Supplementary material: Segment Anything

We recommend reading the full paper with original formatting at: arxiv.org/abs/2304.02643.

Table of contents:

A. Acknowledgments

We would like to thank Aaron Adcock and Jitendra Malik for helpful discussion. We thank Vaibhav Aggarwal and Yanghao Li for help with scaling the model. We thank Cheng-Yang Fu, Jiabo Hu, and Robert Kuo for help with data annotation platform. We thank Allen Goodman and Bram Wasti for help in optimizing web-version of our model. Finally, we thank Morteza Behrooz, Ashley Gabriel, Ahuva Goldstand, Sumanth Gurram, Somya Jain, Devansh Kukreja, Joshua Lane, Lilian Luong, Mallika Malhotra, William Ngan, Omkar Parkhi, Nikhil Raina, Dirk Rowe, Neil Sejoor, Vanessa Stark, Bala Varadarajan, and Zachary Winstrom for their help in making the demo, dataset viewer, and other assets and tooling.

B. Segment Anything Model and Task Details

Image encoder. In general, the image encoder can be any network that outputs a $C \times H \times W$ image embedding. Motivated by scalability and access to strong pre-training, we use an MAE [46] pre-trained Vision Transformer (ViT) [32] with minimal adaptations to process high resolution inputs, specifically a ViT-H/16 with 14×14 windowed attention and four equally-spaced global attention blocks, following [60]. The image encoder's output is a $16\times$ downscaled embedding of the input image. Since our runtime goal is to process each prompt in real-time, we can afford a high number of image encoder FLOPs because they are computed only once per image, *not* per prompt.

Following standard practices (*e.g.*, [39]), we use an input resolution of 1024×1024 obtained by rescaling the image and padding the shorter side. The image embedding is therefore 64×64. To reduce the channel dimension, following [60], we use a 1×1 convolution to get to 256 channels, followed by a 3×3 convolution also with 256 channels. Each convolution is followed by a layer normalization [4].

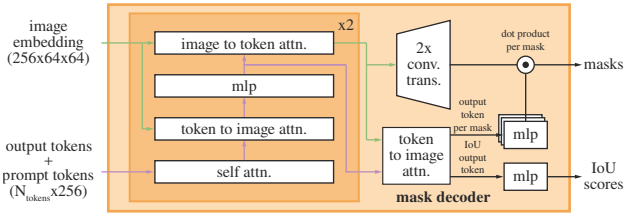


Figure 9: Details of the lightweight mask decoder. A two-layer decoder updates both the image embedding and prompt tokens via cross-attention. Then the image embedding is upscaled, from which the updated output tokens are used to dynamically predict masks. (Not illustrated for figure clarity: At every attention layer, positional encodings are added to the image embedding, and the entire original prompt token (including position encoding) is re-added to the token queries and keys.)

Prompt encoder. Sparse prompts are mapped to 256-dimensional vectorial embeddings as follows. A point is represented as the sum of a positional encoding [93] of the point's location and one of two learned embeddings that indicate if the point is either in the foreground or background. A box is represented by an embedding pair: (1) the positional encoding of its top-left corner summed with a learned embedding representing "top-left corner" and (2) the same structure but using a learned embedding indicating "bottom-right corner". Finally, to represent free-form text we use the text encoder from CLIP [80] (any text encoder is possible in general). We focus on geometric prompts for the remainder of this section and discuss text prompts in depth in §E.5.

Dense prompts (*i.e.*, masks) have a spatial correspondence with the image. We input masks at a $4\times$ lower resolution than the input image, then downscale an additional $4\times$ using two 2×2, stride-2 convolutions with output channels 4 and 16, respectively. A final 1×1 convolution maps the channel dimension to 256. Each layer is separated by GELU activations [48] and layer normalization. The mask and image embedding are then added element-wise. If there is no mask prompt, a learned embedding representing "no mask" is added to each image embedding location.

Lightweight mask decoder. This module efficiently maps the image embedding and a set of prompt embeddings to an output mask. To combine these inputs, we take inspiration from Transformer segmentation models [13, 19] and modify a standard Transformer decoder [101]. Before applying our decoder, we first insert into the set of prompt embeddings a learned output token embedding that will be used at the decoder's output, analogous to the `[class]` token in [32]. For simplicity, we refer to these embeddings (*not* including the image embedding) collectively as "tokens".

Our decoder design is shown in Fig. 9. Each decoder layer performs 4 steps: (1) self-attention on the tokens, (2) cross-attention from tokens (as queries) to the image em-

bedding, (3) a point-wise MLP updates each token, and (4) cross-attention from the image embedding (as queries) to tokens. This last step updates the image embedding with prompt information. During cross-attention, the image embedding is treated as a set of 64^2 256-dimensional vectors. Each self/cross-attention and MLP has a residual connection [47], layer normalization, and a dropout [91] of 0.1 at training. The next decoder layer takes the updated tokens and the updated image embedding from the previous layer. We use a two-layer decoder.

To ensure the decoder has access to critical geometric information the positional encodings are added to the image embedding whenever they participate in an attention layer. Additionally, the *entire* original prompt tokens (including their positional encodings) are re-added to the updated tokens whenever they participate in an attention layer. This allows for a strong dependence on both the prompt token's geometric location and type.

After running the decoder, we upsample the updated image embedding by $4\times$ with two transposed convolutional layers (now it's downscaled $4\times$ relative to the input image). Then, the tokens attend once more to the image embedding and we pass the updated output token embedding to a small 3-layer MLP that outputs a vector matching the channel dimension of the upscaled image embedding. Finally, we predict a mask with a spatially point-wise product between the upscaled image embedding and the MLP's output.

The transformer uses an embedding dimension of 256. The transformer MLP blocks have a large internal dimension of 2048, but the MLP is applied only to the prompt tokens for which there are relatively few (rarely greater than 20). However, in cross-attention layers where we have a 64×64 image embedding, we reduce the channel dimension of the queries, keys, and values by $2\times$ to 128 for computational efficiency. All attention layers use 8 heads.

The transposed convolutions used to upscale the output image embedding are 2×2, stride 2 with output channel dimensions of 64 and 32 and have GELU activations. They are separated by layer normalization.

Making the model ambiguity-aware. As described, a single input prompt may be ambiguous in the sense that it corresponds to multiple valid masks, and the model will learn to average over these masks. We eliminate this problem with a simple modification: instead of predicting a single mask, we use a small number of output tokens and predict multiple masks simultaneously. By default we predict three masks, since we observe that three layers (whole, part, and subpart) are often enough to describe nested masks. During training, we compute the loss (described shortly) between the ground truth and each of the predicted masks, but only backpropagate from the lowest loss. This is a common technique used for models with multiple outputs [14, 44, 62]. For use in applications, we'd like to rank predicted masks, so we add a small head (operating on an additional output token) that estimates the IoU between each predicted mask and the object it covers.

Ambiguity is much rarer with multiple prompts and the three output masks will usually become similar. To minimize computation of degenerate losses at training and ensure the single unambiguous mask receives a regular gradient signal, we only predict a single mask when more than one prompt is given. This is accomplished by adding a fourth output token for an additional mask prediction. This fourth mask is never returned for a single prompt and is the only mask returned for multiple prompts.

Losses. We supervise mask prediction with a linear combination of focal loss [63] and dice loss [71] in a 20:1 ratio of focal loss to dice loss, following [19, 13]. Unlike [19, 13], we observe that auxiliary deep supervision after each decoder layer is unhelpful. The IoU prediction head is trained with mean-square-error loss between the IoU prediction and the predicted mask's IoU with the ground truth mask. It is added to the mask loss with a constant scaling factor of 1.0.

Training algorithm. Following recent approaches [90, 36], we simulate an interactive segmentation setup during training. First, with equal probability either a foreground point or bounding box is selected randomly for the target mask. Points are sampled uniformly from the ground truth mask. Boxes are taken as the ground truth mask's bounding box, with random noise added in each coordinate with standard deviation equal to 10% of the box sidelength, to a maximum of 20 pixels. This noise profile is a reasonable compromise between applications like instance segmentation, which produce a tight box around the target object, and interactive segmentation, where a user may draw a loose box.

After making a prediction from this first prompt, subsequent points are selected uniformly from the error region between the previous mask prediction and the ground truth mask. Each new point is foreground or background if the error region is a false negative or false positive, respectively. We also supply the mask prediction from the previous iteration as an additional prompt to our model. To provide the next iteration with maximal information, we supply the unthresholded mask logits instead of the binarized mask. When multiple masks are returned, the mask passed to the next iteration and used to sample the next point is the one with the highest predicted IoU.

We find diminishing returns after 8 iteratively sampled points (we have tested up to 16). Additionally, to encourage the model to benefit from the supplied mask, we also use two more iterations where no additional points are sampled. One of these iterations is randomly inserted among the 8 iteratively sampled points, and the other is always at the end. This gives 11 total iterations: one sampled initial input prompt, 8 iteratively sampled points, and two iterations where no new external information is supplied to the model

so it can learn to refine its own mask predictions. We note that using a relatively large number of iterations is possible because our lightweight mask decoder requires less than 1% of the image encoder's compute and, therefore, each iteration adds only a small overhead. This is unlike previous interactive methods that perform only one or a few interactive steps per optimizer update [68, 9, 36, 90].

Training recipe. We use the AdamW [66] optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$) and a linear learning rate warmup [41] for 250 iterations and a step-wise learning rate decay schedule. The initial learning rate (*lr*), after warmup, is $8e^{-4}$. We train for 90k iterations (~2 SA-1B epochs) and decrease the *lr* by a factor of 10 at 60k iterations and again at 86666 iterations. The batch size is 256 images. To regularize SAM, we set weight decay (*wd*) to 0.1 and apply drop path [51] (*dp*) with a rate of 0.4. We use a layer-wise learning rate decay [5] (*ld*) of 0.8. No data augmentation is applied. We initialize SAM from an MAE [46] pre-trained ViT-H. We distribute training across 256 GPUs, due to the large image encoder and 1024×1024 input size. To limit GPU memory usage, we train with up to 64 randomly sampled masks per GPU. Additionally, we find that lightly filtering SA-1B masks to discard any that cover more than 90% of the image qualitatively improves results.

For ablations and others variations on training (*e.g.*, text-to-mask §E.5), we deviate from the default recipe above as follows. When training with data from the first and second data engine stages only, we augment the input with large-scale jitter [39] with a scale range of [0.1, 2.0]. Intuitively, data augmentation may be helpful when training data is more limited. To train ViT-B and ViT-L, we use 180k iterations with batch size 128 distributed across 128 GPUs. We set *lr* = $8e^{-4}$/$4e^{-4}$, *ld* = 0.6/0.8, *wd* = 0.1, and *dp* = 0.6/0.4 for ViT-B/L, respectively.

C. Automatic Mask Generation Details

Here we discuss details of the data engine's fully automatic stage that was used to generate the released SA-1B.

Cropping. Masks were generated from a regular grid of 32×32 points on the full image and 20 additional zoomed-in image crops arising from 2×2 and 4×4 partially overlapping windows using 16×16 and 8×8 regular point grids, respectively. The original high-resolution images were used for cropping (this was the only time we used them). We removed masks that touch the inner boundaries of the crops. We applied standard greedy box-based NMS (boxes were used for efficiency) in two phases: first within each crop and second across crops. When applying NMS within a crop, we used the model's predicted IoU to rank masks. When applying NMS across crops, we ranked masks from most zoomed-in (*i.e.*, from a 4×4 crop) to least zoomed-in (*i.e.*, the original image), based on their source crop. In both cases, we used an NMS threshold of 0.7.

Filtering. We used three filters to increase mask quality. First, to keep only *confident* masks we filtered by the model's predicted IoU score at a threshold of 88.0. Second, to keep only *stable* masks we compared two binary masks resulting from the same underlying soft mask by thresholding it at different values. We kept the prediction (*i.e.*, the binary mask resulting from thresholding logits at 0) only if the IoU between its pair of -1 and +1 thresholded masks was equal to or greater than 95.0. Third, we noticed that occasionally an automatic mask would cover the entire image. These masks were generally uninteresting, and we filtered them by removing masks that covered 95% or more of an image. All filtering thresholds were selected to achieve both a large number of masks and high mask quality as judged by professional annotators using the method described in §5.

Postprocessing. We observed two error types that are easily mitigated with postprocessing. First, an estimated 4% of masks include small, spurious components. To address these, we removed connected components with area less than 100 pixels (including removing entire masks if the largest component is below this threshold). Second, another estimated 4% of masks include small, spurious holes. To address these, we filled holes with area less than 100 pixels. Holes were identified as components of inverted masks.

Automatic mask generation model. We trained a special version of SAM for fully automatic mask generation that sacrifices some inference speed for improved mask generation properties. We note the differences between our default SAM and the one used for data generation here: it was trained on manual and semi-automatic data only, it was trained for longer (177656 iterations instead of 90k) with large-scale jitter data augmentation [39], simulated interactive training used only point and mask prompts (no boxes) and sampled only 4 points per mask during training (reducing from our default of 9 to 4 sped up training iterations and had no impact on 1-point performance, though it would harm mIoU if evaluating with more points), and finally the mask decoder used 3 layers instead of 2.

SA-1B examples. We show SA-1B samples in Fig. 2. For more examples, please see our dataset explorer.

D. Segment Anything RAI Analysis

We next perform a Responsible AI (RAI) analysis of our work by investigating potential fairness concerns and biases when using SA-1B and SAM. We focus on the geographic and income distribution of SA-1B and fairness of SAM across protected attributes of people. We also provide dataset, data annotation, and model cards in §G.

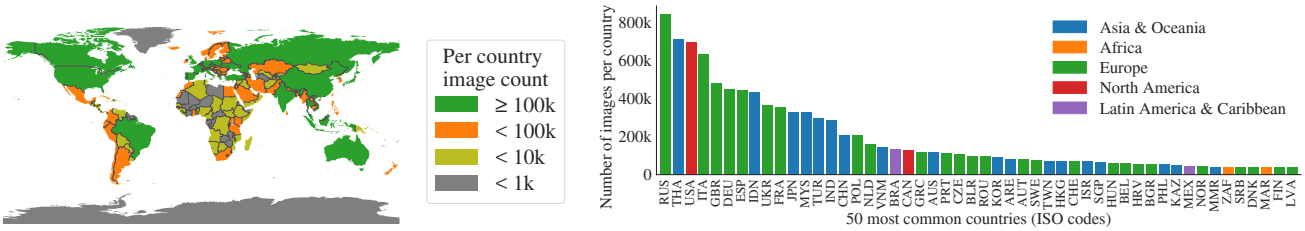


Figure 10: Estimated geographic distribution of SA-1B images. Most of the world's countries have more than 1000 images in SA-1B, and the three countries with the most images are from different parts of the world.

		SA-1B		% images		
	# countries	#imgs	#masks	SA-1B	COCO	O.I.
Africa	54	300k	28M	2.8%	3.0%	1.7%
Asia & Oceania	70	3.9M	423M	36.2%	11.4%	14.3%
Europe	47	5.4M	540M	49.8%	34.2%	36.2%
Latin America & Carib.	42	380k	36M	3.5%	3.1%	5.0%
North America	4	830k	80M	7.7%	48.3%	42.8%
high income countries	81	5.8M	598M	54.0%	89.1%	87.5%
middle income countries	108	4.9M	499M	45.0%	10.5%	12.0%
low income countries	28	100k	9.4M	0.9%	0.4%	0.5%

Table 1: Comparison of geographic and income representation. SA-1B has higher representation in Europe and Asia & Oceania as well as middle income countries. Images from Africa, Latin America & Caribbean, as well as low income countries, are underrepresented in all datasets.

D.1. Geographic and Income Distribution

To perform the analysis we infer the country images were photographed in using standard methods for SA-1B, COCO [64], and Open Images [58] datasets.

Inferring geographic information for SA-1B. While the images in SA-1B are not geo-tagged, each image has a caption describing its contents and where it was taken. We infer approximate image geo-locations from these captions using an Elmo-based named entity recognition model [76]. Each extracted location entity is mapped to every matching country, province, and city. Captions are mapped to a single country by first considering the matching countries, then provinces, and finally cities. We note that there are ambiguities and potential for biases with this method (*e.g.*, "Georgia" may refer to the country or the US state). As such, we use the extracted locations to analyze the dataset as a whole, but do not release the inferred locations. The captions will not be released publicly as required by the image provider.

Inferring geographic information for COCO and Open Images. The COCO [64] and Open Images [58] datasets do not provide geo-locations. Following [28], we retrieve geographic metadata using the Flickr API. We retrieved locations for 24% of the COCO training set (19,562 images) and for Open Images we retrieved 18% of the training set (493,517 images, after only considering images with masks). We note that the geographic information is approximate, and the sample of images with this information may not fully match the full dataset distribution.

Inferring income information. We use each image's inferred country to look up its income level using the levels defined by The World Bank [96]. We collapse the upper-middle and lower-middle levels into a single middle level.

Analysis. In Fig. 10 we visualize the per-country image counts in SA-1B (left) and the 50 countries with the most images (right). We note that the top-three countries are from different parts of the world. Next, in Table 1 we compare the geographic and income representation of SA-1B, COCO [64], and Open Images [58]. SA-1B has a substantially higher percentage of images in Europe and Asia & Oceania as well as in middle income countries. All datasets underrepresent Africa as well as low income countries. We note that in SA-1B, all regions, including Africa, have at least 28 million masks, 10× more than the *total* number of masks of any previous dataset. Finally, we observe that the average number of masks per image (not shown) is fairly consistent across region and income (94-108 per image).

D.2. Fairness Across Protected Attributes of People

Fairness in segmenting people. We investigate potential fairness concerns across perceived gender presentation, perceived age group, and perceived skin tone by measuring the performance discrepancy of SAM between groups. We use the More Inclusive Annotations for People (MIAP) [85] test set annotations for Open Images [58] dataset for gender presentation and age. MIAP provides box annotations, while we need ground truth masks for this analysis. To get ground truth masks, we select each person-category mask from Open Images if its corresponding bounding box is within a 1% margin (based on relative box side lengths) of an annotated bounding box in MIAP, resulting in 3.9k masks. For skin tone we use a proprietary dataset. Our evaluation uses simulated interactive segmentation with random sampling of 1 and 3 points (see §E). Table 2 (top left) shows results for perceived gender presentation. We note that females have been shown to be underrepresented in detection and segmentation datasets [113], but observe that SAM performs similarly across groups. We repeat the analysis for

	mIoU at 1 point	mIoU at 3 points
perceived gender presentation		
feminine	54.4 ±1.7	90.4 ±0.6
masculine	55.7 ±1.7	90.1 ±0.6
perceived age group		
older	62.9 ±6.7	92.6 ±1.3
middle	54.5 ±1.3	90.2 ±0.5
young	54.2 ±2.2	91.2 ±0.7

	mIoU at 1 point	mIoU at 3 points
perceived skin tone		
1	52.9 ±2.2	91.0 ±0.9
2	51.5 ±1.4	91.1 ±0.5
3	52.2 ±1.9	91.4 ±0.7
4	51.5 ±2.7	91.7 ±1.0
5	52.4 ±4.2	92.5 ±1.4
6	56.7 ±6.3	91.2 ±2.4

Table 2: SAM's performance segmenting people across perceived gender presentation, age group, and skin tone. 95% confidence intervals are shown. Within each grouping, all confidence intervals overlap except older *vs*. middle.

	mIoU at 1 point	mIoU at 3 points
perceived gender presentation		
feminine	76.3 ±1.1	90.7 ±0.5
masculine	81.0 ±1.2	92.3 ±0.4

	mIoU at 1 point	mIoU at 3 points
perceived age group		
older	81.9 ±3.8	92.8 ±1.6
middle	78.2 ±0.8	91.3 ±0.3
young	77.3 ±2.7	91.5 ±0.9

Table 3: SAM's performance segmenting clothing across perceived gender presentation and age group. The intervals for perceived gender are disjoint, with mIoU for masculine being higher. Confidence intervals for age group overlap.

perceived age in Table 2 (bottom left), noting that those who are perceived to be younger and older have been shown to be underrepresented in large-scale datasets [108]. SAM performs best on those who are perceived older (although the confidence interval is large). Finally, we repeat the analysis for perceived skin tone in Table 2 (right), noting that those with lighter apparent skin tones have been shown to be overrepresented and those with darker skin tones underrepresented in large-scale datasets [108]. As MIAP does not contain perceived skin tone annotations, we use a proprietary dataset that contains annotations for the perceived Fitzpatrick skin type [35], which ranges from 1 (lightest skin tone) to 6 (darkest skin tone). While the means vary somewhat, we do not find a significant difference across groups. We believe our findings stem from the nature of the task, and acknowledge biases may arise when SAM is used as a component in larger systems.

Fairness in segmenting clothing. We extend our analysis to clothing segmentation. We look at SAM's performance on clothing relative to the attributes of those wearing the clothes. We use all 6.5k ground truth masks from Open Images that have a category under the clothing superclass and reside within a person box from MIAP. In Table 3 we compare performance across perceived gender presentation and age group. We find that SAM is better at segmenting clothing on those who present predominantly masculine, with disjoint 95% confidence intervals. The gap closes when moving from 1 to 3 point evaluation. Differences for perceived age group are not significant. Our results indicate there is a bias when segmenting clothing across perceived gender presentation with a one point prompt, and we encourage users of SAM to be mindful of this limitation.

E. Additional Experiments and Details

E.1. Zero-Shot Single Point Valid Mask Evaluation

Datasets. We built a new segmentation benchmark to evaluate the zero-shot transfer capabilities of our model using a suite of 23 diverse segmentation datasets from prior work. A description of each dataset is given in Table 4. For examples, see main text Fig. **??**. This suite covers a range of domains including egocentric [33, 27, 111], microscopy [11], X-ray [102], underwater [50, 98], aerial [16], simulation [84], driving [24], and painting [23] images. For efficient evaluation we subsampled datasets with more than 15k masks. Specifically, we randomly picked images so that the total number of masks in the sampled images was ~10k. We blurred faces of people in all the datasets.

Point sampling. Our default point sampling follows standard practice in interactive segmentation [107, 62, 90]. The first point is chosen deterministically as the point farthest from the object boundary. Each subsequent point is the farthest from the boundary of the error region between ground truth and the previous prediction. Some experiments (where specified) use a more challenging sampling strategy in which the first point is a *random* point, rather than a deterministically selected "center" point. Each subsequent point is selected as described above. This setting better reflects use cases in which the first point is not reliably near the center of the mask, such as prompting from eye gaze.

Evaluation. We measure IoU between a prediction after N point prompts and a ground truth mask, where $N = \{1, 2, 3, 5, 9\}$ and points are sampled iteratively with either of the strategies described above. The per-dataset mIoU is the per-mask IoU averaged across all objects in the dataset. Finally, we report the top-line metric by averaging the per-dataset mIoUs across all 23 datasets. Our evaluation differs from the standard interactive segmentation evaluation protocol which measures the average number of points needed to achieve $X\%$ IoU, with up to 20 points. We focus on predictions after just one, or possibly a few points, since many of our use cases involve a single or very few prompts. Given our application focus, which requires real-time prompt processing, we expect the best interactive segmentation models to outperform SAM when using a large number of points.

Baselines. We use three recent strong interactive baselines: RITM [90], FocalClick [17], and SimpleClick [65]. For each, we use the largest models trained on the broadest datasets publicly released by the authors. For RITM, we use `HRNet32 IT-M` trained on the combination of COCO [64] and LVIS [43] introduced by the authors.

dataset	abbreviation & link	image type	description	mask type	source split	# images sampled	# masks sampled
Plant Phenotyping Datasets Leaf Segmentation [72]	PPDLS	Plants	Leaf segmentation for images of tobacco and ara plants.	Instance	N/A	182	2347
BBBC038v1 from Broad Bioimage Benchmark Collection [11]	BBBC038v1	Microscopy	Biological images of cells in a variety of settings testing robustness in nuclei segmentation.	Instance	Train	227	10506
Dataset fOr bOuldeRs Segmentation [78]	DOORS	Boulders	Segmentation masks of single boulders positioned on the surface of a spherical mesh.	Instance	DS1	10000	10000
TimberSeg 1.0 [37]	TimberSeg	Logs	Segmentation masks of individual logs in piles of timber in various environments and conditions. Images are taken from an operator's point-of-view.	Instance	N/A	220	2487
Northumberland Dolphin Dataset 2020 [98]	NDD20	Underwater	Segmentation masks of two different dolphin species in images taken above and under water.	Instance	N/A	4402	6100
Large Vocabulary Instance Segmentation [43]	LVIS	Scenes	Additional annotations for the COCO [64] dataset to enable the study of long-tailed object detection and segmentation.	Instance	Validation (v0.5)	945	9642
STREETS [89]	STREETS	Traffic camera	Segmentation masks of cars in traffic camera footage.	Instance	N/A	819	9854
ZeroWaste-f [6]	ZeroWaste-f	Recycling	Segmentation masks in cluttered scenes of deformed recycling waste.	Instance	Train	2947	6155
iShape [109]	iShape	Irregular shapes	Segmentation masks of irregular shapes like antennas, logs, fences, and hangers.	Instance	Validation	754	9742
ADE20K [115]	ADE20K	Scenes	Object and part segmentation masks for images from SUN [105] and Places [114] datasets.	Instance	Validation	302	10128
Occluded Video Instance Segmentation [79]	OVIS	Occlusions	Instance segmentation masks in videos, focusing on objects that are occluded.	Instance	Train	2044	10011
Hypersim [84]	Hypersim	Simulation	Photorealistic synthetic dataset of indoor scenes with instance masks.	Instance	Evermotion archinteriors volumes 1-55 excluding 20,25,40,49	338	9445
Night and Day Instance Segmented Park [21, 22]	NDISPark	Parking lots	Images of parking lots from video footage taken at day and night during different weather conditions and camera angles for vehicle segmentation.	Instance	Train	111	2577
EPIC-KITCHENS VISOR [27, 26]	VISOR	Egocentric	Segmentation masks for hands and active objects in ego-centric video from the cooking dataset EPIC-KITCHENS [26].	Instance	Validation	1864	10141
Plittersdorf dataset [45]	Plittersdorf	Stereo images	Segmentation masks of wildlife in images taken with the SOCRATES stereo camera trap.	Instance	Train, validation, test	187	546
Egocentric Hand-Object Segmentation [111]	EgoHOS	Egocentric	Fine-grained egocentric hand-object segmentation dataset. Dataset contains mask annotations for existing datasets.	Instance	Train (including only Ego4D [42] and THU-READ [95, 94])	2940	9961
InstanceBuilding 2D [16]	IBD	Drones	High-resolution drone UAV images annotated with roof instance segmentation masks.	Instance	Train (2D annotations)	467	11953
WoodScape [110]	WoodScape	Fisheye driving	Fisheye driving dataset with segmentation masks. Images are taken from four surround-view cameras.	Instance	Set 1	107	10266
Cityscapes [24]	Cityscapes	Driving	Stereo video of street scenes with segmentation masks.	Panoptic	Validation	293	9973
PIDray [102]	PIDRay	X-ray	Segmentation masks of prohibited items in X-ray images of baggage.	Instance	Test (hard)	3733	8892
Diverse Realism in Art Movements [23]	DRAM	Paintings	Domain adaptation dataset for semantic segmentation of art paintings.	Semantic	Test	718	1179
TrashCan [50]	TrashCan	Underwater	Segmentation masks of trash in images taken by underwater ROVs. Images are sourced from the J-EDI [67] dataset.	Instance	Train (instance task)	5936	9540
Georgia Tech Egocentric Activity Datasets [33, 61]	GTEA	Egocentric	Videos are composed of four different subjects performing seven types of daily activities with segmentation masks of hands.	Instance	Train (segmenting hands task)	652	1208

Table 4: Segmentation datasets used to evaluate zero-shot segmentation with point prompts. The 23 datasets cover a broad range of domains; see column "image type". To make our evaluation efficient, we subsample datasets that have more than 15k masks. Specifically, we randomly sampled images so that the total number of masks in the images is ~10k.

method	year	ODS	OIS	AP	R50
HED [106]	2015	.788	.808	.840	.923
EDETR [77]	2022	.840	.858	.896	.930
zero-shot transfer methods:					
Sobel filter	1968	.539	-	-	-
Canny [12]	1986	.600	.640	.580	-
Felz-Hutt [34]	2004	.610	.640	.560	-
SAM	2023	.768	.786	.794	.928

Table 5: Zero-shot transfer to edge detection on BSDS500.

	mask AR@1000						
method	all	small	med.	large	freq.	com.	rare
ViTDet-H [60]	63.0	51.7	80.8	87.0	63.1	63.3	58.3
zero-shot transfer methods:							
SAM – single out.	54.9	42.8	76.7	74.4	54.7	59.8	62.0
SAM	59.3	45.5	81.6	86.9	59.1	63.9	65.8

Table 6: Object proposal generation on LVIS v1. SAM is applied zero-shot, *i.e.* it was not trained for object proposal generation nor did it access LVIS images or annotations.

For FocalClick, we use `SegFormerB3-S2` trained on a "combined dataset" that includes 8 different segmentation datasets [17]. For SimpleClick, we use `ViT-H448` trained on a combination of COCO and LVIS. We follow the suggested default strategies for data pre-processing (*i.e.*, data augmentations or image resizing) and do not change or adapt any parameters for our evaluation. In our experiments, we observe that RITM outperforms other baselines on our 23 dataset suite with 1 point evaluation. Therefore, we use RITM as the default baseline. When evaluating with more points we report results for all baselines.

Single point ambiguity and oracle evaluation. In addition to IoU after N points prompts, we report SAM's "oracle" performance at 1 point by evaluating the predicted mask that best matches ground truth from amongst SAM's three predictions (rather than using the one that SAM itself ranks first, as we do by default). This protocol addresses possible single point prompt ambiguity by relaxing the requirement to guess the one right mask among several valid objects.

E.2. Zero-Shot Edge Detection

Dataset and metrics. We perform zero-shot edge detection experiments on BSDS500 [70, 3]. The ground truth for each image comes from the manual annotations of five different subjects. We report results on the 200 image test subset using the four standard metrics for edge detection [3, 31]: optimal dataset scale (ODS), optimal image scale (OIS), average precision (AP), and recall at 50% precision (R50).

Approach. We use a simplified version of our automatic mask generation pipeline. Specifically, we prompt SAM with a 16×16 regular grid of foreground points, which yields 768 predicted masks (three per point). We do not filter by predicted IoU or stability. Redundant masks are removed by NMS. Then we apply a Sobel filter to the remaining masks' unthresholded probability maps and set values to zero if they do not intersect with the outer boundary pixels of a mask. Finally, we take a pixel-wise max over all the predictions, linearly normalize the result to [0,1], and apply edge NMS [12] to thin the edges.

Results. We visualize representative edge maps in Fig. 11. Qualitatively, we observe that even though SAM was not trained for edge detection, it produces reasonable edge maps. Compared to the ground truth, SAM predicts more edges, including sensible ones that are not annotated in BSDS500. This bias is reflected quantitatively in Table 5: recall at 50% precision (R50) is high, at the cost of precision. SAM naturally lags behind state-of-the-art methods that learn the biases of BSDS500, *i.e.*, which edges to suppress. Nevertheless, SAM performs well compared to pioneering deep learning methods such as HED [106] (also trained on BSDS500) and significantly better than prior, though admittedly outdated, zero-shot transfer methods.

E.3. Zero-Shot Object Proposals

Next, we evaluate SAM on the mid-level task of object proposal generation [2, 100]. This task has played an important role in object detection research, serving as an intermediate step in pioneering systems (*e.g.*, [100, 40, 82]).

Dataset and metrics. We report the standard average recall (AR) metric for masks at 1000 proposals on the LVIS v1 validation set [43]. Since LVIS has high-quality masks for 1203 object classes, it provides a challenging test for object proposal generation. We focus on AR@1000 due to the open-world nature of our model, which will likely produce many valid masks outside even the 1203 classes in LVIS. To measure performance on frequent, common, and rare categories, we use AR@1000 but measured against a ground truth set containing just the corresponding LVIS categories.

Baseline. We use cascade ViTDet-H as a baseline, the strongest model from [60] by AP on LVIS. We note that this "baseline" corresponds to the "Detector Masquerading as Proposal generator" (DMP) method [15] that was shown to game AR, making it a stronger baseline than other models that focus on open-world proposals or segmentation [56, 103]. To produce 1000 proposals, we disable score thresholding in the three cascade stages and as raise the maximum number of predictions per stage to 1000.

Approach. We use a modified version of SAM's automatic mask generation pipeline for zero-shot transfer. First, to make inference time comparable to that of ViTDet we do not process image crops. Second, we remove filtering by predicted IoU and stability. This leaves two tunable parameters to get ∼1000 masks per image: the input point grid and the NMS threshold duplicate mask suppression. We choose

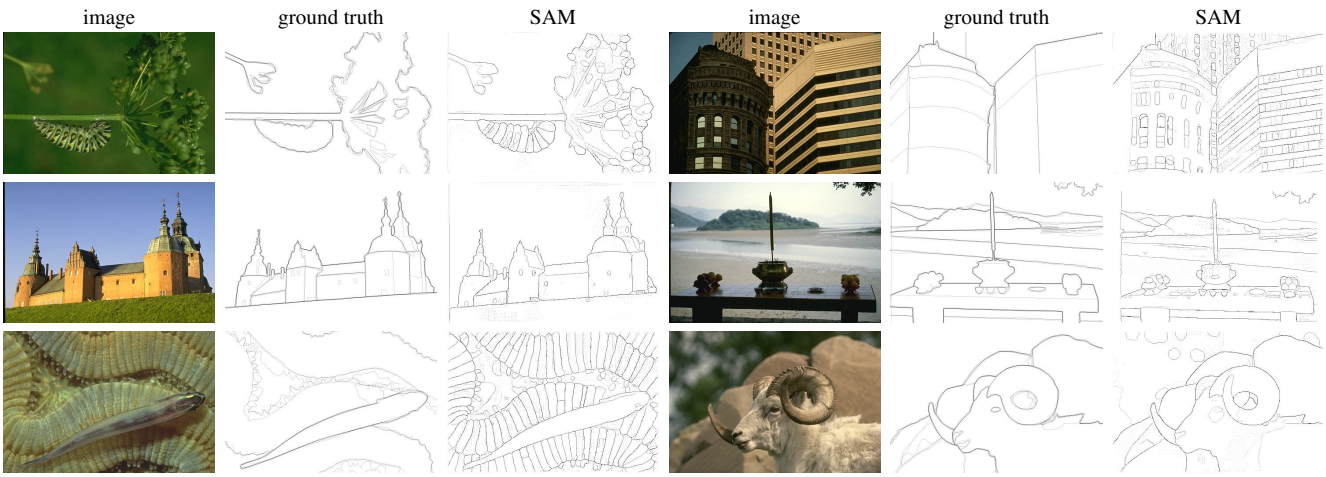


Figure 11: Zero-shot edge prediction on BSDS500. Recall that SAM was not trained to predict edge maps and did not have access to BSDS images and annotations during training.

a 64×64 point grid and an NMS threshold of 0.9, which produces ~900 masks per image on average. At evaluation, if greater than 1000 masks have been proposed in an image, they are ranked by the average of their confidence and stability scores, then truncated to the top 1000 proposals.

We hypothesize that SAM's ability to output multiple masks is especially valuable for this task, since recall should benefit from proposals generated at multiple scales from a single input point. To test this, we compare to an ablated version SAM that only outputs a single mask instead of three (SAM - single-output). Since this model produces fewer masks, we further increase the number of points sampled and NMS threshold to 128×128 and 0.95, respectively, obtaining ~950 masks per image on average. Additionally, single-output SAM does not produce the IoU score used to rank masks for NMS in the automatic mask generation pipeline, so instead masks are ranked randomly. Testing suggests this has similar performance to more sophisticated methods of ranking masks, such as using the max logit value of the mask as a proxy for model confidence.

Results. In Table 6 we see unsurprisingly that using the detections from ViTDet-H as object proposals (*i.e.*, the DMP method [15] that games AR) performs the best overall. However, SAM does remarkably well on several metrics. Notably, it outperforms ViTDet-H on medium and large objects, as well as rare and common objects. In fact, SAM only underperforms ViTDet-H on small objects and frequent objects, where ViTDet-H can easily learn LVIS-specific annotation biases since it was trained on LVIS, unlike SAM. We also compare against an ablated ambiguity-unaware version of SAM ("single out."), which performs significantly worse than SAM on all AR metrics.

E.4. Zero-Shot Instance Segmentation

Approach. Moving to higher-level vision, we use SAM as the segmentation module of an instance segmenter. The implementation is simple: we run a object detector (the ViTDet used before) and prompt SAM with its output boxes. This illustrates *composing* SAM in a larger system. More specifically, we prompt SAM with the boxes output by a fully-supervised ViTDet-H on COCO and LVIS v1 validation splits. We apply an additional mask refinement iteration by feeding the most confident predicted mask, together with the box prompt, back to the mask decoder to produce the final prediction.

Results. We compare the masks predicted by SAM and ViTDet on COCO and LVIS in Table 7. Looking at the mask AP metric we observe gaps on both datasets, where SAM is reasonably close, though certainly behind ViTDet. By visualizing outputs, we observed that SAM masks are often qualitatively better than those of ViTDet, with crisper boundaries (see Fig. 12). To investigate this observation, we conducted an additional human study asking annotators to rate the ViTDet masks and SAM masks on the 1 to 10 quality scale used before. In Fig. 13 we observe that SAM consistently outperforms ViTDet in the human study.

We hypothesize that on COCO, where the mask AP gap is larger and the ground truth quality is relatively low (as borne out by the human study), ViTDet learns the specific biases of COCO masks. SAM, being a zero-shot method, is unable to exploit these (generally undesirable) biases. The LVIS dataset has higher quality ground truth, but there are still specific idiosyncrasies (*e.g.*, masks do not contain holes, they are simple polygons by construction) and biases for modal *vs.* amodal masks. Again, SAM is not trained to learn these biases, while ViTDet can exploit them.

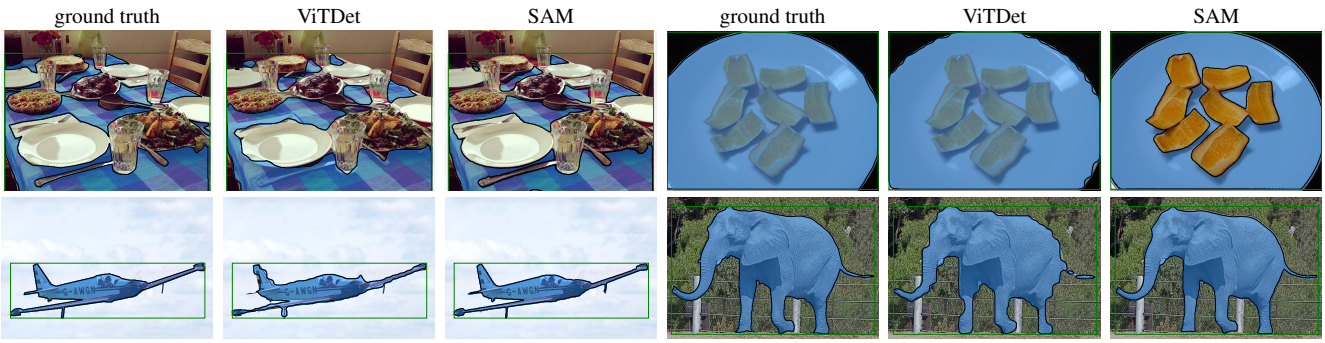


Figure 12: Zero-shot instance segmentation on LVIS v1. SAM produces higher quality masks than ViTDet. As a zero-shot model, SAM does not have the opportunity to learn specific training data biases; see top-right as an example where SAM makes a modal prediction, whereas the ground truth in LVIS is amodal given that mask annotations in LVIS have no holes.

	COCO [64]				LVIS v1 [43]			
method	AP	AP^S	AP^M	AP^L	AP	AP^S	AP^M	AP^L
ViTDet-H [60]	51.0	32.0	54.3	68.9	46.6	35.0	58.0	66.3
zero-shot transfer methods (segmentation module only):								
SAM	46.5	30.8	51.0	61.7	44.7	32.5	57.6	65.5

Table 7: Instance segmentation results. SAM is prompted with ViTDet boxes to do zero-shot segmentation. The fully-supervised ViTDet outperforms SAM, but the gap shrinks on the higher-quality LVIS masks. Interestingly, SAM outperforms ViTDet according to human ratings (see Fig. 13).

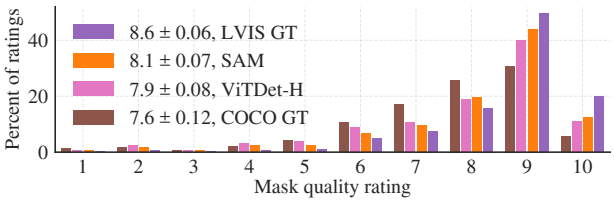


Figure 13: Mask quality rating distribution from our human study for ViTDet and SAM, both applied to LVIS ground truth boxes. We also report LVIS and COCO ground truth quality. The legend shows rating means and 95% confidence intervals. Despite its lower AP (Table 7), SAM has higher ratings than ViTDet, suggesting that ViTDet exploits biases in the COCO and LVIS training data.

Figure 14: Visualization of thresholding the similarities of mask embeddings from SAM's latent space. A query is indicated by the magenta box; top row shows matches at a low threshold, bottom row at a high threshold. The most similar mask embeddings in the same image can often be semantically similar to the query mask embedding, even though SAM is not trained with explicit semantic supervision.

E.5. Zero-Shot Text-to-Mask

Model and training. We use the largest publicly available CLIP model [80] (`ViT-L/14@336px`) to compute text and image embeddings, which we ℓ^2 normalize prior to use. To train SAM, we use masks from the first two stages of our data engine. Moreover, we discard all masks with an area smaller than 100^2 pixels. We train this model with large-scale jitter [39] for 120k iterations with batch size 128. All other training parameters follow our default settings.

Generating training prompts. To extract an input prompt we first expand the bounding box around each mask by a random factor from $1\times$ to $2\times$, square-crop the expanded box to maintain its aspect ratio, and resize it to 336×336 pixels. Before feeding the crop to the CLIP image encoder, with 50% probability we zero-out pixels outside the mask. To ensure the embedding focuses on the object, we use masked attention in the last layer to restrict attention from the output token to the image positions inside the mask. Finally, our prompt is the output token embedding. For training we supply the CLIP-based prompt first, followed by additional iterative point prompts to refine the prediction.

Inference. During inference we use the CLIP text encoder without any modifications to create a prompt for SAM. We rely on the fact that text and image embeddings are aligned by CLIP, which allows us to train without any explicit text supervision while using text-based prompts for inference.

E.6. Probing the Latent Space of SAM

Finally, we perform an initial investigation to qualitatively probe the latent space learned by SAM. In particular, we are interested in whether SAM is able to capture any semantics in its representation even though is not trained with explicit semantic supervision. To do so, we compute *mask embeddings* by extracting an image embedding from SAM from an image crop around a mask and its horizontally flipped version, multiplying the image embedding by the binary mask, and averaging over spatial locations. In Fig. 14, we show 3 examples of a query mask and similar masks (in the latent space) in the same image. We observe that the nearest neighbors for each query show some, albeit imperfect, shape and semantic similarity. Although these results are preliminary, they indicate that the representations from SAM may be useful for a variety of purposes, such as further data labeling, understanding the contents of datasets, or as features for downstream tasks.

E.7. Ablations

We perform several ablations on our 23 dataset suite with the single center point prompt protocol. Recall that a single point may be ambiguous and that ambiguity may not be represented in the ground truth, which contains only a single mask per point. Since SAM is operating in a zero-shot transfer setting there can be systematic biases between SAM's top-ranked mask *vs*. the masks resulting from data annotation guidelines. We therefore additionally report the best mask with respect to the ground truth ("oracle").

Fig. 15 (left) plots SAM's performance when trained on cumulative data from the data engine stages. We observe that each stage increases mIoU. When training with all three stages, the automatic masks vastly outnumber the manual and semi-automatic masks. To address this, we found that oversampling the manual and semi-automatic masks during training by 10× gave best results. This setup complicates training. We therefore tested a fourth setup that uses only the automatically generated masks. With this data, SAM performs only marginally lower than using all data (∼0.5 mIoU). Therefore, by default we use only the automatically generated masks to simplify the training setup.

In Fig. 15 (middle) we look at the impact of data volume. The full SA-1B contains 11M images, which we uniformly subsample to 1M and 0.1M for this ablation. At 0.1M images, we observe a large mIoU decline under all settings. However, with 1M images, about 10% of the full dataset, we observe results comparable to using the full dataset. This data regime, which still includes approximately 100M masks, may be a practical setting for many use cases.

Finally, Fig. 15 (right) shows results with ViT-B, ViT-L, and ViT-H image encoders. ViT-H improves substantially over ViT-B, but has only marginal gains over ViT-L. Further image encoder scaling does not appear fruitful at this time.

F. Human Study Experimental Design

Here we describe details of the human study used to evaluate mask quality in §6.1 and §E.4. The purpose of the human study is to address two limitations of using IoU to ground truth as a measure of predicted mask quality. The first limitation is that, for ambiguous inputs such as a single point, the model may be strongly penalized for returning a valid mask of a different object than the ground truth. The second limitation is that ground truth masks may include various biases, such as systematic errors in the edge quality or decisions to modally or amodally segment occluding objects. A model trained in-domain can learn these biases and obtain a higher IoU without necessarily producing better masks. Human review can obtain a measure of mask quality independent of an underlying ground truth mask in order to alleviate these issues.

Models. For single-point evaluation, we use RITM [90], single-output SAM, and SAM to test two hypotheses. First, we hypothesize that SAM produces visually higher quality masks than baseline interactive segmentation models when given a single point, even when metrics such as IoU with ground truth do not reveal this. Second, we hypothesize that SAM's ability to disambiguate masks improves mask quality for single point inputs, since single output SAM may return masks that average over ambiguous masks.

For instance segmentation experiments, we evaluate cascade ViTDet-H [60] and SAM in order to test the hypothesis that SAM produces visually higher quality masks, even if it obtains a lower AP due to the inability to learn specific annotation biases of the validation dataset.

Datasets. For single-point experiments, we select 7 datasets from our set of 23 datasets, since the full suite is too large for human review. We choose LVIS v0.5 [16], VISOR [27, 26], DRAM [23], IBD [16], NDD20 [98], OVIS [79], and iShape [109], which provide a diverse collection of images, including scene-level, ego-centric, drawn, overhead, underwater, and synthetic imagery. Additionally, this set includes datasets both where SAM outperforms RITM with IoU metrics and vice-versa. For instance segmentation experiments, we use the LVIS v1 validation set, allowing for direct comparison to ViTDet, which was trained on LVIS.

Methodology. We presented masks generated by the models to professional annotators and asked them to rate each mask using provided guidelines (see §H for the complete guidelines). Annotators were sourced from the same company that collected manually annotated masks for the data engine. An annotator was provided access to an image, the predicted mask of a single model, and the input to the model (either a single point or single box) and asked to judge the mask on three criterion: Does the mask correspond to a valid object? Does the mask have a clean boundary? and Does the mask correspond to the input? They then submit-

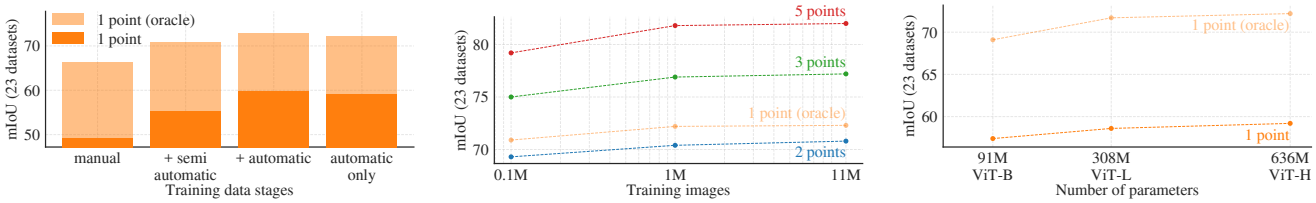


Figure 15: Ablation studies of our data engine stages, image encoder scaling, and training data scaling. (Left) Each data engine stage leads to improvements on our 23 dataset suite, and training with only the automatic data (our default) yields similar results to using data from all three stages. (Middle) SAM trained with ∼10% of SA-1B and full SA-1B is comparable. We train with all 11M images by default, but using 1M images is a reasonable practical setting. (Right) Scaling SAM's image encoder shows meaningful, yet saturating gains. Nevertheless, smaller image encoders may be preferred in certain settings.

ted a rating from 1-10 indicating the overall mask quality.

A score of 1 indicates a mask that corresponds to no object at all; a low score (2-4) indicates that the mask has huge errors, such including huge regions of other objects or having large areas of nonsensical boundaries; a middle score (5-6) indicates masks that are mostly sensible but still have significant semantic or boundary errors; a high score (7-9) indicates masks with only minor boundary errors; and a score of 10 is for masks with no visible errors. Annotators were provided with five different views, each designed to help identify different error types.

For single point experiments, 1000 masks per dataset were selected randomly from the same subsets used for benchmarking zero-shot interactive segmentation (see §E.1 for details on these subsets). The model input was the centermost point, calculated as the largest value of the distance transform from the edge of the mask. For instance segmentation experiments, 1000 masks were selected from the LVIS v1 validation set, and the model input was the LVIS ground truth box. In all experiments, masks with a size smaller than 24^2 pixels were excluded from sampling, to prevent showing raters a mask that was too small to judge accurately. For both memory and display reasons, large images were rescaled to have a max side-length of 2000 before predicting a mask. In all experiments, the same inputs were fed to each model to produce a predicted mask.

For comparison, the ground truth masks from each dataset were also submitted for rating. For single-point experiments, this gave 4000 total rating jobs per dataset (1000 masks each for RITM, SAM single-output, SAM, and ground truth); for instance segmentation experiments, it gave 3000 total jobs (ViTDet, SAM, and ground truth).

For each dataset, these jobs were inserted with random ordering into a queue from which 30 annotators drew jobs. In initial testing of the review study, we provided each job to five different annotators and found reasonable consistency in scores: the average standard deviation in score over the five annotators was 0.83. Additionally, the annotation company deployed quality assurance testers who spot checked a fraction of results for extreme departures from the guidelines. Thus for our experiments each job (*i.e.*, rating one mask in one image) was completed by only a single annotator. Average time spent per annotator per job was 90 seconds, longer than our initial target of 30 seconds, but still sufficiently fast to collect a large number of ratings on each of the 7 selected datasets.

	SAM > baseline		SAM > SAM single out.	
dataset	p-value	$\text{CI}_{99}(\Delta\mu)$	p-value	$\text{CI}_{99}(\Delta\mu)$
point input (RITM [90] baseline):				
LVIS v0.5 [43]	4e-69	(1.40, 1.84)	2e-11	(0.29, 0.64)
VISOR [27, 26]	7e-98	(1.81, 2.24)	7e-26	(0.58, 0.94)
DRAM [23]	1e-76	(1.54, 2.00)	2e-24	(0.62, 1.03)
IBD [16]	2e-57	(1.03, 1.39)	1e-15	(0.32, 0.62)
NDD20 [98]	2e-86	(1.88, 2.37)	5e-08	(0.19, 0.55)
OVIS [79]	2e-64	(1.38, 1.84)	3e-10	(0.27, 0.63)
iShape [109]	2e-88	(1.97, 2.47)	7e-23	(0.65, 1.10)
box input (ViTDet-H [60] baseline):				
LVIS v1 [43]	2e-05	(0.11, 0.42)	N/A	N/A

Table 8: Statistical tests showing significance that SAM has higher mask quality ratings than baseline and single-output SAM. P-values are calculated by paired t-test, while confidence intervals for the difference in mean scores are calculated by paired bootstrap on 10k samples. All p-values are significant, and all confidence intervals exclude zero.

Results. Fig. 16 shows histograms over ratings for each dataset in the single-point experiments. We run statistical tests for two hypotheses: (1) that SAM gets higher scores than the baseline model (RITM or ViTDet) and (2) that SAM gets higher scores than single-output SAM. P-values are calculated via a paired t-test on the means of the model scores, which we supplement with a paired bootstrap test on 10k samples to find the 99% confidence interval for the difference of means. Table 8 shows p-values and confidence intervals for these tests. All statistical tests are strongly significant, and all confidence intervals exclude zero.

For instance segmentation, Fig. 13 of the main text shows the histogram for ratings. To compare to COCO ground truth, we additionally include 794 ratings of COCO ground truth masks that were collected during our testing of

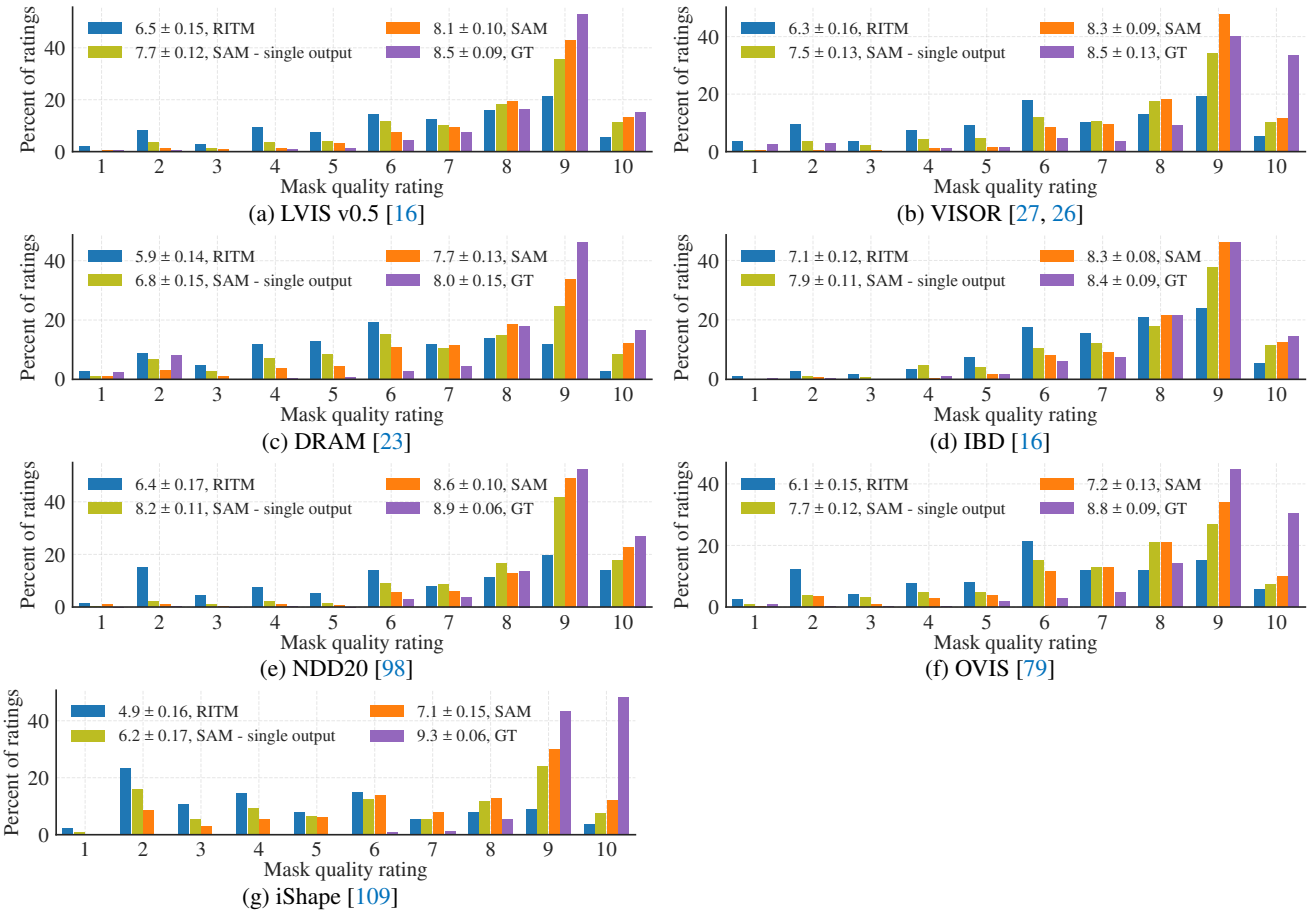


Figure 16: Mask quality rating distributions by dataset from our human evaluation study.

the human review process. These masks were presented to raters using an identical setup as the LVIS results. For fair comparison, results for LVIS in Fig. 13 were subsampled to the same 794 inputs for each model and ground truth. For Table 8, the full 1000 ratings are used to run statistical tests, which show that SAM's mask quality improvement over ViTDet is statistically significant.

G. Dataset, Annotation, and Model Cards

In §G.1 we provide a Dataset Card for SA-1B, following [38], in a list of questions and answers. Next, we provide a Data Annotation Card in §G.2 for the first two stages of our data engine described in §4, following CrowdWorkSheets [29], again as a list of questions and answers. We provide a Model Card following [73] in Table 9.

G.1. Dataset Card for SA-1B

Motivation

1. *For what purpose was the dataset created? Was there a specific task in mind? Was there a specific gap that needed to be filled? Please provide a description.* The contributions of our dataset to the vision community are fourfold: (1) We release a dataset of 11M images and 1.1B masks, by far the largest segmentation dataset to date. (2) The dataset we release is privacy protecting: we have blurred faces and license plates in all images. (3) The dataset is licensed under a broad set of terms of use which can be found at https://ai.facebook.com/datasets/segment-anything. (4) The data is more geographically diverse than its predecessors, and we hope it will bring the community one step closer to creating fairer and more equitable models.
2. *Who created the dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?* The dataset was created by the FAIR team of Meta AI. The underlying images were collected and licensed from a third party photo company.
3. *Who funded the creation of the dataset? If there is an associated grant, please provide the name of the grantor and the grant name and number.* Meta AI funded the creation of the dataset.
4. *Any other comments?* No.

Composition

1. *What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)? Are there multiple types of instances (e.g., movies, users, and ratings; people and interactions between them; nodes and edges)? Please provide a description.* All of the instances in the dataset are photos. The photos vary in subject matter; common themes of the photo include: locations, objects, scenes. All of the photos are distinct, however there are some sets of photos that were taken of the same subject matter.
2. *How many instances are there in total (of each type, if appropriate)?* There are 11 million images.

3. *Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set? If the dataset is a sample, then what is the larger set? Is the sample representative of the larger set (e.g., geographic coverage)? If so, please describe how this representativeness was validated/verified. If it is not representative of the larger set, please describe why not (e.g., to cover a more diverse range of instances, because instances were withheld or unavailable).* The dataset is composed of images licensed from a photo provider. The dataset contains all instances licensed. The images are photos, *i.e.* not artwork, although there are a few exceptions. The dataset includes all generated masks for each image in the dataset. We withheld ∼2k randomly selected images for testing purposes.

4. *What data does each instance consist of? "Raw" data (e.g., unprocessed text or images) or features? In either case, please provide a description.* Each instance in the dataset is an image. The images were processed to blur faces and license plates to protect the identities of those in the image.

5. *Is there a label or target associated with each instance? If so, please provide a description.* Each image is annotated with masks. There are no categories or text associated with the masks. The average image has ∼100 masks, and there are ∼1.1B masks in total.

6. *Is any information missing from individual instances? If so, please provide a description, explaining why this information is missing (e.g., because it was unavailable). This does not include intentionally removed information, but might include, e.g., redacted text.* Yes. Each image is accompanied by a short caption that describes the content and place of the photo in a free form text. Per our agreement with the photo provider we are not allowed to release these captions. However, we use them in our paper to analyze the geographical distribution of the dataset.

7. *Are relationships between individual instances made explicit (e.g., users' movie ratings, social network links)? If so, please describe how these relationships are made explicit.* No, there are no known relationships between instances in the dataset.

8. *Are there any errors, sources of noise, or redundancies in the dataset? If so, please provide a description. **Errors:*** The masks are generated by a segmentation model, so there may be errors or inconsistencies in the masks. ***Redundancies:*** While no two images are the same, there are instances of images of the same subject taken close together in time.

9. *Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)? If it links to or relies on external resources, a) are there guarantees that they will exist, and remain constant, over time; b) are there official archival versions of the complete dataset (i.e., including the external resources as they existed at the time the dataset was created); c) are there any restrictions (e.g., licenses, fees) associated with any of the external resources that might apply to a dataset consumer? Please provide descriptions of all external resources and any restrictions associated with them, as well as links or other access points, as appropriate.* The dataset is self-contained.

10. *Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor-patient confidentiality, data that includes the content of individuals' non-public communications)? If so, please provide a description.* No.

11. *Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety? If so, please describe why.* We have two safety measures to prevent objectionable content: (1) Photos are licensed from a photo provider and had to meet the terms of service of the photo provider. We requested that all objectionable content be filtered from the images we licensed. (2) If a user observes objectionable image(s) in the dataset, we invite them to report the image(s) at segment-anything@meta.com for removal. Despite the measures taken, we observe that a small portion of images contains scenes of protests or other gatherings that focus on a diverse spectrum of religious beliefs or political opinions that may be offensive. We were not able to produce a filtering strategy that removes all such images and rely on users to report this type of content.

12. *Does the dataset identify any subpopulations (e.g., by age, gender)? If so, please describe how these subpopulations are identified and provide a description of their respective distributions within the dataset.* The dataset does not identify any subpopulations of the people in the photos.

13. *Is it possible to identify individuals (i.e., one or more natural persons), either directly or indirectly (i.e., in combination with other data) from the dataset? If so, please describe how.* No. Images were subjected to a face blurring model to remove any personally identifiable information. If a user observes any anonymization issue, we invite them to report the issue and the image id(s) at segment-anything@meta.com.

14. *Does the dataset contain data that might be considered sensitive in any way (e.g., data that reveals race or ethnic origins, sexual orientations, religious beliefs, political opinions or union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)? If so, please provide a description.* The dataset contains scenes of protests, or other gatherings that may suggest religious beliefs, political opinions or union memberships. However, the faces of all people in the dataset have been anonymized via facial blurring, so it is not possible to identify any person in the dataset.

15. *Any other comments?* No.

Collection Process

1. *How was the data associated with each instance acquired? Was the data directly observable (e.g., raw text, movie ratings), reported by subjects (e.g., survey responses), or indirectly inferred/derived from other data (e.g., part-of-speech tags, model-based guesses for age or language)? If the data was reported by subjects or indirectly inferred/derived from other data, was the data validated/verified? If so, please describe how.* The released masks associated with each image were automatically inferred by our segmentation model, SAM. The masks that were collected using model-assisted manual annotation will not be released. Quality was validated as described in §5.

2. *What mechanisms or procedures were used to collect the data (e.g., hardware apparatuses or sensors, manual human curation, software programs, software APIs)? How were these mechanisms or procedures validated?* The images in the dataset are licensed from an image provider. They are all photos taken by photographers with different cameras.

3. *If the dataset is a sample from a larger set, what was the sampling strategy (e.g., deterministic, probabilistic with specific sampling probabilities)?* We withheld ∼2k randomly selected images for testing purposes. The rest of the licensed images are included in the dataset.

4. *Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)?* The released masks were automatically inferred by SAM. For details on our model-assisted manual annotation process see our Data Annotation Card in §G.2. Note these masks will not be released.

5. *Over what timeframe was the data collected? Does this timeframe match the creation timeframe of the data associated with the instances (e.g., recent crawl of old news articles)? If not, please describe the timeframe in which the data associated with the instances was created.* The licensed photos vary in their date taken over a wide range of years up to 2022.

6. *Were any ethical review processes conducted (e.g., by an institutional review board)? If so, please provide a description of these review processes, including the outcomes, as well as a link or other access point to any supporting documentation. If the dataset does not relate to people, you may skip the remaining questions in this section.* We underwent an internal privacy review to evaluate and determine how to mitigate any potential risks with respect to the privacy of people in the photos. Blurring faces and license plates protects the privacy of the people in the photos.

7. *Did you collect the data from the individuals in question directly, or obtain it via third parties or other sources (e.g., websites)?* We licensed the data from a third party photo provider.

8. *Were the individuals in question notified about the data collection? If so, please describe (or show with screenshots or other information) how notice was provided, and provide a link or other access point to, or otherwise reproduce, the exact language of the notification itself.* The images are licensed from a third party who provided appropriate representations regarding the collection of any notices and consents as required from individuals. In addition, all identifiable information (*e.g.* faces, license plates) was blurred. Under the terms of the dataset license it is prohibited to attempt to identify or associate an image with a particular individual.

9. *Did the individuals in question consent to the collection and use of their data? If so, please describe (or show with screenshots or other information) how consent was requested and provided, and provide a link or other access point to, or otherwise reproduce, the exact language to which the individuals consented.* The images are licensed from a third party who provided appropriate representations regarding the collection of any notices and consents as required from individuals. In addition, all identifiable information (*e.g.* faces, license plates) was blurred from all images. For avoidance of doubt, under the terms of the dataset license it is prohibited to attempt to identify or associate an image with a particular individual.

10. *If consent was obtained, were the consenting individuals provided with a mechanism to revoke their consent in the future or for certain uses? If so, please provide a description, as well as a link or other access point to the mechanism (if appropriate).* We invite users to report at segment-anything@meta.com for image(s) removal.
11. *Has an analysis of the potential impact of the dataset and its use on data subjects (*e.g.*, a data protection impact analysis) been conducted? If so, please provide a description of this analysis, including the outcomes, as well as a link or other access point to any supporting documentation.* To eliminate any potential impact on people whose photos are included in the dataset, identifiable information (faces, license plates) has been blurred.
12. *Any other comments?* No.

Preprocessing / Cleaning / Labeling

1. *Was any preprocessing / cleaning / labeling of the data done (*e.g.*, discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)? If so, please provide a description. If not, you may skip the remaining questions in this section.* We resized the high-resolution licensed images such that the shorter side is 1500 pixels and only processed the images to remove any identifiable and personal information from the photos (faces, license plates).
2. *Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (*e.g.*, to support unanticipated future uses)? If so, please provide a link or other access point to the "raw" data.* No, as we removed the data for safety reasons and to respect privacy, we do not release the unaltered photos.
3. *Is the software that was used to preprocess/clean/label the data available? If so, please provide a link or other access point.* We used the RetinaFace [86, 87] model (https://github.com/serengil/retinaface) to detect faces. The model used to blur license plates has not been made public.

Uses

1. *Has the dataset been used for any tasks already? If so, please provide a description.* The dataset was used to train our segmentation model, SAM.
2. *Is there a repository that links to any or all papers or systems that use the dataset? If so, please provide a link or other access point.* No. However, all users of the dataset must cite it, so its use is trackable via citation explorers.
3. *What (other) tasks could the dataset be used for?* We intend the dataset to be a large-scale segmentation dataset. However, we invite the research community to gather additional annotations for the dataset.
4. *Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses? For example, is there anything that a dataset consumer might need to know to avoid uses that could result in unfair treatment of individuals or groups (*e.g.*, stereotyping, quality of service issues) or other risks or harms (*e.g.*, legal risks, financial harms)? If so, please provide a description. Is there anything a dataset consumer could do to mitigate these risks or harms?* We have an analysis of the approximate geographic and income level coverage of our dataset in §**??**. While we believe our dataset to be more representative than most of the publicly existing datasets at this time, we acknowledge that we do not have parity across all groups, and we encourage users to be mindful of potential biases their models have learned using this dataset.
5. *Are there tasks for which the dataset should not be used? If so, please provide a description.* Full terms of use for the dataset including prohibited use cases can be found at https://ai.facebook.com/datasets/segment-anything.
6. *Any other comments?* No.

Distribution

1. *Will the dataset be distributed to third parties outside of the entity (*e.g.*, company, institution, organization) on behalf of which the dataset was created? If so, please provide a description.* The dataset will be available for the research community.
2. *How will the dataset will be distributed (*e.g.*, tarball on website, API, GitHub)? Does the dataset have a digital object identifier (DOI)?* The dataset is available at https://ai.facebook.com/datasets/segment-anything.
3. *When will the dataset be distributed?* The dataset will be released in 2023.
4. *Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)? If so, please describe this license and/or ToU, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms or ToU, as well as any fees associated with these restrictions.* Yes. The license agreement and terms of use for the dataset can be found at https://ai.facebook.com/datasets/segment-anything. Users must agree to the terms of use before downloading or using the dataset.
5. *Have any third parties imposed IP-based or other restrictions on the data associated with the instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms, as well as any fees associated with these restrictions.* Full terms of use and restrictions on use of the SA-1B dataset can be found at https://ai.facebook.com/datasets/segment-anything.
6. *Do any export controls or other regulatory restrictions apply to the dataset or to individual instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any supporting documentation.* The license and restrictions on use of the SA-1B dataset can be found at https://ai.facebook.com/datasets/segment-anything.
7. *Any other comments?* No.

Maintenance

1. *Who will be supporting/hosting/maintaining the dataset?* The dataset will be hosted at https://ai.facebook.com/datasets/segment-anything and maintained by Meta AI.
2. *How can the owner/curator/manager of the dataset be contacted (*e.g.*, email address)?* Please email segment-anything@meta.com.
3. *Is there an erratum? If so, please provide a link or other access point.* No.
4. *Will the dataset be updated (*e.g.*, to correct labeling errors, add new instances, delete instances)? If so, please describe how often, by whom, and how updates will be communicated to dataset consumers (*e.g.*, mailing list, GitHub)?* To aid reproducibility of research using SA-1B, the only updates will be to remove reported images.
5. *If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (*e.g.*, were the individuals in question told that their data would be retained for a fixed period of time and then deleted)? If so, please describe these limits and explain how they will be enforced.* There are no limits on data retention. We took measures to remove personally identifiable information from any images of people. Users may report content for potential removal here: segment-anything@meta.com.
6. *Will older versions of the dataset continue to be supported/hosted/maintained? If so, please describe how. If not, please describe how its obsolescence will be communicated to dataset consumers.* No, as the only updates will be to remove potentially harmful content, we will not keep older versions with the content.
7. *If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so? If so, please provide a description. Will these contributions be validated/verified? If so, please describe how. If not, why not? Is there a process for communicating/distributing these contributions to dataset consumers? If so, please provide a description.* We encourage users to gather further annotations for SA-1B. Any users who generate annotations will be liable for hosting and distributing their annotations.
8. *Any other comments?* No.

G.2. Data Annotation Card

Task Formulation

1. *At a high level, what are the subjective aspects of your task?* Segmenting objects present in an image is inherently a subjective task. For instance, one annotator may segment two boots as one mask, whereas another may segment each boot separately. Depending on annotators's skills, the quality of the mask and the number of masks per image are different between annotators. Despite these subjective aspects of the task, we believed efficient annotation was possible as the data was annotated in a per-mask fashion with the main focus on the diversity of the data rather than completeness.
2. *What assumptions do you make about annotators?* Our annotators worked full time on our annotation task with very small attrition rate. This made it possible to train the annotators providing feedback and answering their questions on a regular basis. Specifically: (1) By giving a clear understanding of the goals of this work and providing clear guidelines, including visuals and video recordings of the tasks, annotators had enough context to understand and perform the tasks reasonably. (2) Sharing objectives and key results and meeting weekly with annotators increased the likelihood that annotators improved annotation quality and quantity over time.

3. *How did you choose the specific wording of your task instructions? What steps, if any, were taken to verify the clarity of task instructions and wording for annotators?* As our task was annotating images, the annotation guidelines included visual examples. Our research team completed 30 annotation tasks to identify any obvious challenges using the annotation tool, collectively decide how to handle complex cases, and refine the guidelines. The research team met with the annotators weekly for feedback sessions. Videos of the research team performing the task were shared live with the annotators, followed by Q&A sessions. Annotators were able to give feedback on unclear aspects, both during the feedback session and asynchronously.
4. *What, if any, risks did your task pose for annotators and were they informed of the risks prior to engagement with the task?* No identified risks. Images were filtered for objectionable content prior to the annotation phase.
5. *What are the precise instructions that were provided to annotators?* We provide only high-level instructions: Given an image, we aim at segmenting every possible object. Annotators generate a mask for every potential object they can identify. An object can be segmented using our interactive segmentation tool either by using corrective foreground/background clicks to add/remove parts of the mask or by drawing a bounding box around the object. Masks can be refined using pixel-precise tools.

Selecting Annotations

1. *Are there certain perspectives that should be privileged? If so, how did you seek these perspectives out?* We chose to work with annotators that have worked on other vision annotation tasks before.
2. *Are there certain perspectives that would be harmful to include? If so, how did you screen these perspectives out?* No.
3. *Were sociodemographic characteristics used to select annotators for your task? If so, please detail the process.* No.
4. *If you have any aggregated socio-demographic statistics about your annotator pool, please describe. Do you have reason to believe that sociodemographic characteristics of annotators may have impacted how they annotated the data? Why or why not?* We worked with 130 annotators. The annotators were all based in Kenya. We do not believe sociodemographic characteristics of annotators meaningfully impacted the annotated data.
5. *Consider the intended context of use of the dataset and the individuals and communities that may be impacted by a model trained on this dataset. Are these communities represented in your annotator pool?* The Segment Anything 1B (SA-1B) dataset is to be used for research purposes only. The SA-1B dataset is one of the most geographically diverse segmentation dataset, as discussed in §**??**. In addition, we analyze the responsible AI axes of a model trained on the dataset in §**??**.

Platform and Infrastructure Choices

1. *What annotation platform did you utilize? At a high level, what considerations informed your decision to choose this platform? Did the chosen platform sufficiently meet the requirements you outlined for annotator pools? Are any aspects not covered?* We used a proprietary annotation platform.
2. *What, if any, communication channels did your chosen platform offer to facilitate communication with annotators? How did this channel of communication influence the annotation process and/or resulting annotations?* We manually reviewed annotations and shared feedback with the annotators on a weekly basis. We communicated common mistakes or inconsistencies and the corresponding corrections. In addition, the annotators were given feedback for improvements daily by the annotation QA team. Outside the weekly feedback sessions, annotators had access to a spreadsheet and chat group to facilitate communication with the research team. This process greatly improved the average speed and quality of the annotations.
3. *How much were annotators compensated? Did you consider any particular pay standards, when determining their compensation? If so, please describe.* Annotators were compensated with an hourly wage set by the vendor. The vendor is a Certified B Corporation.

Dataset Analysis and Evaluation

1. *How do you define the quality of annotations in your context, and how did you assess the quality in the dataset you constructed?* Annotators were first placed into training. They followed a 1-day training session led by the vendor and then were asked to annotate a large number of examples from a training queue. Annotators graduated from training to production after the vendor QA team, in collaboration with the research team, manually spot-checked the annotator's masks to ensure quality. On average, annotators spent one week in training before graduating. Production quality assessment followed a similar process: the vendor QA team and the research team manually reviewed the annotations weekly, sharing feedback weekly.
2. *Have you conducted any analysis on disagreement patterns? If so, what analyses did you use and what were the major findings? Did you analyze potential sources of disagreement?* We pointed out common mistakes during weekly meetings with the annotators.
3. *How do the individual annotator responses relate to the final labels released in the dataset?* The annotations were only used to train early versions of the SAM model and we do not currently plan to release them.

Dataset Release and Maintenance

1. *Do you have reason to believe the annotations in this dataset may change over time? Do you plan to update your dataset?* No, except to remove objectionable images.
2. *Are there any conditions or definitions that, if changed, could impact the utility of your dataset?* We do not believe so.
3. *Will you attempt to track, impose limitations on, or otherwise influence how your dataset is used? If so, how?* The SA-1B dataset will be released under a license agreement allowing use for certain research purposes and protections for researchers. Researchers must agree to the terms of the license agreement to access the dataset.
4. *Were annotators informed about how the data is externalized? If changes to the dataset are made, will they be informed?* No, we do not plan to release the manual annotations at the moment.
5. *Is there a process by which annotators can later choose to withdraw their data from the dataset? If so, please detail.* No.

H. Annotation Guidelines

We provide the complete guidelines given to annotations for the human review of mask quality in Fig. 17 and Fig. 18.

References

[1] Edward H Adelson. On seeing stuff: the perception of materials by humans and machines. *Human vision and electronic imaging VI*, 2001. 5

[2] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. What is an object? *CVPR*, 2010. 4, 19

[3] Pablo Arbeláez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *TPAMI*, 2010. 4, 19, 28

[4] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv:1607.06450*, 2016. 13

[5] Hangbo Bao, Li Dong, and Furu Wei. BEiT: BERT pre-training of image transformers. *arXiv:2106.08254*, 2021. 15

[6] Dina Bashkirova, Mohamed Abdelfattah, Ziliang Zhu, James Akl, Fadi Alladkani, Ping Hu, Vitaly Ablavsky, Berk Calli, Sarah Adel Bargal, and Kate Saenko. ZeroWaste dataset: Towards deformable object segmentation in cluttered scenes. *CVPR*, 2022. 8, 18

[7] Stuart Berg, Dominik Kutra, Thorben Kroeger, Christoph N. Straehle, Bernhard X. Kausler, Carsten Haubold, Martin Schiegg, Janez Ales, Thorsten Beier, Markus Rudy, Kemal Eren, Jaime I. Cervantes, Buote Xu, Fynn Beuttenmueller, Adrian Wolny, Chong Zhang, Ullrich Koethe, Fred A. Hamprecht, and Anna Kreshuk. ilastik: interactive machine learning for (bio)image analysis. *Nature Methods*, 2019. 9

[8] 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:2108.07258*, 2021. 1, 9

[9] Gustav Bredell, Christine Tanner, and Ender Konukoglu. Iterative interaction training for segmentation editing networks. *MICCAI*, 2018. 15

Model Overview

Name	SAM or Segment Anything Model
Version	1.0
Date	2023
Organization	The FAIR team of Meta AI
Mode type	Promptable segmentation model
Architecture	See §3
Repository	https://github.com/facebookresearch/segment-anything
Citation	https://research.facebook.com/publications/segment-anything
License	Apache 2.0

Intended Use

Primary intended uses	SAM is intended to be used for any prompt-based segmentation task. We explored its use in *segmenting objects from a point* (§6.1), *edge detection* (§**??**), *segmenting all objects* (§**??**), and *segmenting detected objects* (§**??**). We explored how SAM can integrate with other vision models to *segment objects from text* (§6.2).
Primary intended users	SAM was primarily developed for research. The license for SAM can be found at https://github.com/facebookresearch/segment-anything.
Out-of-scope use cases	See terms of use for SAM found at https://github.com/facebookresearch/segment-anything. See *Use Cases* under *Ethical Considerations*.
Caveats and recommendations	SAM has impressive zero-shot performance across a wide range of tasks. We note, however, that in the zero-shot setting there may be multiple valid ground truth masks for a given input. We recommend users take this into consideration when using SAM for zero-shot segmentation. SAM can miss fine structures and can hallucinate small disconnected components. See §7 for a discussion of limitations.

Relevant Factors

Groups	SAM was designed to segment any object. This includes *stuff* and *things*.
Instrumentation and environment	We benchmarked SAM on a diverse set of datasets and found that SAM can handle a variety of visual data including *simulations, paintings, underwater images, microscopy images, driving data, stereo images, fish-eye images*. See §E.1 and Table 4 for information on the benchmarks used.

Metrics

Model performance measures	We evaluated SAM on a variety of metrics based on the downstream task in our experiments. • *mIoU*: We used the mean intersection-over-union after a given number of prompts to evaluate the segmentation quality of a mask when prompted with points. • *Human evaluation*: We performed a human study (detailed in §F) to evaluate the real world performance of SAM. We compared the masks generated by SAM to a baseline state-of-the-art interactive segmentation model, RITM [90], using a perceptual quality scale from 1 to 10. • *AP*: We used average precision to evaluate instance segmentation for a given box and edge detection. • *AR@1000*: We used average recall to evaluate object proposal generation. • *ODS, OIS, AP, R50*: We used the standard edge detection evaluation metrics from BSDS500 [70, 3].

Evaluation Data

Data sources	See §E.1.

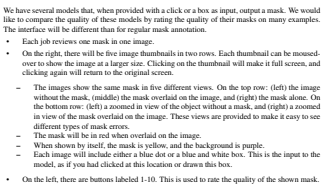
Training Data

Data source	See Data Card in §G.1.

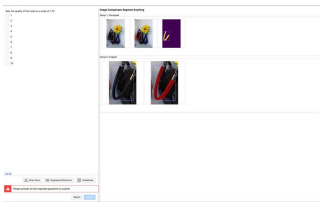
Ethical Considerations

Data	We trained SAM on licensed images. The images were filtered for objectionable content by the provider, but we acknowledge the possibility of false negatives. We performed a geographic analysis of the SA-1B dataset in §**??**. While SA-1B is more geographically diverse than many of its predecessors, we acknowledge that some geographic regions and economic groups are underrepresented.
Cost and impact of compute	SAM was trained on 256 A100 GPUS for 68 hours. We acknowledge the environmental impact and cost of training large scale models. The environmental impact of training the released SAM model is approximately 6963 kWh resulting in an estimated 2.8 metric tons of carbon dioxide given the specific data center used, using the calculation described in [75] and the ML CO_2 Impact calculator [59]. This is equivalent to $\sim$7k miles driven by the average gasoline-powered passenger vehicle in the US [99]. We released the SAM models to both reduce the need for retraining and lower the barrier to entry for large scale vision research.
Risks and harms	We evaluated SAM for fairness in §**??**. Downstream use cases of SAM will create their own potential for biases and fairness concerns. As such we recommend users run their own fairness evaluation when using SAM for their specific use case.
Use cases	We implore users to use their best judgement for downstream use of the model.

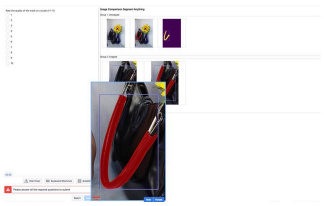
Table 9: Model Card for SAM, following the procedure detailed in [73].



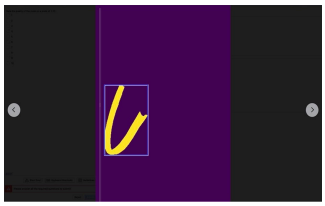
Objective and Setup



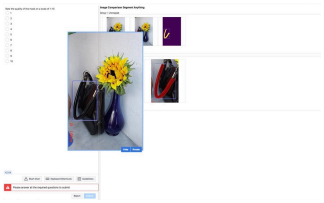
Example interface page. There will be five images on the right and a question box on the left.



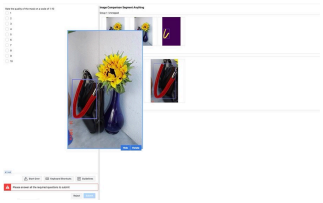
Mouse over an image to show the full image.



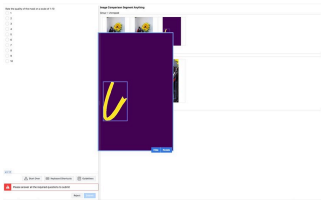
Click on an image to make it full screen. The arrows will cycle between images. Click again to return to previous view.



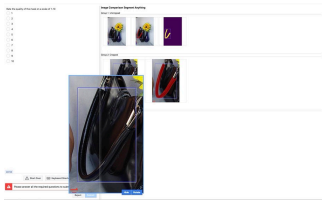
The first image on the top row shows the image without a mask. A blue point will be on the object of interest, or a blue and white box will surround it.



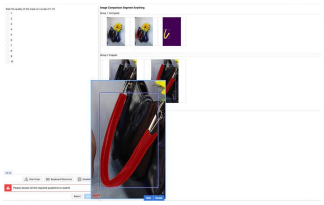
The second image on the top row shows the mask for the object in red.



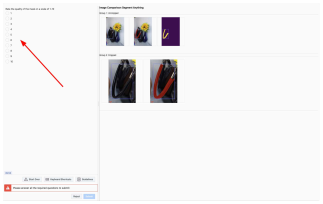
The third image on the top row shows the mask only. The mask is in yellow and the background is purple.



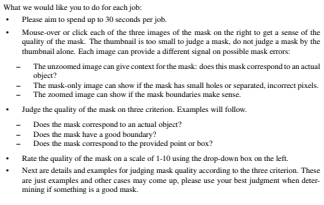
The first image on the bottom row shows a zoomed in view of the object without a mask.



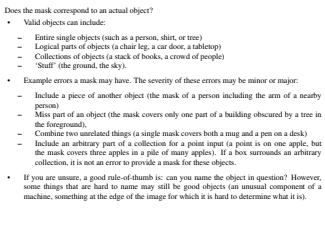
The second image on the bottom row shows a zoomed in view of the object with a mask. The mask is in red.



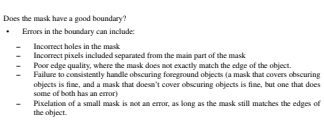
On the left are buttons to rate the mask quality, with selections 1-10.



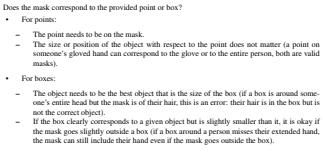
Task



Judging Mask Quality (1 of 3)



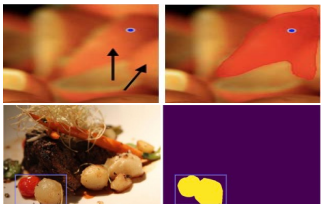
Judging Mask Quality (2 of 3)



Judging Mask Quality (3 of 3)

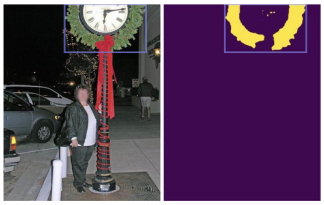
Example error of 'Include a piece of another object': The elephant mask contains a piece of another nearby elephant.

Example error of 'Missing a part of an object': the mask is missing a disconnected part of the object: the back half of the zebra, and the right portion of the plate.



Example error of 'Include an arbitrary part of a collection': In top top image, the point is on one orange rind, but the mask covers two orange rinds. This is a mask error: the mask covers an arbitrary number of objects in the collection, and should either cover one orange rind or all of them. In the bottom image, the box is around both vegetables. Since this is the best match to the box, this is not a mask error.

Example error for 'Incorrect holes in the mask': This mask has holes in the upper left and on the left sides (black arrows). These holes are much easier to see on the 'mask only' image.



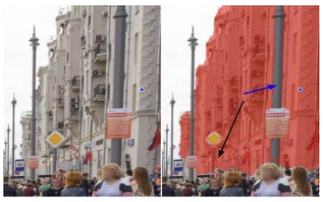
Example error for 'Incorrect pixels included separated from the main part of the mask': The 'mask only' view reveals a few stray incorrect pixels on the clock face.

Example error for 'Poor edge quality': The mask has poor edge quality, both along the edge of the umbrella, as well as along the thin pole.

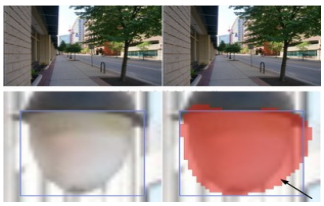
Figure 17: Here we provide the complete guidelines given to annotations for the human review of mask quality. Some images been edited slightly and faces have been blurred to enable release. Best viewed with zoom (part 1 of 2).

[10] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models

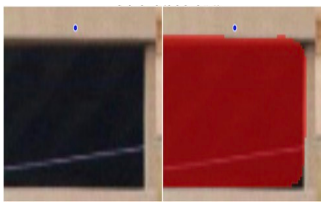
Example for ‘Combine two unrelated things’: The point indicates the lizard, but the mask covers both the lizard and a bird. This is a mask error.



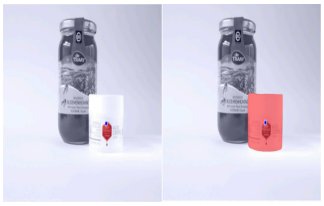
Example error for ‘Failure to consistently handle obscuring foreground objects’: The pole on the right (blue arrow) is excluded from the mask, while the pole on the left is included in the object (black arrow). The mask should either include or exclude both of these.



Example of ‘Pixelation of a small mask’: this mask has an imperfect boundary, since it extends beyond the object at the black arrow. However, the ‘blocky’ pattern of the mask is not an error, since, when zoomed in this much, the image is also blocky the same way.

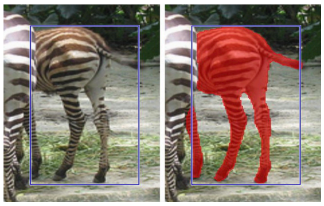


Example error for consistency with the provided point: The mask does not agree with the blue point, so this is a mask error.



Example for consistency with the provided point: For this input point, but the logo (left) and the container (right) are valid objects, since the blue point lies on both of them. Neither mask has a mask error.

Example for consistency with a box: The box surrounds the bowl of oranges, but the mask is only of a single orange. This is a mask error.



Example for consistency with a box: The box’s shape fits the zebra. Even though the mask extends slightly outside the box to include the zebra’s left leg, this is not an error.

Overall mask quality is subjective, each of the above errors may hurt mask quality only a little or a lot, depending on how large the error is. Please use your best judgment when choosing mask scores, and try to stay consistent from mask-to-mask. Here are some general guidelines for what different scores should correspond to:

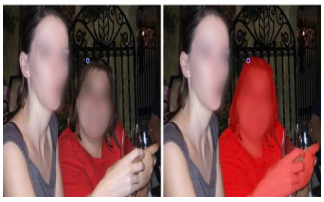
- A score of 1: It is not possible to tell what object this mask corresponds to. This includes the case that there is no mask visible at all.
- A low score (2-4): The object is mostly identifiable, but the mask quality is extremely poor (*e.g.* large regions of the mask cover other objects; large regions of the object missing; extremely splotchy mask boundaries that cut through the middle of the object).
- A mid score (5-6): The object is identifiable and the boundary is mostly correct, but there are major errors (missing a significant disconnected part of the object; containing a significant part of another object; very poor boundary quality in one area of the object but not the entire object).
- A high score (7-9): The object is identifiable and errors are small and rare (missing a small, heavily obscured disconnected component, having small regions where the mask boundary does not quite match the object boundary).
- A score of 10: The mask is pixel-perfect; it has no identifiable errors at all.

Mask Scoring

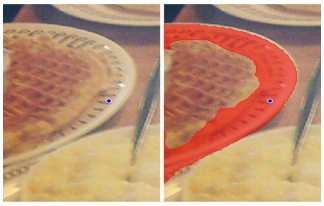
Example of a mask with a score of 1: It is not clear what object this mask corresponds to.

Example of a mask with a low score (2-4): The main object is identifiable, but the mask includes a large, incorrect portion of another object.

Example of a mask with a low score (2-4): The main object is identifiable, but a large, random part of the object is missing.



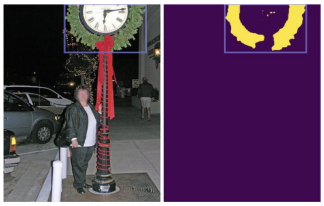
Example of a mask with a low-to-medium score (4-5): The object is identifiable and the edges are all correct, but the mask incorrectly includes the hand of the person on the left.



Example of a mask with a medium score (5-6): The mask clearly corresponds to the plate, but the boundary with the waffle is quite poor.

Example of a mask with a medium score (5-6): the object is easy to identify, and most of the edges make sense. However, there is a significant disconnected part (their arm inside the frame) that is mostly missing, as well as splotchy pixels in this region.

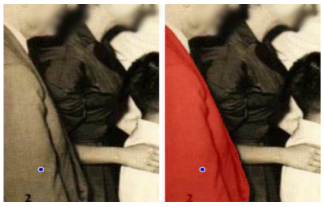
Example of a mask with a medium-to-high score (6-8): the mask has two small-ish regions of poor boundary, at the top of the mask and on the bottom right.



Example of a mask with a medium-to-high score (6-8): The wreath is a valid object that is the size of the box (the entire wreath + clock would also be a valid object). However, there are incorrect stray mask pixels on the clock.

Example of a mask with a high score (7-9): The boundary of the horse is almost entirely correct, except for the right side of its back leg. The mask consistently includes all of the equipment that horse is wearing, and has logical boundaries.

Example of a mask with a very high score (∼9): There are only minor errors around the edge of the mask. The blocky ‘pixelation’ is not an error, since the image is also blocky at this scale.



Example of a mask with a very high score (9-10): the mask has only very minor errors in the edge on the bottom right.

Example of a mask with a very high score (9-10): There are only minor errors around the edge of the mask.

Figure 18: Here we provide the complete guidelines given to annotations for the human review of mask quality. Some images been edited slightly and faces have been blurred to enable release. Best viewed with zoom (part 2 of 2).

are few-shot learners. *NeurIPS*, 2020. 1, 4

[11] Juan C. Caicedo, Allen Goodman, Kyle W. Karhohs, Beth A. Cimini, Jeanelle Ackerman, Marzieh Haghighi, CherKeng Heng, Tim Becker, Minh Doan, Claire McQuin, Mohammad Rohban, Shan-

tanu Singh, and Anne E. Carpenter. Nucleus segmentation across imaging experiments: the 2018 data science bowl. *Nature Methods*, 2019. 8, 17, 18

[12] John Canny. A computational approach to edge detection. *TPAMI*, 1986. 19

[13] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with Transformers. *ECCV*, 2020. 5, 13, 14

[14] Guillaume Charpiat, Matthias Hofmann, and Bernhard Schölkopf. Automatic image colorization via multimodal predictions. *ECCV*, 2008. 5, 14

[15] Neelima Chavali, Harsh Agrawal, Aroma Mahendru, and Dhruv Batra. Object-proposal evaluation protocol is' gameable'. *CVPR*, 2016. 19, 20

[16] Jiazhou Chen, Yanghui Xu, Shufang Lu, Ronghua Liang, and Liangliang Nan. 3D instance segmentation of MVS buildings. *IEEE Transactions on Geoscience and Remote Sensing*, 2022. 8, 17, 18, 22, 23, 24

[17] Xi Chen, Zhiyan Zhao, Yilei Zhang, Manni Duan, Donglian Qi, and Hengshuang Zhao. FocalClick: towards practical interactive image segmentation. *CVPR*, 2022. 7, 8, 9, 17, 19

[18] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. *CVPR*, 2022. 4

[19] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *NeurIPS*, 2021. 5, 13, 14

[20] 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:2204.02311*, 2022. 1

[21] Luca Ciampi, Carlos Santiago, Joao Costeira, Claudio Gennaro, and Giuseppe Amato. Domain adaptation for traffic density estimation. *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2021. 8, 18

[22] Luca Ciampi, Carlos Santiago, Joao Costeira, Claudio Gennaro, and Giuseppe Amato. Night and day instance segmented park (NDIS-Park) dataset: a collection of images taken by day and by night for vehicle detection, segmentation and counting in parking areas. *Zenodo*, 2022. 8, 18

[23] Nadav Cohen, Yael Newman, and Ariel Shamir. Semantic segmentation in art paintings. *Computer Graphics Forum*, 2022. 8, 17, 18, 22, 23, 24

[24] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes dataset for semantic urban scene understanding. *CVPR*, 2016. 8, 17, 18

[25] Bruno da Silva, George Konidaris, and Andrew Barto. Learning parameterized skills. *ICML*, 2012. 4

[26] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *IJCV*, 2022. 8, 18, 22, 23, 24

[27] Ahmad Darkhalil, Dandan Shan, Bin Zhu, Jian Ma, Amlan Kar, Richard Higgins, Sanja Fidler, David Fouhey, and Dima Damen. EPIC-KITCHENS VISOR benchmark: Video segmentations and object relations. *NeurIPS*, 2022. 8, 17, 18, 22, 23, 24

[28] Terrance De Vries, Ishan Misra, Changhan Wang, and Laurens Van der Maaten. Does object recognition work for everyone? *CVPR workshops*, 2019. 16

[29] Mark Díaz, Ian Kivlichan, Rachel Rosen, Dylan Baker, Razvan Amironesei, Vinodkumar Prabhakaran, and Emily Denton. CrowdWorkSheets: Accounting for individual and collective identities underlying crowdsourced dataset annotation. *ACM Conference on Fairness, Accountability, and Transparency*, 2022. 24

[30] Henghui Ding, Scott Cohen, Brian Price, and Xudong Jiang. PhraseClick: toward achieving flexible interactive segmentation by phrase and click. *ECCV*, 2020. 9

[31] Piotr Dollár and C Lawrence Zitnick. Fast edge detection using structured forests. *TPAMI*, 2014. 19

[32] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 5, 7, 13

[33] Alireza Fathi, Xiaofeng Ren, and James M. Rehg. Learning to recognize objects in egocentric activities. *CVPR*, 2011. 8, 17, 18

[34] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 2004. 19

[35] Thomas B. Fitzpatrick. The validity and practicality of sun-reactive skin types i through vi. *Archives of Dermatology*, 1988. 17

[36] Marco Forte, Brian Price, Scott Cohen, Ning Xu, and François Pitié. Getting to 99% accuracy in interactive segmentation. *arXiv:2003.07932*, 2020. 5, 14, 15

[37] Jean-Michel Fortin, Olivier Gamache, Vincent Grondin, François Pomerleau, and Philippe Giguère. Instance segmentation for autonomous log grasping in forestry operations. *IROS*, 2022. 8, 18

[38] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 2021. 24

[39] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. *CVPR*, 2021. 13, 15, 21

[40] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *CVPR*, 2014. 19

[41] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: Training ImageNet in 1 hour. *arXiv:1706.02677*, 2017. 15

[42] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebreselasie, Cristina Gonzalez, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jachym Kolar, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbelaez, David Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4D: Around the World in 3,000 Hours of Egocentric Video. *CVPR*, 2022. 18

[43] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. *CVPR*, 2019. 2, 6, 7, 8, 17, 18, 19, 21, 23

[44] Abner Guzman-Rivera, Dhruv Batra, and Pushmeet Kohli. Multiple choice learning: Learning to produce multiple structured outputs. *NeurIPS*, 2012. 5, 14

[45] Timm Haucke, Hjalmar S. Kühl, and Volker Steinhage. SOCRATES: Introducing depth in visual wildlife monitoring using stereo vision. *Sensors*, 2022. 8, 18

[46] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *CVPR*, 2022. 5, 7, 9, 13, 15

[47] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CVPR*, 2016. 14

[48] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv:1606.08415*, 2016. 13

[49] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv:2203.15556*, 2022. 1

[50] Jungseok Hong, Michael Fulton, and Junaed Sattar. TrashCan: A semantically-segmented dataset towards visual detection of marine debris. *arXiv:2007.08097*, 2020. 8, 17, 18

[51] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. *ECCV*, 2016. 15

[52] Jitesh Jain, Jiachen Li, MangTik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. Oneformer: One transformer to rule universal image segmentation. *arXiv:2211.06220*, 2022. 4

[53] 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. *ICML*, 2021. 1

[54] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv:2001.08361*, 2020. 1

[55] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *IJCV*, 1988. 4

[56] Dahun Kim, Tsung-Yi Lin, Anelia Angelova, In So Kweon, and Weicheng Kuo. Learning open-world object proposals without learning to classify. *IEEE Robotics and Automation Letters*, 2022. 19

[57] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. *CVPR*, 2019. 4

[58] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 2, 6, 7, 16

[59] Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *arXiv:1910.09700*, 2019. 28

[60] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. *ECCV*, 2022. 5, 13, 19, 21, 22, 23

[61] Yin Li, Zhefan Ye, and James M. Rehg. Delving into egocentric actions. *CVPR*, 2015. 8, 18

[62] Zhuwen Li, Qifeng Chen, and Vladlen Koltun. Interactive image segmentation with latent diversity. *CVPR*, 2018. 5, 14, 17

[63] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *ICCV*, 2017. 5, 14

[64] 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. *ECCV*, 2014. 2, 4, 6, 7, 16, 17, 18, 21

[65] Qin Liu, Zhenlin Xu, Gedas Bertasius, and Marc Niethammer. SimpleClick: Interactive image segmentation with simple vision transformers. *arXiv:2210.11006*, 2022. 7, 8, 9, 17

[66] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *ICLR*, 2019. 15

[67] Cathy H Lucas, Daniel OB Jones, Catherine J Hollyhead, Robert H Condon, Carlos M Duarte, William M Graham, Kelly L Robinson, Kylie A Pitt, Mark Schildhauer, and Jim Regetz. Gelatinous zooplankton biomass in the global oceans: geographic variation and environmental drivers. *Global Ecology and Biogeography*, 2014. 18

[68] Sabarinath Mahadevan, Paul Voigtlaender, and Bastian Leibe. Iteratively trained interactive segmentation. *BMVC*, 2018. 4, 15

[69] Kevis-Kokitsi Maninis, Sergi Caelles, Jordi Pont-Tuset, and Luc Van Gool. Deep extreme cut: From extreme points to object segmentation. *CVPR*, 2018. 6

[70] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *ICCV*, 2001. 19, 28

[71] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. *3DV*, 2016. 5, 14

[72] Massimo Minervini, Andreas Fischbach, Hanno Scharr, and Sotirios A. Tsaftaris. Finely-grained annotated datasets for image-based plant phenotyping. *Pattern Recognition Letters*, 2016. 8, 18

[73] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. *Proceedings of the conference on fairness, accountability, and transparency*, 2019. 24, 28

[74] Dim P Papadopoulos, Jasper RR Uijlings, Frank Keller, and Vittorio Ferrari. Extreme clicking for efficient object annotation. *ICCV*, 2017. 6

[75] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. Carbon emissions and large neural network training. *arXiv:2104.10350*, 2021. 28

[76] Matthew E Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. Semi-supervised sequence tagging with bidirectional language models. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 2017. 16

[77] Mengyang Pu, Yaping Huang, Yuming Liu, Qingji Guan, and Haibin Ling. EDTER: Edge detection with transformer. *CVPR*, 2022. 19

[78] Mattia Pugliatti and Francesco Topputo. DOORS: Dataset fOr bOuldeRs Segmentation. *Zenodo*, 2022. 8, 18

[79] Jiyang Qi, Yan Gao, Yao Hu, Xinggang Wang, Xiaoyu Liu, Xiang Bai, Serge Belongie, Alan Yuille, Philip Torr, and Song Bai. Occluded video instance segmentation: A benchmark. *IJCV*, 2022. 8, 18, 22, 23, 24

[80] 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. *ICML*, 2021. 1, 2, 4, 5, 7, 9, 13, 21

[81] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *ICML*, 2021. 1, 4, 9

[82] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *NeurIPS*, 2015. 6, 19

[83] Xiaofeng Ren and Jitendra Malik. Learning a classification model for segmentation. *ICCV*, 2003. 4

[84] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. *ICCV*, 2021. 8, 17, 18

[85] Candice Schumann, Susanna Ricco, Utsav Prabhu, Vittorio Ferrari, and Caroline Pantofaru. A step toward more inclusive people annotations for fairness. *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, 2021. 16

[86] Sefik Ilkin Serengil and Alper Ozpinar. LightFace: A hybrid deep face recognition framework. *ASYU*, 2020. 26

[87] Sefik Ilkin Serengil and Alper Ozpinar. HyperExtended LightFace: A facial attribute analysis framework. *ICEET*, 2021. 26

[88] Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi. TextonBoost: Joint appearance, shape and context modeling for mulit-class object recognition and segmentation. *ECCV*, 2006. 4

[89] Corey Snyder and Minh Do. STREETS: A novel camera network dataset for traffic flow. *NeurIPS*, 2019. 8, 18

[90] Konstantin Sofiiuk, Ilya A Petrov, and Anton Konushin. Reviving iterative training with mask guidance for interactive segmentation. *ICIP*, 2022. 5, 7, 8, 14, 15, 17, 22, 23, 28

[91] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 2014. 14

[92] Chris Stauffer and W Eric L Grimson. Adaptive background mixture models for real-time tracking. *CVPR*, 1999. 4

[93] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *NeurIPS*, 2020. 5, 13

[94] Yansong Tang, Yi Tian, Jiwen Lu, Jianjiang Feng, and Jie Zhou. Action recognition in RGB-D egocentric videos. *ICIP*, 2017. 18

[95] Yansong Tang, Zian Wang, Jiwen Lu, Jianjiang Feng, and Jie Zhou. Multi-stream deep neural networks for RGB-D egocentric action recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019. 18

[96] The World Bank. The world by income and regions, 2022. https://datatopics.worldbank.org/world-development-indicators/the-world-by-income-and-region.html. 16

[97] Sebastian Thrun. Is learning the n-th thing any easier than learning the first? *NeurIPS*, 1995. 9

[98] Cameron Trotter, Georgia Atkinson, Matt Sharpe, Kirsten Richardson, A. Stephen McGough, Nick Wright, Ben Burville, and Per Berggren. NDD20: A large-scale few-shot dolphin dataset for coarse and fine-grained categorisation. *arXiv:2005.13359*, 2020. 8, 17, 18, 22, 23, 24

[99] United States Environmental Protection Agency. Greenhouse Gas Equivalencies Calculator. https://www.epa.gov/energy/greenhouse-gas-equivalencies-calculator, 2022. 28

[100] Koen EA van de Sande, Jasper RR Uijlings, Theo Gevers, and Arnold WM Smeulders. Segmentation as selective search for object recognition. *ICCV*, 2011. 19

[101] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 5, 13

[102] Boying Wang, Libo Zhang, Longyin Wen, Xianglong Liu, and Yanjun Wu. Towards real-world prohibited item detection: A large-scale x-ray benchmark. *CVPR*, 2021. 8, 17, 18

[103] Weiyao Wang, Matt Feiszli, Heng Wang, Jitendra Malik, and Du Tran. Open-world instance segmentation: Exploiting pseudo ground truth from learned pairwise affinity. *CVPR*, 2022. 19

[104] Chao-Yuan Wu, Justin Johnson, Jitendra Malik, Christoph Feichtenhofer, and Georgia Gkioxari. Multiview compressive coding for 3D reconstruction. *CVPR*, 2023. 9

[105] Jianxiong Xiao, James Hays, Krista Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. *CVPR*, 2010. 18

[106] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. *ICCV*, 2015. 19

[107] Ning Xu, Brian Price, Scott Cohen, Jimei Yang, and Thomas S Huang. Deep interactive object selection. *CVPR*, 2016. 4, 17

[108] Kaiyu Yang, Klint Qinami, Li Fei-Fei, Jia Deng, and Olga Russakovsky. Towards fairer datasets: Filtering and balancing the distribution of the people subtree in the imagenet hierarchy. *Proceedings of the 2020 conference on fairness, accountability, and transparency*, 2020. 17

[109] Lei Yang, Yan Zi Wei, Yisheng HE, Wei Sun, Zhenhang Huang, Haibin Huang, and Haoqiang Fan. iShape: A first step towards irregular shape instance segmentation. *arXiv:2109.15068*, 2021. 8, 18, 22, 23, 24

[110] Senthil Yogamani, Ciarán Hughes, Jonathan Horgan, Ganesh Sistu, Padraig Varley, Derek O'Dea, Michal Uricár, Stefan Milz, Martin Simon, Karl Amende, et al. WoodScape: A multi-task, multi-camera fisheye dataset for autonomous driving. *ICCV*, 2019. 8, 18

[111] Lingzhi Zhang, Shenghao Zhou, Simon Stent, and Jianbo Shi. Fine-grained egocentric hand-object segmentation: Dataset, model, and applications. *ECCV*, 2022. 8, 17, 18

[112] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-Net: Towards unified image segmentation. *NeurIPS*, 2021. 4

[113] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. *arXiv:1707.09457*, 2017. 16

[114] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *TPAMI*, 2017. 18

[115] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ADE20K dataset. *IJCV*, 2019. 2, 7, 8, 18