COCONut: Modernizing COCO Segmentation

Xueqing Deng Qihang Yu Peng Wang Xiaohui Shen Liang-Chieh Chen
ByteDance

https://github.com/xdeng7/coconut.git

Abstract

*In recent decades, the vision community has witnessed remarkable progress in visual recognition, partially owing to advancements in dataset benchmarks. Notably, the established COCO benchmark has propelled the development of modern detection and segmentation systems. However, the COCO segmentation benchmark has seen comparatively slow improvement over the last decade. Originally equipped with coarse polygon annotations for 'thing' instances, it gradually incorporated coarse superpixel annotations for 'stuff' regions, which were subsequently heuristically amalgamated to yield panoptic segmentation annotations. These annotations, executed by different groups of raters, have resulted not only in coarse segmentation masks but also in inconsistencies between segmentation types. In this study, we undertake a comprehensive reevaluation of the COCO segmentation annotations. By enhancing the annotation quality and expanding the dataset to encompass 383K images with more than 5.18M panoptic masks, we introduce COCONut, the **COCO** **N**ext **U**niversal segmen**T**ation dataset. COCONut harmonizes segmentation annotations across semantic, instance, and panoptic segmentation with meticulously crafted high-quality masks, and establishes a robust benchmark for all segmentation tasks. To our knowledge, COCONut stands as the inaugural large-scale universal segmentation dataset, verified by human raters. We anticipate that the release of COCONut will significantly contribute to the community's ability to assess the progress of novel neural networks.*

1. Introduction

Over the past decades, significant advancements in computer vision have been achieved, partially attributed to the establishment of comprehensive benchmark datasets. The COCO dataset [35], in particular, has played a pivotal role in the development of modern vision models, addressing a wide range of tasks such as object detection [3, 18, 22, 36, 46, 48, 68], segmentation [5–7, 10, 28, 40, 56–58, 64], keypoint detection [20, 24, 45, 54], and image captioning [8, 47, 62]. Despite the advent of large-scale neural network models [4, 14, 39] and extensive datasets [30, 53], COCO continues to be a primary benchmark across various tasks, including image-to-text [33, 37, 62] and text-to-image [51, 63] multi-modal models. It has also been instrumental in the development of novel models, such as those fine-tuning on COCO for image captioning [49, 63] or open-vocabulary recognition [17, 19, 31, 61, 66, 67]. However, nearly a decade since its introduction, the suitability of COCO as a benchmark for contemporary models warrants reconsideration. This is particularly pertinent given the potential nuances and biases embedded within the dataset, reflective of the early stages of computer vision research.

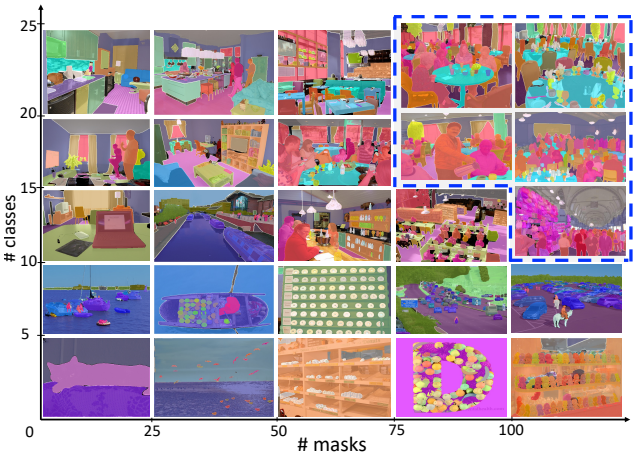


Figure 1. **COCONut Annotation Masks.** Comprising images from COCO and Objects365, COCONut represents a diverse collection annotated with high-quality masks and semantic classes. Notably, images sourced from Objects365 (marked in blue boundaries) contribute to the dataset's richness by featuring a higher count of classes and masks per image. Best zoomed-in.

COCO's early design inevitably encompassed certain annotation biases, including issues like imprecise object boundaries and incorrect class labels (Fig. 2). While these limitations were initially acceptable in the nascent stages of computer vision research (*e.g.*, bounding boxes are invariant to the coarsely annotated masks as long as the extreme points are the same [44]), the rapid evolution of model architectures has led to a performance plateau on the COCO

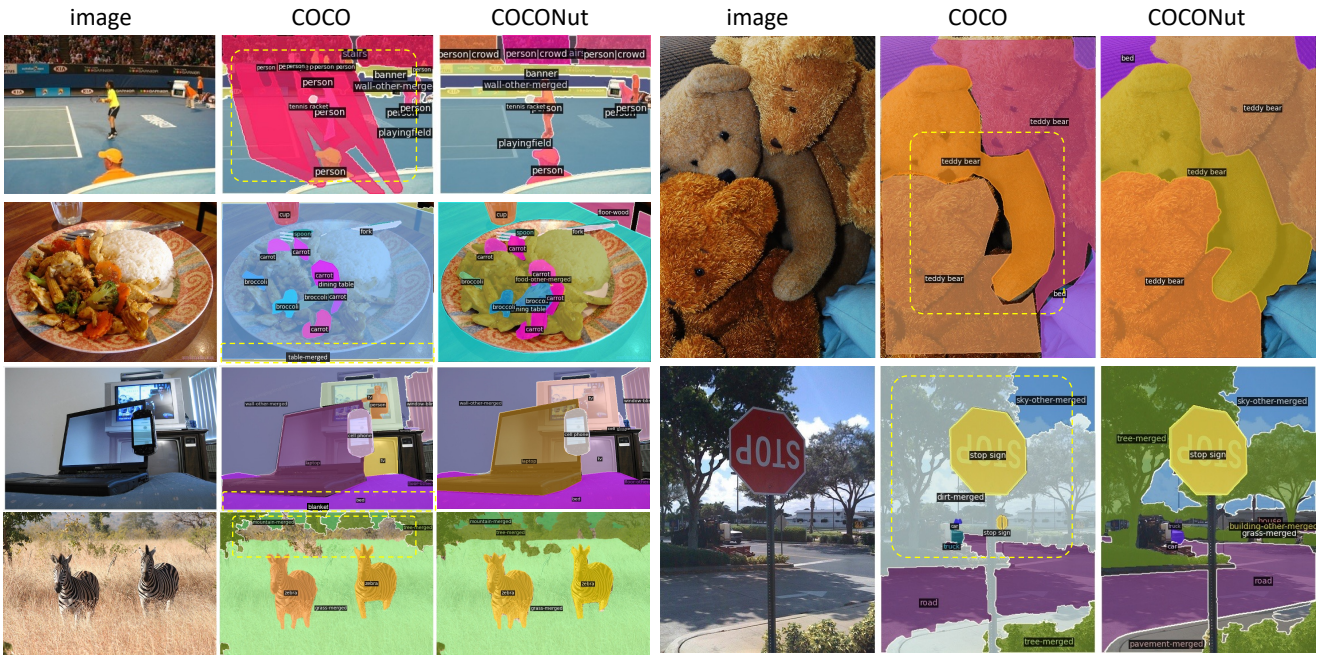


Figure 2. **Annotation Comparison:** We delineate erroneous annotations from COCO using yellow dotted line boxes, juxtaposed with our COCONut corrected annotations. Common COCO annotation errors include over-annotations (*e.g.*, 'person crowd' erroneously extends into 'playingfield'), incomplete mask fragments (*e.g.*, 'table-merged' and 'blanket' are annotated in small isolated segments), missing annotations (*e.g.*, 'tree-merged' remains unannotated), coarse segmentations (especially noticeable in 'stuff' regions annotated by superpixels and in 'thing' regions by loose polygons), and wrong semantic categories (*e.g.*, 'tree-merged' is incorrectly tagged as 'dirt-merged').

benchmark[1]. This stagnation suggests a potential overfitting to the dataset's specific characteristics, raising concerns about the models' applicability to real-world data. Furthermore, despite COCO's diverse annotations supporting various tasks, its annotation is neither exhaustive nor consistent. This in-exhaustiveness is evident in the segmentation annotations, where instances of incomplete labeling are commonplace. Additionally, discrepancies between semantic, instance, and panoptic annotations within the dataset present challenges in developing a comprehensive segmentation model. Moreover, in the context of the ongoing shift towards even larger-scale datasets [16, 52], COCO's repository of approximately 120K images and 1.3M masks appears increasingly inadequate. This limitation hampers its utility in training and evaluating models designed to process and learn from substantially larger and more varied datasets.

To modernize COCO segmentation annotations, we propose the development of a novel, large-scale universal segmentation dataset, dubbed COCONut for the **COCON**ext **U**niversal segmen**T**ation dataset. Distinct in its approach to ensuring high-quality annotations, COCONut features human-verified mask labels for 383K images. Unlike previous attempts at creating large-scale datasets, which often compromise on label accuracy for scale [52, 59], our focus is on maintaining human verification as a standard for dataset quality. To realize this ambition, our initial step involves meticulously developing an assisted-manual annotation pipeline tailored for high-quality labeling on the subset of 118K COCO images, designated as COCONut-S split. The pipeline benefits from modern neural networks (bounding box detector [43] and mask segmenter [55, 65]), allowing our annotation raters to efficiently edit and refine those proposals. Subsequently, to expand the data size while preserving quality, we develop a data engine, leveraging the COCONut-S dataset as a high-quality training datae to upgrade the neural networks. The process iteratively generates various sizes of COCONut training sets, yielding COCONut-B (242K images and 2.78M masks), and COCONut-L (358K images and 4.75M masks).

We adhere to a principle of consistency in annotation, aiming to establish a universal segmentation dataset (*i.e.*, consistent annotations for all panoptic/instance/semantic segmentation tasks). Additionally, COCONut includes a meticulously curated high-quality validation set, COCONut-val, comprising 5K images carefully re-labeled from the COCO validation set, along with an additional 20K images from Objects365 [53] (thus, totally 25K images and 437K masks).

To summarize, our contributions are threefold:

- We introduce COCONut, a modern, universal segmentation dataset that encompasses about 383K images and 5.18M human-verified segmentation masks. This dataset

[1] https://paperswithcode.com/dataset/coco

	COCONut	COCO-17 [35]	EntitySeg [41]	ADE20K [69]	Sama-COCO [70]	LVIS [21]	Open Images [32]	COCO-Stuff [2]	PAS-21 [15]	PC-59 [42]
# images (train/val/test)	358K / 25K / -	118K / 5K / 41K	10K / 1.5K / -†	20K / 2K / 3K‡	118K / 5K / -	100K / 20K / 40K	944K / 13K / 40K	118K / 5K / 41K	1.4K / 1.4K / 1.4K	5K / 5K / -
# masks / image	13.2 / 17.4 / -	11.2 / 11.3 / -	16.8 / 16.4 / -	13.4 / 15.1 / -	9.0 / 9.5 / -	12.7 / 12.4 / -	2.8 / 1.8 / 1.8	8.6 / 8.9 / -	2.5 / 2.5/ -	4.9 / 4.8 / -
# masks	4.75M / 437K / -	1.3M / 57K / -	0.17M / 24K / -	0.27M / 30K / -	1.07M / 47K / -	1.27M / 0.24M / -	2.7M / 25K / 74K	1.02M / 44K / -	3.6K / 3.6K / -	24K / 24K / -
# thing classes	80	80	535	115	80	1203	350	-	-	-
# stuff classes	53	53	109	35	-	-	-	91	21	59
panoptic segmentation	✓	✓	✓	✓						
instance segmentation	✓	✓	✓	△	✓	✓	✓			
semantic segmentation	✓	△	✓	✓				✓	✓	✓
object detection	✓	✓	△	△	✓	✓	✓			

Table 1. **Dataset Comparison:** We compare existing segmentation datasets that focus on daily images (street-view images are not our focus). The definition of 'thing' and 'stuff' classes are different across datasets, where the 'stuff' classes are not annotated with instance identities. †: EntitySeg dataset comprises 33K images, of which only 11K are equipped with panoptic annotations. ‡: ADE20K test server only supports semantic segmentation, and its panoptic annotations are derived by merging separately annotated instance and semantic segmentation maps, introducing minor inconsistencies between segmentation types. △: task supported, but not typically used.

represents a significant expansion in both scale and quality of annotations compared to existing datasets. Additionally, COCONut-val, featuring meticulously curated high-quality annotations for validation, stands as a novel and challenging testbed for the research community.

- Our study includes an in-depth error analysis of the COCO dataset's annotations. This analysis not only reveals various inconsistencies and ambiguities in the existing labels but also informs our approach to refining label definitions. As a result, COCONut features ground-truth annotations with enhanced consistency and reduced label map ambiguity.
- With the COCONut dataset as our foundation, we embark on a comprehensive analysis. Our experimental results not only underscore the efficacy of scaling up datasets with high-quality annotations for both training and validation sets, but also highlight the superior value of human annotations compared to pseudo-labels.

2. Related Work

In this work, we focus on segmentation datasets, featuring daily images (Tab. 1). A prime example of this is the COCO dataset [35], which has been a cornerstone in computer vision for over a decade. Initially, COCO primarily focused on detection and captioning tasks [8]. Subsequent efforts have expanded its scope, refining annotations to support a wider array of tasks. For instance, COCO-Stuff [2] added semantic masks for 91 'stuff' categories, later integrated with instance masks to facilitate panoptic segmentation [29]. In addition to these expansions, several initiatives have aimed at enhancing the quality of COCO's annotations. The LVIS dataset [21] extends the number of object categories from 80 to 1,203, providing more comprehensive annotations for each image. Similarly, Sama-COCO [70] addresses the issue of low-quality masks in COCO by re-annotating instances at a finer granularity. Beyond the COCO-related datasets, there are other notable datasets contributing to diverse research scenarios, including ADE20K [69], PASCAL [15], and PASCAL-Context [42]. While these datasets have significantly advanced computer vision research, they still fall short in either annotation quality or quantity when it comes to meeting the demands for high-quality large-scale datasets.

In the realm of recent dataset innovations, SA-1B [30] stands out with its unprecedented scale, comprising 11M images and 1B masks. However, a critical aspect to consider is the feasibility of human annotation at such an immense scale. Consequently, a vast majority (99.1%) of SA-1B's annotations are machine-generated and lack specific class designations. Additionally, its human annotations are not publicly released. Contrasting with scaling dataset size, the EntitySeg dataset [41] prioritizes enhancing annotation quality. This dataset features high-resolution images accompanied by meticulously curated high-quality mask annotations. However, the emphasis on the quality of annotations incurs significant resource demands, which in turn limits the dataset's scope. As a result, EntitySeg encompasses a relatively modest collection of 33K images, of which only approximately one-third are annotated with panoptic classes. Along the same direction of scaling up datasets, we present COCONut, a new large scale dataset with high quality mask annotations and semantic tags.

3. Constructing the COCONut Dataset

In this section, we first revisit COCO's class map definition (Sec. 3.1) and outline our image sources and varied training data sizes (Sec. 3.2). The construction of COCONut centers on two key objectives: high quality and large scale. To achieve these, we establish an efficient annotation pipeline ensuring both mask quality and accurate semantic tags (Sec. 3.3). This pipeline facilitates scalable dataset expansion while upholding annotation quality (Sec. 3.4).

3.1. COCO's Class Map Definition

In alignment with the COCO panoptic set [29], COCONut encompasses 133 semantic classes, with 80 categorized as 'thing' and 53 as 'stuff.' Adopting the same COCO class map ensures backward compatibility, enabling the initial use of models trained on COCO-related datasets [2, 29, 35, 70] to generate pseudo labels in our annotation pipeline.

Notably, COCONut refines class map definitions compared to COCO, offering greater clarity in our annotation

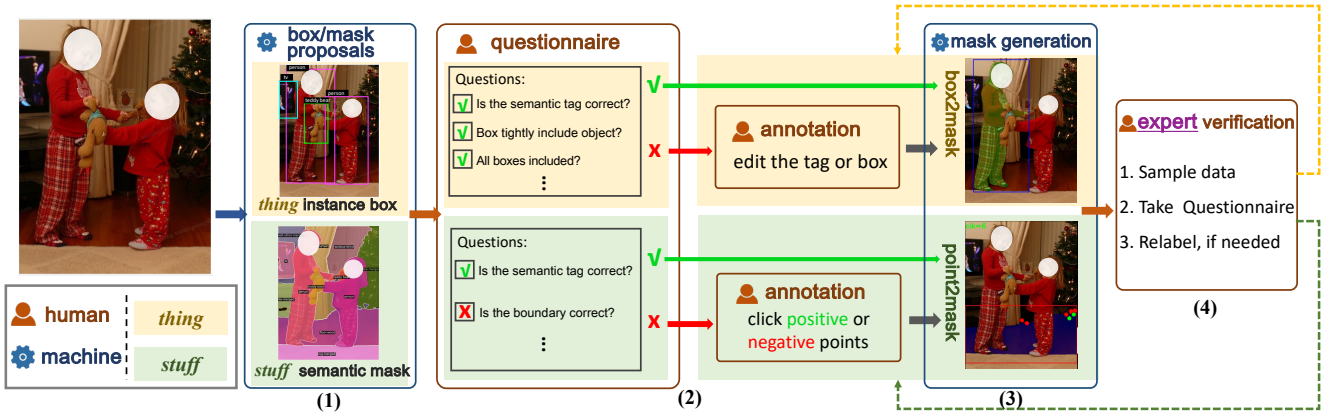


Figure 3. **Overview of the Proposed Assisted-Manual Annotation Pipeline:** To streamline the labor-intensive labeling task, our annotation pipeline encompasses four pivotal stages: (1) machine-generated pseudo labels, (2) human inspection and editing, (3) mask generation or refinement, and (4) quality verification. Acknowledging the inherent distinctions between 'thing' and 'stuff' classes, we systematically address these intricacies at each stage. Stage 1: Machines are employed to generate box and mask proposals for 'thing' and 'stuff', respectively. Stage 2: Raters assess the proposal qualities using a meticulously crafted questionnaire. For proposals falling short of requirements, raters can update them by editing boxes or adding positive/negative points for 'thing' and 'stuff', respectively. Stage 3: We utilize Box2Mask and Point2Mask modules to generate masks based on the inputs from stage 2. Stage 4: Experts perform a comprehensive verification of annotation quality, with relabeling done if the quality falls below our stringent standards.

instruction protocol. Building upon COCO's class map, we introduce additional definitions and instructions for labeling segmentation masks. To mitigate the annotation confusion, we meticulously define label map details and provide clear instructions to our annotation raters. For comprehensive definitions and annotation instructions for all 133 classes, please refer to the supplementary materials.

3.2. Image Sources and Data Splits

The images comprising COCONut are sourced from public datasets. Primarily, we aggregate images from the original COCO training and validation sets as well as its unlabeled set. Additionally, we select approximately 136K images from Objects365 dataset [53], each annotated with bounding boxes and containing at least one COCO class. This comprehensive collection results in a total of 358K and 25K images for training and validation, respectively. As illustrated in Tab. 2, we meticulously define diverse training datasets for COCONut, spanning from 118K images to 358K images. COCONut-S (small) encompasses the same images as the original COCO training set, totaling 118K images. We adopt COCO panoptic [35] and Sama-COCO [70] masks as our starting point. COCONut-B (base) incorporates additional images from the COCO unlabeled set, totaling 242K images. Finally, with extra 116K images from the Objects365 dataset, COCONut-L (large) comprises 358K images. Additionally, COCONut-val contains 5K images from the COCO validation set along with an additional 20K Objects365 images.

dataset splits	image sources	#images	#masks	#masks/image
COCONut-S	COCO training set [35]	118K	1.54M	13.1
COCONut-B	+ COCO unlabeled set [35]	242K	2.78M	11.5
COCONut-L	+ subset of Objects365 [53]	358K	4.75M	13.2
relabeled COCO-*val*	COCO validation set [35]	5K	67K	13.4
COCONut-*val*	+ subset of Objects365 [53]	25K	437K	17.4

Table 2. **Definition of COCONut Dataset Splits:** Statistics are shown accumulatively. Notably, our COCONut-val contains large #masks/image, preseting a more challenging testbed.

3.3. Assisted-Manual Annotation Pipeline

Annotation Challenges: The task of densely annotating images with segmentation masks, coupled with their semantic tags (*i.e.*, classes), is exceptionally labor-intensive. Our preliminary studies reveal that, on average, it takes one expert rater approximately 5 minutes to annotate a single mask. Extrapolating this to annotate images at a scale of 10M masks would necessitate 95 years with just one expert rater. Even with a budget to employ 100 expert raters, the annotation process would still require about a year to complete. Given the extensive time and cost involved, this challenge underscores the need to explore a more effective and efficient annotation pipeline.

Annotation Pipeline: In response to the challenges, we introduce the assisted-manual annotation pipeline, utilizing neural networks to augment human annotators. As illustrated in Fig. 3, the pipeline encompasses four key stages: (1) machine-generated prediction, (2) human inspection and editing, (3) mask generation or refinement, and (4) quality verification. Recognizing the inherent differences between 'thing' (countable objects) and 'stuff' (amorphous regions), we meticulously address them at every stage.

Machine-Generated Prediction: In handling 'thing' classes, we utilize the bounding box object detector DETA [43], and for 'stuff' classes, we deploy the mask segmenter kMaX-DeepLab [65]. This stage yields a set of box proposals for 'thing' and mask proposals for 'stuff'.

Human Inspection and Editing: With the provided box and mask proposals, raters meticulously evaluate them based on a prepared questionnaire (*e.g.*, Is the box/mask sufficiently accurate? Is the tag correct? Any missing boxes?) The raters adhere to stringent standards during inspection to ensure proposal quality. In cases where proposals fall short, raters are directed to perform further editing. Specifically, for 'thing' classes, raters have the flexibility to add or remove boxes along with their corresponding tags (i.e., classes). In the case of 'stuff' classes, raters can refine masks by clicking positive or negative points, indicating whether the points belong to the target instance or not.

Mask Generation or Refinement: Utilizing the provided boxes and masks from the preceding stage, we employ the **Box2Mask** and **Point2Mask** modules to generate segmentation masks for 'thing' and 'stuff' classes, respectively. The **Box2Mask** module extends kMaX-DeepLab, resulting in the box-kMaX model, which generates masks based on provided bounding boxes. This model incorporates additional box queries in conjunction with the original object queries. The added box queries function similarly to the original object queries, except that they are initialized using features pooled from the backbone within the box regions (original object queries are randomly initialized). As shown in Fig. 4, leveraging object-aware box queries enables box-kMaX to effectively segment 'thing' objects with the provided bounding boxes. The **Point2Mask** module utilizes the interactive segmenter CFR [55], taking positive/negative points as input and optionally any initial mask (from either kMaX-DeepLab or the previous round's output mask). This stage allows us to amass a collection of masks generated from boxes and refined by points.

It is worth noting that there are other interactive segmenters that are also capable of generating masks using box and point as inputs (*e.g.*, SAM [30], SAM-HQ[27]). However, our analyses (in Sec. 4) indicate that the tools we have developed suffice for our raters to produce high-quality annotations. The primary focus of our work is to conduct a comprehensive analysis between the original COCO dataset and our newly annotated COCONut. Improving interactive segmenters lies outside the scope of this study.

Quality Verification by Experts: Armed with the amassed masks from the preceding stage, we task *expert raters* with quality verification. Unlike the general human raters in stage 2, our expert raters boast extensive experience in dense pixel labeling (5 years of proficiency in Photoshop). To manage the extensive volume of annotated masks with only two experts, we opt for a random sampling of 50%. The experts meticulously assess these masks, along with their associated tags, using the same carefully crafted questionnaire as in the previous stage. Furthermore, recognizing the Box2Mask module's reliance on provided bounding boxes, we additionally instruct experts to verify the accuracy of box proposals, selecting 30% samples for a thorough quality check. Should any fall short of our stringent requirements, they undergo relabeling using the time-intensive Photoshop tool to ensure high annotation quality.

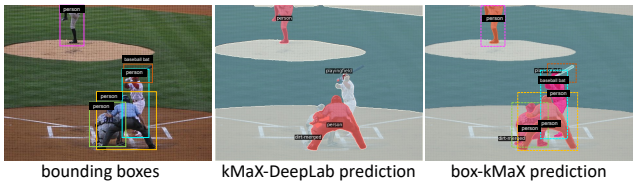
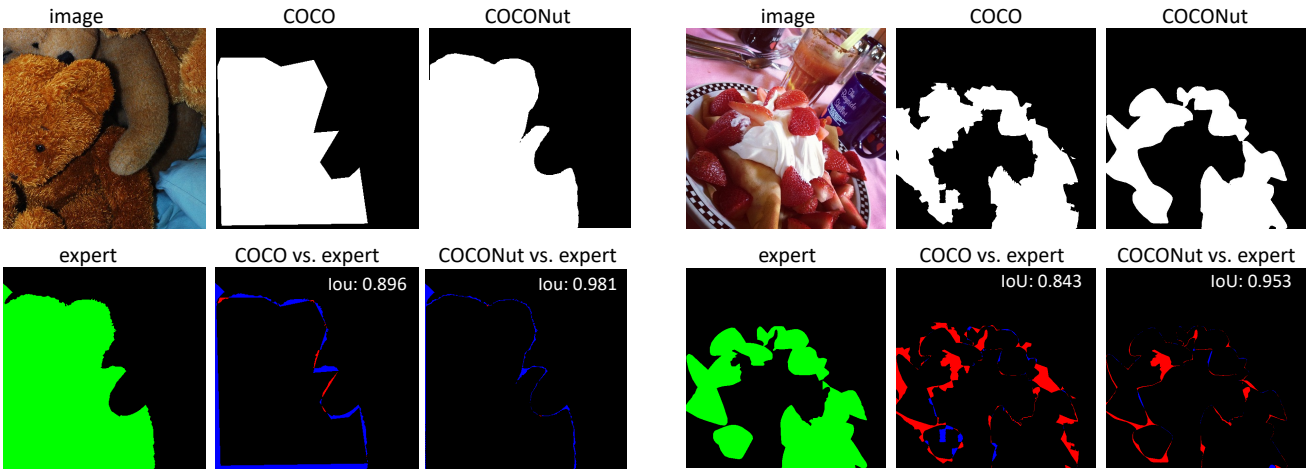


Figure 4. **Mask Prediction Comparison:** In contrast to kMaX-DeepLab, box-kMaX (Box2Mask module) leverages box queries, initialized with features pooled from the backbone within the box regions, enabling more accurate segmentation of 'thing' objects. Notably, kMaX-DeepLab falls short in capturing the challenging 'baseball bat' and the heavily occluded 'person' in the figure.

3.4. Data Engine for Scaling Up Dataset Size

Overview: With the streamlined assisted-manual annotation pipeline in place, we build a data engine to facilitate the dataset expansion. Our data engine capitalizes on the annotation pipeline to accumulate extensive, high-quality annotations, subsequently enhancing the training of new neural networks for improved pseudo-label generation. This positive feedback loop is iteratively applied multiple times.

Data Engine: Machines play a crucial role in generating box/mask proposals (stage 1) and refined masks (stage 3) in the assisted-manual annotation pipeline. Initially, publicly available pre-trained neural networks are employed to produce proposals. Specifically, DETA [43] (utilizing a Swin-L backbone [38] trained with Objects365 [53] and COCO detection set [35]) and kMaX-DeepLab [65] (featuring a ConvNeXt-L backbone [39] trained with COCO panoptic set [29]) are utilized to generate box and mask proposals for 'thing' and 'stuff', respectively. The Point2Mask module (built upon CFR [55]) remains fixed throughout the COCONut construction, while the Box2Mask module (box-kMaX, a variant of kMaX-DeepLab using box queries) is trained on COCO panoptic set. The annotation pipeline initially produces the COCONut-S dataset split. Subsequently, COCONut-S is used to re-train kMaX-DeepLab and box-kMaX, enhancing mask proposals for 'stuff' and Box2Mask capabilities, respectively. Notably, DETA and the Point2Mask module are not re-trained, as DETA is already pre-trained on a substantial dataset, and CFR exhibits robust generalizability. The upgraded neural networks yield improved proposals and mask generations, enhancing the assisted-manual annotation pipeline and leading to the cre-



(a) annotation comparison for 'thing' **(b) annotation comparison for 'stuff'**

Figure 5. **Annotation Comparison:** We show annotations obtained by COCO, COCONut (Box2Mask for 'thing' in (a) or Point2Mask for 'stuff' in (b)), and our expert rater. COCONut's annotation exhibits sharper boundaries, closely resembling expert results, as evident from higher IoU values. The blue and red regions correspond to extra and missing regions, respectively, compared to the expert mask.

	'thing'	'stuff'
expert-1 *vs*. expert-2	98.1%	97.3%
raters *vs*. experts	96.3%	96.7%

(a) **Annotation Agreement**

	'thing'	'stuff'
purely-manual	10 min	5 min
assisted-manual	10 sec	42 sec

(b) **Annotation Speed**

Table 3. **Annotation Analysis:** (a) Our two experts and raters demonstrate a high level of agreement in their annotations. (b) The assisted-manual pipeline expedites the annotation.

constructed dataset	mean	median
COCONut-S	78%	75%
COCONut-B	51%	55%
COCONut-L	43%	45%

(a) **Non-Pass Rate in Stage 2**

constructed dataset	mean	median
COCONut-S	2.4	2
COCONut-B	0.8	1
COCONut-L	0.5	1

(b) **#Rounds of Relabeling in Stage 4**

Table 4. **Data Engine Analysis:** During the creation of the current dataset split, the mask proposals stem from models trained on datasets from preceding stages, such as COCONut-S utilizing proposal models from COCO, and so forth.

ation of COCONut-B. This process is iterated to generate the final COCONut-L, which also benefits from the ground-truth boxes provided by Objects365.

4. Annotation and Data Engine Analysis

In this section, we scrutinize the annotations produced through our proposed assisted-manual annotation pipeline (Sec. 4.1). Subsequently, we delve into the analysis of the improvement brought by our data engine (Sec. 4.2).

4.1. Annotation Analysis

Assisted-Manual *vs*. Purely-Manual: We conduct a thorough comparison in this study between annotations generated by our assisted-manual and purely-manual annotation pipelines. Our assessment is based on two metrics: annotation quality and processing speed.

The purely-manual annotation pipeline involves two in-house experts, each with over 5 years of experience using Photoshop for labeling dense segmentation maps. They received detailed instructions based on our annotation guidelines and subsequently served as tutorial training mentors for our annotation raters. Additionally, they played a crucial role in the quality verification of masks during stage 4.

To conduct the "agreement" experiments, we randomly selected 1000 segmentation masks and tasked our two in-house experts with annotating each mask. An "agreement" was achieved when both annotations exhibited an IoU (Intersection-over-Union) greater than 95%. As presented in Tab. 3a, our experts consistently demonstrated a high level of agreement in annotating both 'thing' and 'stuff' masks. Comparatively, minor disparities were observed in the annotations provided by our raters, highlighting their proficiency. Additionally, Tab. 3b showcases the annotation speed. The assisted-manual pipeline notably accelerates the annotation process by editing boxes and points, particularly beneficial for 'thing' annotations. Annotating 'stuff', however, involves additional time due to revising the coarse superpixel annotations by COCO. Finally, Fig. 5 presents annotation examples from COCO, our experts, and COCONut (our raters with the assisted-manual pipeline), underscoring the high-quality masks produced.

4.2. Data Engine Analysis

The data engine enhances neural networks using annotated high-quality data, resulting in improved pseudo masks and decreased workload for human raters. To measure its impact, we present non-pass rates in stage 2 human inspection. These rates indicate the percentage of machine-generated proposals that failed our questionnaire's standards and re-

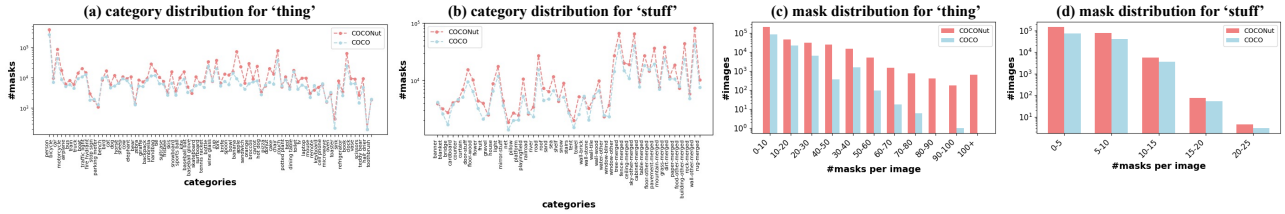


Figure 6. **Dataset Statistics:** In subfigures (a) and (b), depicting category distributions for 'thing' and 'stuff', COCONut consistently displays a higher number of masks across all categories compared to COCO. Subfigures (c) and (d) show mask distribution for 'thing' and 'stuff', respectively, demonstrating that COCONut contains a greater number of images with a higher density of masks per image.

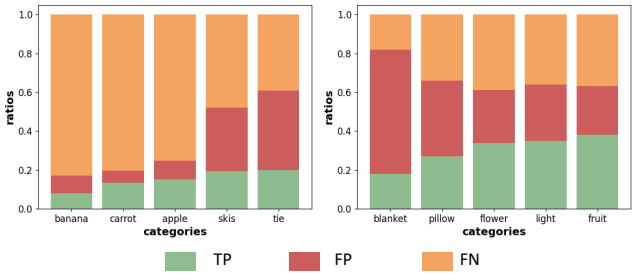


Figure 7. **Top 5 Disagreed Categories Between COCO-val and relabeled COCO-val:** COCO-val is treated as the prediction, while relabeled COCO-val serves as ground truth. The comparison showcases True Positive (TP), False Positive (FP), and False Negative (FN) rates for both 'thing' (left) and 'stuff' (right).

	PQ	SQ	RQ	PQ^{bdry}	SQ^{bdry}	RQ^{bdr}
all	67.1	86.2	77.4	59.2	79.4	74.5
thing	65.0	86.0	75.2	58.6	80.7	72.4
stuff	70.2	86.5	80.8	60.1	77.3	77.6

Table 5. **Quantitative Comparison Between COCO-val and relabeled COCO-val:** COCO-val serves as the prediction, contrasting with relabeled COCO-val as the ground-truth.

quired further editing. Tab. 4a demonstrates that including more high-quality training data improves non-pass rates, signifying enhanced proposal quality. Furthermore, Tab. 4b showcases the number of relabeling rounds in stage 4 expert verification, reflecting additional iterations required for annotations failing expert verification. Consistently, we observed reduced relabeling rounds with increased inclusion of high-quality training data.

5. Dataset Statistics

Class and Mask Distribution: Fig. 6 depicts the category and mask distribution within COCONut. Panels (a) and (b) demonstrate that COCONut surpasses COCO in the number of masks across all categories. Additionally, panels (c) and (d) feature histograms depicting the frequency of 'masks per image'. These histograms highlight a notable trend in COCONut, indicating a higher prevalence of images with denser mask annotations compared to COCO.

COCO-val *vs*. relabeled COCO-val: We conducted a comparative analysis between the original COCO-val annotations and our relabeled COCO-val. Exploiting the Panoptic Quality (PQ) metric, we employed its True Positive (TP), False Positive (FP), and False Negative (FN) rates to assess each category. TP signifies agreement between annotations, while FP and FN highlight additional or missing masks, respectively. In Fig. 7, we present the top 5 categories displaying discrepancies for both 'thing' and 'stuff'. All these categories exhibit notably low TP rates, indicating substantial differences between COCO-val and our relabeled version. In 'thing' categories, high FN rates (around 0.8) are observed for 'banana', 'carrot', and 'apple', suggesting numerous missing masks. Conversely, 'stuff' categories exhibit high FP rates for 'blanket' and 'pillow', indicating numerous small isolated masks, echoing our earlier findings regarding 'bed' and 'blanket' conflicts, as depicted in Fig. 2 (row 3). Finally, Tab. 5 provides a quantitative analysis comparing COCO-val and our relabeled COCO-val. The results emphasize the notable divergence between the two sets, underscoring our dedicated efforts to improve the annotation quality of validation set. The discrepancy is particularly evident in boundary metrics [11]. Notably, the divergence in stuff SQ^{bdry} reflects our enhancements to the original 'stuff' annotations by superpixels [1, 2].

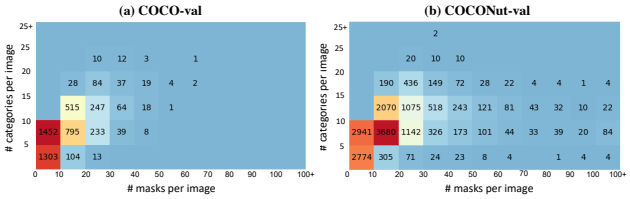


Figure 8. **Mask and Class Frequency Distribution:** COCONut-val introduces a more challenging testbed compared to the original COCO-val. It features a greater number of images that contain higher quantities of both masks and distinct categories per image.

COCONut-val (a new challenging testbed): To augment our relabeled COCO-val, we introduced an additional 20K annotated images from Objects365, forming COCONut-val. Fig. 8 illustrates 2D histograms comparing COCO-val and COCONut-val, where we count the number of images w.r.t. their #masks and #categories per image. The figure showcases that COCO-val annotations are concentrated around a smaller number of masks and cate-

backbone	training set	COCO-val			relabeled COCO-val			COCONut-val		
		PQ	AP^{mask}	mIoU	PQ	AP^{mask}	mIoU	PQ	AP^{mask}	mIoU
ResNet50	COCO	53.3	39.6	61.7	55.1	40.6	63.9	53.1	37.1	62.5
	COCONut-S	51.7	37.5	59.4	58.9	44.4	64.4	56.7	41.2	63.6
	COCONut-B	53.4	39.3	62.6	60.2	45.2	65.7	58.1	42.9	64.7
	COCONut-L	54.1	40.2	63.1	60.7	45.8	66.1	60.7	44.8	68.3
ConvNeXt-L	COCO	57.9	45.0	66.9	60.4	46.4	69.9	58.3	44.1	66.4
	COCONut-S	55.9	41.9	66.1	64.4	50.8	71.4	59.4	45.7	67.8
	COCONut-B	57.8	44.8	66.6	64.9	51.2	71.8	61.3	46.5	69.5
	COCONut-L	58.1	45.3	67.3	65.1	51.4	71.9	62.7	47.6	70.6

Table 6. **Training Data and Backbones:** The evaluations are conducted on three different validation sets: original COCO-val, relabeled COCO-val (by our raters), and COCONut-val.

gories, whereas COCONut-val demonstrates a broader distribution, with more images having over 30 masks. On average, COCONut-val boasts 17.4 masks per image, significantly exceeding COCO-val's average of 11.3 masks.

6. Discussion

In light of the COCONut dataset, we undertake a meticulous analysis to address the following inquiries. We employ kMaX-DeepLab [65] throughout the experiments, benchmarked with several training and validation sets.

COCO encompasses only 133 semantic classes. Is an extensive collection of human annotations truly necessary? We approach this query from two vantage points: the training and validation sets. Tab. 6 showcases consistent improvements across various backbones (ResNet50 [23] and ConvNeXt-L [39]) and three evaluated validation sets (measured in PQ, AP, and mIoU) as the training set size increases from COCONut-S to COCONut-L. Interestingly, relying solely on the original small-scale COCO training set yields unsatisfactory performance on both relabeled COCO-val and COCONut-val sets, emphasizing the need for more human annotations in training. Despite annotation biases between COCO and COCONut (Fig. 9), training with COCONut-B achieves performance akin to the original COCO training set on the COCO validation set, hinting that a larger training corpus might mitigate inter-dataset biases.

Shifting our focus from the training set to the validation set, the results in Tab. 6 indicate performance saturation on both COCO-val and relabeled COCO-val as the training set expands from COCONut-B to COCONut-L. This saturation phenomenon in COCO-val, consisting of only 5K images, is also observed in the literature[2], suggesting its inadequacy in evaluating modern segmenters. Conversely, the newly introduced COCONut-val, comprising 25K images with denser mask annotations, significantly improves benchmarking for models trained with varied data amounts. This outcome underscores the significance of incorporating more human-annotated, challenging validation images for robust model assessment. Therefore, the inclusion of additional human-annotated images is pivotal for both training and validation, significantly impacting the performance of modern segmentation models.

[2] https://paperswithcode.com/dataset/coco

backbone	training set	COCO-val			relabeled COCO-val			COCONut-val		
		PQ	AP^{mask}	mIoU	PQ	AP^{mask}	mIoU	PQ	AP^{mask}	mIoU
ConvNeXt-L	COCO	57.9	45.0	66.9	60.4	46.4	69.9	58.3	44.1	66.4
	COCO-B_M	58.0	44.9	67.1	60.7	46.3	70.5	58.5	44.2	66.4
	COCONut-S	55.9	41.9	66.1	64.4	50.8	71.4	59.4	45.7	67.8
	COCONut-B_M	56.2	41.8	66.3	64.5	50.9	71.4	59.5	45.3	67.7
	COCONut-B	57.8	44.8	66.6	64.9	51.2	71.8	61.3	46.5	69.5

Table 7. **Pseudo-Labels *vs*. Human Labels:** COCO-B_M comprises the original COCO training set plus the machine pseudo-labeled COCO unlabeled set. COCONut-B_M contains COCONut-S and machine pseudo-labeled COCO unlabeled set (in contrast to the fully human-labeled COCONut-B).

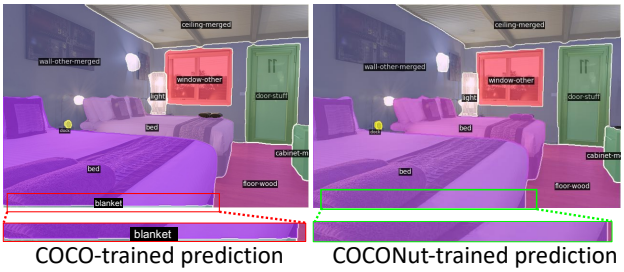


Figure 9. **Influence of Training Data on Predictions:** We present predictions from two models: one trained on original COCO (left) and the other on COCONut (right). The COCO-trained model predicts a small isolated mask, influenced by the biases inherent in the COCO coarse annotations (see Fig. 2, row 3). Best zoomed-in.

Are pseudo-labels a cost-effective alternative to human annotations? While expanding datasets using machine-generated pseudo-labels seems promising for scaling models trained on large-scale data, its effectiveness remains uncertain. To address this, we conducted experiments outlined in Tab. 7. Initially, leveraging a checkpoint (row 1: 57.9% PQ on COCO-val), we generated pseudo-labels for the COCO unlabeled set, augmenting the original COCO training set to create the COCO-B_M dataset. Surprisingly, training on COCO-B_M resulted in only a marginal 0.1% PQ improvement on COCO-val, consistent across all tested validation sets (1st and 2nd rows in the table).

We hypothesized that the annotation quality of the pre-trained dataset might influence pseudo-label quality. To investigate, we then utilized a different checkpoint (row 3: 64.4% PQ on relabeled COCO-val) to generate new pseudo-labels for the COCO unlabeled set. Combining these with COCONut-S produced the COCONut-B_M dataset, yet still yielded a mere 0.1% PQ improvement on the relabeled COCO-val. Notably, employing the fully human-labeled COCONut-B resulted in the most significant improvements (last row in the table). Our findings suggest limited benefits from incorporating pseudo-labels. Training with pseudo-labels seems akin to distilling knowledge from a pre-trained network [26], offering minimal additional information for training new models.

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *TPAMI*, 34 (11):2274–2282, 2012. 7

[2] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *CVPR*, 2018. 3, 7

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1

[4] Jieneng Chen, Qihang Yu, Xiaohui Shen, Alan Yuille, and Liang-Chieh Chen. ViTamin: Designing Scalable Vision Models in the Vision-Language Era. In *CVPR*, 2024. 1

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. 1

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 40(4):834–848, 2017.

[7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 1

[8] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 1, 3

[9] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022. 3

[10] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *CVPR*, 2020. 1

[11] Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C. Berg, and Alexander Kirillov. Boundary IoU: Improving object-centric image segmentation evaluation. In *CVPR*, 2021. 7

[12] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 5

[13] Zheng Ding, Jieke Wang, and Zhuowen Tu. Open-vocabulary panoptic segmentation with maskclip. In *ICML*, 2023. 1

[14] 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. In *ICLR*, 2021. 1, 3

[15] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88:303–338, 2010. 3, 1

[16] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *arXiv preprint arXiv:2304.14108*, 2023. 2

[17] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *ECCV*, 2022. 1

[18] Ross Girshick. Fast r-cnn. In *ICCV*, 2015. 1

[19] Xiuye Gu, Yin Cui, Jonathan Huang, Abdullah Rashwan, Xuan Yang, Xingyi Zhou, Golnaz Ghiasi, Weicheng Kuo, Huizhong Chen, Liang-Chieh Chen, and David A Ross. Dataseg: Taming a universal multi-dataset multi-task segmentation model. *NeurIPS*, 2023. 1

[20] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, 2018. 1

[21] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 3

[22] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *ECCV*, 2014. 1

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 8, 1

[24] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 1

[25] Xuming He, Richard S Zemel, and Miguel Á Carreira-Perpiñán. Multiscale conditional random fields for image labeling. In *CVPR*, 2004. 1

[26] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 8

[27] Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Segment anything in high quality. In *NeurIPS*, 2023. 5

[28] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, 2019. 1

[29] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *CVPR*, 2019. 3, 5, 1

[30] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *ICCV*, 2023. 1, 3, 5

[31] Weicheng Kuo, Yin Cui, Xiuye Gu, AJ Piergiovanni, and Anelia Angelova. F-vlm: Open-vocabulary object detection upon frozen vision and language models. In *ICLR*, 2023. 1

[32] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object

detection, and visual relationship detection at scale. *IJCV*, 128(7):1956–1981, 2020. 3

[33] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023. 1

[34] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *CVPR*, 2023. 4, 5

[35] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1, 3, 4, 5

[36] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 1

[37] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS*, 2023. 1

[38] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 5, 1

[39] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *CVPR*, 2022. 1, 5, 8

[40] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1

[41] Qi Lu, Jason Kuen, Shen Tiancheng, Gu Jiuxiang, Guo Weidong, Jia Jiaya, Lin Zhe, and Yang Ming-Hsuan. High-quality entity segmentation. In *ICCV*, 2023. 3

[42] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, 2014. 3, 1

[43] Jeffrey Ouyang-Zhang, Jang Hyun Cho, Xingyi Zhou, and Philipp Krähenbühl. Nms strikes back. *arXiv preprint arXiv:2212.06137*, 2022. 2, 5, 1

[44] Dim P Papadopoulos, Jasper RR Uijlings, Frank Keller, and Vittorio Ferrari. Extreme clicking for efficient object annotation. In *ICCV*, 2017. 1

[45] George Papandreou, Tyler Zhu, Liang-Chieh Chen, Spyros Gidaris, Jonathan Tompson, and Kevin Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In *ECCV*, 2018. 1

[46] Siyuan Qiao, Liang-Chieh Chen, and Alan Yuille. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. In *CVPR*, 2021. 1

[47] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1

[48] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NeurIPS*, 2015. 1

[49] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1

[50] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 5

[51] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *NeurIPS*, 2022. 1

[52] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 2022. 2

[53] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *ICCV*, 2019. 1, 2, 4, 5

[54] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 1

[55] Shoukun Sun, Min Xian, Fei Xu, Tiankai Yao, and Luca Capriotti. Cfr-icl: Cascade-forward refinement with iterative click loss for interactive image segmentation. *arXiv preprint arXiv:2303.05620*, 2023. 2, 5

[56] Shuyang Sun, Weijun Wang, Andrew Howard, Qihang Yu, Philip Torr, and Liang-Chieh Chen. Remax: Relaxing for better training on efficient panoptic segmentation. *NeurIPS*, 2024. 1

[57] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *ECCV*, 2020.

[58] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *CVPR*, 2021. 1

[59] Weiyun Wang, Min Shi, Qingyun Li, Wenhai Wang, Zhenhang Huang, Linjie Xing, Zhe Chen, Hao Li, Xizhou Zhu, Zhiguo Cao, et al. The all-seeing project: Towards panoptic visual recognition and understanding of the open world. *arXiv preprint arXiv:2308.01907*, 2023. 2

[60] Mark Weber, Huiyu Wang, Siyuan Qiao, Jun Xie, Maxwell D Collins, Yukun Zhu, Liangzhe Yuan, Dahun Kim, Qihang Yu, Daniel Cremers, et al. Deeplab2: A tensorflow library for deep labeling. *arXiv preprint arXiv:2106.09748*, 2021. 1

[61] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *CVPR*, 2023. 1

[62] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *TMLR*, 2022. 1

[63] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *TMLR*, 2022. 1

[64] Qihang Yu, Huiyu Wang, Dahun Kim, Siyuan Qiao, Maxwell Collins, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Cmt-deeplab: Clustering mask transformers for panoptic segmentation. In *CVPR*, 2022. 1

[65] Qihang Yu, Huiyu Wang, Siyuan Qiao, Maxwell Collins, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. k-means Mask Transformer. In *ECCV*, 2022. 2, 5, 8, 1

[66] Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Convolutions die hard: Open-vocabulary segmentation with single frozen convolutional clip. *NeurIPS*, 2023. 1, 5

[67] Qihang Yu, Xiaohui Shen, and Liang-Chieh Chen. Towards open-ended visual recognition with large language model. *arXiv preprint arXiv:2311.08400*, 2023. 1

[68] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. In *ICLR*, 2023. 1

[69] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 3, 1

[70] Eric Zimmermann, Justin Szeto, Jerome Pasquero, and Frederic Ratle. Benchmarking a benchmark: How reliable is ms-coco? In *ICCV Datacomp Workshop*, 2023. 3, 4