

# STABLE SEGMENT ANYTHING MODEL

**Anonymous authors**

Paper under double-blind review

## 1 MORE IMPLEMENTATION DETAILS

**Training Details.** During training, we only train DSP and DRP on HQSeg-44K dataset while fixing the model parameters of the pre-trained SAM model. We train Stable-SAM on 8 NVIDIA Tesla V100 GPUs with a total batch size of 32, using Adam optimizer with zero weight decay and 0.001 learning rate. The training images are augmented using large-scale jittering (Ghiasi et al., 2021). The input prompts are randomly sampled from mixed prompt types, including ground truth bounding boxes, randomly sampled points (1, 3, 5, 10 positive points randomly selected from the ground truth mask), noisy boxes (generated by adding noise (noise scale 0.4) to the ground truth bounding boxes), where we ensure the generated noisy boxes have at least 0.5 overlap IoU with the ground truth boxes), and coarse masks (generated by adding Gaussian noise in the boundary regions of the ground truth masks). The model is optimized using cross entropy loss and dice loss (Milletari et al., 2016).

**Inference Details.** We follow the same inference pipeline of the original SAM. The mask decoder first predicts a small mask in  $256 \times 256$  spatial resolution for each prompt, which is then up-sampled to the original resolution  $1024 \times 1024$  as the output mask.

**Evaluation Metrics.** We select suitable evaluation metrics depending on testing datasets, *i.e.*, 1) mask mIoU, boundary mBIoU and ST for DIS, ThinObject-5K, COIFT, and HR-SOD, which usually contain only one object in each image; 2) mask mAP and mAP<sub>50</sub> for COCO and SGINW, which usually contain multiple objects in each image.

**Occlusion Image Synthesis.** For each training image, we randomly select another image with random scale jittering and random horizontal flipping augmentation. Then we select a random object from the selected image as the “occluder” and paste it onto the training image to occlude the “occludee” object. Specifically, we ensure the center of the occluder is strategically placed within the bounding box of the occluded object in the training image. Finally, we remove the fully occluded objects and update the ground-truth mask annotations of the partially occluded objects. The occlusion image synthesis is enabled with a probability of 0.5 in the training stage.

**More implementation details of LoRA and adapters in SAM.** We introduce Adapter/LoRA modules to the feed-forward network (FFN) of each ViT layer in SAM’s encoder for tuning. During training, we fine-tune only the adapter/LoRA modules and SAM’s prediction layer, with all other parameters frozen. In line with AdaptFormer (Chen et al., 2022), the adapters consist of small bottleneck layers inserted in parallel into the FFN, containing two MLPs and a GELU activation function between them. The bottleneck’s middle dimension is set to 64 to balance model performance and computational efficiency. In line with LoRA (Hu et al., 2022), the module uses an encoder-decoder structure to impose a low-rank constraint on FFN weight updates, injecting small trainable rank decomposition matrices into each layer. In our experiments, the rank of LoRA is set to 4 for efficiency and performance optimization. All other experimental settings remain the same as those of the baseline and full model.

**More implementation details for SAM-based interactive segmentation.** In the SAM-based interactive segmentation experiments, setting the hyperparameter ‘multimask\_output = False’ will yield better performance, especially on SBD dataset. For a fair comparison, we use the default ‘multimask\_output = True’ setting to align with the SAM-based interaction segmentation implementations potentially adopted by other works.

Table 1: Comparison on Multi-domain Evaluation of Semantic Segmentation (MESS) benchmark, consisting of 22 downstream datasets, 448 classes, and 25,079 images.

Model	Prompt	General	Earth Monitoring	Medical Sciences	Engineering	Agriculture & Biology	Mean
SAM	Oracle Point	46.51	42.31	55.92	51.57	57.43	49.67
HQ-SAM		44.05	40.82	<b>61.0</b>	<b>54.18</b>	53.92	49.58
Stable-SAM (Ours)		<b>49.10</b>	<b>43.23</b>	54.45	53.52	<b>60.92</b>	<b>51.15</b>
SAM	Oracle Box	77.85	<b>73.03</b>	64.89	73.03	<b>86.75</b>	<b>74.74</b>
HQ-SAM		76.63	68.03	<b>65.1</b>	<b>74.78</b>	85.34	73.43
Stable-SAM (Ours)		<b>77.91</b>	72.84	62.93	73.12	85.63	74.21
SAM	Random Point	40.21	36.74	53.20	47.15	49.75	44.34
HQ-SAM		38.81	36.62	<b>58.27</b>	<b>51.43</b>	44.51	44.92
Stable-SAM (Ours)		<b>42.43</b>	<b>37.30</b>	50.89	48.96	<b>53.42</b>	<b>45.49</b>
SAM	Noisy Box	55.97	55.12	62.45	62.34	64.26	59.24
HQ-SAM		52.02	47.15	<b>62.57</b>	62.02	60.56	55.81
Stable-SAM (Ours)		<b>59.82</b>	<b>58.86</b>	62.19	<b>64.56</b>	<b>68.84</b>	<b>62.13</b>

Table 2: Dataset and comparison details on MESS benchmark. Models are prompted with noisy boxes. “HQ” denotes HQ-SAM.

Dataset	Domain	Sensor type	Mask size	# Classes	# Images	Task	SAM	HQ	Ours
BDD100K (Yu et al., 2020)	General	Visible spectrum	Medium	19 (Medium)	1,000	Driving	48.9	43.66	<b>51.84</b>
Dark Zurich (Sakaridis et al., 2019)		Visible spectrum	Medium	20 (Medium)	50	Driving	54.34	50.85	<b>56.81</b>
MHP v1 (Li et al., 2017)		Visible spectrum	Small	19 (Medium)	980	Body parts	66.64	62.63	<b>69.11</b>
FoodSeg103 (Ghosh et al., 2021)		Visible spectrum	Medium	104 (Many)	2,135	Ingredients	57.98	56.22	<b>64.12</b>
ATLANTIS (Erfani et al., 2022)		Visible spectrum	Small	56 (Many)	1,295	Maritime	56.23	48.7	<b>60.04</b>
DRAM (Cohen et al., 2022)		Visible spectrum	Medium	12 (Medium)	718	Paintings	51.73	50.06	<b>57.03</b>
iSAID (Waqas Zamir et al., 2019)	Earth Monitoring	Visible spectrum	Small	16 (Medium)	4,055	Objects	65.2	62.6	<b>67.46</b>
ISPRS Potsdam (Khoshelham et al., 2017)		Multispectral	Small	6 (Few)	504	Land use	47.32	38.43	<b>50.92</b>
WorldFloods (Mateo-Garcia et al., 2021)		Multispectral	Medium	3 (Binary)	160	Floods	57.61	49.46	<b>60.81</b>
FloodNet (Rahmemonfar et al., 2021)		Visible spectrum	Medium	10 (Few)	5,571	Floods	51.85	39.82	<b>58.31</b>
UAVid (Lyu et al., 2020)		Visible spectrum	Small	8 (Few)	840	Objects	53.62	45.43	<b>56.82</b>
Kvasir-Inst. (Jha et al., 2021)	Medical Sciences	Visible spectrum	Medium	2 (Binary)	118	Endoscopy	83.62	81.32	<b>85.87</b>
CHASE DB1 (Fraz et al., 2012)		Microscopic	Small	2 (Binary)	20	Retina scan	34.59	<b>37.5</b>	33.88
CryoNuSeg (Mahbod et al., 2021)		Microscopic	Small	2 (Binary)	30	WSI	72.87	72.78	<b>73.09</b>
PAXRay-4 (Seibold et al., 2022)		Electromagnetic	Large	4x2 (Binary)	180	X-Ray	<b>58.72</b>	58.66	55.94
Corrosion CS (Bianchi & Hebdon, 2021)	Engineering	Visible spectrum	Medium	4 (Few)	44	Corrosion	53.05	51.86	<b>55.47</b>
DeepCrack (Liu et al., 2019)		Visible spectrum	Small	2 (Binary)	237	Cracks	56.36	<b>60.84</b>	58.2
ZeroWaste-f (Bashkirova et al., 2022)		Visible spectrum	Medium	5 (Few)	929	Conveyor	70.23	69.52	<b>74.01</b>
PST900 (Shivakumar et al., 2020)		Electromagnetic	Small	5 (Few)	288	Thermal	69.71	65.88	<b>70.55</b>
SUIM (Islam et al., 2020)	Agriculture & Biology	Visible spectrum	Medium	8 (Few)	110	Underwater	56.98	47.51	<b>62.35</b>
CUB-200 (Welinder et al., 2010)		Visible spectrum	Medium	201 (Many)	5,794	Bird species	56.3	54.54	<b>64.57</b>
CWFID (Haug & Ostermann, 2015)		Visible spectrum	Small	3 (Few)	21	Crops	79.49	<b>79.63</b>	79.61
Mean IoU	-	-	-	-	-	-	59.24	55.81	<b>62.13</b>

## 2 MORE EXPERIMENTAL RESULTS

### 2.1 MULTI-DOMAIN EVALUATION OF SEMANTIC SEGMENTATION (MESS)

The recently released Multi-domain Evaluation of Semantic Segmentation (MESS) (Blumenstiel et al., 2023) is a large-scale benchmark for holistic analysis of zero-shot segmentation performance. MESS consists of 22 downstream tasks, a total of 448 classes, and 25079 images, covering a wide range of domain-specific datasets in the fields of earth monitoring, medical sciences, engineering, agriculture and biology and other general domains. We evaluate SAM (Kirillov et al., 2023), HQ-SAM (Ke et al., 2023) and our Stable-SAM on MESS benchmark using the official MESS evaluation code, and report the mean of class-wise intersection over union (mIoU).

Following MESS’s model settings, our Stable-SAM selects the first mask of the predicted multiple masks as the output. For a fair comparison, our Stable-SAM follows HQ-SAM to fuse the SAM’s original prediction map into our predicted segmentation map. We provide four prompt types for evaluation. The *oracle point* refers to a single point sampled from the ground-truth mask using the point sampling approach RITM (Sofiuk et al., 2022). The *random point* refers to a single point

Table 3: Comparison on PartImageNet. All models (except for SAM) are trained on PartImageNet train set and evaluated on PartImageNet val set with various prompts.

	GT box	Noisy box	1 Point	3 Points
SAM	68.2	38.3	35.4	45.7
PT-SAM	72.6	51.6	36.8	55.2
HQ-SAM	72.9	52.4	37.3	56.1
Stable-SAM	73.4	60.7	44.2	61.6

Table 4: User study of noisy boxes for realistic application scenarios.

Model	Noisy Box			1 Point		
	mIoU	mBIOU	ST	mIoU	mBIOU	ST
SAM (baseline)	48.8	42.1	39.5	58.3	51.7	49.3
HQ-SAM	72.4	62.8	65.5	76.1	66.4	69.2
Ours	82.3	74.1	82.3	84.2	76.3	84.0

Table 5: Comparison on MS COCO and four HQ datasets for different backbone variants.

Model	Epoch	MS COCO		Four HQ Datasets						Params (M)		FPS	Mem.
		mAP	mAP <sub>50</sub>	Noisy Box			1 Point			Total	Trainable		
SAM-Huge	-	25.6	56.8	50.1	43.2	40.4	44.5	38.3	46.5	2446	2446	3.5	10.3G
HQ-SAM-Huge	12	30.0	62.6	75.2	65.5	69.3	48.0	41.1	49.5	2452.1	6.1	3.4	10.3G
Stable-SAM-Huge	1	<b>43.9</b>	<b>75.7</b>	<b>81.8</b>	<b>73.5</b>	<b>82.3</b>	<b>77.2</b>	<b>68.9</b>	<b>74.6</b>	2446.08	0.08	3.5	10.3G
SAM-Large	-	27.3	60.2	48.8	42.1	39.5	43.3	37.4	45.1	1191	1191	5.0	7.6G
HQ-SAM-Large	12	31.9	65.5	72.4	62.8	65.5	43.2	44.6	37.4	1196.1	5.1	4.8	7.6G
Stable-SAM-Large	1	<b>44.8</b>	<b>76.4</b>	<b>82.3</b>	<b>74.1</b>	<b>82.3</b>	<b>76.9</b>	<b>68.4</b>	<b>71.1</b>	1191.08	0.08	5.0	7.6G
SAM-Base	-	19.7	49.2	41.6	35.8	33.4	35.1	29.2	36.7	358	358	10.1	5.1G
HQ-SAM-Base	12	24.7	56.1	68.7	59.1	63.2	40.6	35.7	42.7	362.1	4.1	9.8	5.1G
Stable-SAM-Base	1	<b>31.2</b>	<b>63.3</b>	<b>74.7</b>	<b>64.8</b>	<b>75.9</b>	<b>68.9</b>	<b>59.5</b>	<b>67.1</b>	358.08	0.08	10.1	5.1G

randomly sampled from the ground-truth mask of the target object. The *oracle box* refers to a single box tightly enclosing the ground-truth mask of the target object. The *noisy box* refers to a single box generated by adding noise (noise scale 0.4) to the *oracle box*.

Table 1 tabulates the zero-shot semantic segmentation performance comparison on MESS. Our Stable-SAM performs best when prompted with *oracle point*, *random point* and *noisy box*, and achieves comparable performance when provided with *oracle box*. Our competitive performance on the large-scale MESS benchmark further consolidates the powerful zero-shot generalization ability inherent in our Stable-SAM. Table 2 shows the dataset and comparison details on 22 tasks of MESS benchmark. Our Stable-SAM performs best on 19 out of 22 datasets.

## 2.2 PART-LEVEL SEGMENTATION

Our method can handle segmentation tasks when trained with segmentation targets which are typically clear and unambiguous, object or part level. We conduct comparison experiments on PartImageNet (He et al., 2022) dataset to validate our part-level segmentation ability. All models, using the ViT-L backbone, are trained on PartImageNet train set and evaluated on the val set. Table 3 shows that our method outperforms the original SAM and finetuned SAMs (PT-SAM and HQ-SAM).

## 2.3 COMPARISON BASED ON USER-ANNOTATED BOX PROMPTS

Table 6: Ablation studies on the number of prompt points. All models (except for SAM) are trained on HQSeg-44K dataset, and evaluated on four HQ datasets with various prompt points.

# of Points	1	3	5	7	10	15	20
SAM	43.3 $\pm$ 1.8	78.7 $\pm$ 0.8	83.3 $\pm$ 0.7	84.1 $\pm$ 0.7	84.8 $\pm$ 0.6	85.3 $\pm$ 0.6	85.7 $\pm$ 0.5
PT-SAM	43.0 $\pm$ 1.9	80.1 $\pm$ 0.7	84.3 $\pm$ 0.6	85.2 $\pm$ 0.6	85.6 $\pm$ 0.4	86.1 $\pm$ 0.4	86.3 $\pm$ 0.3
Stable-SAM	76.9 $\pm$ 0.8	84.0 $\pm$ 0.6	85.8 $\pm$ 0.5	86.5 $\pm$ 0.4	87.0 $\pm$ 0.4	87.5 $\pm$ 0.3	87.8 $\pm$ 0.3

Table 7: The DRP routing weight  $\alpha_1$  is affected by the number of the prompt points. Fewer prompt points needs more deformable attention indicated by larger  $\alpha_1$ . The models is trained on HQSeg-44K dataset and evaluated on four HQ datasets.

# of Points	1	2	3	4	5	10	20
$\alpha_1$	0.614	0.552	0.469	0.427	0.398	0.371	0.359

We conduct a user study to provide a more realistic evaluation of our method. Five participants are asked to provide box annotations for the highlighted target object in each image. Participants are instructed to complete each box annotation within 5 seconds to ensure consistency throughout the process. Annotations are collected using the Label Studio platform. The user-annotated boxes are subsequently used as prompts to evaluate the performance of each segmentation method. Table 4 shows that user-annotated boxes provide better segmentation performance across all methods compared to generated noisy boxes. We also assess the quality of the user-annotated boxes by comparing them to the ground truth boxes derived from the mask annotations. The average IoU between user-annotated boxes and ground truth boxes is approximately 0.753, indicating that user-annotated boxes are more accurate than generated noisy boxes. Under the user-provided box prompt, our method continues to outperform other methods by a large margin.

## 2.4 BACKBONE VARIANTS

Table 5 tabulates the performance comparison on different backbone variants. Our Stable-SAM consistently performs better than other methods on all backbone variants.

## 2.5 ABLATION STUDIES ON THE NUMBER OF PROMPT POINTS

Table 6 shows the performance curve of SAM, PT-SAM and Stable-SAM when handling various number of prompt points. We also show the performance standard deviation to indicate the segmentation stability. The results show that our Stable-SAM has larger performance gains when handling lower-quality prompts, *i.e.*, fewer prompt points.

## 2.6 ANALYSIS ON THE DRP ROUTING WEIGHT $\alpha_1$

Table 7 shows the DRP routing weight  $\alpha_1$  is increased from 0.469 to 0.614 when we change the point prompt from three points to one point. It indicates that lower-quality prompts rely more on DSP features to shift attention to the desirable regions.

We further conduct additional experiments to manipulate the output strength of the DRP and examine its impact on segmentation quality for different prompt qualities. Specifically, we evaluate segmentation performance at different  $\alpha_1$  values for both ambiguous (one point) and precise (ten points) prompts. For a fair comparison, we use the same set of point prompts for each  $\alpha_1$  value.

Table 8 show that larger  $\alpha_1$  values (indicating stronger DSP activation) lead to better segmentation performance for the ambiguous one point prompt. This suggests that increasing the output strength of the DRP is particularly beneficial for less informative prompts. In contrast, segmentation performance is less sensitive to variations in  $\alpha_1$  for precise prompts, suggesting that when prompts provide clearer guidance, the system achieves satisfactory results even with lower activation levels.

Table 8: The DRP routing weight  $\alpha_1$  is affected by the number of the prompt points. Fewer prompt points needs more deformable attention indicated by larger  $\alpha_1$ . The models is trained on HQSeg-44K dataset and evaluated on four HQ datasets.

$\alpha_1$	0.0	0.2	0.4	0.5	0.6	0.8	1.0
1 point	43.3	73.2	75.9	76.5	76.9	76.5	76.0
10 points	84.8	86.1	87.0	86.9	86.7	86.7	86.5

Table 9: Comparison on MS COCO and four HQ datasets for different Stable-SAM variants. The “finetuning decoder” denotes finetuning the mask decoder when training Stable-SAM.

Model	MS COCO		Four HQ Datasets						
	N-Box (0.5-0.6) mAP	mAP <sub>50</sub>	Noisy Box			1 Point			
	mIoU	mBIoU	mSF	mIoU	mBIoU	mSF	mIoU	mBIoU	mSF
SAM	27.3	60.2	48.8	42.1	39.5	43.3	37.4	45.1	
Stable-SAM (finetuning decoder)	25.7	56.5	78.5	69.2	79.9	76.0	67.1	<b>78.2</b>	
Stable-SAM (spatial attention)	29.8	64.9	69.3	59.8	57.8	51.6	44.5	49.1	
Stable-SAM	<b>44.8</b>	<b>76.4</b>	<b>82.3</b>	<b>74.1</b>	<b>82.3</b>	<b>76.9</b>	<b>68.4</b>	71.1	

### 3 RELATION TO OTHER METHODS

**Deformable Attention.** Our method is unique in its idea and design on solely adjusting the feature sampling locations and amplitudes by training the offset network, without involving the original model parameters. In contrast, conventional deformable attention methods (Dai et al., 2017; Xia et al., 2022) train both the offset network and original network parameters, which is undesirable when adapting powerful foundation models in deployment, especially in finetuning large foundation models. Figure 1 shows the difference between our deformable sampling plugin and conventional deformable attention.

We apply the conventional deformable attention in our Stable-SAM by finetuning the mask decoder during training. Table 9 shows that the conventional deformable attention (Stable-SAM (finetuning decoder)) exhibits the worst generalization ability on MS COCO, even worse than the original SAM model. This further validates the necessity and better performance of our deformable sampling plugin paradigm, *i.e.*, adapting the foundation model by only adjusting the feature sampling locations and amplitudes, while fixing the original model features and parameters.

**Spatial Attention.** The spatial attention (Woo et al., 2018; Hou et al., 2021) can adjust the image spatial feature weights, and thus can be regarded as a soft feature sampling method. We directly replace DSP with spatial attention in our Stable-SAM to investigate if spatial attention offers comparable effectiveness. Table 9 shows that spatial attention performs much worse than our DSP, although it consistently improves the segmentation performance and stability on all datasets. This indicates that simply adjusting the feature weights is insufficient to adapt SAM for handling suboptimal prompts.

## 4 MORE DISCUSSIONS

### 4.1 DISCUSSION ON MODEL SCALABILITY TO NOISY TRAINING DATA

Increasing the amount of training data could improve the model’s robustness and generalization. More noisy data could help the model learn to handle a wider variety of input prompts, especially in real-world scenarios where user-provided prompts are often imprecise or ambiguous. As the model is exposed to more diverse and challenging inputs, it may become better at distinguishing relevant features and handling uncertainty in segmentation. However, introducing excessive noisy data could cause the model to overfit to the noise, leading to instability in some cases. Our method,

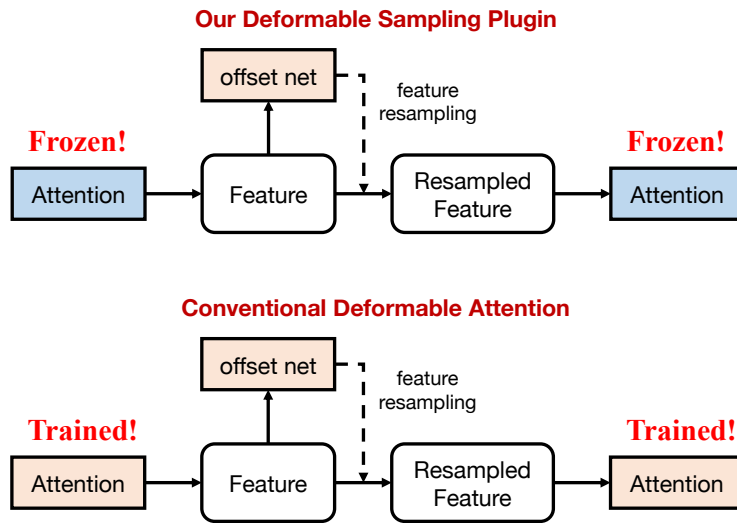


Figure 1: Method difference between our deformable sampling plugin and conventional deformable attention.

which dynamically calibrates attention based on prompt quality, mitigates some risks of noisy data by guiding the model to focus on relevant image regions without being overwhelmed by noise.

We also understand the concerns about the scalability of our method when scaling up the training data. While our method is designed to efficiently handle noisy data with limited resources, it can still benefit from larger training sets. As the training dataset grows, we can fine-tune additional parameters of Stable-SAM to further improve segmentation stability. Our method’s lightweight design, with only 0.08M learnable parameters, allows for fine-tuning additional parameters without significantly increasing model complexity or computational cost. This additional fine-tuning of more parameters applied to larger datasets, can further boost performance and stability, as the model adapts more precisely to the increased diversity of data and ambiguous prompts. This highlights the flexibility of our approach, which performs well with limited data but can also benefit from additional fine-tuning of more parameters when the training dataset increases in size and noise.

#### 4.2 DISCUSSION ON POTENTIAL SEGMENTATION BIAS

We emphasize that our method is designed without inherent bias towards large or small objects. If the prompt explicitly specifies the background, our method adapts accordingly, without constraining the model to prioritize the foreground. Although our method avoids introducing bias, we acknowledge that the model may still be influenced by dataset bias. For instance, if the training dataset predominantly contains foreground objects, the model may skew predictions towards the foreground, potentially neglecting background regions. Thus, we also highlight the flexibility of our method. If users wish to personalize segmentation targets, such as focusing on specific background regions, Stable-SAM can be easily fine-tuned to meet this requirement. This adaptability is a key strength, enabling our approach to effectively mitigate dataset bias and address a wide range of user needs and scenarios beyond typical foreground segmentation.

#### 4.3 DISCUSSION ON THE MOTIVATION BEHIND DSP (DYNAMIC SAMPLING PLUGIN)

**Problem motivation.** The primary motivation for adjusting the attention sampling positions through learnable offsets is to overcome the inherent limitations of traditional fixed-grid attention, especially in the presence of noisy prompts. In the original SAM architecture, the mask decoder uses a regular grid sampling strategy, assuming that the initial prompt provides an accurate indication of the target region. However, in real-world scenarios where prompts are imprecise, fixed-grid sampling can cause

324 the attention mechanism to focus on unintended areas, often capturing background or partially correct  
325 features.

326 **Method motivation.** Our approach introduces learnable offsets to adapt the sampling positions  
327 dynamically, enabling attention to more precisely target the correct regions, even with ambiguous  
328 or noisy prompts. The learnable offsets enable the attention mechanism to "deform" the regions  
329 it focuses on in a data-driven manner, akin to how a person might adjust their focus when given  
330 unclear instructions. Essentially, the model learns from diverse training examples how to "interpret"  
331 and refine ambiguous or noisy prompts, using the learned offsets to direct attention to the most  
332 contextually relevant regions of the image.

#### 333 4.4 DISCUSSION ON THE CHOICE OF PARAMETER-EFFICIENT FINE-TUNING (PEFT) AND FULL 334 RETRAINING

335 The decision to use PEFT rather than retraining all components of SAM primarily depends on the  
336 scale and type of the training dataset. SAM was pretrained on the SA-1B dataset, containing 1 billion  
337 masks and 11 million images, which endows it with a generalized ability to segment a wide range  
338 of objects across various domains. In contrast, our downstream dataset, HQSeg-44K, contains only  
339 44,000 images. Due to the significant difference in dataset scale, PEFT is a strategic choice to prevent  
340 overfitting while effectively adapting the pre-trained SAM to our target segmentation tasks with a  
341 small training budget.

342 Numerous studies have demonstrated that PEFT methods are particularly effective when adapting  
343 large pretrained models to downstream tasks with limited data. This approach retains the generalizable  
344 features learned during pre-training, fine-tuning only a small subset of additional parameters. This  
345 paradigm is well-established in parameter-efficient fine-tuning research (as summarized in the PEFT  
346 surveys (Han et al., 2024; Xin et al., 2024)). By using PEFT, we aim to retain SAM's foundational  
347 representation power while mitigating overfitting, which is more likely to occur if we were to retrain  
348 the entire model on a relatively small dataset.

349 In our case, finetuning only the DSP and DRP modules ensures that SAM retains its generality and  
350 adaptability while becoming more robust to noisy and ambiguous prompts, maintaining computational  
351 efficiency. In contrast, retraining SAM's entire model compromises its integrity and significantly  
352 impairs performance, as shown by the large performance drop of FT-SAM (finetuning SAM's whole  
353 model) in Table 2 of the main paper.

## 354 5 STABILITY VISUALIZATION

355 Figure 2-12 show extensive visualization comparisons between SAM and Stable-SAM, under box,  
356 1-point and 3-points prompts of diverse qualities. We also visualize the image activation map for  
357 the token-to-image cross-attention in SAM's second mask decoder layer to better understand its  
358 response to low-quality prompts. The important features are highlighted by the orange circles, with  
359 larger radius indicating higher attention score. SAM yields unsatisfactory segmentation results  
360 when provided with low-quality prompts, and even a minor prompt modification leads to unstable  
361 segmentation output. In contrast, our Stable-SAM produces consistent and accurate mask predictions  
362 even under prompts of diverse qualities, by shifting more feature sampling attention to the target  
363 object.

378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431

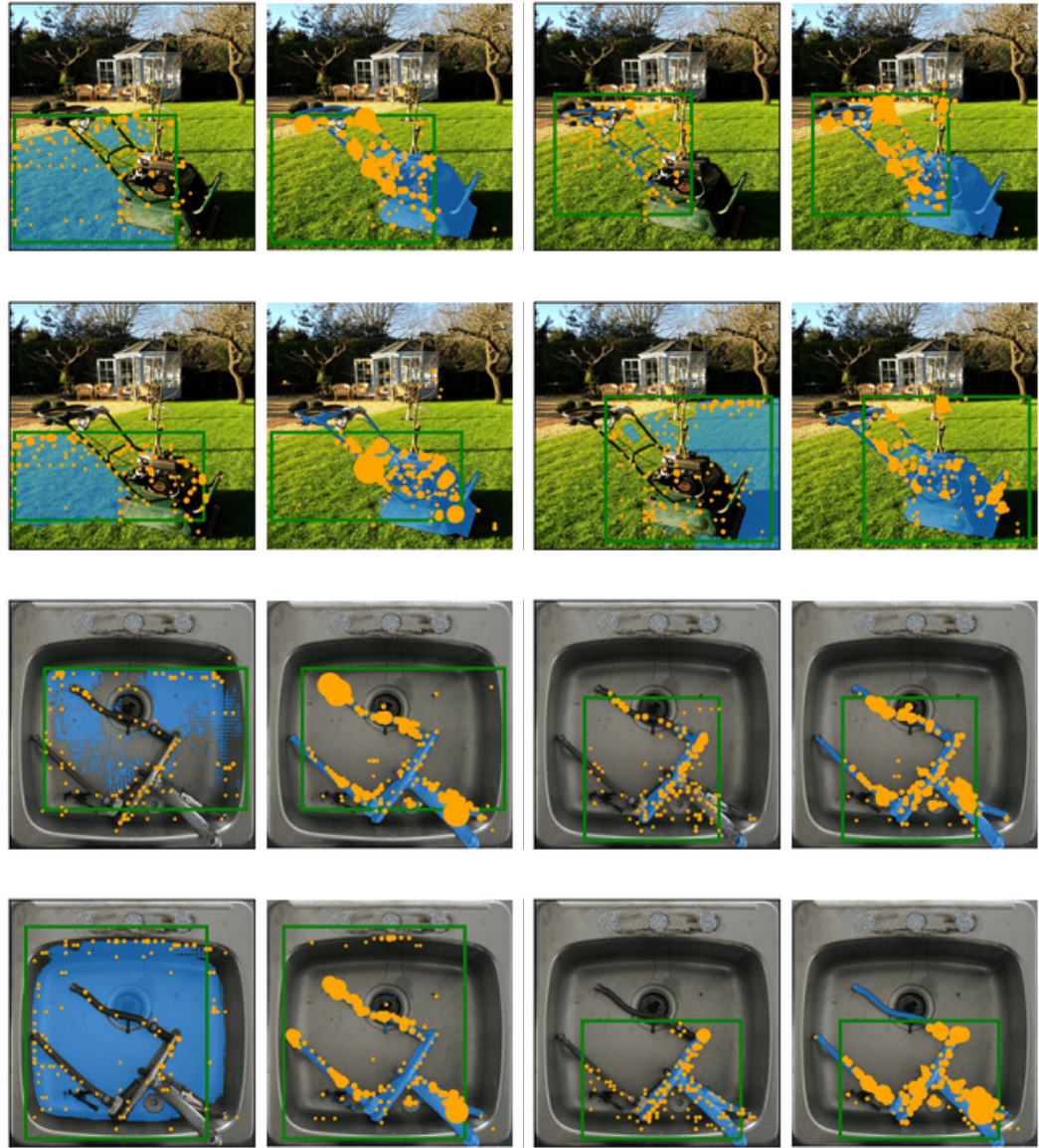


Figure 2: Visual results for box prompts. Within each image pair given the same prompt (green box), the subfigures represent the results of SAM and Stable-SAM, respectively.



432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485

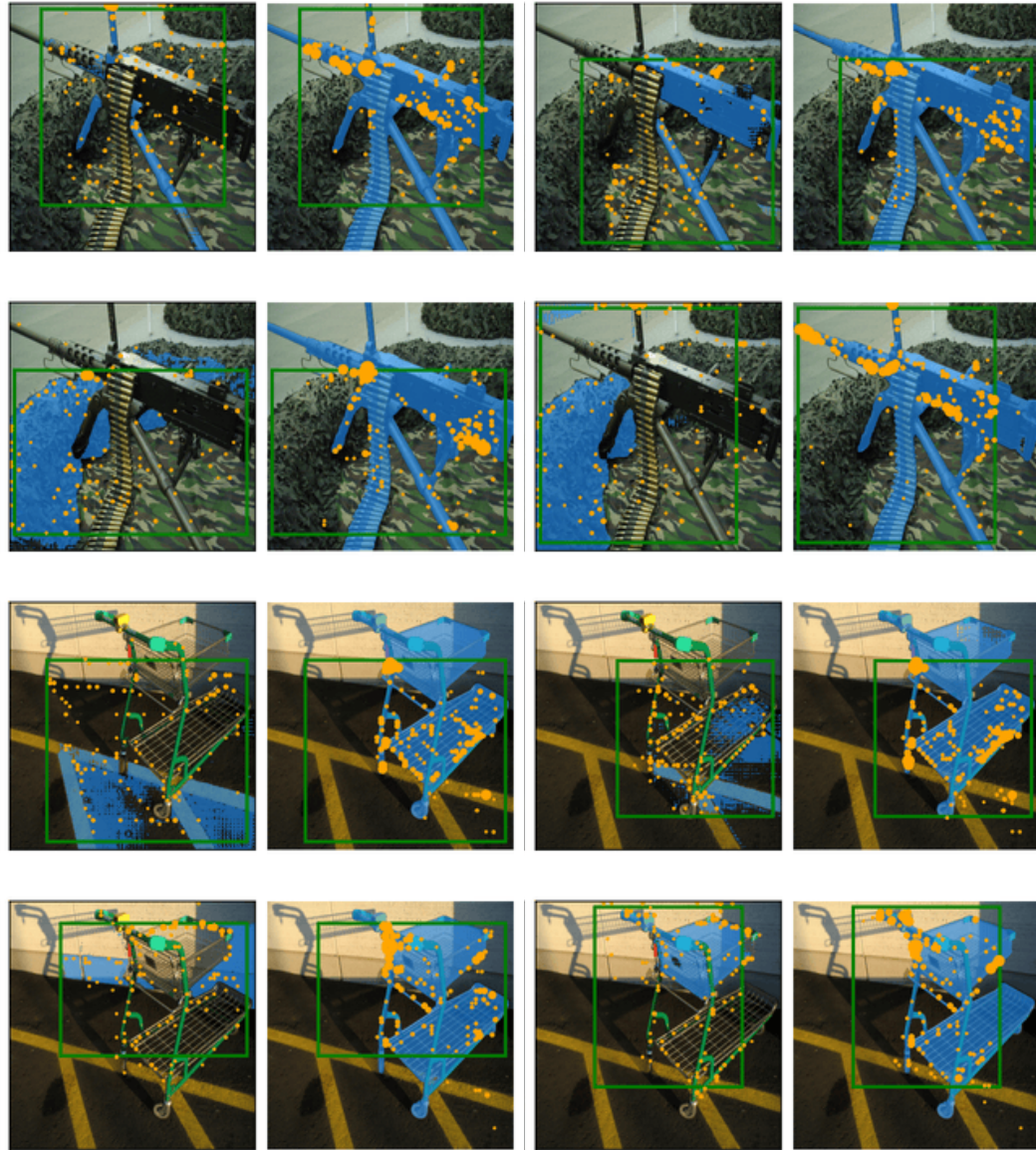


Figure 3: Visual results for box prompts.

486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539

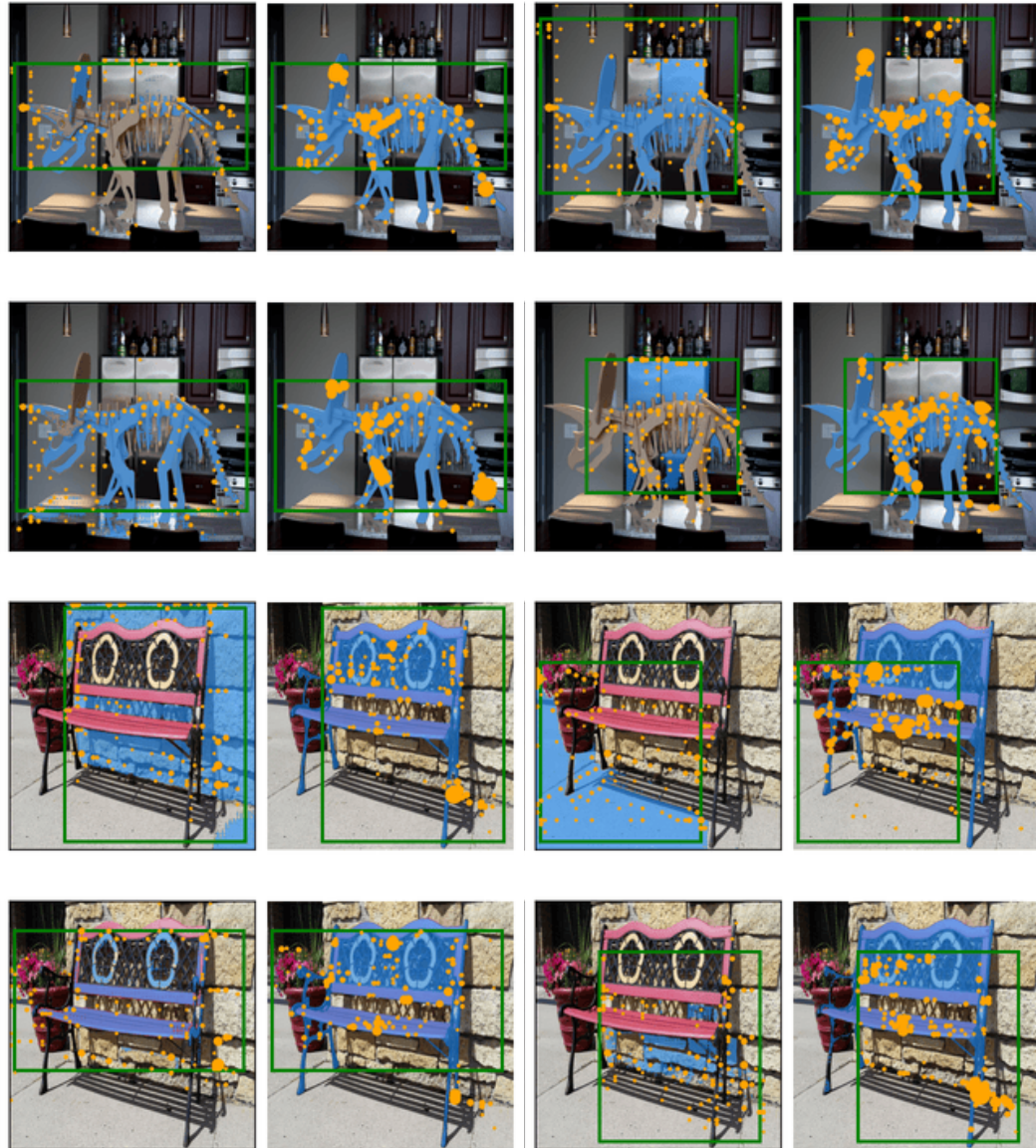


Figure 4: Visual results for box prompts.

540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593

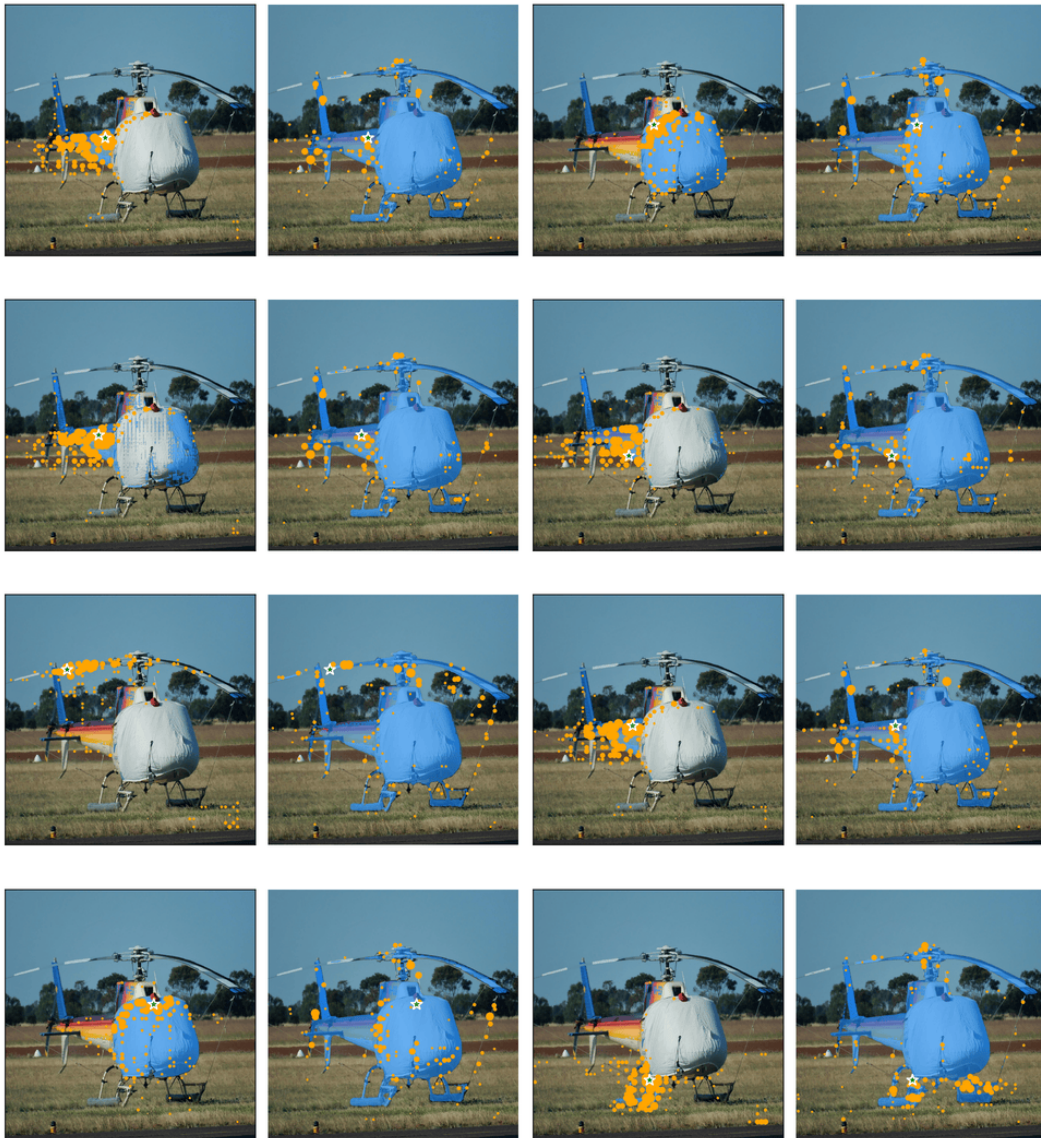


Figure 5: Visual results for 1-point prompt.

594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647



Figure 6: Visual results for 1-point prompt.

648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701

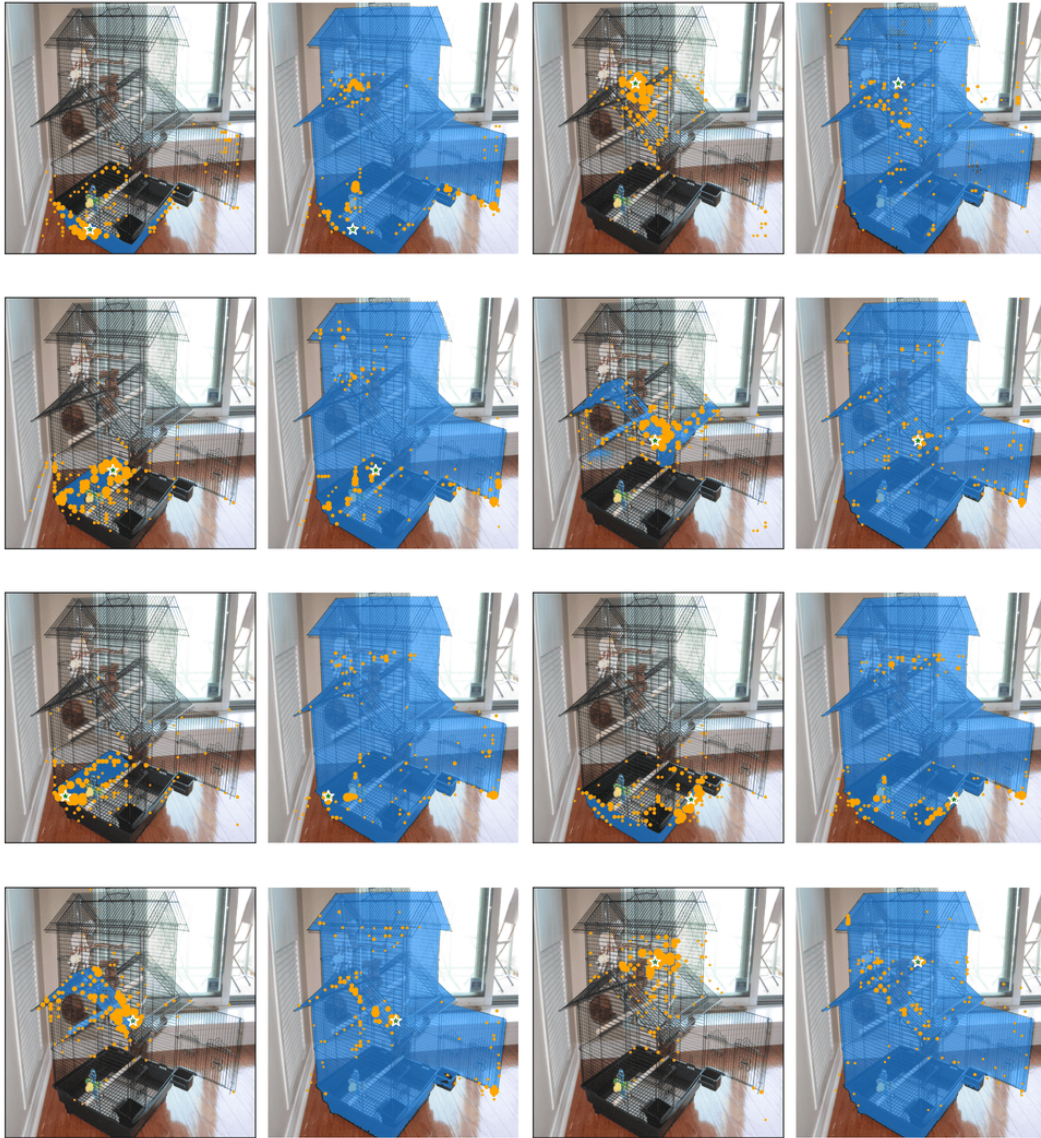


Figure 7: Visual results for 1-point prompt.

702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755

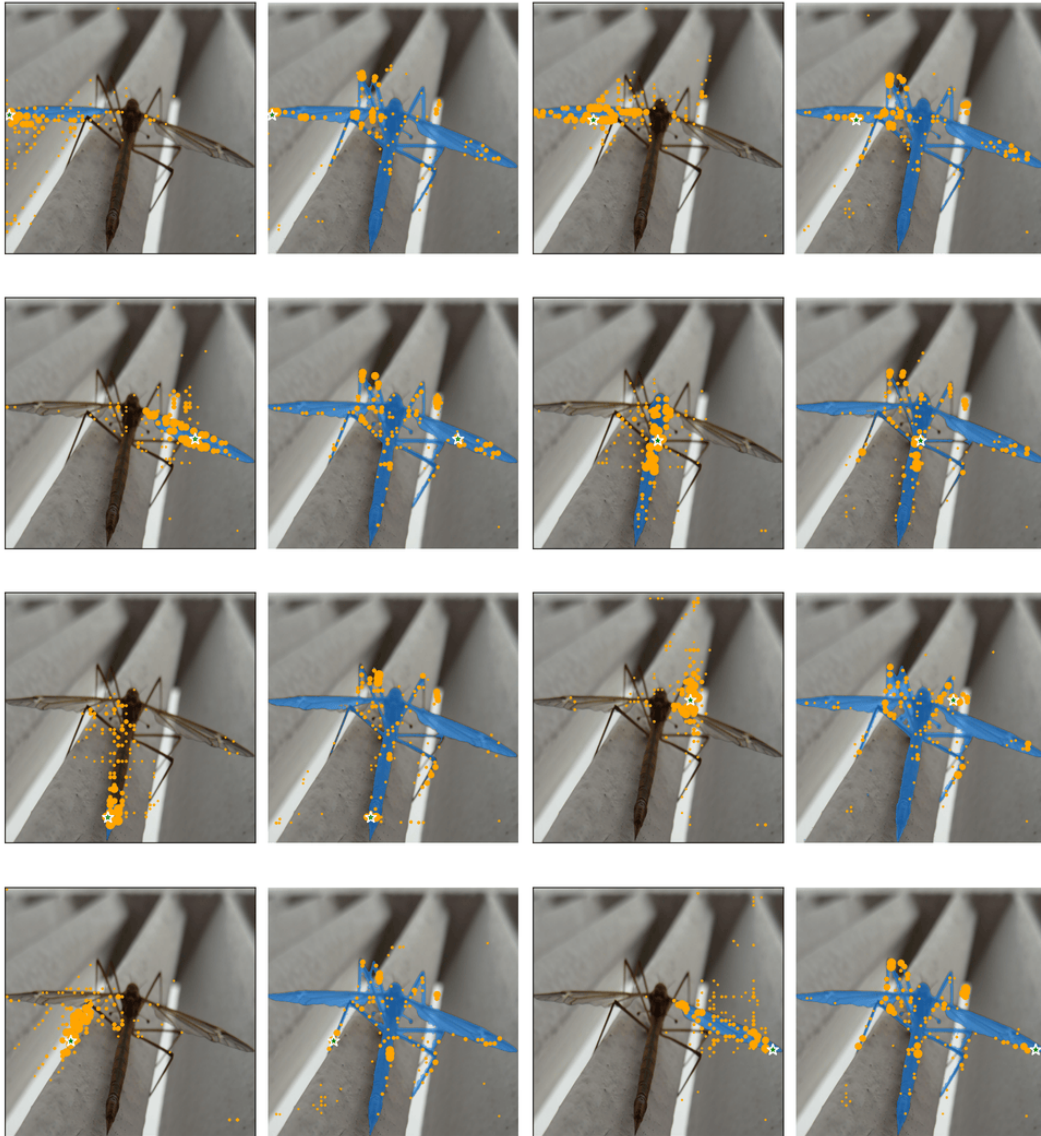


Figure 8: Visual results for 1-point prompt.

756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809

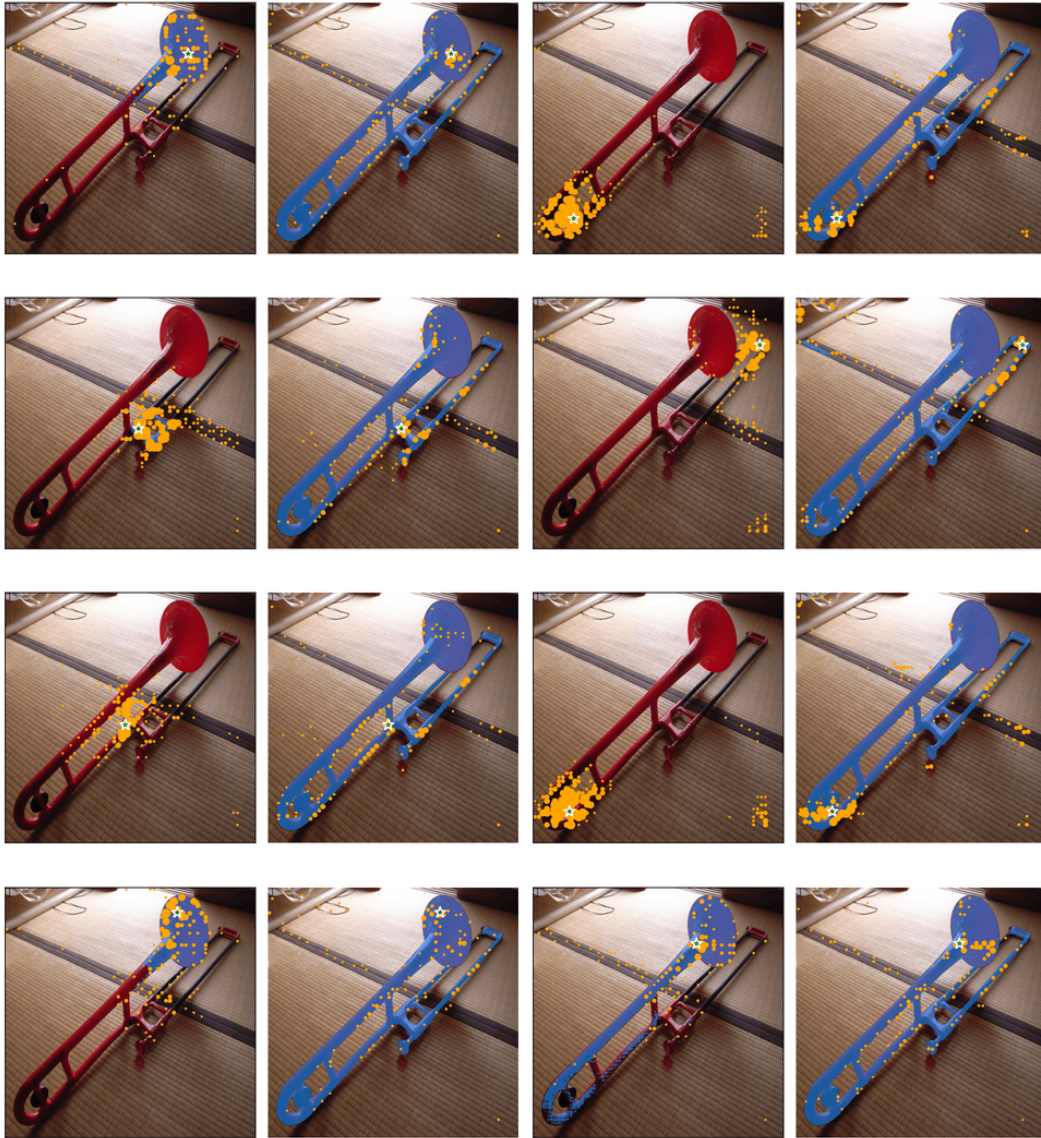


Figure 9: Visual results for 1-point prompt.

810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863

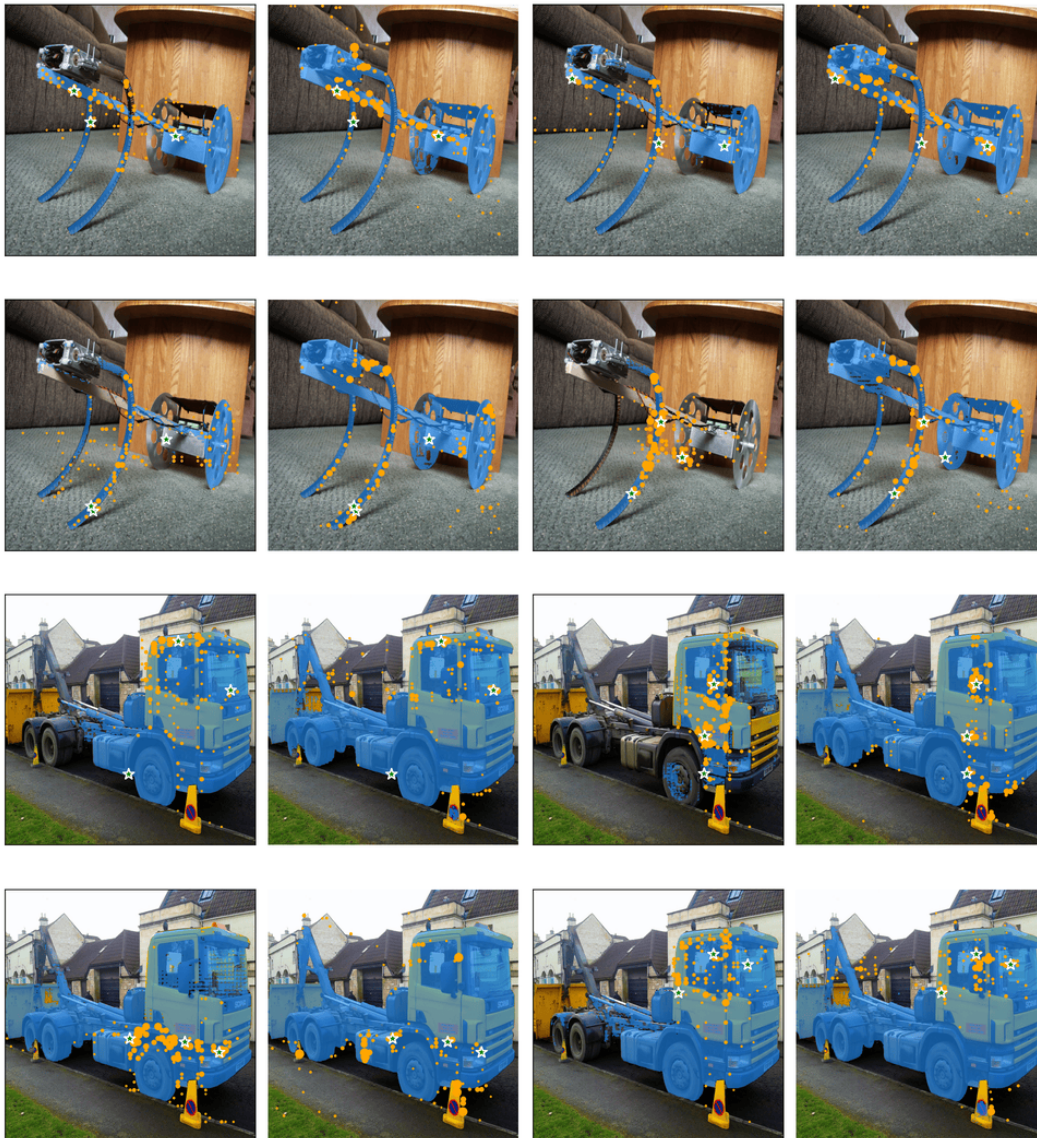


Figure 10: Visual results for 3-points prompt.



864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917

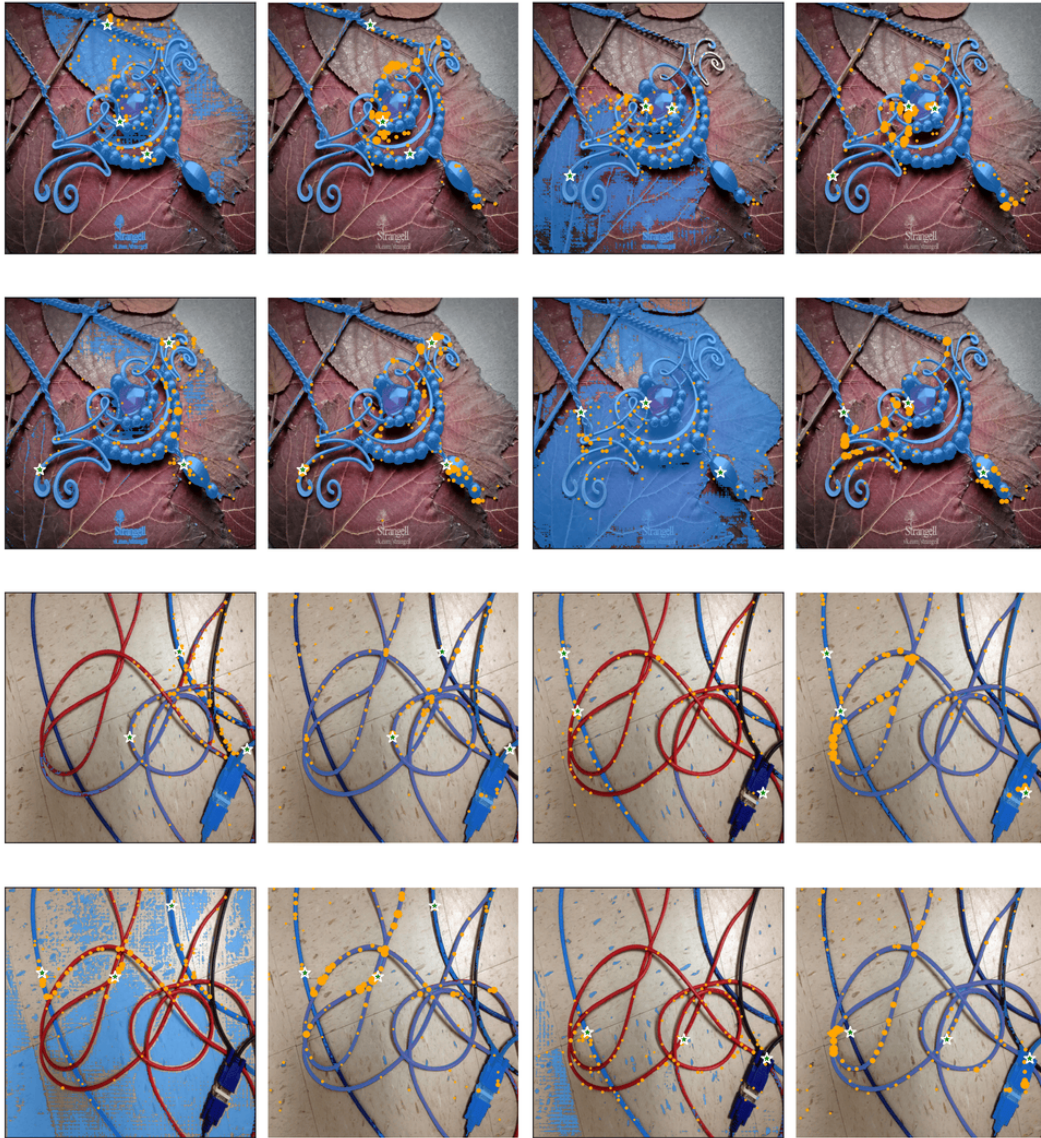


Figure 11: Visual results for 3-points prompt.

918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971

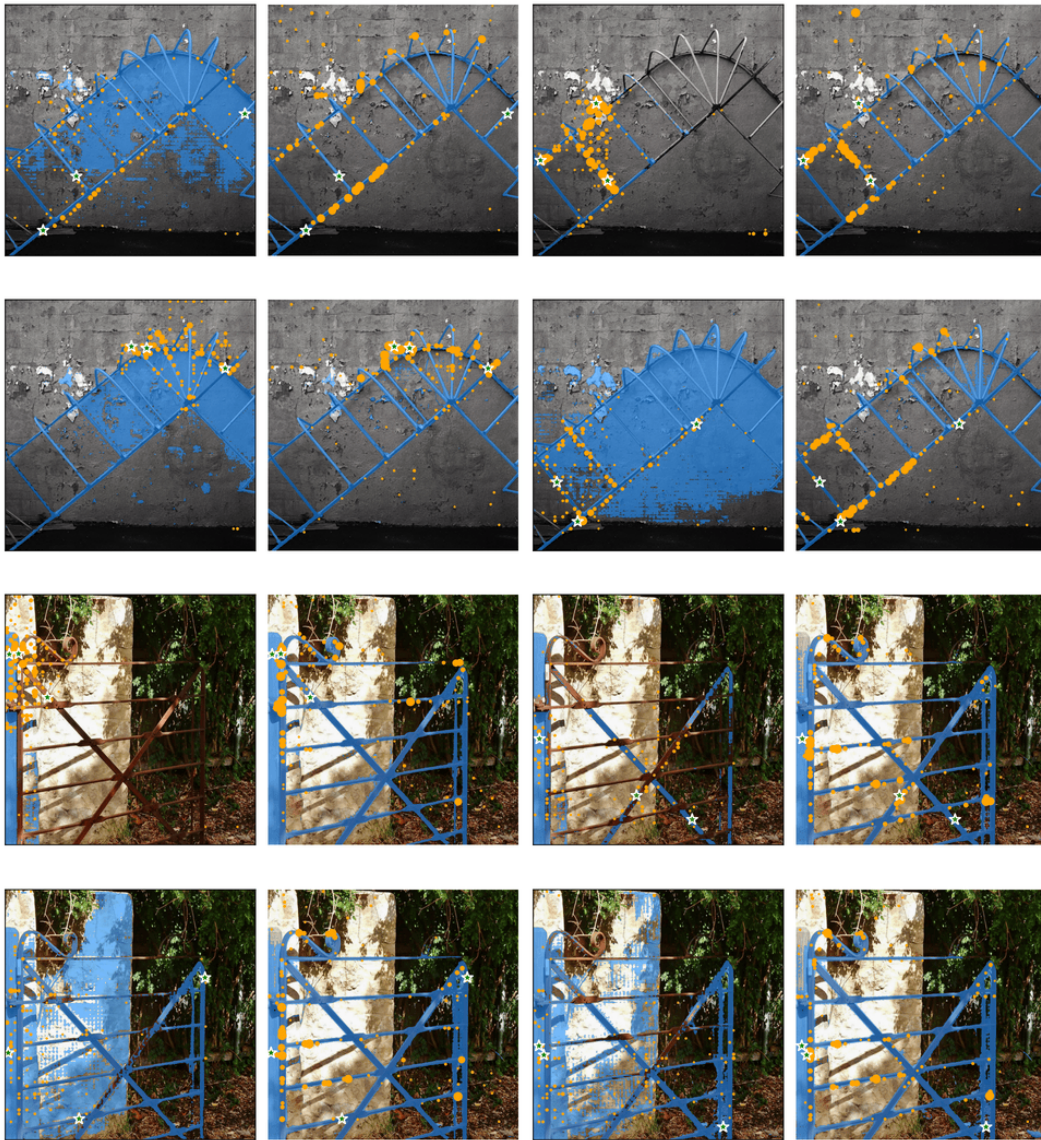


Figure 12: Visual results for 3-points prompt.

## REFERENCES

- 972  
973  
974 Dina Bashkirova, Mohamed Abdelfattah, Ziliang Zhu, James Akl, Fadi Alladkani, Ping Hu, Vitaly  
975 Ablavsky, Berk Calli, Sarah Adel Bargal, and Kate Saenko. Zerowaste dataset: towards deformable  
976 object segmentation in cluttered scenes. In *CVPR*, 2022.
- 977 Eric Bianchi and Matthew Hebdon. Corrosion condition state semantic segmentation dataset. *Univer-*  
978 *sity Libraries, Virginia Tech: Blacksburg, VA, USA*, 2021.
- 979 Benedikt Blumenstiel, Johannes Jakubik, Hilde Kühne, and Michael Vössing. What a MESS:  
980 Multi-Domain Evaluation of Zero-shot Semantic Segmentation. In *NeurIPS Workshop*, 2023.
- 981 Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo.  
982 Adaptionformer: Adapting vision transformers for scalable visual recognition. *NeurIPS*, 2022.
- 983 Nadav Cohen, Yael Newman, and Ariel Shamir. Semantic segmentation in art paintings. In *Computer*  
984 *Graphics Forum*, 2022.
- 985 Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable  
986 convolutional networks. In *ICCV*, 2017.
- 987 Seyed Mohammad Hassan Erfani, Zhenyao Wu, Xinyi Wu, Song Wang, and Erfan Goharian. Atlantis:  
988 A benchmark for semantic segmentation of waterbody images. *Environmental Modelling &*  
989 *Software*, 2022.
- 990 Muhammad Moazam Fraz, Paolo Remagnino, Andreas Hoppe, Bunyarit Uyyanonvara, Alicja R  
991 Rudnicka, Christopher G Owen, and Sarah A Barman. An ensemble classification-based approach  
992 applied to retinal blood vessel segmentation. *IEEE Transactions on Biomedical Engineering*, 2012.
- 993 Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and  
994 Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation.  
995 In *CVPR*, 2021.
- 996 Rahul Ghosh, Praveen Ravirathinam, Xiaowei Jia, Ankush Khandelwal, David Mulla, and Vipin  
997 Kumar. Calcrop21: A georeferenced multi-spectral dataset of satellite imagery and crop labels. In  
998 *ICBD*, 2021.
- 999 Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning  
1000 for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608*, 2024.
- 1001 Sebastian Haug and Jörn Ostermann. A crop/weed field image dataset for the evaluation of computer  
1002 vision based precision agriculture tasks. In *ECCV Workshops*, 2015.
- 1003 Ju He, Shuo Yang, Shaokang Yang, Adam Kortylewski, Xiaoding Yuan, Jie-Neng Chen, Shuai Liu,  
1004 Cheng Yang, Qihang Yu, and Alan Yuille. Partimagenet: A large, high-quality dataset of parts. In  
1005 *ECCV*, 2022.
- 1006 Qibin Hou, Daquan Zhou, and Jiashi Feng. Coordinate attention for efficient mobile network design.  
1007 In *CVPR*, 2021.
- 1008 Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yanzhi Li, Shean Wang, Lu Wang,  
1009 and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022.
- 1010 Md Jahidul Islam, Chelsey Edge, Yuyang Xiao, Peigen Luo, Muntaqim Mehtaz, Christopher Morse,  
1011 Sadman Sakib Enan, and Junaed Sattar. Semantic segmentation of underwater imagery: Dataset  
1012 and benchmark. In *IROS*, 2020.
- 1013 Debesh Jha, Sharib Ali, Krister Emanuelsen, Steven A Hicks, Vajira Thambawita, Enrique Garcia-  
1014 Ceja, Michael A Riegler, Thomas de Lange, Peter T Schmidt, Håvard D Johansen, et al. Kvasir-  
1015 instrument: Diagnostic and therapeutic tool segmentation dataset in gastrointestinal endoscopy. In  
1016 *MultiMedia Modeling*, 2021.
- 1017 Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu.  
1018 Segment anything in high quality. In *NeurIPS*, 2023.
- 1019  
1020  
1021  
1022  
1023  
1024  
1025

- 1026 Kourosh Khoshelham, L Díaz Vilariño, Michael Peter, Zhizhong Kang, and Debaditya Acharya. The  
1027 isprs benchmark on indoor modelling. *The International Archives of the Photogrammetry, Remote*  
1028 *Sensing and Spatial Information Sciences*, 2017.
- 1029 Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete  
1030 Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick.  
1031 Segment anything. In *ICCV*, 2023.
- 1032 Jianshu Li, Jian Zhao, Yunchao Wei, Congyan Lang, Yidong Li, Terence Sim, Shuicheng Yan, and  
1033 Jiashi Feng. Multiple-human parsing in the wild. *arXiv preprint arXiv:1705.07206*, 2017.
- 1034 Yahui Liu, Jian Yao, Xiaohu Lu, Renping Xie, and Li Li. Deepcrack: A deep hierarchical feature  
1035 learning architecture for crack segmentation. *Neurocomputing*, 2019.
- 1036 Ye Lyu, George Vosselman, Gui-Song Xia, Alper Yilmaz, and Michael Ying Yang. Uavid: A semantic  
1037 segmentation dataset for uav imagery. *ISPRS journal of photogrammetry and remote sensing*,  
1038 2020.
- 1039 Amirreza Mahbod, Gerald Schaefer, Benjamin Bancher, Christine Löw, Georg Dorffner, Rupert Ecker,  
1040 and Isabella Ellinger. Cryonuseg: A dataset for nuclei instance segmentation of cryosectioned  
1041 h&e-stained histological images. *Computers in biology and medicine*, 2021.
- 1042 Gonzalo Mateo-Garcia, Joshua Veitch-Michaelis, Lewis Smith, Silviu Vlad Oprea, Guy Schumann,  
1043 Yarin Gal, Atılım Güneş Baydin, and Dietmar Backes. Towards global flood mapping onboard low  
1044 cost satellites with machine learning. *Scientific reports*, 2021.
- 1045 Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural  
1046 networks for volumetric medical image segmentation. In *3DV*, 2016.
- 1047 Maryam Rahnemoonfar, Tashnim Chowdhury, Argho Sarkar, Debvrat Varshney, Masoud Yari, and  
1048 Robin Roberson Murphy. Floodnet: A high resolution aerial imagery dataset for post flood scene  
1049 understanding. *IEEE Access*, 2021.
- 1050 Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided curriculum model adaptation and  
1051 uncertainty-aware evaluation for semantic nighttime image segmentation. In *ICCV*, 2019.
- 1052 Constantin Seibold, Simon Reiß, Saquib Sarfraz, Matthias A Fink, Victoria Mayer, Jan Sellner,  
1053 Moon Sung Kim, Klaus H Maier-Hein, Jens Kleesiek, and Rainer Stiefelhagen. Detailed annota-  
1054 tions of chest x-rays via ct projection for report understanding. In *BMVC*, 2022.
- 1055 Shreyas S Shivakumar, Neil Rodrigues, Alex Zhou, Ian D Miller, Vijay Kumar, and Camillo J Taylor.  
1056 Pst900: Rgb-thermal calibration, dataset and segmentation network. In *ICRA*, 2020.
- 1057 Konstantin Sofiiuk, Ilya A Petrov, and Anton Konushin. Reviving iterative training with mask  
1058 guidance for interactive segmentation. In *ICIP*, 2022.
- 1059 Syed Waqas Zamir, Aditya Arora, Akshita Gupta, Salman Khan, Guolei Sun, Fahad Shahbaz Khan,  
1060 Fan Zhu, Ling Shao, Gui-Song Xia, and Xiang Bai. isaid: A large-scale dataset for instance  
1061 segmentation in aerial images. In *CVPR Workshops*, 2019.
- 1062 Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and  
1063 Pietro Perona. Caltech-ucsd birds 200. 2010.
- 1064 Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block  
1065 attention module. In *ECCV*, 2018.
- 1066 Zhuofan Xia, Xuran Pan, Shiji Song, Li Erran Li, and Gao Huang. Vision transformer with deformable  
1067 attention. In *CVPR*, 2022.
- 1068 Yi Xin, Siqi Luo, Haodi Zhou, Junlong Du, Xiaohong Liu, Yue Fan, Qing Li, and Yuntao  
1069 Du. Parameter-efficient fine-tuning for pre-trained vision models: A survey. *arXiv preprint*  
1070 *arXiv:2402.02242*, 2024.
- 1071 Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan,  
1072 and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In  
1073 *CVPR*, 2020.