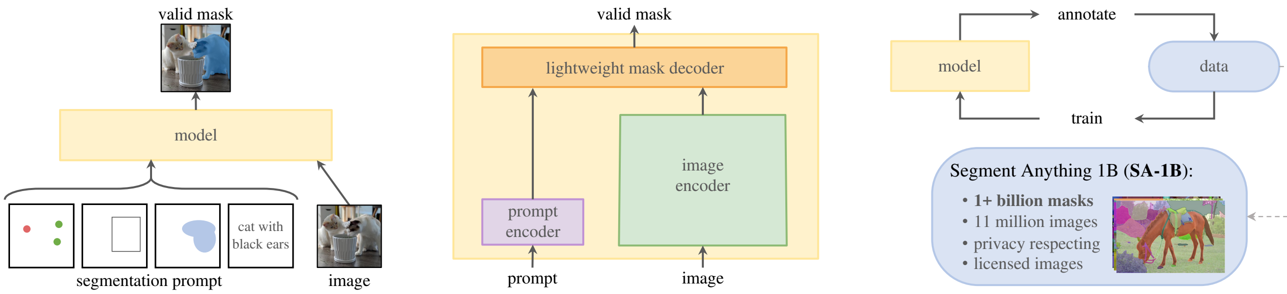
Segment Anything

Alexander Kirillov[1,2,4] Eric Mintun[2] Nikhila Ravi[1,2] Hanzi Mao[2] Chloe Rolland[3] Laura Gustafson[3]
Tete Xiao[3] Spencer Whitehead Alexander C. Berg Wan-Yen Lo Piotr Dollár[4] Ross Girshick[4]
[1]project lead [2]joint first author [3]equal contribution [4]directional lead
Meta AI Research, FAIR



(a) **Task**: promptable segmentation (b) **Model**: Segment Anything Model (**SAM**) (c) **Data**: data engine (top) & dataset (bottom)

Figure 1: We aim to build a foundation model for segmentation by introducing three interconnected components: a promptable segmentation *task*, a segmentation *model* (SAM) that powers data annotation and enables zero-shot transfer to a range of tasks via prompt engineering, and a *data* engine for collecting SA-1B, our dataset of over 1 billion masks.

Abstract

We introduce the Segment Anything (SA) project: a new task, model, and dataset for image segmentation. Using our efficient model in a data collection loop, we built the largest segmentation dataset to date (by far), with over 1 ***billion*** *masks on 11M licensed and privacy respecting images. The model is designed and trained to be promptable, so it can transfer zero-shot to new image distributions and tasks. We evaluate its capabilities on numerous tasks and find that its zero-shot performance is impressive – often competitive with or even superior to prior fully supervised results. We are releasing the Segment Anything Model (SAM) and corresponding dataset (SA-1B) of 1B masks and 11M images at https://segment-anything.com to foster research into foundation models for computer vision.*

1. Introduction

Large language models pre-trained on web-scale datasets are revolutionizing NLP with strong zero-shot and few-shot generalization [10]. These "foundation models" [8] can generalize to tasks and data distributions beyond those seen during training. This capability is often implemented with *prompt engineering* in which hand-crafted text is used to prompt the language model to generate a valid textual response for the task at hand. When scaled and trained with abundant text corpora from the web, these models' zero and few-shot performance compares surprisingly well to (even matching in some cases) fine-tuned models [10, 21]. Empirical trends show this behavior improving with model scale, dataset size, and total training compute [56, 10, 21, 51].

Foundation models have also been explored in computer vision, albeit to a lesser extent. Perhaps the most prominent illustration aligns paired text and images from the web. For example, CLIP [82] and ALIGN [55] use contrastive learning to train text and image encoders that align the two modalities. Once trained, engineered text prompts enable zero-shot generalization to novel visual concepts and data distributions. Such encoders also compose effectively with other modules to enable downstream tasks, such as image generation (*e.g.*, DALL·E [83]). While much progress has been made on vision and language encoders, computer vision includes a wide range of problems beyond this scope, and for many of these, abundant training data does not exist.

In this work, our goal is to build *a foundation model for image segmentation*. That is, we seek to develop a promptable model and pre-train it on a broad dataset using a task that enables powerful generalization. With this model, we aim to solve a range of downstream segmentation problems on new data distributions using prompt engineering.

The success of this plan hinges on three components: **task**, **model**, and **data**. To develop them, we address the following questions about image segmentation:

1. What **task** will enable zero-shot generalization?
2. What is the corresponding **model** architecture?
3. What **data** can power this task and model?

These questions are entangled and require a comprehensive solution. We start by defining a *promptable segmentation* **task** that is general enough to provide a powerful pre-training objective and to enable a wide range of downstream applications. This task requires a **model** that supports flexible prompting and can output segmentation masks in real-time when prompted to allow for interactive use. To train our model, we need a diverse, large-scale source of **data**. Unfortunately, there is no web-scale data source for segmentation; to address this, we build a "data engine", *i.e.*, we iterate between using our efficient model to assist in data collection and using the newly collected data to improve the model. We introduce each interconnected component next, followed by the dataset we created and the experiments that demonstrate the effectiveness of our approach.

Task (§2). In NLP and more recently computer vision, foundation models are a promising development that can perform zero-shot and few-shot learning for new datasets and tasks often by using "prompting" techniques. Inspired by this line of work, we propose the *promptable segmentation task*, where the goal is to return a *valid* segmentation mask given any segmentation *prompt* (see Fig. 1a). A prompt simply specifies what to segment in an image, *e.g.*, a prompt can include spatial or text information identifying an object. The requirement of a valid output mask means that even when a prompt is ambiguous and could refer to multiple objects (for example, a point on a shirt may indicate either the shirt or the person wearing it), the output should be a reasonable mask for at least one of those objects. We use the promptable segmentation task as both a pre-training objective and to solve general downstream segmentation tasks via prompt engineering.

Model (§3). The promptable segmentation task and the goal of real-world use impose constraints on the model architecture. In particular, the model must support *flexible prompts*, needs to compute masks in amortized *real-time* to allow interactive use, and must be *ambiguity-aware*. Surprisingly, we find that a simple design satisfies all three constraints: a powerful image encoder computes an image embedding, a prompt encoder embeds prompts, and then the two information sources are combined in a lightweight mask decoder that predicts segmentation masks. We refer to this model as the Segment Anything Model, or SAM (see Fig. 1b). By separating SAM into an image encoder and a fast prompt encoder / mask decoder, the same image embedding can be reused (and its cost amortized) with different prompts. Given an image embedding, the prompt encoder and mask decoder predict a mask from a prompt in ~50ms in a web browser. We focus on point, box, and mask prompts, and also present initial results with free-form text prompts. To make SAM ambiguity-aware, we design it to predict multiple masks for a single prompt allowing SAM to naturally handle ambiguity, such as the shirt *vs*. person example.

Data engine (§4). To achieve strong generalization to new data distributions, we found it necessary to train SAM on a large and diverse set of masks, beyond any segmentation dataset that already exists. While a typical approach for foundation models is to obtain data online [82], masks are not naturally abundant and thus we need an alternative strategy. Our solution is to build a "data engine", *i.e.*, we co-develop our model with model-in-the-loop dataset annotation (see Fig. 1c). Our data engine has three stages: *assisted-manual*, *semi-automatic*, and *fully automatic*. In the first stage, SAM assists annotators in annotating masks, similar to a classic interactive segmentation setup. In the second stage, SAM can automatically generate masks for a subset of objects by prompting it with likely object locations and annotators focus on annotating the remaining objects, helping increase mask diversity. In the final stage, we prompt SAM with a regular grid of foreground points, yielding on average ~100 high-quality masks per image.

Dataset (§5). Our final dataset, SA-1B, includes more than *1B* masks from *11M* licensed and privacy-preserving images (see Fig. 2). SA-1B, collected fully automatically using the final stage of our data engine, has 400× more masks than any existing segmentation dataset [66, 44, 117, 60], and as we verify extensively, the masks are of high quality and diversity. Beyond its use in training SAM to be robust and general, we hope SA-1B becomes a valuable resource for research aiming to build new foundation models.

Responsible AI (§6). We study and report on potential fairness concerns and biases when using SA-1B and SAM. Images in SA-1B span a geographically and economically diverse set of countries and we found that SAM performs similarly across different groups of people. Together, we hope this will make our work more equitable for real-world use cases. We provide model and dataset cards in the appendix.

Experiments (§7). We extensively evaluate SAM. First, using a diverse new suite of 23 segmentation datasets, we find that SAM produces high-quality masks from a single foreground point, often only slightly below that of the manually annotated ground truth. Second, we find consistently strong quantitative and qualitative results on a variety of downstream tasks under a zero-shot transfer protocol using prompt engineering, including edge detection, object proposal generation, instance segmentation, and a preliminary exploration of text-to-mask prediction. These results suggest that SAM can be used out-of-the-box with prompt engineering to solve a variety of tasks involving object and image distributions beyond SAM's training data. Nevertheless, room for improvement remains, as we discuss in §8.

Release. We are releasing the SA-1B dataset for research purposes and making SAM available under a permissive open license (Apache 2.0) at https://segment-anything.com. We also showcase SAM's capabilities with an online demo.

2. Segment Anything Task

We take inspiration from NLP, where the next token prediction task is used for foundation model pre-training *and* to solve diverse downstream tasks via prompt engineering [10]. To build a foundation model for segmentation, we aim to define a task with analogous capabilities.

Task. We start by translating the idea of a prompt from NLP to segmentation, where a prompt can be a set of foreground / background points, a rough box or mask, free-form text, or, in general, any information indicating what to segment in an image. The *promptable segmentation task*, then, is to return a *valid* segmentation mask given any *prompt*. The requirement of a "valid" mask simply means that even when a prompt is *ambiguous* and could refer to multiple objects (*e.g.*, recall the shirt *vs.* person example, and see Fig. 3), the output should be a reasonable mask for at least *one* of those objects. This requirement is similar to expecting a language model to output a coherent response to an ambiguous prompt. We choose this task because it leads to a natural pre-training algorithm *and* a general method for zero-shot transfer to downstream segmentation tasks via prompting.

Pre-training. The promptable segmentation task suggests a natural pre-training algorithm that simulates a sequence of prompts (*e.g.*, points, boxes, masks) for each training sample and compares the model's mask predictions against the ground truth. We adapt this method from interactive segmentation [109, 70], although unlike interactive segmentation whose aim is to eventually predict a valid mask after enough user input, our aim is to always predict a *valid mask* for *any prompt* even when the prompt is *ambiguous*. This ensures that a pre-trained model is effective in use cases that involve ambiguity, including automatic annotation as required by our data engine §4. We note that performing well at this task is challenging and requires specialized modeling and training loss choices, which we discuss in §3.

Zero-shot transfer. Intuitively, our pre-training task endows the model with the ability to respond appropriately to any prompt at inference time, and thus downstream tasks can be solved by engineering appropriate prompts. For example, if one has a bounding box detector for cats, cat instance segmentation can be solved by providing the detector's box output as a prompt to our model. In general, a wide array of practical segmentation tasks can be cast as prompting. In addition to automatic dataset labeling, we explore five diverse example tasks in our experiments in §7.

Related tasks. Segmentation is a broad field: there's interactive segmentation [57, 109], edge detection [3], super pixelization [85], object proposal generation [2], foreground segmentation [94], semantic segmentation [90], instance segmentation [66], panoptic segmentation [59], *etc.* The goal of our promptable segmentation task is to produce a broadly capable model that can adapt to *many* (though not all) existing and *new* segmentation tasks via prompt engineering. This capability is a form of task generalization [26]. Note that this is different than previous work on multi-task segmentation systems. In a multi-task system, a single model performs a *fixed* set of tasks, *e.g.*, joint semantic, instance, and panoptic segmentation [114, 19, 54], but the training and test tasks are the same. An important distinction in our work is that a model trained for promptable segmentation can perform a new, different task at inference time by acting as a *component* in a larger system, *e.g.*, to perform instance segmentation, a promptable segmentation model is *combined* with an existing object detector.

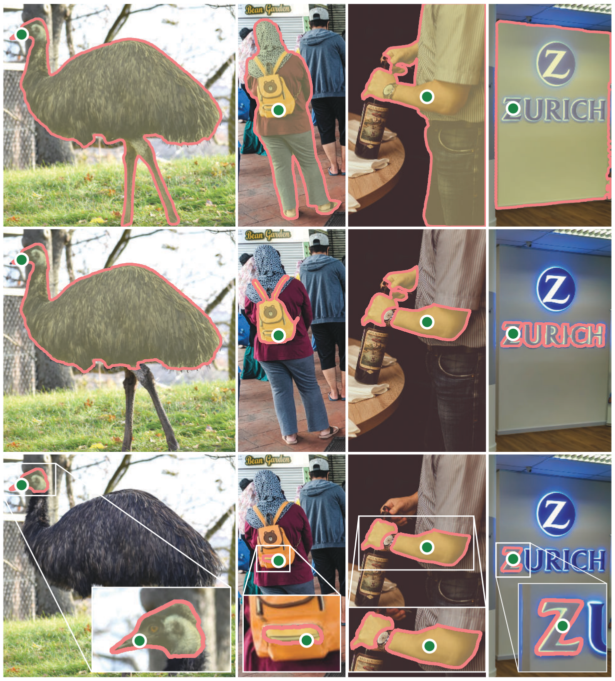


Figure 3: Each column shows 3 valid masks generated by SAM from a single ambiguous point prompt (green circle).

Discussion. Prompting and composition are powerful tools that enable a single model to be used in extensible ways, potentially to accomplish tasks unknown at the time of model design. This approach is analogous to how other foundation models are used, *e.g.*, how CLIP [82] is the text-image alignment component of the DALL·E [83] image generation system. We anticipate that composable system design, powered by techniques such as prompt engineering, will enable a wider variety of applications than systems trained specifically for a fixed set of tasks. It's also interesting to compare promptable and interactive segmentation through the lens of composition: while interactive segmentation models are designed with human users in mind, a model trained for promptable segmentation can also be composed into a larger algorithmic system as we will demonstrate.

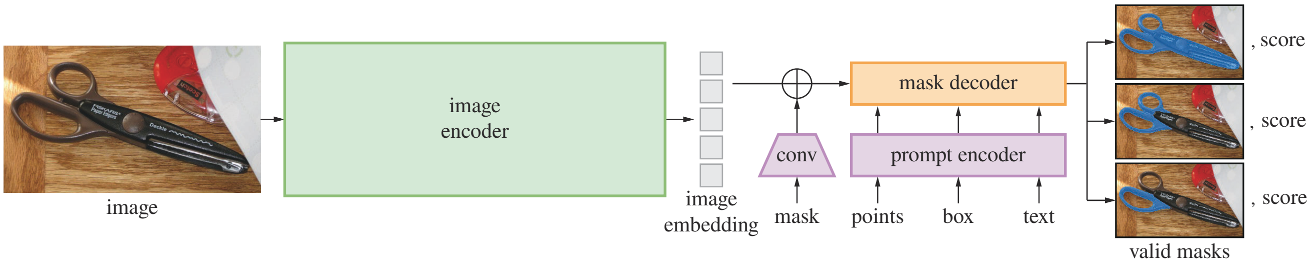


Figure 4: Segment Anything Model (SAM) overview. A heavyweight image encoder outputs an image embedding that can then be efficiently queried by a variety of input prompts to produce object masks at amortized real-time speed. For ambiguous prompts corresponding to more than one object, SAM can output multiple valid masks and associated confidence scores.

3. Segment Anything Model

We next describe the Segment Anything Model (SAM) for promptable segmentation. SAM has three components, illustrated in Fig. 4: an image encoder, a flexible prompt encoder, and a fast mask decoder. We build on Transformer vision models [14, 33, 20, 62] with specific tradeoffs for (amortized) real-time performance. We describe these components at a high-level here, with details in §A.

Image encoder. Motivated by scalability and powerful pretraining methods, we use an MAE [47] pre-trained Vision Transformer (ViT) [33] minimally adapted to process high resolution inputs [62]. The image encoder runs once per image and can be applied prior to prompting the model.

Prompt encoder. We consider two sets of prompts: *sparse* (points, boxes, text) and *dense* (masks). We represent points and boxes by positional encodings [95] summed with learned embeddings for each prompt type and free-form text with an off-the-shelf text encoder from CLIP [82]. Dense prompts (*i.e.*, masks) are embedded using convolutions and summed element-wise with the image embedding.

Mask decoder. The mask decoder efficiently maps the image embedding, prompt embeddings, and an output token to a mask. This design, inspired by [14, 20], employs a modification of a Transformer decoder block [103] followed by a dynamic mask prediction head. Our modified decoder block uses prompt self-attention and cross-attention in two directions (prompt-to-image embedding and vice-versa) to update *all* embeddings. After running two blocks, we upsample the image embedding and an MLP maps the output token to a dynamic linear classifier, which then computes the mask foreground probability at each image location.

Resolving ambiguity. With one output, the model will average multiple valid masks if given an ambiguous prompt. To address this, we modify the model to predict multiple output masks for a single prompt (see Fig. 3). We found 3 mask outputs is sufficient to address most common cases (nested masks are often at most three deep: whole, part, and subpart). During training, we backprop only the minimum loss [15, 45, 64] over masks. To rank masks, the model predicts a confidence score (*i.e.*, estimated IoU) for each mask.

Efficiency. The overall model design is largely motivated by efficiency. Given a precomputed image embedding, the prompt encoder and mask decoder run in a web browser, on CPU, in ∼50ms. This runtime performance enables seamless, real-time interactive prompting of our model.

Losses and training. We supervise mask prediction with the linear combination of focal loss [65] and dice loss [73] used in [14]. We train for the promptable segmentation task using a mixture of geometric prompts (for text prompts see §7.5). Following [92, 37], we simulate an interactive setup by randomly sampling prompts in 11 rounds per mask, allowing SAM to integrate seamlessly into our data engine.

4. Segment Anything Data Engine

As segmentation masks are not abundant on the internet, we built a data engine to enable the collection of our 1.1B mask dataset, SA-1B. The data engine has three stages: (1) a model-assisted manual annotation stage, (2) a semi-automatic stage with a mix of automatically predicted masks and model-assisted annotation, and (3) a fully automatic stage in which our model generates masks without annotator input. We go into details of each next.

Assisted-manual stage. In the first stage, resembling classic interactive segmentation, a team of professional annotators labeled masks by clicking foreground / background object points using a browser-based interactive segmentation tool powered by SAM. Masks could be refined using pixel-precise "brush" and "eraser" tools. Our model-assisted annotation runs in real-time directly inside a browser (using precomputed image embeddings) enabling a truly interactive experience. We did not impose semantic constraints for labeling objects, and annotators freely labeled both "stuff" and "things" [1]. We suggested annotators label objects they could name or describe, but did not collect these names or descriptions. Annotators were asked to label objects in order of prominence and were encouraged to proceed to the next image once a mask took over 30 seconds to annotate.

At the start of this stage, SAM was trained using common public segmentation datasets. After sufficient data annotation, SAM was retrained using only newly annotated masks. As more masks were collected, the image encoder was scaled from ViT-B to ViT-H and other architectural details evolved; in total we retrained our model 6 times. Average annotation time per mask decreased from 34 to 14 seconds as the model improved. We note that 14 seconds is 6.5× faster than mask annotation for COCO [66] and only 2× slower than bounding-box labeling with extreme points [76, 71]. As SAM improved, the average number of masks per image increased from 20 to 44 masks. Overall, we collected 4.3M masks from 120k images in this stage.

Semi-automatic stage. In this stage, we aimed to increase the *diversity* of masks in order to improve our model's ability to segment anything. To focus annotators on less prominent objects, we first automatically detected confident masks. Then we presented annotators with images prefilled with these masks and asked them to annotate any additional unannotated objects. To detect confident masks, we trained a bounding box detector [84] on all first stage masks using a generic "object" category. During this stage we collected an additional 5.9M masks in 180k images (for a total of 10.2M masks). As in the first stage, we periodically retrained our model on newly collected data (5 times). Average annotation time per mask went back up to 34 seconds (excluding the automatic masks) as these objects were more challenging to label. The average number of masks per image went from 44 to 72 masks (including the automatic masks).

Fully automatic stage. In the final stage, annotation was *fully automatic*. This was feasible due to two major enhancements to our model. First, at the start of this stage, we had collected enough masks to greatly improve the model, including the diverse masks from the previous stage. Second, by this stage we had developed the ambiguity-aware model, which allowed us to predict valid masks even in ambiguous cases. Specifically, we prompted the model with a 32×32 regular grid of points and for each point predicted a set of masks that may correspond to valid objects. With the ambiguity-aware model, if a point lies on a part or subpart, our model will return the subpart, part, and whole object. The IoU prediction module of our model is used to select *confident* masks; moreover, we identified and selected only *stable* masks (we consider a mask stable if thresholding the probability map at $0.5 - \delta$ and $0.5 + \delta$ results in similar masks). Finally, after selecting the confident and stable masks, we applied non-maximal suppression (NMS) to filter duplicates. To further improve the quality of smaller masks, we also processed multiple overlapping zoomed-in image crops. For further details of this stage, see §B. We applied fully automatic mask generation to all 11M images in our dataset, producing a total of 1.1B high-quality masks. We describe and analyze the resulting dataset, SA-1B, next.

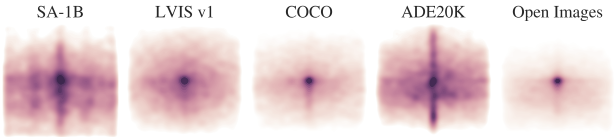


Figure 5: Image-size normalized mask center distributions.

5. Segment Anything Dataset

Our dataset, SA-1B, consists of 11M diverse, high-resolution, licensed, and privacy protecting images and 1.1B high-quality segmentation masks collected with our data engine. We compare SA-1B with existing datasets and analyze mask quality and properties. We are releasing SA-1B to aid future development of foundation models for computer vision. We note that SA-1B will be released under a favorable license agreement for certain research uses and with protections for researchers.

Images. We licensed a new set of 11M images from a provider that works directly with photographers. These images are high resolution (3300×4950 pixels on average), and the resulting data size can present accessibility and storage challenges. Therefore, we are releasing downsampled images with their shortest side set to 1500 pixels. Even after downsampling, our images are significantly higher resolution than many existing vision datasets (*e.g.*, COCO [66] images are ~480×640 pixels). Note that most models today operate on much lower resolution inputs. Faces and vehicle license plates have been blurred in the released images.

Masks. Our data engine produced 1.1B masks, 99.1% of which were generated fully automatically. Therefore, the quality of the automatic masks is centrally important. We compare them directly to professional annotations and look at how various mask properties compare to prominent segmentation datasets. Our main conclusion, as borne out in the analysis below and the experiments in §7, is that our automatic masks are high quality and effective for training models. Motivated by these findings, SA-1B *only includes automatically generated masks.*

Mask quality. To estimate mask quality, we randomly sampled 500 images (~50k masks) and asked our professional annotators to improve the quality of all masks in these images. Annotators did so using our model and pixel-precise "brush" and "eraser" editing tools. This procedure resulted in pairs of automatically predicted and professionally corrected masks. We computed IoU between each pair and found that 94% of pairs have greater than 90% IoU (and 97% of pairs have greater than 75% IoU). For comparison, prior work estimates inter-annotator consistency at 85-91% IoU [44, 60]. Our experiments in §7 confirm by human ratings that mask quality is high relative to a variety of datasets and that training our model on automatic masks is nearly as good as using all masks produced by the data engine.