Matcher: Segment Anything with One Shot Using All-Purpose Feature Matching

Yang Liu[1*†] **Muzhi Zhu**[1*] **Hengtao Li**[1*] **Hao Chen**[1] **Xinlong Wang**[2] **Chunhua Shen**[1]

[1] Zhejiang University, China [2] Beijing Academy of Artificial Intelligence

Code: https://github.com/aim-uofa/Matcher

Abstract

Powered by large-scale pre-training, vision foundation models exhibit significant potential in open-world image understanding. Even though individual models have limited capabilities, combining multiple such models properly can lead to positive synergies and unleash their full potential. In this work, we present **Matcher**, which segments anything with one shot by integrating an all-purpose feature extraction model and a class-agnostic segmentation model. Naively connecting the models results in unsatisfying performance, *e.g.*, the models tend to generate matching outliers and false-positive mask fragments.To address these issues, we design a bidirectional matching strategy for accurate cross-image semantic dense matching and a robust prompt sampler for mask proposal generation. In addition, we propose a novel instance-level matching strategy for controllable mask merging. The proposed Matcher method delivers impressive generalization performance across various segmentation tasks, all without training. For example, it achieves 52.7% mIoU on COCO-20^i for one-shot semantic segmentation, surpassing the state-of-the-art specialist model by 1.6%. In addition, our visualization results show open-world generality and flexibility on images in the wild.

1 Introduction

Pre-trained on web-scale datasets, large language models (LLMs) [BMR+20, OWJ+22, CND+22, ZRG+22, ZLD+22, TLI+23], like ChatGPT [Ope23], have revolutionized natural language processing (NLP). These foundation models [BHA+21] show remarkable transfer capability on tasks and data distributions beyond their training scope. LLMs demonstrate powerful zero-shot and few-shot generalization [BMR+20] and solve various language tasks well, *e.g.*, language understanding, generation, interaction, and reasoning.

Research of foundation models in computer vision is catching up with NLP. Driven by large-scale image-text contrastive pre-training, CLIP [RKH+21] and ALIGN [JYX+21] perform strong zero-shot transfer ability to downstream tasks. Instead of guiding by text, DINOv2 [ODM+23] learns all-purpose visual features by capturing complex information at the image and pixel level from raw image data alone. It achieves better or comparable performance with CLIP on downstream tasks. However, using these foundation models as image encoders requires task-specific heads for downstream tasks, which limits their generalization for real-world applications. Recently, the Segment Anything Model (SAM) [KMR+23] has achieved impressive zero-shot segmentation performance, which exhibits significant potential in open-world image perception. However, as a class-agnostic segmenter, SAM can not extract high-level semantic features, which limits its capability for open-world image understanding. In this paper, we demonstrate that even though individual foundation

[*]Equal contribution. [†]Part of the work was done when YL was an intern at Beijing Academy of Artificial Intelligence. CS is the corresponding author.

models have limited capabilities, integrating them leads to positive synergies, improving both the segmentation quality and open-set generality.

Recently, a line of research [WYQ+23, SST+23, ZCH+23, ZCS+23] attempts to solve complicated AI tasks by conjoining various foundation models. For example, Grounded-SAM [KMR+23, LZR+23] builds a strong pipeline for open-world understanding by combining the strengths of different models [KMR+23, RBL+22, LLXH22, WYQ+23]. These models collaborate in the AI system by directly taking the output results of one foundation as the inputs of another. However, each component works independently, and the cumulative error cannot be easily reduced. Therefore, we rethink the connection of different vision foundation models. The practicability of the Segment Anything Model (SAM) is limited due to the lack of semantic information and the presence of ambiguous mask fragments. To address this limitation, we explore segmenting anything using a single in-context example without training. We consider semantic diversity and structural diversity for different segmentation requirements. Semantic diversity encompasses instance-level and semantic-level perception, including tasks such as video object segmentation and semantic segmentation. Structural diversity refers to various semantic granularity, from parts and whole to multiple instances. To achieve semantic diversity, we leverage all-purpose feature matching of a pre-trained model. To address structural diversity, we employ prompt-based SAM. In summary, we introduce a new paradigm that utilizes the all-purpose features of a pre-trained model for feature matching and leverages this matching to enable efficient and effective segmentation.

We present **Matcher**, a training-free framework combining an all-purpose feature extraction model and a class-agnostic segmentation model. Specifically, we devise a bidirectional matching strategy for accurate cross-image semantic dense matching and a robust prompt sampler for mask proposal generation. This strategy increases the diversity of mask proposals and suppresses fragmented false-positive masks induced by matching outliers. Furthermore, we perform instance-level matching between the reference mask and mask proposals to select high-quality masks. We utilize three effective metrics, *i.e.*, *emd*, *purity*, and *coverage*, to estimate the mask proposals based on semantic similarity and the quality of the mask proposals, respectively. Finally, by controlling the number of merged masks, Matcher can produce controllable mask output to instances of the same semantics in the target image.

Our comprehensive experiments demonstrate that Matcher has superior generalization performance across various segmentation tasks, all without the need for training. For one-shot semantic segmentation, Matcher achieves **52.7%** mIoU on COCO-20^i [NT19], surpassing the state-of-the-art specialist model by 1.6%. And Matcher outperforms recent PerSAM [ZJG+23] by a large margin (+**29.2%** mean mIoU on COCO-20^i, +**11.4%** mIoU on FSS-1000 [LWC+20], and +**10.7%** mean mIoU on LVIS-92^i), suggesting that depending solely on SAM limits the generalization capabilities for semantically-driven tasks, *e.g.*, semantic segmentation. Moreover, evaluated on two proposed benchmarks, Matcher shows outstanding generalization on one-shot object part segmentation tasks. Specifically, Matcher outperforms other methods by about **10.0%** mean mIoU on both benchmarks. Matcher also achieves competitive performance for video object segmentation on both DAVIS 2017 val [PTPC+17] and DAVIS 2016 val [PPTM+16]. In addition, exhaustive ablation studies verify the effectiveness of the proposed components of Matcher. Finally, our visualization results show robust generality and flexibility never seen before.

Our main contributions are summarized as follows:

- We present Matcher, a training-free framework integrating an all-purpose feature extraction model and a class-agnostic segmentation model for solving various few-shot segmentation tasks.
- We design three components for Matcher, *i.e.*, bidirectional matching, robust prompt sampler, and instance-level matching, which can effectively unleash the ability of these foundation models to improve both the segmentation quality and open-set generality.
- Our comprehensive results demonstrate the powerful generalization of Matcher. Significantly, Matcher surpasses the state-of-the-art specialist model on COCO-20^i for one-shot semantic segmentation.

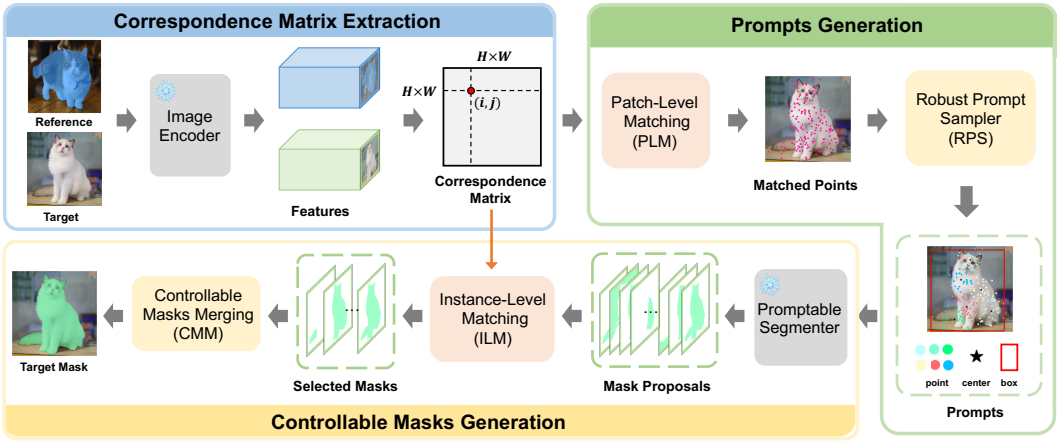


Figure 1: An overview of Matcher. Our training-free framework addresses various segmentation tasks through three operations: Correspondence Matrix Extraction, Prompts Generation, and Controllable Masks Generation.

2 Related Work

Foundation Models Powered by large-scale pre-training, vision foundation models have achieved great success in computer vision. Motivated by masked language modeling [DCLT19, LOG+19] in natural language processing, MAE [HCX+22] uses an asymmetric encoder-decoder and conducts masked image modeling to effectively and efficiently train scalable vision Transformer [DBK+20] models. MAE shows excellent fine-tuning performance in various downstream tasks. Contrastive Language-Image Pre-training (CLIP) [RKH+21] learns image representations from scratch on 400 million image-text pairs and demonstrates impressive zero-shot image classification ability. By performing image and patch level discriminative self-supervised learning, DINOv2 [ODM+23] learns all-purpose visual features for various downstream tasks. Moreover, DINOv2 demonstrates impressive patch-matching ability, capturing information about semantic parts that perform similar intents across different objects or animals. Recently, pre-trained with 1B masks and 11M images, Segment Anything Model (SAM) [KMR+23] emerges with impressive zero-shot class-agnostic segmentation performance. Although vision foundation models have shown exceptional fine-tuning performance, they have limited capabilities in open-world image understanding. However, large language models [BMR+20, OWJ+22, CND+22, ZRG+22, ZLD+22, TLI+23], like ChatGPT [Ope23], can solve various language tasks without training. Motivated by this, this work shows that various few-shot perception tasks can be solved training-free by integrating an all-purpose feature extraction model and a class-agnostic segmentation model.

Vision Generalist for Segmentation Recently, a growing effort has been made to unify various segmentation tasks under a single model using Transformer architecture [VSP+17]. The generalist Painter [WWC+23] redefines the output of different vision tasks as images and utilizes masked image modeling on continuous pixels to perform in-context training with supervised datasets. As a variant of Painter, SegGPT [WZC+23] introduces a novel random coloring approach for in-context training to improve the model's generalization ability. By prompting spatial queries, *e.g.*, points, and text queries, *e.g.*, textual prompts, SEEM [ZYZ+23] performs various segmentation tasks effectively. More recently, PerSAM and PerSAM-F [ZJG+23] adapt SAM for personalized segmentation and video object segmentation without training or with two trainable parameters. This work presents Matcher, a training-free framework for segmenting anything with one shot. Unlike these methods, Matcher demonstrates impressive generalization performance across various segmentation tasks by integrating different foundation models.

3 Method

Matcher is a training-free framework that segments anything with one shot by integrating an all-purpose feature extraction model (*e.g.*, DINOv2 [ODM+23], CLIP [RKH+21], and MAE [HCX+22]) and a class-agnostic segmentation model (SAM) [KMR+23]. For a given reference image $\mathbf{x}_r$ and

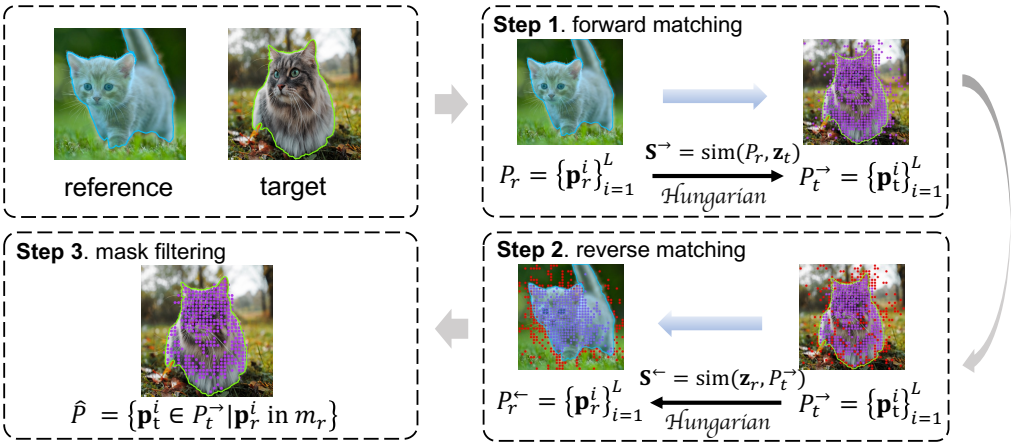


Figure 2: Illustration of the proposed bidirectional matching. Bidirectional matching consists of three steps: forward matching, reverse matching, and mask filtering. Purple points denote the matched points. Red points denote the outliers.

mask m_r, Matcher can segment the objects or parts of a target image $\mathbf{x}_t$ with the same semantics. The overview of Matcher is depicted in Fig. 1. Our framework consists of three components: Correspondence Matrix Extraction (CME), Prompts Generation (PG), and Controllable Masks Generation (CMG). First, Matcher extracts a correspondence matrix by calculating the similarity between the image features of $\mathbf{x}_r$ and $\mathbf{x}_t$. Then, we conduct patch-level matching, followed by sampling multiple groups of prompts (including points and boxes) from the matched points. These prompts serve as inputs to SAM, enabling the generation of mask proposals. Finally, we perform an instance-level matching between the reference mask and mask proposals to select high-quality masks. We elaborate on the three components in the following subsections.

3.1 Correspondence Matrix Extraction

We rely on off-the-self image encoders to extract features for both the reference and target images. Given inputs $\mathbf{x}_r$ and $\mathbf{x}_t$, the encoder outputs patch-level features $\mathbf{z}_r, \mathbf{z}_t \in \mathbb{R}^{H \times W \times C}$. Patch-wise similarity between the two features is computed to discovery the best matching regions of the reference mask on the target image. We define a correspondence matrix $\mathbf{S} \in \mathbb{R}^{HW \times HW}$ as follows,

$$(\mathbf{S})_{ij} = \frac{\mathbf{z}_r^i \cdot \mathbf{z}_t^j}{\|\mathbf{z}_r^i\| \times \|\mathbf{z}_t^j\|}, \tag{1}$$

where $(\mathbf{S})_{ij}$ denotes the cosine similarity between i-th patch feature $\mathbf{z}_r^i$ of $\mathbf{z}_r$ and j-th patch feature $\mathbf{z}_t^j$ of $\mathbf{z}_t$. We can denote the above formulation in a compact form as $\mathbf{S} = \mathrm{sim}(\mathbf{z}_r, \mathbf{z}_t)$.

Ideally, the matched patches should have the highest similarity. This could be challenging in practice, since the reference and target objects could have different appearances or even belong to different categories. This requires the encoder to embed rich and detailed information in these features.

3.2 Prompts Generation

Given the dense correspondence matrix, we can get a coarse segmentation mask by selecting the most similar patches in the target image. However, this naive approach leads to inaccurate, fragmented result with many outliers. Hence, we use the correspondence feature to generate high quality point and box guidance for promptable segmentation. The process involves a bidirectional patch matching and a diverse prompt sampler.

Patch-Level Matching The encoder tends to produce wrong matches in hard cases such as ambiguous context and multiple instances. We propose a bidirectional matching strategy to eliminate the matching outliers.

- As shown in Fig. 2, we first perform bipartite matching between the points on the reference mask $P_r = \{\mathbf{p}_r^i\}_{i=1}^L$ and $\mathbf{z}_t$ to obtain the forward matched points on the target image $P_t^{\rightarrow} = \{\mathbf{p}_t^i\}_{i=1}^L$ using the forward correspondence matrix $\mathbf{S}^{\rightarrow} = \mathrm{sim}(P_r, \mathbf{z}_t)$.
- Then, we perform another bipartite matching, named the reverse matching between $P_t^{\rightarrow}$ and $\mathbf{z}_r$ to obtain the reverse matched points on the reference image $P_r^{\leftarrow} = \{\mathbf{p}_r^i\}_{i=1}^L$ using the reverse correspondence matrix $\mathbf{S}^{\leftarrow} = \mathrm{sim}(\mathbf{z}_r, P_t^{\rightarrow})$.
- Finally, we filter out the points in the forward set if the corresponding reverse points are not on the reference mask. The final matched points are $\hat{P} = \{\mathbf{p}_t^i \in P_t^{\rightarrow} | \mathbf{p}_r^i \text{ in } m_r\}$.

Robust Prompt Sampler

To support robust segmentation with various semantic granularity, from parts and whole to multiple instances, we propose a robust prompt sampler to encourage diverse and meaningful mask proposals. We first cluster the matched points $\hat{P}$ based on their locations into K clusters $\hat{P}_k$ with k-means++ [AV07]. Then the following three types of subsets are sampled as prompts:

- Part-level prompts are sampled within each cluster $P^p \subset \hat{P}_k$;
- Instance-level prompts are sampled within all matched points $P^i \subset \hat{P}$;
- Global prompts are sampled within the set of cluster centers $P^g \subset C$ to encourage coverage, where $C = \{c_1, c_2, \ldots, c_k\}$ are the cluster centers.

Finally, we add the bounding box of $\hat{P}$ as a box proposal. In practice, we find this strategy not only increases the diversity of mask proposals but also suppresses fragmented false-positive masks induced by matching outliers.

3.3 Controllable Masks Generation

The edge features of an object extracted by the image encoder can confuse background information, inducing some indistinguishable outliers. These outliers can be selected to generate some false-positive masks. To overcome this difficulty, we further select high-quality masks from the mask proposals via an instance-level matching module and then merge the selected masks to obtain the final target mask.

Instance-Level Matching We perform the instance-level matching between the reference mask and mask proposals to select great masks. We formulate the matching to the Optimal Transport (OT) problem and employ the Earth Mover's Distance (EMD) to compute a structural distance between dense semantic features inside the masks to determine mask relevance. The cost matrix of the OT problem can be calculated by $\mathbf{C} = \frac{1}{2}(1 - \mathbf{S})$. We use the method proposed in [BVDPPH11] to calculate the EMD, noted as *emd*.

In addition, we propose two other mask proposal metrics, *i.e.*, *purity* $= \frac{Num(\hat{P}_{mp})}{Area(m_p)}$ and *coverage* $= \frac{Num(\hat{P}_{mp})}{Num(\hat{P})}$, to assess the quality of the mask proposals simultaneously, where $\hat{P}_{mp} = \{\mathbf{p}_t^i \in P_t^{\rightarrow} | \mathbf{p}_t^i \text{ in } m_p\}$, $Num(\cdot)$ represents the number of points, $Area(\cdot)$ represents the area of the mask, and m_p is the mask proposal. A higher degree of *purity* promotes the selection of part-level masks, while a higher degree of *coverage* promotes the selection of instance-level masks. The false-positive mask fragments can be filtered using the proposed metrics through appropriate thresholds, followed by a score-based selection process to identify the top-k highest-quality masks

$$\textit{score} = \alpha \cdot (1 - \textit{emd}) + \beta \cdot \textit{purity} \cdot \textit{coverage}^{\lambda}, \tag{2}$$

where α, β, and λ are regulation coefficients between different metrics.

Controllable Masks Merging By manipulating the number of merged masks, Matcher can produce controllable mask output to instances of the same semantics in the target image.

Methods	Venue	COCO-20^i F0	F1	F2	F3	mean	FSS-1000 mIoU	LVIS-92^i mean mIoU
specialist model								
HSNet [MKC21]	ICCV'21	37.2	44.1	42.4	41.3	41.2	86.5	17.4
VAT [HCN+22]	ECCV'22	39.0	43.8	42.6	39.7	41.3	90.3	18.5
FPTrans [ZSYC22]	NeurIPS'22	44.4	48.9	50.6	44.0	47.0	-	-
MSANet* [ISB22]	arXiv'22	47.8	57.4	48.7	50.5	51.1	-	-
generalist model								
Painter [WWC+23]	CVPR'23	31.2	35.3	33.5	32.4	33.1	61.7	10.5
SegGPT [WZC+23]	arXiv'23	56.3	57.4	58.9	51.7	56.1	85.6	18.6
PerSAM†‡ [ZJG+23]	arXiv'23	23.1	23.6	22.0	23.4	23.0	71.2	11.5
PerSAM-F‡	arXiv'23	22.3	24.0	23.4	24.1	23.5	75.6	12.3
Matcher†‡	this work	**52.7**	**53.5**	**52.6**	**52.1**	**52.7**	**87.0**	**33.0**

Table 1: Results of one-shot semantic segmentation on COCO-20^i, FSS-1000, and LVIS-92^i. Gray indicates the model is trained by in-domain datasets. * indicates the state-of-the-art specialist model. † indicates the training-free method. ‡ indicates the method using SAM.

4 Experiments

4.1 Experiments Setting

Vision Foundation Models We use DINOv2 [ODM+23] with a ViT-L/14 [DBK+20] as the default image encoder of Matcher. Benefiting from large-scale discriminative self-supervised learning at both the image and patch level, DINOv2 has impressive patch-level representation ability, which promotes exact patch matching between different images. We also conduct comparison experiments on CLIP [RKH+21] ViT-L/14 and MAE [HCX+22] ViT-H/14. We use the Segment Anything Model (SAM) [KMR+23] with ViT-H as the segmenter of Matcher. Pre-trained with 1B masks and 11M images, SAM emerges with impressive zero-shot segmentation performance. Combining these vision foundation models has the enormous potential to touch open-world image understanding. **In all experiments, we do not perform any training for the Matcher.** More implementation details are provided in Appendix A.

4.2 One-shot Semantic Segmentation

Datasets For one-shot semantic segmentation, we evaluate the performance of Matcher on COCO-20^i [NT19], FSS-1000 [LWC+20], and LVIS-92^i. COCO-20^i partitions the 80 categories of the MSCOCO dataset [LMB+14] into four cross-validation folds, each containing 60 training classes and 20 test classes. FSS-1000 consists of mask-annotated images from 1,000 classes, with 520, 240, and 240 classes in the training, validation, and test sets, respectively. We verify Matcher on the test sets of COCO-20^i and FSS-1000 following the evaluation scheme of [MKC21]. Note that, different from specialist models, we do not train Matcher on these datasets. In addition, based on the LVIS dataset [GDG19], we create LVIS-92^i, a more challenging benchmark for evaluating the generalization of a model across datasets. After removing the classes with less than two images, we retained a total of 920 classes for further analysis. These classes were then divided into 10 equal folds for testing purposes. For each fold, we randomly sample a reference image and a target image for evaluation and conduct 2,300 episodes.

Results We compare the Matcher against a variety of specialist models, such as HSNet [MKC21], VAT [HCN+22], FPTrans [ZSYC22], and MSANet [ISB22], as well as generalist models like Painter [WWC+23], SegGPT [WZC+23], and PerSAM [ZJG+23]. As shown in Table 1, for COCO-20^i, Matcher achieves **52.7%** mean mIoU without training, surpassing the state-of-the-art specialist model MSANet by 1.6% and achieving comparable with SegGPT. Note that the training data of SegGPT include COCO. For FSS-1000, Matcher exhibits highly competitive performance compared with specialist models and surpasses all generalist models. Furthermore, Matcher outperforms training-free PerSAM and fine-tuning PerSAM-F by a significant margin (**+29.2%** mean mIoU on COCO-20^i, **+11.4%** mIoU on FSS-1000, and **+10.7%** mean mIoU on LVIS-92^i), suggesting that depending solely on SAM results in limited generalization capabilities for semantic tasks. For LVIS-92^i, we compare the cross-dataset generalization abilities of Matcher and other models. For specialist models, we report the average performance of four pre-trained models on COCO-20^i. Our

Methods	Venue	PASCAL-Part					PACO-Part				
		animals	indoor	person	vehicles	mean	F0	F1	F2	F3	mean
HSNet [MKC21]	ICCV'21	21.2	53.0	20.2	35.1	32.4	20.8	21.3	25.5	22.6	22.6
VAT [HCN+22]	ECCV'22	21.5	55.9	20.7	36.1	33.6	22.0	22.9	26.0	23.1	23.5
Painter [WWC+23]	CVPR'23	20.2	49.5	17.6	34.4	30.4	13.7	12.5	15.0	15.1	14.1
PerSAM†‡ [ZJG+23]	arXiv'23	19.9	51.8	18.6	32.0	30.1	19.4	20.5	23.8	21.2	21.2
Matcher†‡	this work	**37.1**	**56.3**	**32.4**	**45.7**	**42.9**	**32.7**	**35.6**	**36.5**	**34.1**	**34.7**

Table 2: Results of one-shot part segmentation on PASCAL-Part and PACO-Part. † indicates the training-free method. ‡ indicates the method using SAM.

results indicate that Matcher exhibits robust generalization capabilities that are not present in the other models. The detailed results of LVIS-92^i are provided in Appendix C.

4.3 One-shot Object Part Segmentation

Datasets Requiring a fine-grained understanding of objects, object part segmentation is a more challenging task than segmenting an object. We build two benchmarks to evaluate the performance of Matcher on one-shot part segmentation, *i.e.*, PASCAL-Part and PACO-Part. Based on PASCAL VOC 2010 [EVGW+10] and its body part annotations [CML+14], we build the PASCAL-Part dataset following [MAV20]. The dataset consists of four superclasses, *i.e.*, animals, indoor, person, and vehicles. There are five subclasses for animals, three for indoor, one for person, and six for vehicles. There are 56 different object parts in total. PACO [RKP+23] is a newly released dataset that provides 75 object categories and 456 object part categories. Based on the PACO dataset, we build the more difficult PACO-Part benchmark for one-shot object part segmentation. We filter the object parts whose area is minimal and those with less than two images, resulting in 303 remaining object parts. We split these parts into four folds, each with about 76 different object parts. We crop all objects out with their bounding box to evaluate the one-shot part segmentation on both two datasets. More details are provided in Appendix B.

Results We compare our Matcher with HSNet, VAT, Painter, and PerSAM. For HSNet and VAT, we use the models pre-trained on PASCAL-5^i [SBL+17] and COCO-20^i for PASCAL-Part and PACO-Part, respectively. As shown in Table 2, the results demonstrate that Matcher outperforms all previous methods by a large margin. Specifically, Matcher outperforms the SAM-based PerSAM +**12.8%** mean mIoU on PASCAL-Part and +**13.5%** on PACO-Part, respectively. SAM has shown the potential to segment any object into three levels: whole, part, and subpart [KMR+23]. However, it cannot distinguish these ambiguity masks due to the lack of semantics. This suggests that SAM alone cannot work well on one-shot object part segmentation. Our method empowers SAM for semantic tasks by combining it with an all-purpose feature extractor and performs effective generalization performance on fine-grained object part segmentation tasks with one-shot.

4.4 Video Object Segmentation

Datasets Video object segmentation (VOS) aims to segment a specific object in video frames. Following [WZC+23], we evaluate Matcher on the validation split of two datasets, *i.e.*, DAVIS 2017 val [PTPC+17], and DAVIS 2016 val [PPTM+16], under the semi-supervised VOS setting. Two commonly used metrics in VOS, the J score and the F score, are used for evaluation.

Details In order to track particular moving objects in a video, we maintain a reference memory containing features and the intermediate predictions of the previous frames in Matcher. We determine which frame to retain in the memory according to the *score* (see subsection 3.3) of the frames. Considering that objects are more likely to be similar to those in adjacent frames, we apply a decay ratio decreasing by time to the *score*. We fix the given reference image and mask in the memory to avoid failing when some objects disappear in intermediate frames and reappear later.

Results We compare Matcher with the models trained with or without video data on different datasets in Table 3. The results show that Matcher can achieve competitive performance compared with the models trained with video data. Moreover, Matcher outperforms the models trained without video data, *e.g.*, SegGPT and PerSAM-F, on both two datasets. These results suggest that Matcher can effectively generalize to VOS tasks without training.

Methods	Venue	DAVIS 2017 val			DAVIS 2016 val		
		$J\&F$	J	F	$J\&F$	J	F
with video data							
AGAME [JDB+19]	CVPR'19	70.0	67.2	72.7	-	-	-
AGSS [LQJ19]	ICCV'19	67.4	64.9	69.9	-	-	-
AFB-URR [LLJC20]	NeurIPS'20	74.6	73.0	76.1	-	-	-
SWEM [LYL+22]	CVPR'22	84.3	81.2	87.4	91.3	89.9	92.6
XMem [CS22]	ECCV'22	87.7	84.0	91.4	92.0	90.7	93.2
AOT [YWY21]	NeurIPS'22	85.4	82.4	88.4	92.0	90.7	93.3
without video data							
Painter [WWC+23]	CVPR'23	34.6	28.5	40.8	70.3	69.6	70.9
SegGPT [WZC+23]	arXiv'23	75.6	72.5	78.6	83.7	83.6	83.8
PerSAM†‡ [ZJG+23]	arXiv'23	60.3	56.6	63.9	-	-	-
PerSAM-F‡		71.9	69.0	74.8	-	-	-
Matcher†‡	this work	**79.5**	**76.5**	**82.6**	**86.1**	**85.2**	**86.7**

Table 3: Results of video object segmentation on DAVIS 2017 val, and DAVIS 2016 val. Gray indicates the model is trained on target datasets with video data. † indicates the training-free method. ‡ indicates the method using SAM.

Encoder	COCO-20^i mean mIoU	FSS-1000 mIoU	DAVIS 2017 $J\&F$
MAE	18.8	71.9	69.5
CLIP	32.2	77.4	73.9
DINOv2	**52.7**	**87.0**	**79.5**

(a) Effect of different image encoders.

ILM	COCO-20^i mean mIoU	FSS-1000 mIoU	DAVIS 2017 $J\&F$
	29.0	76.2	39.9
✓	**52.7**	**87.0**	**79.5**

(b) Ablation study of ILM.

Strategy	COCO-20^i mean mIoU	FSS-1000 mIoU	DAVIS 2017 $J\&F$
forward	50.6	81.1	73.5
reverse	21.4	47.7	41.3
bidirectional	**52.7**	**87.0**	**79.5**

(c) Ablation study of bidirectional matching.

emd	*p&c*	COCO-20^i mean mIoU	FSS-1000 mIoU	DAVIS 2017 $J\&F$
✓		51.3	86.3	67.5
	✓	35.3	86.3	76.3
✓	✓	**52.7**	**87.0**	**79.5**

(d) Ablation study of different mask proposal metrics.

Frames	DAVIS 2017			
	1	2	4	6
$J\&F$	73.5	74.4	**79.5**	78.0
J	70.0	70.5	**76.5**	74.9
F	77.5	78.2	**82.6**	81.1

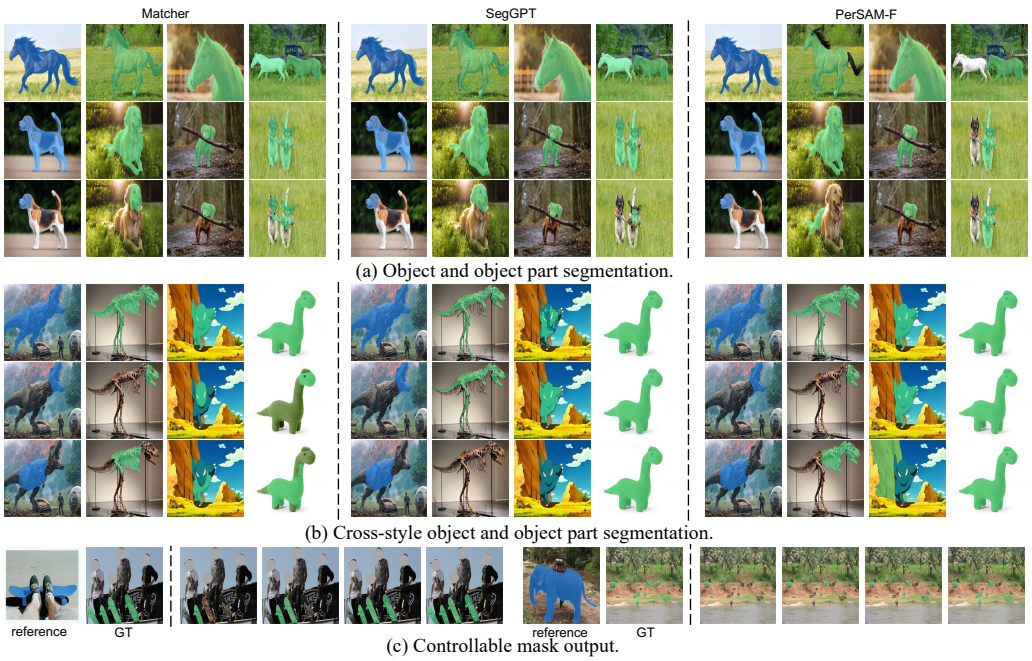
(e) Effect of the number of frames for VOS.

Table 4: Ablation study. We report the mean mIoU of four folds on COCO-20^i, mIoU on FSS-1000, and $J\&F$ on DAVIS 2017 val. Default setting settings are marked in Gray.

4.5 Ablation Study

As shown in Table 4, we conduct ablation studies on both the difficult COCO-20^i dataset and the simple FSS-1000 dataset for one-shot semantic segmentation and DAVIS 2017 val for video object segmentation to sufficiently verify the effectiveness of our proposed components. In this subsection, we explore the effects of different image encoders, matching modules (ILM), patch-level matching strategies, and different mask proposal metrics.

Effect of Different Image Encoders Table 4a shows the comparison experiments of CLIP, MAE, and DINOv2. DINOv2 achieves the best performance on all datasets. Because the text-image contrastive pre-training limits learning complex pixel-level information, CLIP cannot precisely match image patches. Although MAE can extract pixel-level features by masked image modeling, it performs poorly. We suspect that the patch-level features extracted by MAE confuse the information about the surrounding patches, resulting in mistaken feature matching. In contrast, pre-trained by image-level and patch-level discriminative self-supervised learning, DIVOv2 extracts all-purpose visual features and exhibit impressive patch-level feature matching ability. Based on the all-purpose visual features, Matcher exhibits robust generalization performance without training.



(a) Object and object part segmentation.

(b) Cross-style object and object part segmentation.

(c) Controllable mask output.

Figure 3: Qualitative results of one-shot segmentation.

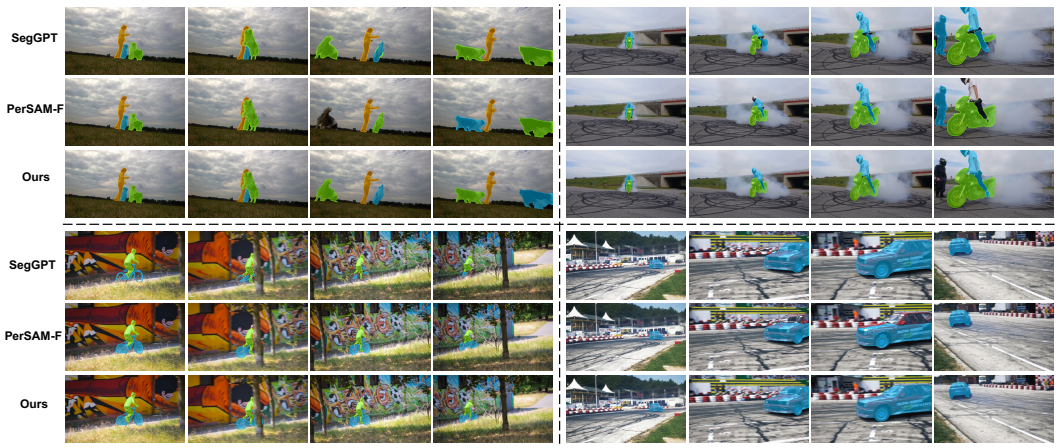


Figure 4: Qualitative results of video object segmentation on DAVIS 2017.

Ablation Study of ILM Patch-level matching (PLM) and instance-level matching (ILM) are the vital components of Matcher that bridge the gap between the image encoder and SAM to solve various few-shot perception tasks training-free. As shown in Table 4b, PLM builds the connection between matching and segmenting and empowers Matcher with the capability of performing various few-shot perception tasks training-free. And ILM enhances this capability by a large margin.

Ablation Study of Bidirectional Matching As shown in Table 4c, we explore the effects of the forward matching and the reverse matching of the proposed bidirectional matching. For the reverse matching, because the matched points $P_t^{\rightarrow}$ (see subsection 3.2) are unavailable when performing reverse matching directly, we perform the reverse matching between $\mathbf{z}_t$ and $\mathbf{z}_r$. The reverse matching (line 2) achieves poor performance in all segmentation tasks. Compared with the forward matching (line 1), our bidirectional matching strategy improves the performance by +2.1% mean mIoU on COCO-20^i, by +5.9% mIoU on FSS-1000, and by +6.0% $J\&F$ on DAVIS 2017. These significant improvements show the effectiveness of the proposed bidirectional matching strategy.

Ablation Study of Different Mask Proposal Metrics. As shown in Table 4d, *emd* is more effective on the complex COCO-20^i dataset. *emd* evaluates the patch-level feature similarity between the

mask proposals and the reference mask that encourages matching all mask proposals with the same category. In contrast, by using *purity* and *coverage*, Matcher can achieve great performance on DAVIS 2017. Compared with *emd*, *purity* and *coverage* are introduced to encourage selecting high-quality mask proposals. Combining these metrics to estimate mask proposals, Matcher can achieve better performance in various segmentation tasks without training.

Effect of the Number of Frames for VOS As shown in Table 4e, we also explore the effect of the number of frames on DAVIS 2017 val. The performance of Matcher can be improved as the number of frames increases, and the optimal performance is achieved when using four frames.

4.6 Qualitative Results

To demonstrate the generalization of our Matcher, we visualize the qualitative results of one-shot segmentation in Fig. 3 from three views, *i.e.*, object and object part segmentation, cross-style object and object part segmentation, and controllable mask output. Our Matcher can achieve higher-quality objects and parts masks than SegGPT and PerSAM-F. Better results on cross-style segmentation show the impressive generalization of Matcher due to effective all-feature matching. In addition, by manipulating the number of merged masks, Macther supports multiple instances with the same semantics. Fig. 4 shows qualitative results of VOS on DAVIS 2017. The remarkable results demonstrate that Matcher can effectively unleash the ability of foundation models to improve both the segmentation quality and open-set generality.

5 Conclusion

In this paper, we present Matcher, a training-free framework integrating an all-purpose feature extraction model and a class-agnostic segmentation model for solving various few-shot segmentation tasks. Combining these foundation models properly leads to positive synergies, and Matcher emerges complex capabilities beyond individual models. The introduced universal components, *i.e.*, bidirectional matching, robust prompt sampler, and instance-level matching, can effectively unleash the ability of these foundation models. Our experiments demonstrate the powerful performance of Matcher for various few-shot segmentation tasks, and our visualization results show open-world generality and flexibility on images in the wild.

Limitation and Broader Impact While Matcher demonstrates impressive performance for semantic-level segmentation, *e.g.*, one-shot semantic segmentation and one-shot object part segmentation, it has relatively limited instance-level matching inherited from the image encoder, which restrains its performance for instance segmentation. However, the comparable VOS performance and the visualization of controllable mask output demonstrates that Matcher has the potential for instance-level segmentation. We will explore it in future work. Our work can unleash the potential of different foundation models for various visual tasks. In addition, our Matcher is built upon open-source foundation models without training, significantly reducing carbon emissions. We do not foresee any obvious undesirable ethical or social impacts now.

Acknowledgment

This work was supported by National Key R&D Program of China (No. 2022ZD0118700).

Appendix

A Implementation Details

We use DINOv2 [ODM+23] with a ViT-L/14 [DBK+20] as the default image encoder of Matcher. And we use the Segment Anything Model (SAM) [KMR+23] with ViT-H as the segmenter of Matcher. In all experiments, we do not perform any training for the Matcher. We set input image sizes are 518×518 for one-shot semantic segmentation and object part segmentation and 896×504 for video object segmentation. We conduct experiments from three semantic granularity for one-shot semantic segmentation, *i.e.*, parts (PASCAL-Part and PACO-Part), whole (FSS-1000), and multiple instances (COCO-20^i and LVIS-92^i). For COCO-20^i and LVIS-92^i, we sample the instance-level

Methods	Venue	LVIS-92^i										
		F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	mean
HSNet [MKC21]	ICCV'21	6.8	18.2	18.0	18.7	16.1	18.5	19.1	16.4	18.0	15.6	17.4
VAT [HCN+22]	ECCV'22	18.0	18.8	19.2	19.1	16.1	19.7	20.1	17.8	19.3	16.8	18.5
Painter [WWC+23]	CVPR'23	10.5	11.3	10.8	8.9	11.0	11.3	13.4	8.5	10.2	9.3	10.5
SegGPT [WZC+23]	arXiv'23	17.5	20.1	19.9	22.4	16.2	20.0	16.7	17.9	18.7	16.8	18.6
PerSAM†‡ [ZJG+23]	arXiv'23	12.5	13.0	11.6	11.7	8.5	12.8	12.1	11.4	11.9	9.5	11.5
PerSAM-F‡		13.5	13.8	12.1	12.3	10.0	13.2	13.2	12.1	12.1	10.2	12.3
Matcher†‡	this work	**31.4**	**30.9**	**33.7**	**38.1**	**30.5**	**32.5**	**35.9**	**34.2**	**33.0**	**29.7**	**33.0**

Table 5: One-shot semantic segmentation on LVIS-92^i. † indicates training-free method. ‡ indicates the method using SAM.

points from the matched points and dense image points to encourage SAM to output more instance masks. We set the filtering thresholds *emd* and *purity* to 0.67, 0.02 and set α, β and λ to 1.0, 0.0, and 0.0, respectively. For FSS-1000, we sample the global prompts from centers. We set α, β, and λ to 0.8, 0.2, and 1.0, respectively. We sample the points from the matched points for PASCAL-Part and PACO-Part. We set the filtering threshold *coverage* to 0.3 and set α, β and λ to 0.5, 0.5, and 0.0, respectively. For video object segmentation We sample the global prompts from centers. We set the filtering threshold *emd* to 0.75 and set α, β, and λ to 0.4, 1.0, and 1.0.

B Dataset Details

PASCAL-Part Based on PASCAL VOC 2010 [EVGW+10] and its body part annotations [CML+14], we build the PASCAL-Part dataset following [MAV20]. Table 6 shows the part taxonomy of PASCAL-Part dataset. The dataset consists of four superclasses, *i.e.*, animals, indoor, person, and vehicles. There are five subclasses for animals (bird, cat, cow, dog, horse, sheep), three for indoor (bottle, potted plant, tv monitor), one for person (person), and six for vehicles (aeroplane, bicycle, bus, car, motorbike, train). There are 56 different object parts in total.

PACO-Part Based on the PACO [RKP+23] dataset, we build the more difficult PACO-Part benchmark for one-shot object part segmentation. We filter the object parts whose area is minimal and those with less than two images, resulting in 303 remaining object parts. Table 7 shows the part taxonomy of PACO-Part dataset. We split these parts into four folds, each with about 76 different object parts.

Superclasses	Subclasses	Parts
animals	bird	face, leg, neck, tail, torso, wings
	cat	face, leg, neck, tail, torso
	cow	face, leg, neck, tail, torso
	dog	face, leg, neck, tail, torso
	horse	face, leg, neck, tail, torso
	sheep	face, leg, neck, tail, torso
indoor	bottle	body
	potted plant	plant, pot
	tv monitor	screen
person	person	face, arm & hand, leg, neck, torso
vehicles	aeroplane	body, engine, wheel, wings
	bicycle	wheel
	bus	door, vehicle side, wheel, windows
	car	door, vehicle side, wheel, windows
	motorbike	wheel
	train	train coach, train head

Table 6: Part taxonomy of PASCAL-Part

C Additional Results

One-shot Semantic Segmentation on LVIS-92^i We compare the cross-dataset generalization abilities between the Matcher with HSNet [MKC21], VAT [HCN+22], Painter [WWC+23], SegGPT [WZC+23], and PerSAM [ZJG+23]. The results are shown in Table 5. For HSNet and VAT, we report the average performance of four pre-trained models on COCO-20^i. Our results indicate that Matcher exhibits robust generalization capabilities that are not present in the other models.

Visualizations We provide more visualizations for one-shot semantic segmentation in Fig. 5, one-shot object part segmentation in Fig. 6 and Fig. 7, controllable mask output in Fig. 8, and video object segmentation in Fig. 9. The remarkable results demonstrate that Matcher can effectively unleash the ability of foundation models to improve both the segmentation quality and open-set generality.

Fold	Parts
0	bench:arm, laptop_computer:back, bowl:base, handbag:base, basket:base, chair:base, glass_(drink_container):base, cellular_telephone:bezel, guitar:body, bucket:body, can:body, soap:body, vase:body, crate:bottom, box:bottom, glass_(drink_container):bottom, basket:bottom, lamp:bulb, television_set:button, watch:case, bottle:closure, book:cover, table:drawer, pillow:embroidery, car_(automobile):fender, dog:foot, bicycle:fork, bicycle:gear, clock:hand, bucket:handle, basket:handle, spoon:handle, bicycle:handlebar, guitar:headstock, sweater:hem, trash_can:hole, bucket:inner_body, hat:inner_side, microwave_oven:inner_side, tray:inner_side, pliers:jaw, laptop_computer:keyboard, shoe:lace, bench:leg, can:lid, fan:light, car_(automobile):mirror, spoon:neck, sweater:neckband, tray:outer_side, bicycle:pedal, can:pull_tab, shoe:quarter, can:rim, mug:rim, pan_(for_cooking):rim, tray:rim, basket:rim, car_(automobile):runningboard, laptop_computer:screen, chair:seat, bicycle:seat_stay, lamp:shade_inner_side, sweater:shoulder, television_set:side, sweater:sleeve, blender:spout, jar:sticker, helmet:strap, table:stretcher, blender:switch, bench:table_top, plastic_bag:text, shoe:tongue, television_set:top, bicycle:top_tube, hat:visor, car_(automobile):wheel, car_(automobile):wiper
1	chair:apron, chair:back, bench:back, fan:base, cup:base, pan_(for_cooking):base, laptop_computer:base_panel, knife:blade, scissors:blade, bowl:body, sweater:body, handbag:body, mouse_(computer_equipment):body, towel:body, dog:body, bowl:bottom, plate:bottom, television_set:bottom, spoon:bowl, car_(automobile):bumper, cellular_telephone:button, laptop_computer:cable, fan:canopy, bottle:cap, clock:case, pipe:colied_tube, sweater:cuff, microwave_oven:dial, mug:drawing, vase:foot, car_(automobile):grille, plastic_bag:handle, scissors:handle, handbag:handle, mug:handle, cup:handle, pan_(for_cooking):handle, dog:head, bicycle:head_tube, towel:hem, car_(automobile):hood, plastic_bag:inner_body, wallet:inner_body, glass_(drink_container):inner_body, crate:inner_side, pan_(for_cooking):inner_side, plate:inner_wall, soap:label, chair:leg, crate:lid, laptop_computer:logo, broom:lower_bristles, fan:motor, vase:neck, dog:nose, shoe:outsole, lamp:pipe, chair:rail, bucket:rim, bowl:rim, car_(automobile):rim, tape_(sticky_cloth_or_paper):roll, bicycle:saddle, scissors:screw, bench:seat, bicycle:seat_tube, soap:shoulder, box:side, carton:side, earphone:slider, bicycle:stem, chair:stile, bench:stretcher, dog:tail, mug:text, bottle:top, table:top, laptop_computer:touchpad, shoe:vamp, helmet:visor, car_(automobile):window, mouse_(computer_equipment):wire
2	table:apron, telephone:back_cover, plate:base, kettle:base, blender:base, bicycle:basket, fan:blade, plastic_bag:body, trash_can:body, plate:body, mug:body, kettle:body, towel:border, mug:bottom, telephone:button, microwave_oven:control_panel, microwave_oven:door_handle, dog:ear, helmet:face_shield, scissors:finger_hole, wallet:flap, mirror:frame, kettle:handle, blender:handle, earphone:headband, earphone:housing, bowl:inner_body, trash_can:inner_body, helmet:inner_side, basket:inner_side, calculator:key, bottle:label, mouse_(computer_equipment):left_button, dog:leg, box:lid, trash_can:lid, vase:mouth, pipe:nozzle, slipper_(footwear):outsole, fan:pedestal_column, ladder:rail, hat:rim, plate:rim, trash_can:rim, bottle:ring, car_(automobile):roof, telephone:screen, mouse_(computer_equipment):scroll_wheel, stool:seat, lamp:shade, bottle:shoulder, microwave_oven:side, basket:side, chair:spindle, hat:strap, belt:strap, car_(automobile):taillight, towel:terry_bar, newspaper:text, microwave_oven:time_display, shoe:toe_box, microwave_oven:top, car_(automobile):trunk, slipper_(footwear):vamp, car_(automobile):windowpane, sweater:yoke
3	chair:arm, remote_control:back, cellular_telephone:back_cover, bottle:base, bucket:base, television_set:base, jar:base, tray:base, lamp:base, telephone:bezel, bottle:body, pencil:body, scarf:body, calculator:body, jar:body, glass_(drink_container):body, bottle:bottom, pan_(for_cooking):bottom, tray:bottom, remote_control:button, bucket:cover, basket:cover, bicycle:down_tube, earphone:ear_pads, dog:eye, guitar:fingerboard, blender:food_cup, stool:footrest, scarf:fringes, knife:handle, vase:handle, car_(automobile):headlight, mug:inner_body, jar:inner_body, cup:inner_body, box:inner_side, carton:inner_side, trash_can:label, table:leg, stool:leg, jar:lid, kettle:lid, car_(automobile):logo, bucket:loop, bottle:neck, dog:neck, pipe:nozzle_stem, book:page, mouse_(computer_equipment):right_button, handbag:rim, jar:rim, glass_(drink_container):rim, cup:rim, cellular_telephone:screen, blender:seal_ring, lamp:shade_cap, table:shelf, crate:side, pan_(for_cooking):side, mouse_(computer_equipment):side_button, chair:skirt, car_(automobile):splashboard, bottle:spout, ladder:step, watch:strap, chair:stretcher, chair:swivel, can:text, jar:text, spoon:tip, slipper_(footwear):toe_box, blender:vapour_cover, chair:wheel, bicycle:wheel, car_(automobile):windshield, handbag:zip

Table 7: Part taxonomy of PACO-Part

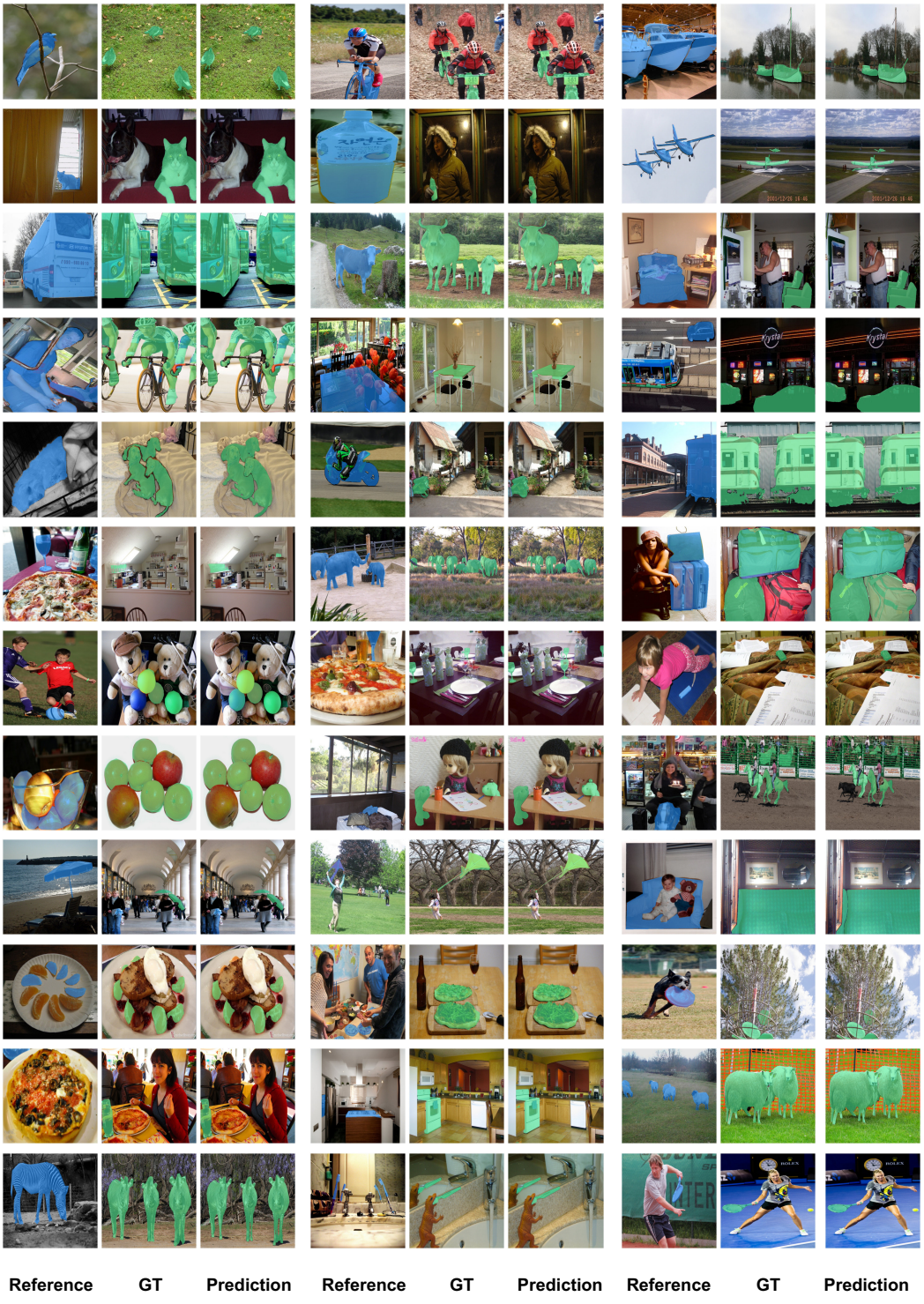


Figure 5: Visualization of one-shot semantic segmentation.

Figure 6: Visualization of one-shot object part segmentation on PASCAL-Part.

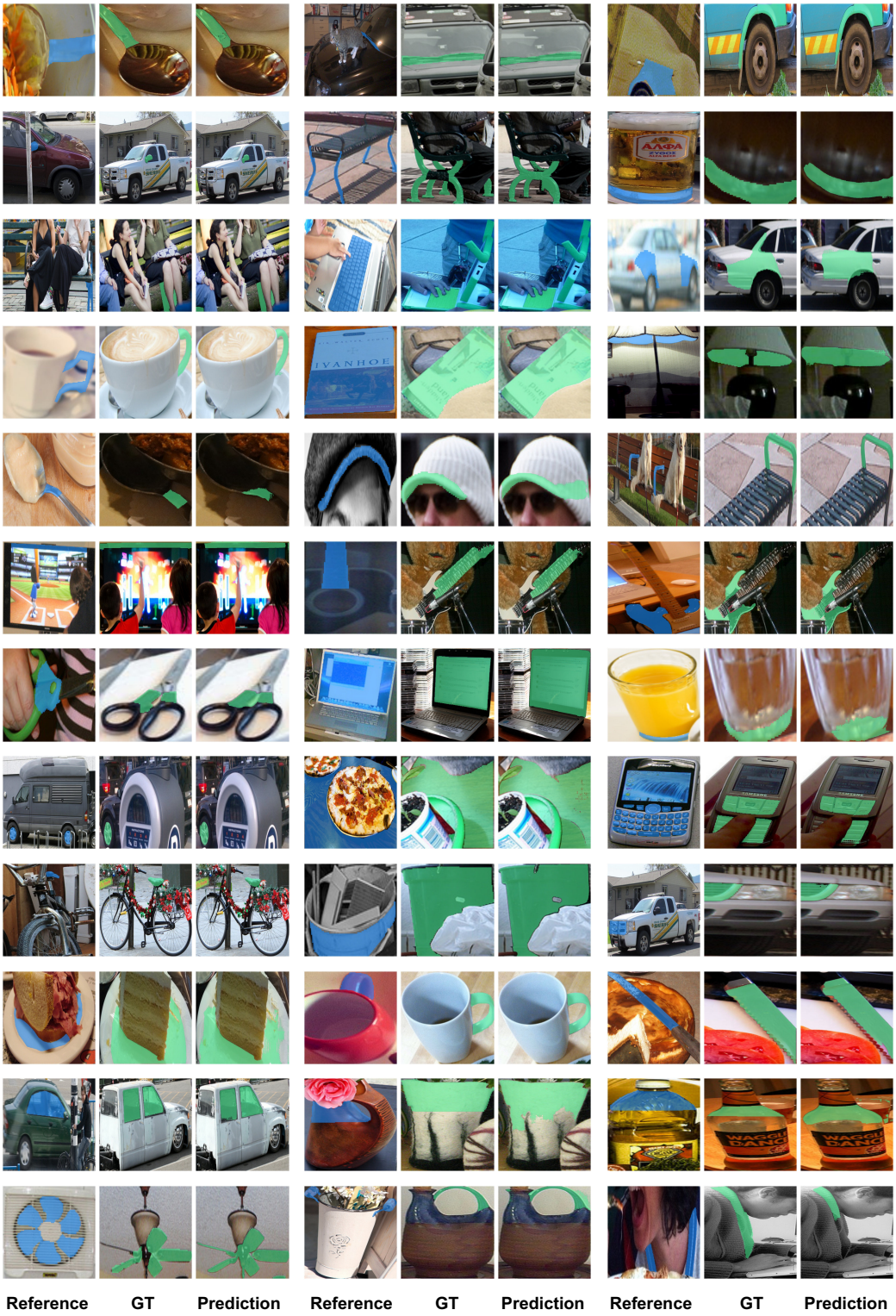


Figure 7: Visualization of one-shot object part segmentation on PACO-Part.

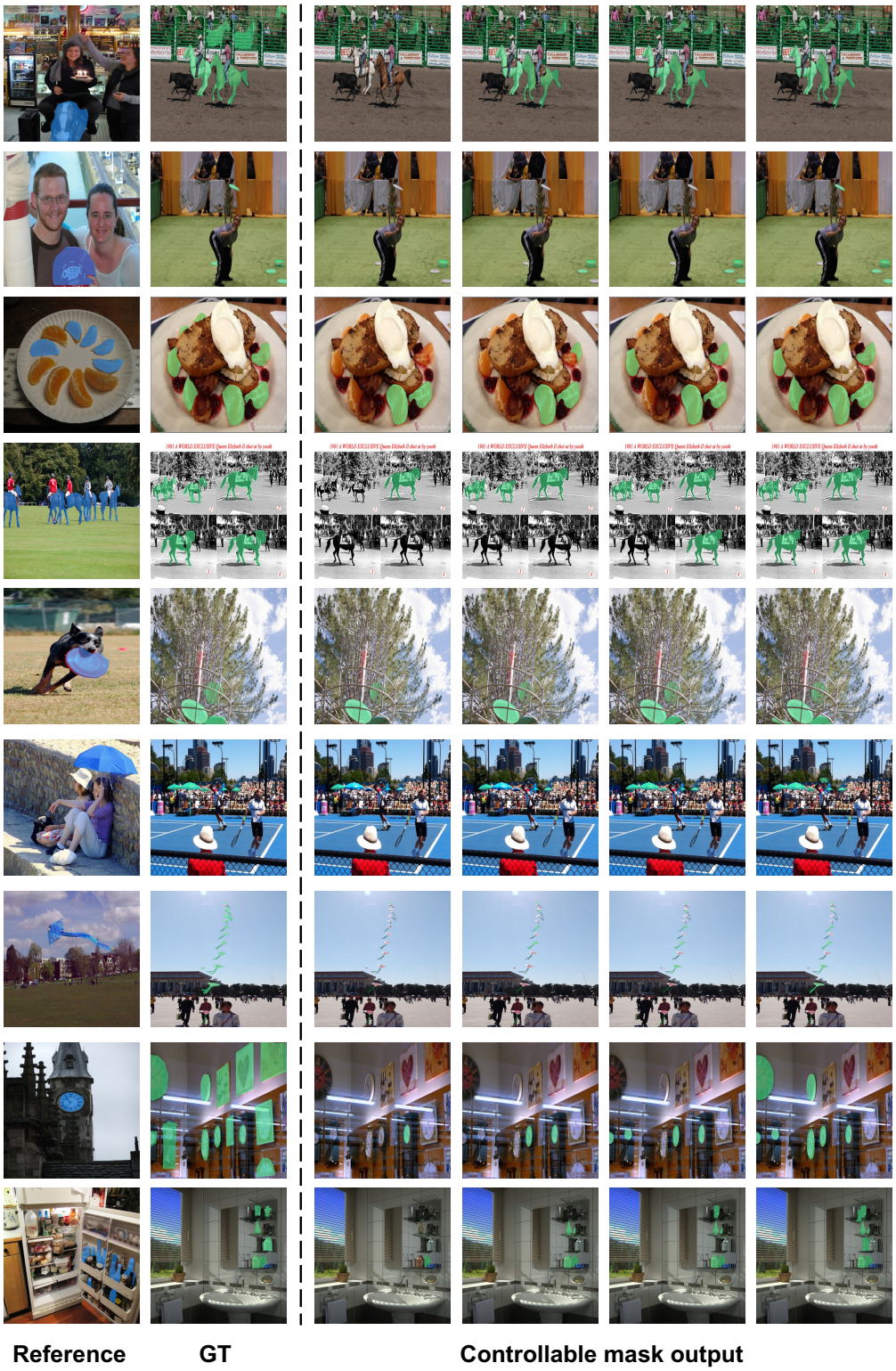


Figure 8: Visualization of controllable mask output.

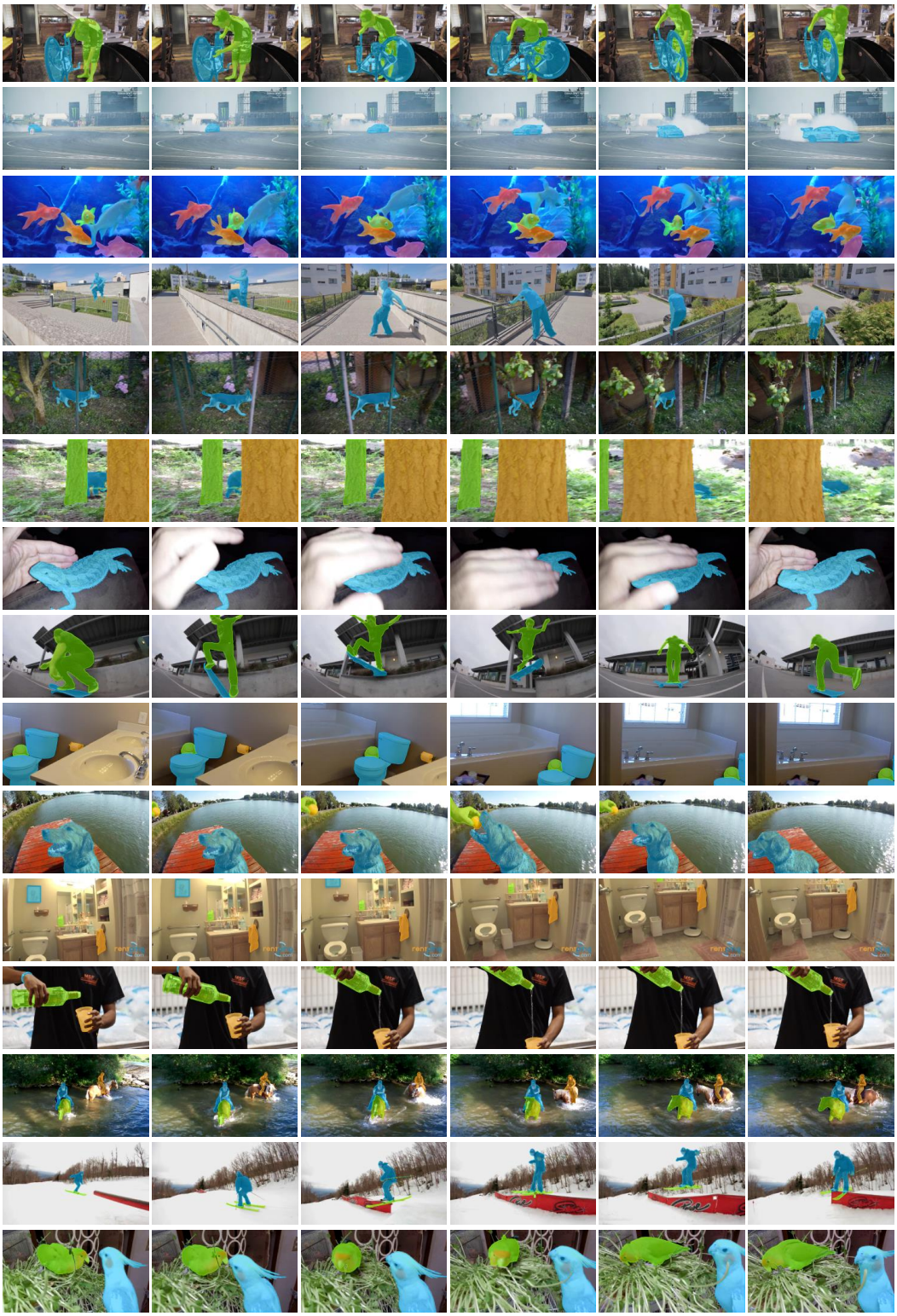


Figure 9: Visualization of video object segmentation.

References

[AV07] David Arthur and Sergei Vassilvitskii. K-means++ the advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, 2007.

[BHA+21] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[BMR+20] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NIPS*, 2020.

[BVDPPH11] Nicolas Bonneel, Michiel Van De Panne, Sylvain Paris, and Wolfgang Heidrich. Displacement interpolation using lagrangian mass transport. In *SIGGRAPH Asia*, 2011.

[CML+14] Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *CVPR*, 2014.

[CND+22] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

[CS22] Ho Kei Cheng and Alexander G Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *ECCV*, 2022.

[DBK+20] 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. In *ICLR*, 2020.

[DCLT19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.

[EVGW+10] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010.

[GDG19] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019.

[HCN+22] Sunghwan Hong, Seokju Cho, Jisu Nam, Stephen Lin, and Seungryong Kim. Cost aggregation with 4d convolutional swin transformer for few-shot segmentation. In *ECCV*, 2022.

[HCX+22] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

[ISB22] Ehtesham Iqbal, Sirojbek Safarov, and Seongdeok Bang. Msanet: Multi-similarity and attention guidance for boosting few-shot segmentation. *arXiv preprint arXiv:2206.09667*, 2022.

[JDB+19] Joakim Johnander, Martin Danelljan, Emil Brissman, Fahad Shahbaz Khan, and Michael Felsberg. A generative appearance model for end-to-end video object segmentation. In *CVPR*, 2019.

[JYX+21] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021.

[KMR+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. *arXiv preprint arXiv:2304.02643*, 2023.

[LLJC20] Yongqing Liang, Xin Li, Navid Jafari, and Jim Chen. Video object segmentation with adaptive feature bank and uncertain-region refinement. In *NIPS*, 2020.

[LLXH22] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.

[LMB+14] 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*, 2014.

[LOG+19] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[LQJ19] Huaijia Lin, Xiaojuan Qi, and Jiaya Jia. Agss-vos: Attention guided single-shot video object segmentation. In *ICCV*, 2019.

[LWC+20] Xiang Li, Tianhan Wei, Yau Pun Chen, Yu-Wing Tai, and Chi-Keung Tang. Fss-1000: A 1000-class dataset for few-shot segmentation. In *CVPR*, 2020.

[LYL+22] Zhihui Lin, Tianyu Yang, Maomao Li, Ziyu Wang, Chun Yuan, Wenhao Jiang, and Wei Liu. Swem: Towards real-time video object segmentation with sequential weighted expectation-

maximization. In *CVPR*, 2022.

[LZR+23] 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.

[MAV20] Keval Morabia, Jatin Arora, and Tara Vijaykumar. Attention-based joint detection of object and semantic part. *arXiv preprint arXiv:2007.02419*, 2020.

[MKC21] Juhong Min, Dahyun Kang, and Minsu Cho. Hypercorrelation squeeze for few-shot segmentation. In *CVPR*, 2021.

[NT19] Khoi Nguyen and Sinisa Todorovic. Feature weighting and boosting for few-shot segmentation. In *ICCV*, 2019.

[ODM+23] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[Ope23] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[OWJ+22] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *NIPS*, 2022.

[PPTM+16] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016.

[PTPC+17] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

[RBL+22] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[RKH+21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[RKP+23] Vignesh Ramanathan, Anmol Kalia, Vladan Petrovic, Yi Wen, Baixue Zheng, Baishan Guo, Rui Wang, Aaron Marquez, Rama Kovvuri, Abhishek Kadian, et al. Paco: Parts and attributes of common objects. *arXiv preprint arXiv:2301.01795*, 2023.

[SBL+17] Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots. One-shot learning for semantic segmentation. In *BMVC*, 2017.

[SST+23] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in huggingface. *arXiv preprint arXiv:2303.17580*, 2023.

[TLI+23] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017.

[WWC+23] Xinlong Wang, Wen Wang, Yue Cao, Chunhua Shen, and Tiejun Huang. Images speak in images: A generalist painter for in-context visual learning. In *CVPR*, 2023.

[WYQ+23] Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*, 2023.

[WZC+23] Xinlong Wang, Xiaosong Zhang, Yue Cao, Wen Wang, Chunhua Shen, and Tiejun Huang. Seggpt: Segmenting everything in context. *arXiv preprint arXiv:2304.03284*, 2023.

[YWY21] Zongxin Yang, Yunchao Wei, and Yi Yang. Associating objects with transformers for video object segmentation. In *NIPS*, 2021.

[ZCH+23] Deyao Zhu, Jun Chen, Kilichbek Haydarov, Xiaoqian Shen, Wenxuan Zhang, and Mohamed Elhoseiny. Chatgpt asks, blip-2 answers: Automatic questioning towards enriched visual descriptions. *arXiv preprint arXiv:2303.06594*, 2023.

[ZCS+23] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

[ZJG+23] Renrui Zhang, Zhengkai Jiang, Ziyu Guo, Shilin Yan, Junting Pan, Hao Dong, Peng Gao, and Hongsheng Li. Personalize segment anything model with one shot. *arXiv preprint*

arXiv:2305.03048, 2023.

[ZLD+22] Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*, 2022.

[ZRG+22] 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.

[ZSYC22] Jian-Wei Zhang, Yifan Sun, Yi Yang, and Wei Chen. Feature-proxy transformer for few-shot segmentation. In *NIPS*, 2022.

[ZYZ+23] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *arXiv preprint arXiv:2304.06718*, 2023.