alignment. Let $K \in \mathbb{R}^{D\times D'\times 1\times 1}$ be the kernel parameters of last convolution layer with input features $I \in \mathbb{R}^{D'\times H\times W}$ in the object embedding head, $\circledast$ be the convolution operator, and R be the reshape function, we have

$$\text{Label} = R_{D\times H\times W\to HW\times D}(I \circledast K)\cdot(f_\theta(P))^T. \quad (2)$$

Moreover, after training, the auxiliary network can be re-parameterized with the object embedding head into the identical classification head of YOLOs. The new kernel parameters $K' \in \mathbb{R}^{C\times D'\times 1\times 1}$ for last convolution layer after re-parameterization can be derived by

$$K' = R_{C\times D\to C\times D\times 1\times 1}(f_\theta(P)) \circledast K^T. \quad (3)$$

The final predication can be obtained by $\text{Label} = I \circledast K'$, which is identical to the original YOLO architecture, leading to zero overhead for deployment and transferring to downstream closed-set tasks.

3.3. Semantic-activated visual prompt encoder

Visual prompts are designed to indicate the object category of interest through visual cues, *e.g.*, box and mask. To produce the visual prompt embedding, prior works often employ transformer-heavy design [20, 30], *e.g.*, deformable attention [78], or additional CLIP vision encoder [44, 67]. These ways, however, introduce challenges in deployment and efficiency due to complex operators or high computational demands. Considering this, we introduce Semantic-Activated Visual Prompt Encoder (SAVPE) for efficiently processing visual cues. It features two decoupled lightweight branches: (1) Semantic branch outputs prompt-agnostic semantic features in D channels without overhead of fusing visual cues, and (2) Activation branch

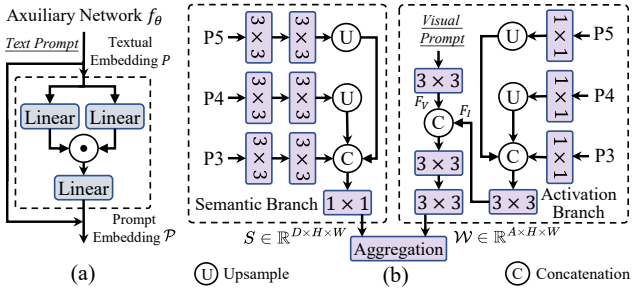


Figure 3. (a) The structure of lightweight auxiliary network in RepRTA, which consists of one SwiGLU FFN block [53]. (b) The structure of SAVPE, which consists of semantic branch to generate prompt-agnostic semantic features and activation branch to provide grouped prompt-aware weights. Visual prompt embedding can thus be efficiently derived by their aggregation.

produces grouped prompt-aware weights by interacting visual cues with image features in much fewer channels under low costs. Their aggregation then leads to informative prompt embedding under minimal complexity.

As shown in Fig. 3.(b), in the semantic branch, we adopt the similar structure as object embedding head. With multi-scale features $\{P_3, P_4, P_5\}$ from PAN, we employ two 3×3 convs for each scale, respectively. After upsampling, features are concatenated and projected to derive semantic features $S \in \mathbb{R}^{D\times H\times W}$. In the activation branch, we formalize visual prompt as mask with 1 for indicated region and 0 for others. We downsample it and leverage 3×3 conv to derive prompt feature $F_V \in \mathbb{R}^{A\times H\times W}$. Besides, we obtain image features $F_I \in \mathbb{R}^{A\times H\times W}$ for fusion with it from $\{P_3, P_4, P_5\}$ by convs. F_V and F_I are then concatenated and utilized to output prompt-aware weights $\mathcal{W} \in \mathbb{R}^{A\times H\times W}$, which is normalized using softmax within prompt-indicated region. Moreover, we divide the channels of S into A groups with $\frac{D}{A}$ channels in each. The channels in the i-th group share the weight $\mathcal{W}_{i:i+1}$ from the i-th channel of $\mathcal{W}$. With $A \ll D$, we can process visual cues with image features in low dimension, bringing minimal cost. Furthermore, prompt embedding can be derived with aggregation of two branches by

$$\mathcal{P} = \text{Concat}(G_1, ..., G_A); G_i = \mathcal{W}_{i:i+1}\cdot S^T_{\frac{D}{A}*i:\frac{D}{A}*(i+1)}. \quad (4)$$

It can thus contrast with anchor points' object embeddings to identify objects with category of interest.

3.4. Lazy region-prompt contrast

In prompt-free scenario without explicit guidance, models are expected to identity all objects with names in the image. Prior works usually formulate such setting as a generative problem, where language model is employed to generate categories for dense found objects [33, 49, 62]. However, this introduces notable overhead, where language models, *e.g.*, FlanT5-base [6] with 250M parameters in GenerateU [33] and OPT-125M [73] in DINO-X [49], are far from

meeting high efficiency requirement. Given this, we reformulate such setting as a retrieval problem and present Lazy Region-Prompt Contrast (LRPC) strategy. It lazily retrieves category names from a built-in large vocabulary for anchor points with objects in the cost-effective way. Such paradigm enjoys zero dependency on language models, meanwhile with favorable efficiency and performance.

Specifically, with pretrained YOLOE, we introduce a specialized prompt embedding and train it exclusively to find all objects, where objects are treated as one category. Meanwhile, we follow [16] to collect a large vocabulary which covers various categories and serve as the built-in data source for retrieval. One may directly leverage the large vocabulary as text prompts for YOLOE to identify all objects, which, however, incurs notable computational cost by contrasting abundant anchor points' object embeddings with numerous textual embeddings. Instead, we employ the specialized prompt embedding $\mathcal{P}_s$ to find the set $\mathcal{O}'$ of anchor points corresponding to objects by

$$\mathcal{O}' = \{o \in \mathcal{O} \mid o \cdot \mathcal{P}_s^T > \delta\}, \tag{5}$$

where $\mathcal{O}$ denotes all anchor points and δ is the threshold hyperparameter for filtering. Then, only anchor points in $\mathcal{O}'$ are lazily matched against the built-in vocabulary to retrieve category names, bypassing the cost for irrelevant anchor points. This further improves efficiency without performance drop, facilitating the real world application.

3.5. Training objective

During training, we follow [5] to obtain an online vocabulary for each mosaic sample with the texts involved in the images as positive labels. Following [21], we leverage task-aligned label assignment to match predictions with ground truths. The binary cross entropy loss is employed for classification, with IoU loss and distributed focal loss adopted for regression. For segmentation, we follow [3] to utilize binary cross-entropy loss for optimizing masks.

4. Experiments

4.1. Implementation details

Model. For fair comparison with [5], we employ the same YOLOv8 architecture [21] for YOLOE. Besides, to verify its good generalizability on other YOLOs, we also experiment with YOLO11 architecture [21]. For both of them, we provide three model scales, *i.e.*, small (S), medium (M), and large (L), to suit various application needs. Text prompts are encoded using the pretrained MobileCLIP-B(LT) [57] text encoder. We empirically use $A = 16$ in SAVPE, by default.

Data. We follow [5] to utilize detection and grounding datasets, including Objects365 (V1) [52], GoldG [22] (includes GQA [17] and Flickr30k [43]), where images from COCO [34] are excluded. Beside, we leverage advanced SAM-2.1 [46] model to generate pseudo instance masks using ground truth bounding boxes from the detection and grounding datasets for segmentation data. These masks undergo filtering and simplification to eliminate noise [9]. For visual prompt data, we follow [20] to leverage ground truth bounding boxes for visual cues. In prompt-free tasks, we reuse the same datasets, but annotate all objects as a single category to learn a specialized prompt embedding.

Training. Due to limited computational resource, unlike YOLO-World's training for 100 epochs, we first train YOLOE with text prompts for 30 epochs. Then, we only train the SAVPE for merely 2 epochs with visual prompts, which avoids additional significant training cost that comes with supporting visual prompts. At last, we train the specialized prompt embedding for only 1 epoch for prompt-free scenarios. During the text prompt training stage, we adopt the same settings as [5]. Notably, YOLOE-v8-S / M / L can be trained on 8 Nvidia RTX4090 GPUs in 12.0 / 17.0 / 22.5 hours, with 3× less cost compared with YOLO-World. For visual prompt training, we freeze all other parts and adopt the same setting as in text prompt training. To enable prompt-free capability, we leverage the same data to train a specialized embedding. We can see that YOLOE not only enjoys low training costs but also show exceptional zero-shot performance. Besides, to verify YOLOE's good transferability on downstream tasks, we fine-tune our YOLOE on COCO [34] for closed-set detection and segmentation. We experiment with two distinct practical fine-tuning strategies: (1) *Linear probing*: Only the classification head is learnable and (2) *Full tuning*: All parameters are trainable. For *Linear probing*, we train all models for only 10 epochs. For *Full tuning*, we train small scale models including YOLOE-v8-S / 11-S for 160 epochs, and medium and large scale models including YOLOE-v8-M / L and YOLOE-11-M / L for 80 epochs, respectively.

Metric. For text prompt evaluation, we utilize all category names from the benchmark as inputs, adhering to the standard protocol for open-vocabulary object detection tasks. For visual prompt evaluation, following [20], for each category, we randomly sample N training images (N=16 by default), extract visual embeddings using their ground truth bounding boxes, and compute the average prompt embedding. For prompt-free evaluation, we employ the same protocol as [33]. A pretrained text encoder [57] is employed to map open-ended predictions to semantically similar category names within the benchmark. In contrast to [33], we streamline the mapping process by selecting the most confident prediction, and eliminating the need for top-k selection and beam search. We use the tag list from [16] as the built-in large vocabulary with total 4585 category names, and empirically use $\delta = 0.001$ for LRPC, by default. For all three prompt types, following [5, 20, 33], evaluations are conducted on LVIS [14] in a zero-shot manner, which con-

Table 1. **Zero-shot detection evaluation on LVIS.** For fair comparisons, *Fixed AP* is reported on LVIS `minival` set in a zero-shot manner. The training time is for text prompts, based on 8 Nvidia V100 GPUs for [32, 65] and 8 RTX4090 GPUs for YOLO-World and YOLOE. The FPS is measured on Nvidia T4 GPU using TensorRT and on iPhone 12 using CoreML, respectively. Results are provided with text prompt (T) and visual prompt (V) type. For training data, OI, HT, and CH indicates OpenImages [24], HierText [39], and CrowdHuman [51], respectively. OG indicates Objects365 [52] and GoldG [22], and G-20M represents Grounding-20M [50].

Model	Prompt Type	Params	Training Data	Training Time	FPS T4 / iPhone	AP	AP_r	AP_c	AP_f
GLIP-T [32]	T	232M	OG,Cap4M	1337.6h	- / -	26.0	20.8	21.4	31.0
GLIPv2-T [70]	T	232M	OG,Cap4M	-	- / -	29.0	-	-	-
GDINO-T [37]	T	172M	OG,Cap4M	-	- / -	27.4	18.1	23.3	32.7
DetCLIP-T [65]	T	155M	OG	250.0h	- / -	34.4	26.9	33.9	36.3
G-1.5 Edge [50]	T	-	G-20M	-	- / -	33.5	28.0	34.3	33.9
T-Rex2 [20]	V	-	O365,OI,HT CH,SA-1B	-	- / -	37.4	29.9	33.9	41.8
YWorldv2-S [5]	T	13M	OG	41.7h	216.4 / 48.9	24.4	17.1	22.5	27.3
YWorldv2-M [5]	T	29M	OG	60.0h	117.9 / 34.2	32.4	28.4	29.6	35.5
YWorldv2-L [5]	T	48M	OG	80.0h	80.0 / 22.1	35.5	25.6	34.6	38.1
YOLOE-v8-S	T / V	12M / 13M	OG	**12.0h**	**305.8 / 64.3**	**27.9** / 26.2	**22.3** / 21.3	**27.8** / 27.7	**29.0** / 25.7
YOLOE-v8-M	T / V	27M / 30M	OG	**17.0h**	**156.7 / 41.7**	**32.6** / 31.0	26.9 / **27.0**	**31.9** / 31.7	**34.4** / 31.1
YOLOE-v8-L	T / V	45M / 50M	OG	**22.5h**	**102.5 / 27.2**	**35.9** / 34.2	**33.2 / 33.2**	**34.8** / 34.6	**37.3** / 34.1
YOLOE-11-S	T / V	10M / 12M	OG	**13.0h**	**301.2 / 73.3**	**27.5** / 26.3	21.4 / **22.5**	26.8 / **27.1**	**29.3** / 26.4
YOLOE-11-M	T / V	21M / 27M	OG	**18.5h**	**168.3 / 39.2**	**33.0** / 31.4	26.9 / **27.1**	**32.5** / 31.9	**34.5** / 31.7
YOLOE-11-L	T / V	26M / 32M	OG	**23.5h**	**130.5 / 35.1**	**35.2** / 33.7	**29.1** / 28.1	**35.0** / 34.6	**36.5** / 33.8

tains 1,203 categories. By default, *Fixed AP* [7] on LVIS `minival` subset is reported. For transferring to COCO, standard AP is evaluated, following [1, 21]. Besides, we measure the FPS for all models on Nvidia T4 GPU with TensorRT and mobile device iPhone 12 with CoreML.

4.2. Text and visual prompt evaluation

As shown in Tab. 1, for detection on LVIS, YOLOE exhibits favorable trade-offs between efficiency and zero-shot performance across different model scales. We also note that such results are achieved under much less training time, *e.g.*, 3× faster than YOLO-Worldv2. Specifically, YOLOE-v8-S / M / L outperforms YOLOv8-Worldv2-S / M / L by 3.5 / 0.2 / 0.4 AP, along with 1.4× / 1.3× / 1.3× and 1.3× / 1.2× / 1.2× inference speedups on T4 and iPhone 12, respectively. Besides, for rare category which is challenging, our YOLOE-v8-S and YOLOE-v8-L obtains significant improvements of 5.2% and 7.6% AP_r. Besides, compared with YOLO-Worldv2, while YOLOE-v8-M / L achieves lower AP_f, this performance gap primarily stems from YOLOE's integration of both detection and segmentation in one model. Such multi-task learning introduces a trade-off that adversely impact detection performance on frequent categories, as shown in Tab. 5. Besides, YOLOE with YOLO11 architecture also exhibits favorable performance and efficiency. For example, YOLOE-11-L achieves comparable AP with YOLO-Worldv2-L, but with notably 1.6× inference speedups on T4 and iPhone 12, highlighting the strong generalizability of our YOLOE.

Moreover, the inclusion of visual prompts further amplifies YOLOE's versatility. Compared with T-Rex2, YOLOE-v8-L yield the improvements of 3.3 AP_r and 0.9 AP_c, with 2× less training data (3.1 M vs. Our: 1.4 M) and much lower training resource (16 Nvidia A100 GPUs vs. Our: 8 Nvidia RTX4090 GPUs). Besides, for visual prompts, while we only train SAVPE with other parts frozen for 2 epochs, we note that it can achieve comparable AP_r and AP_c with the text prompts for various model scales. This shows the efficacy of visual prompts in less frequent objects that text prompts often struggle to accurately describe, which is similar to the observation in [20].

Furthermore, for segmentation, we present the evaluation results on the LVIS `val` set with the standard AP^m reported in Tab. 2. It shows that YOLOE exhibits strong performance by leveraging both text prompts and visual prompts. Specifically, YOLOE-v8-M / L achieves 20.8 and 23.5 AP^m in the zero-shot manner, significantly outperforming YOLO-Worldv2-M / L that is fine-tuned on `LVIS-Base` dataset, by 3.0 and 3.7 AP^m, respectively. These results well show the superiority of YOLOE.

4.3. Prompt-free evaluation

As shown in Tab. 3, for prompt-free scenario, YOLOE also exhibits superior performance and efficiency. Specifically, YOLO-v8-L achieves 27.2 AP and 23.5 AP_r, outperforming GenerateU with Swin-T backbone by 0.4 AP and 3.5 AP_r, along with 6.3× fewer parameters and 53× inference speedups. It shows the effectiveness of YOLOE by

Table 2. **Segmentation evaluation on LVIS.** We evaluate all models on LVIS `val` set with the standard AP^m reported. YOLOE supports both text (T) and visual cues (V) as inputs. † indicates that the pretrained models are fine-tuned on `LVIS-Base` data for segmentation head. In contrast, we evaluate YOLOE in a zero-shot manner without utilizing any images from LVIS during training.

Model	Prompt	AP^m	AP^m_r	AP^m_c	AP^m_f
YWorld-M†	T	16.7	12.6	14.6	20.8
YWorld-L†	T	19.1	14.2	17.2	23.5
YWorldv2-M†	T	17.8	13.9	15.5	22.0
YWorldv2-L†	T	19.8	17.2	17.5	23.6
YOLOE-v8-S	T / V	**17.7** / 16.8	**15.5** / 13.5	16.3 / **16.7**	**20.3** / 18.2
YOLOE-v8-M	T / V	**20.8** / 20.3	**17.2** / 17.0	19.2 / **20.1**	**24.2** / 22.0
YOLOE-v8-L	T / V	**23.5** / 22.0	**21.9** / 16.5	21.6 / **22.1**	**26.4** / 24.3
YOLOE-11-S	T / V	**17.6** / 17.1	**16.1** / 14.4	15.6 / **16.8**	**20.5** / 18.6
YOLOE-11-M	T / V	**21.1** / 21.0	17.2 / **18.3**	19.6 / **20.6**	**24.4** / 22.6
YOLOE-11-L	T / V	**22.6** / 22.5	19.3 / **20.5**	20.9 / **21.7**	**26.0** / 24.1

Table 3. **Prompt-free evaluation on LVIS.** *Fixed AP* is reported on the LVIS `minival` set, following the protocol in [33]. The FPS is measured on Nvidia T4 GPU with Pytorch [42].

Model	Backbone	Params	AP	AP_r	AP_c	AP_f	FPS
GenerateU [33]	Swin-T	297M	26.8	20.0	24.9	29.8	0.48
GenerateU [33]	Swin-L	467M	27.9	22.3	25.2	31.4	0.40
YOLOE-v8-S	YOLOv8-S	13M	**21.0**	19.1	21.3	21.0	**95.8**
YOLOE-v8-M	YOLOv8-M	29M	**24.7**	22.2	24.5	25.3	**45.9**
YOLOE-v8-L	YOLOv8-L	47M	**27.2**	23.5	27.0	28.0	**25.3**
YOLOE-11-S	YOLO11-S	11M	**20.6**	18.4	20.2	21.3	**93.0**
YOLOE-11-M	YOLO11-M	24M	**25.5**	21.6	25.5	26.1	**42.5**
YOLOE-11-L	YOLO11-L	29M	**26.3**	22.7	25.8	27.5	**34.9**

reformulating the open-ended problem as the retrieval task for a built-in large vocabulary and underscores its potential in generalizing across a wide range of categories without replying on explicit prompts. Such functionality also enhances YOLOE's practicality, enabling its application in a broader range of real-world scenarios.

4.4. Downstream transferring

As shown in Tab. 4, when transferring to COCO for downstream closed-set detection and segmentation, YOLOE exhibits favorable performance under limited training epochs in both two fine-tuning strategies. Specifically, for *Linear probing*, with less than 2% of the training time, YOLOE-11-M / L can achieve over 80% of the performance of YOLO11-M / L, respectively. This highlights the strong transferability of YOLOE. For *Full tuning*, YOLOE can further enhance the performance under limited training cost. For example, with nearly 4× less training epochs, YOLOE-v8-M / L outperforms YOLOv8-M / L by 0.4 AP^m and 0.6 AP^b, respectively. Under 3× less training time, YOLO-v8-S also obtains better performance compared with YOLOv8-S for both detection and segmentation. These results well demonstrate that YOLOE can serve as a strong starting point for transferring to downstream task.

Table 4. **Downstream transfer on COCO.** We fine-tune YOLOE on COCO and report the standard AP for both detection and segmentation. We experiment with two practical fine-tuning strategies, *i.e.*, *Linear probing* and *Full tuning*.

Model	Epochs	AP^b	AP^b_{50}	AP^b_{75}	AP^m	AP^m_{50}	AP^m_{75}
Training from scratch							
YOLOv8-S	500	44.7	61.4	48.7	36.6	58.0	38.6
YOLOv8-M	300	50.0	66.8	54.8	40.5	63.4	43.3
YOLOv8-L	300	52.4	69.3	57.2	42.3	66.0	44.9
YOLO11-S	500	46.6	63.3	50.6	37.8	59.7	40.0
YOLO11-M	600	51.5	68.5	55.7	41.5	65.0	43.9
YOLO11-L	600	53.3	70.1	58.2	42.8	66.8	45.5
Linear probing							
YOLOE-v8-S	**10**	**35.6**	51.5	38.9	**30.3**	48.2	32.0
YOLOE-v8-M	**10**	**42.2**	59.2	46.3	**35.5**	55.6	37.7
YOLOE-v8-L	**10**	**45.4**	63.3	50.0	**38.3**	59.6	40.8
YOLOE-11-S	**10**	**37.0**	52.9	40.4	**31.5**	49.7	33.5
YOLOE-11-M	**10**	**43.1**	60.6	47.4	**36.5**	56.9	39.0
YOLOE-11-L	**10**	**45.1**	62.8	49.5	**38.0**	59.2	40.6
Full tuning							
YOLOE-v8-S	**160**	**45.0**	61.6	49.1	**36.7**	58.3	39.1
YOLOE-v8-M	**80**	**50.4**	67.0	55.2	**40.9**	63.7	43.5
YOLOE-v8-L	**80**	**53.0**	69.8	57.9	**42.7**	66.5	45.6
YOLOE-11-S	**160**	**46.2**	62.9	50.0	**37.6**	59.3	40.1
YOLOE-11-M	**80**	**51.3**	68.3	56.0	**41.5**	64.8	44.3
YOLOE-11-L	**80**	**52.6**	69.7	57.5	**42.4**	66.2	45.2

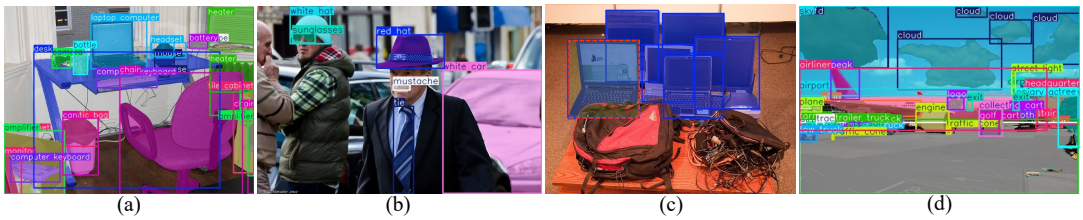
Table 5. Roadmap to YOLOE in terms of text prompts. The standard AP is reported on LVIS `minival` set in the zero-shot manner. The FPS is is measured on Nvidia T4 GPU and iPhone 12 with TensorRT (T) and CoreML (C), respectively.

Model	Epochs	AP	AP_r	AP_c	AP_f	FPS (T / C)
YOLO-Worldv2-L	100	33.0	22.6	32.0	35.8	80.0 / 22.1
+ Fewer train. epochs	30	31.0	22.6	28.8	34.2	80.0 / 22.1
+ Global negative dict.	30	31.9	22.8	31.0	34.4	80.0 / 22.1
- Cross-modal. fusion	30	30.0	19.1	28.0	33.9	102.5 / 27.2
+ MobileCLIP encoder	30	31.5	20.2	30.5	34.4	102.5 / 27.2
+ RepRTA	30	33.5	29.5	32.0	35.5	102.5 / 27.2
+ Segment. (YOLOE)	30	33.3	30.8	32.2	34.6	102.5 / 27.2

4.5. Ablation study

We further provide extensive analyses for the effectiveness of designs in our YOLOE. Experiments are conducted on YOLOE-v8-L and standard AP is reported on LVIS `minival` set for zero-shot evaluation, by default.

Roadmap to YOLOE. We outline the stepwise progression from the baseline model YOLOv8-Worldv2-L to our YOLOE-v8-L in terms of text prompts in Tab. 5. With the initial baseline metric of 33.0% AP, due to limited computational resource, we first reduce the training epochs to 30, leading to 31.0% AP. Besides, instead of using empty



(a) (b) (c) (d)

Figure 4. (a) Zero-shot inference on LVIS. (b) Results with customized text prompt, where "white hat, red hat, white car, sunglasses, mustache, tie" are provided as text prompts. (c) Results with visual prompt, where the red dashed bounding box serves as the visual cues. (d) Results in prompt-free scenario, where no explicit prompt is provided. Please refer to the supplementary for more examples.

Table 6. Effective. of SAVPE.

Model	AP	AP_r	AP_c	AP_f
Mask pool	30.4	27.6	31.3	30.2
SAVPE	31.9	29.4	32.5	31.7
$A = 1$	30.9	28.2	31.9	30.4
$A = 16$	31.9	29.4	32.5	31.7
$A = 32$	31.9	28.2	33.0	31.7

Table 7. Effective. of LRPC.

Model	LRPC	AP	AP_r	AP_c	AP_f	FPS
v8-S	✗	21.0	19.1	21.4	21.0	56.5
	$\delta = 1e^{-3}$	21.0	19.1	21.3	21.0	95.8
	$\delta = 1e^{-4}$	21.0	19.1	21.3	21.0	66.1
	$\delta = 1e^{-2}$	20.8	19.1	21.2	20.8	106
v8-L	✗	27.2	23.5	27.0	28.0	19.9
	$\delta = 1e^{-3}$	27.2	23.5	27.0	28.0	25.3

string as negative texts for grounding data, we follow [65] by maintaining a global dictionary to sample more diverse negative prompts. The global dictionary is constructed by selecting category names that appear more than 100 times in the training data. This leads to 0.9% AP improvement. Next, we remove the cross-modality fusion to avoid costly visual-textual feature interaction, which results in 1.9% AP degradation but with 1.28× and 1.23× inference speedups on T4 and iPhone 12, respectively. To address this drop, we utilize stronger MobileCLIP-B(LT) text encoder [57] to obtain better pretrained textual embeddings, which recovers AP to 31.5%. Furthermore, we employ RepRTA to enhance the alignment between anchor points' object and textual embeddings, which leads to notable 2.3% AP enhancement with zero inference overhead, showing its effectiveness. At last, we introduce the segmentation head and train YOLOE for detection and segmentation simultaneously. Although this leads to 0.2% AP and 0.9 AP_f drop due to multi-task learning, YOLOE gains ability to segment arbitrary objects.

Effectiveness of SAVPE. To verify the effectiveness of SAVPE for visual inputs, we remove the activation branch and simply leverage mask pooling to aggregate semantic features with the formulated visual prompt mask. As shown in Tab. 6, SAVPE significantly outperforms "Mask pool" by 1.5 AP. This is because "Mask pool" neglects the varying semantic importance at different positions within prompt-indicated region, while our activation branch effectively models such difference, leading to improved aggregation of semantic features and better prompt embedding for contrast. We also examine the impact of different group numbers, *i.e.*, A, in the activation branch. As shown in Tab. 6, performance can also be enhanced with only a group, *i.e.*, $A = 1$. Besides, we can achieve the strong performance of 31.9 AP under $A = 16$, obtaining the favorable balance, where more groups lead to marginal performance difference.

Effectiveness of LRPC. To verify the effectiveness of LRPC for prompt-free setting, we introduce the baseline that directly leverage the built-in large vocabulary as text prompts for YOLOE to identify all objects. Tab. 7 presents the comparison results. We observe that with the same performance, our LRPC obtains notably 1.7× / 1.3× inference speedups for YOLOE-v8-S / L, respectively, by lazily retrieving the categories for anchor points with found objects and skipping the numerous irrelevant ones. These results well highlight its efficacy and practicality. Besides, with different threshold δ for filtering, LRPC can achieve different performance and efficiency trade-offs, *e.g.*, enabling 1.9× speedup for YOLOE-v8-S with only 0.2 AP drop.

4.6. Visualization analyses

We conduct visualization analyses for YOLOE in four scenarios: (1) Zero-shot inference on LVIS in Fig. 4.(a), where its category names are text prompts, (2) Text prompts in Fig. 4.(b), where arbitrary texts can be input as prompts, (3) Visual prompts in Fig. 4.(c), where visual cues can be drawn as prompts, and (4) No explicit prompt in Fig. 4.(d), where model identifies all objects. We can see that YOLOE performs well and can accurately detect and segment various objects in these diverse scenarios, further showing its efficacy and practicality in various applications.

5. Conclusion

In this paper, we present YOLOE, a single highly efficient model that seamlessly integrates object detection and segmentation across diverse open prompt mechanisms. Specifically, we introduce RepRTA, SAVPE, and LRPC to enable YOLOs to process textual prompt, visual cues, and prompt-free paradigm with favorable performance and low cost. Thanks to them, YOLOE enjoys strong capabilities and high efficiency for various prompt ways, enabling real-time seeing anything. We hope that it can serve as a strong baseline to inspire further advancements.

6. Acknowledgments

This work was supported by National Natural Science Foundation of China (Nos. 62525103, 624B2082, 62271281, 62441235).

References

[1] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020. 1, 2, 3, 6

[2] Daniel Bogdoll, Maximilian Nitsche, and J Marius Zöllner. Anomaly detection in autonomous driving: A survey. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4488–4499, 2022. 1

[3] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9157–9166, 2019. 1, 2, 3, 5

[4] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018. 2

[5] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16901–16911, 2024. 1, 2, 4, 5, 6

[6] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024. 4

[7] Achal Dave, Piotr Dollár, Deva Ramanan, Alexander Kirillov, and Ross Girshick. Evaluating large-vocabulary object detectors: The devil is in the details. *arXiv preprint arXiv:2102.01066*, 2021. 6

[8] Douglas Henke Dos Reis, Daniel Welfer, Marco Antonio De Souza Leite Cuadros, and Daniel Fernando Tello Gamarra. Mobile robot navigation using an object recognition software with rgbd images and the yolo algorithm. *Applied Artificial Intelligence*, 33(14):1290–1305, 2019. 1

[9] David H Douglas and Thomas K Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: the international journal for geographic information and geovisualization*, 10 (2):112–122, 1973. 5

[10] Chengjian Feng, Yujie Zhong, Yu Gao, Matthew R Scott, and Weilin Huang. Tood: Task-aligned one-stage object detection. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3490–3499. IEEE Computer Society, 2021. 2

[11] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European conference on computer vision*, pages 540–557. Springer, 2022. 2

[12] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 2

[13] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *International Conference on Learning Representation*, 2022. 2

[14] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5356–5364, 2019. 2, 5

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 1, 2

[16] Xinyu Huang, Yi-Jie Huang, Youcai Zhang, Weiwei Tian, Rui Feng, Yuejie Zhang, Yanchun Xie, Yaqian Li, and Lei Zhang. Open-set image tagging with multi-grained text supervision. *arXiv preprint arXiv:2310.15200*, 2023. 5

[17] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019. 5

[18] Dat Huynh, Jason Kuen, Zhe Lin, Jiuxiang Gu, and Ehsan Elhamifar. Open-vocabulary instance segmentation via robust cross-modal pseudo-labeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7020–7031, 2022. 2

[19] Qing Jiang, Feng Li, Tianhe Ren, Shilong Liu, Zhaoyang Zeng, Kent Yu, and Lei Zhang. T-rex: Counting by visual prompting. *arXiv preprint arXiv:2311.13596*, 2023. 1, 3

[20] Qing Jiang, Feng Li, Zhaoyang Zeng, Tianhe Ren, Shilong Liu, and Lei Zhang. T-rex2: Towards generic object detection via text-visual prompt synergy. In *European Conference on Computer Vision*, pages 38–57. Springer, 2024. 1, 2, 3, 4, 5, 6

[21] Glenn Jocher, Jing Qiu, and Ayush Chaurasia. Ultralytics YOLO, 2023. 1, 2, 3, 5, 6

[22] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1780–1790, 2021. 5, 6

[23] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023. 3

[24] 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. 6

[25] Weicheng Kuo, Yin Cui, Xiuye Gu, AJ Piergiovanni, and Anelia Angelova. F-vlm: Open-vocabulary object detection upon frozen vision and language models. *International Conference on Learning Representation*, 2022. 2

[26] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. *International Conference on Learning Representation*, 2022. 2

[27] Chuyi Li, Lulu Li, Hongliang Jiang, Kaiheng Weng, Yifei Geng, Liang Li, Zaidan Ke, Qingyuan Li, Meng Cheng,

Weiqiang Nie, et al. Yolov6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*, 2022. 2

[28] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13619–13627, 2022. 2

[29] Feng Li, Hao Zhang, Huaizhe Xu, Shilong Liu, Lei Zhang, Lionel M Ni, and Heung-Yeung Shum. Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3041–3050, 2023. 2

[30] Feng Li, Qing Jiang, Hao Zhang, Tianhe Ren, Shilong Liu, Xueyan Zou, Huaizhe Xu, Hongyang Li, Jianwei Yang, Chunyuan Li, et al. Visual in-context prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12861–12871, 2024. 1, 2, 3, 4

[31] Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Chunyuan Li, Jianwei Yang, Lei Zhang, and Jianfeng Gao. Segment and recognize anything at any granularity. In *European Conference on Computer Vision*, pages 467–484. Springer, 2024. 3

[32] 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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10965–10975, 2022. 1, 2, 6

[33] Chuang Lin, Yi Jiang, Lizhen Qu, Zehuan Yuan, and Jianfei Cai. Generative region-language pretraining for open-ended object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13958–13968, 2024. 1, 2, 3, 4, 5, 7

[34] 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 *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755. Springer, 2014. 2, 5

[35] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 2

[36] Lihao Liu, Juexiao Feng, Hui Chen, Ao Wang, Lin Song, Jungong Han, and Guiguang Ding. Yolo-uniow: Efficient universal open-world object detection. *arXiv preprint arXiv:2412.20645*, 2024. 2

[37] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2024. 1, 2, 4, 6

[38] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer, 2016. 2

[39] Shangbang Long, Siyang Qin, Dmitry Panteleev, Alessandro Bissacco, Yasuhisa Fujii, and Michalis Raptis. Towards end-to-end unified scene text detection and layout analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1049–1059, 2022. 6

[40] Yanxin Long, Youpeng Wen, Jianhua Han, Hang Xu, Pengzhen Ren, Wei Zhang, Shen Zhao, and Xiaodan Liang. Capdet: Unifying dense captioning and open-world detection pretraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15233–15243, 2023. 3

[41] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection. In *European conference on computer vision*, pages 728–755. Springer, 2022. 3

[42] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 7

[43] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015. 5

[44] 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 *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 4

[45] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18082–18091, 2022. 2

[46] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 5

[47] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 1, 2, 3

[48] 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

[49] Tianhe Ren, Yihao Chen, Qing Jiang, Zhaoyang Zeng, Yuda Xiong, Wenlong Liu, Zhengyu Ma, Junyi Shen, Yuan Gao, Xiaoke Jiang, et al. Dino-x: A unified vision model for open-world object detection and understanding. *arXiv preprint arXiv:2411.14347*, 2024. 1, 2, 3, 4

[50] Tianhe Ren, Qing Jiang, Shilong Liu, Zhaoyang Zeng, Wenlong Liu, Han Gao, Hongjie Huang, Zhengyu Ma, Xiaoke Jiang, Yihao Chen, Yuda Xiong, Hao Zhang, Feng Li, Peijun Tang, Kent Yu, and Lei Zhang. Grounding dino 1.5: Advance the "edge" of open-set object detection, 2024. 6

[51] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018. 6

[52] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8430–8439, 2019. 5, 6

[53] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020. 4

[54] Yunhang Shen, Chaoyou Fu, Peixian Chen, Mengdan Zhang, Ke Li, Xing Sun, Yunsheng Wu, Shaohui Lin, and Rongrong Ji. Aligning and prompting everything all at once for universal visual perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13193–13203, 2024. 2

[55] Joseph Sobek, Jose R Medina Inojosa, Betsy J Medina Inojosa, SM Rassoulinejad-Mousavi, Gian Marco Conte, Francisco Lopez-Jimenez, and Bradley J Erickson. Medyolo: a medical image object detection framework. *Journal of Imaging Informatics in Medicine*, pages 1–9, 2024. 1

[56] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019. 2

[57] Pavan Kumar Anasosalu Vasu, Hadi Pouransari, Fartash Faghri, Raviteja Vemulapalli, and Oncel Tuzel. Mobileclip: Fast image-text models through multi-modal reinforced training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15963–15974, 2024. 4, 5, 8

[58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[59] Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, et al. Yolov10: Real-time end-to-end object detection. *Advances in Neural Information Processing Systems*, 37:107984–108011, 2025. 2

[60] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7464–7475, 2023. 2

[61] Yu Wang, Xiangbo Su, Qiang Chen, Xinyu Zhang, Teng Xi, Kun Yao, Errui Ding, Gang Zhang, and Jingdong Wang. Ovlw-detr: Open-vocabulary light-weighted detection transformer. *arXiv preprint arXiv:2407.10655*, 2024. 2

[62] Jialian Wu, Jianfeng Wang, Zhengyuan Yang, Zhe Gan, Zicheng Liu, Junsong Yuan, and Lijuan Wang. Grit: A generative region-to-text transformer for object understanding. In *European Conference on Computer Vision*, pages 207–224. Springer, 2024. 1, 3, 4

[63] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18134–18144, 2022. 2

[64] Yifan Xu, Mengdan Zhang, Chaoyou Fu, Peixian Chen, Xiaoshan Yang, Ke Li, and Changsheng Xu. Multi-modal queried object detection in the wild. *Advances in Neural Information Processing Systems*, 36:4452–4469, 2023.

[65] Lewei Yao, Jianhua Han, Youpeng Wen, Xiaodan Liang, Dan Xu, Wei Zhang, Zhenguo Li, Chunjing Xu, and Hang Xu. Detclip: Dictionary-enriched visual-concept paralleled pre-training for open-world detection. *Advances in Neural Information Processing Systems*, 35:9125–9138, 2022. 1, 2, 6, 8

[66] Lewei Yao, Renjie Pi, Jianhua Han, Xiaodan Liang, Hang Xu, Wei Zhang, Zhenguo Li, and Dan Xu. Detclipv3: Towards versatile generative open-vocabulary object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27391–27401, 2024. 3

[67] Yuhang Zang, Wei Li, Kaiyang Zhou, Chen Huang, and Chen Change Loy. Open-vocabulary detr with conditional matching. In *European conference on computer vision*, pages 106–122. Springer, 2022. 2, 3, 4

[68] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14393–14402, 2021. 2

[69] 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. 2

[70] Haotian Zhang, Pengchuan Zhang, Xiaowei Hu, Yen-Chun Chen, Liunian Harold Li, Xiyang Dai, Lijuan Wang, Lu Yuan, Jenq-Neng Hwang, and Jianfeng Gao. Glipv2: Unifying localization and vision-language understanding, 2022. 6

[71] Hao Zhang, Feng Li, Xueyan Zou, Shilong Liu, Chunyuan Li, Jianwei Yang, and Lei Zhang. A simple framework for open-vocabulary segmentation and detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1020–1031, 2023. 2

[72] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9759–9768, 2020. 2

[73] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab,

Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022. 4

[74] Tiancheng Zhao, Peng Liu, Xuan He, Lu Zhang, and Kyusong Lee. Real-time transformer-based open-vocabulary detection with efficient fusion head. *arXiv preprint arXiv:2403.06892*, 2024. 2

[75] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16793–16803, 2022.

[76] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *European conference on computer vision*, pages 350–368. Springer, 2022. 2

[77] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *International Conference on Learning Representation*, 2021. 2

[78] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection, 2021. 4

[79] Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, et al. Generalized decoding for pixel, image, and language. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15116–15127, 2023. 2

[80] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *Advances in neural information processing systems*, 36:19769–19782, 2023. 1, 3