

Objaverse: A Universe of Annotated 3D Objects

Matt Deitke^{†ψ}, Dustin Schwenk[†], Jordi Salvador[†], Luca Weihs[†], Oscar Michel[†]
 Eli VanderBilt[†], Ludwig Schmidt^ψ, Kiana Ehsani[†], Aniruddha Kembhavi^{†ψ}, Ali Farhadi^ψ
[†]PRIOR @ Allen Institute for AI, ^ψUniversity of Washington, Seattle
objaverse.allenai.org

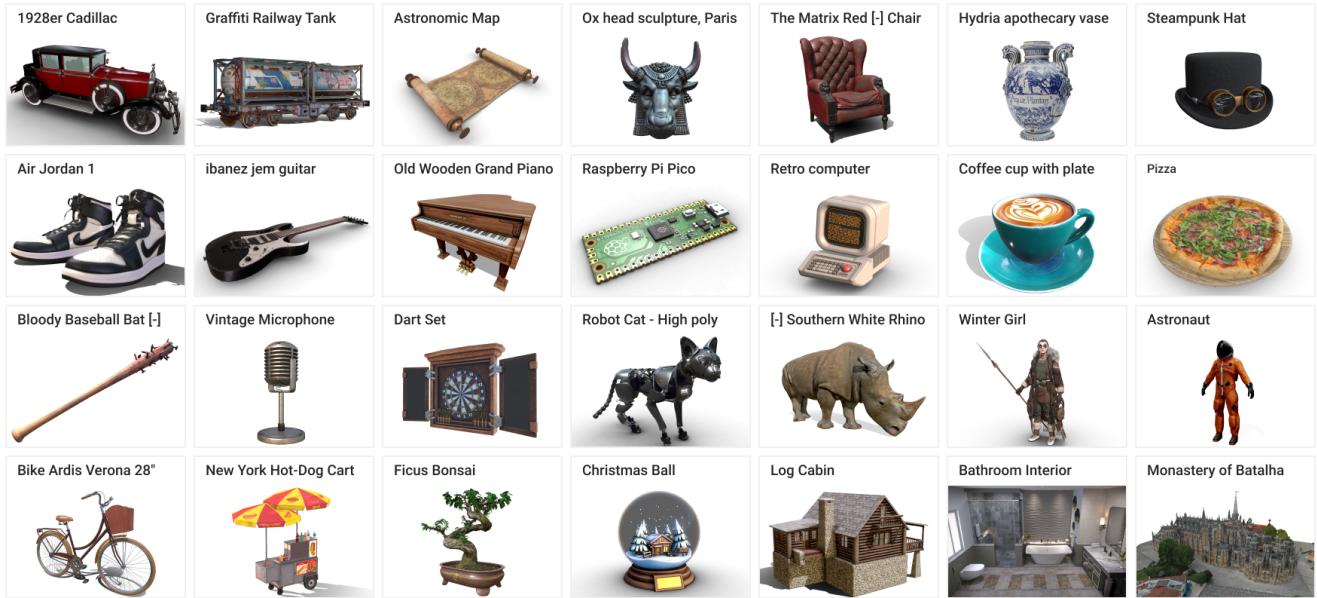


Figure 1. Example instances from our large-scale 3D asset dataset OBJAVERSE. OBJAVERSE 3D assets are semantically diverse, high-quality, and paired with natural-language descriptions.

Abstract

Massive data corpora like WebText, Wikipedia, Conceptual Captions, WebImageText, and LAION have propelled recent dramatic progress in AI. Large neural models trained on such datasets produce impressive results and top many of today’s benchmarks. A notable omission within this family of large-scale datasets is 3D data. Despite considerable interest and potential applications in 3D vision, datasets of high-fidelity 3D models continue to be mid-sized with limited diversity of object categories. Addressing this gap, we present Objaverse 1.0, a large dataset of objects with 800K+ (and growing) 3D models with descriptive captions, tags, and animations. Objaverse improves upon present day 3D repositories in terms of scale, number of categories, and in the visual diversity of instances within a category. We demonstrate the large potential of Objaverse via four diverse applications: training generative 3D models, im-

proving tail category segmentation on the LVIS benchmark, training open-vocabulary object-navigation models for Embodied AI, and creating a new benchmark for robustness analysis of vision models. Objaverse can open new directions for research and enable new applications across the field of AI.

1. Introduction

Massive datasets have enabled and driven rapid progress in AI. Language corpora on the web led to large language models like GPT-3 [4]; paired image and text datasets like Conceptual Captions [68] led to vision-and-language pre-trained models like ViLBERT [45]; YouTube video datasets led to video capable models like Merlot-Reserve [87]; and massive multimodal datasets like WebImageText [70] and LAION [66, 67] led to models like CLIP [60] and StableDiffusion [64]. These leaps in dataset scale and diversity were triggered by moving from manually curated datasets to harnessing the power of the web and its creative content.

In contrast to the datasets described above, the size of

Correspondence to <mattd@allenai.org>.

the datasets we are feeding to our, data-hungry, deep learning models in many other areas of research is simply not comparable. For instance, the number of 3D assets used in training generative 3D models is, maximally, on the order of thousands [24] and the simulators used to train embodied AI models typically have only between a few dozen to a thousand unique scenes [39, 42, 63, 72]. The startling advances brought about by developing large-scale datasets for images, videos, and natural language, demand that an equivalent dataset be built for 3D assets.

We present OBJAVERSE 1.0, a large scale corpus of high-quality, richly annotated, 3D objects; see Fig. 1. Objects in our dataset are free to use¹ and sourced from Sketchfab, a leading online platform for managing, viewing, and distributing 3D models. In total, OBJAVERSE contains over **800K** 3D assets designed by over **100K** artists which makes this data large and diversely sourced. Assets not only belong to varied categories like animals, humans, and vehicles, but also include interiors and exteriors of large spaces that can be used, *e.g.*, to train embodied agents. OBJAVERSE is a universe of rich 3D data with detailed metadata that can support many different annotations to enable new applications. With this remarkable increase in scale, we see an incredible opportunity for OBJAVERSE to impact research progress across domains. In this work, we provide promising results to answer three questions.

Can 3D vision benefit from a large-scale dataset?

First, as a 3D asset resource, OBJAVERSE can support the exciting field of 3D generative modeling. We use data extracted from OBJAVERSE to train generative models for single and multiple categories using GET3D [24] and find that we are able to generate high-quality objects and, moreover, that our generated objects are found by human annotators to be more diverse than those generated by a model trained on ShapeNet objects in 91% of cases.

Can the diversity of 3D models help improve classical 2D vision task performance? To answer this question, we use the diversity of OBJAVERSE to improve the performance of long tail instance segmentation models. Instance segmentation data can be expensive to obtain owing to the cost of annotating contours around objects. The recent LVIS dataset contains annotations for 1,230 categories but the task remains very challenging for present day models, particularly on tail categories that have few examples. We show that increasing the volume of data by leveraging a simple Copy+Paste augmentation method with OBJAVERSE assets can improve the performance of state-of-the-art segmentation methods.

We also use OBJAVERSE to build a benchmark for evaluating the robustness of state-of-the-art visual classification models to perspective shifts. We render objects in OBJAVERSE from random orientations, which is how one

might expect to see them in the real world and test the ability of CLIP-style visual backbones to correctly classify these images. Our experiments show that current state-of-the-art models’ performance degrades dramatically in this setting when viewing objects from arbitrary views. OBJAVERSE allows us to build benchmarks to test (and potentially train) for orientation robustness for a long tail distribution of asset categories. Building such benchmarks is made uniquely possible by the scale and diversity of 3D assets in OBJAVERSE. This would simply not be feasible to create in the real world nor can they be generated from existing 2D images.

Can a large-scale 3D dataset help us train embodied agents performant embodied agents? We use assets in OBJAVERSE to populate procedurally generated simulated environments in ProcTHOR [17] that are used to train Embodied AI agents. This results in an orders of magnitude increase in the number of unique assets available for use in ProcTHOR scenes (previously limited to AI2-THOR’s [39] asset library of a few thousand unique instances each assigned to one of 108 object categories). Using OBJAVERSE populated scenes enables open vocabulary object navigation from any text description. In this paper, we provide quantitative results for navigating to 1.1K semantic object categories, roughly a 50x increase.

These findings represent just a small fraction of what can be accomplished using OBJAVERSE. We are excited to see how the research community will leverage OBJAVERSE to enable fast and exciting progress in 2D and 3D computer vision applications and beyond.

2. Related Work

Large scale datasets. Scaling the size and scope of training datasets has widely been demonstrated to be an effective avenue of improvement for model performance. In computer vision, the adoption of early large scale datasets such as Imagenet [18, 65] and MS-COCO [44] has dramatically accelerated progress on a variety of tasks including classification, object detection, captioning, and more. Ever since, the diversity and scale of datasets have continued to grow. YFCC100M is a dataset of 99.2M images and 800K videos [77]. OpenImages [40] is a large scale dataset of 9M images that contains labeled subsets bounding boxes, visual relationships, segmentation masks, localized narratives, and categorical annotations. Massive web-scraped datasets containing image-text pairs such as Conceptual Captions [68], WIT [70], and LAION [66, 67] have seen increased popularity recently as they have been used to train impressive models for vision-language representation learning [29, 32, 60], text-to-image generation [32, 61, 62, 64], and vision-language multitasking [9, 10, 73, 79].

3D datasets. Current large-scale 2D image datasets offer three crucial components that benefit learning: scale,

¹Creative Commons license

Dataset	# Objects	# Classes
YCB [5]	77	5
BigBIRD [69]	125	–
KIT [35]	219	145
IKEA [43]	219	11
Pix3D [71]	395	9
GSO [19]	1K	17
EGAD [51]	2K	–
PhotoShape [53]	5K	1
ABO [12]	8K	63
3D-Future [22]	10K	34
ShapeNet [7]	51K	55
Objaverse	818K	21K

Figure 2. Comparison between OBJAVERSE and existing 3D object datasets. (*Left:*) Visual comparison of instances from OBJAVERSE and ShapeNet for the categories of CAR, BED, VASE, and BAG. OBJAVERSE instances are substantially more diverse since objects can come from many 3D content creation platforms, whereas ShapeNet models look more similar and all come from SketchUp, a 3D modeling platform built for simple architectural modeling. (*Right:*) Scale comparison table between existing 3D object datasets.

diversity, and realism. Ideally, models that reason about 3D objects should have access to datasets that meet these same criteria. However, of the numerous 3D object datasets that currently exist, none are able to excel in all three categories to the same degree as their 2D counterparts. Datasets such as KIT [35], YCB [5], BigBIRD [69], IKEA [43], and Pix3D [71] provide image-calibrated models over a diverse set of household objects, but severely lack in scale with only a few hundred objects at most. EGAD [51] procedurally generates 2K objects for grasping, but produces objects that are not that realistic or diverse. Slightly larger datasets of photo-realistic objects include GSO [19], PhotoShape [53], ABO [12] and 3D-Future [22], and ShapeNet [7] with object counts in the tens of thousands, see Fig. 2 for comparisons between OBJAVERSE and these datasets. Datasets for CAD models, such as ModelNet [83] and DeepCAD [82], and ABC [38] do not include textures or materials, which limits their ability to represent objects that could plausibly be found in the real world. Datasets of scanned 3D objects and environments are valuable for real-world understanding [11, 14, 15, 41], but are quite small and limited. In addition to containing numerous artist designed objects, OBJAVERSE contains many scanned assets, making it a useful source of data for learning from real-world distributions.

While rapid progress has been made in developing datasets that combine image and text, in contrast, only a few datasets that pair language and 3D data exist. Text2Shape [8] released a dataset of 15,038 chairs and tables from ShapeNet each with around 5 text captions, giving 75,344 total text-shape pairs. ShapeGlot [1] released the CiC (Chairs in Context) dataset which contains 4,511 chairs from ShapeNet along with 78,789 descriptive utterances generated from a referential game. Due to the small scale and limited diversity of these datasets, current SoTA text-to-3D models [31, 47, 57] forgo the use of 3D datasets entirely

and instead rely on 2D image-text supervision.

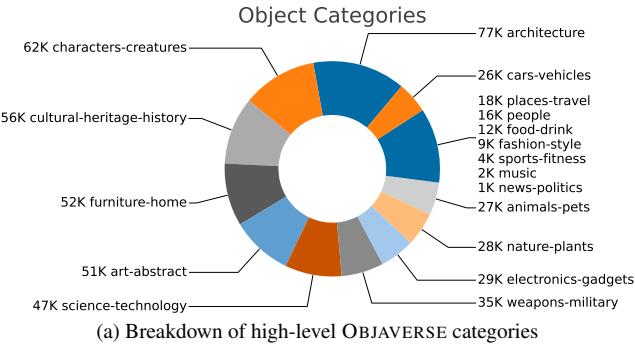
3. Objaverse

OBJAVERSE is a massive annotated 3D dataset that can be used to enable research in a wide range of areas across computer vision. The objects are sourced from Sketchfab, an online 3D marketplace where users can upload and share models for both free and commercial use. Objects selected for OBJAVERSE have a distributable Creative Commons license and were obtained using Sketchfab’s public API. Aside from licensing consideration, models marked as restricted due to objectionable or adult thematic content were excluded from the dataset.

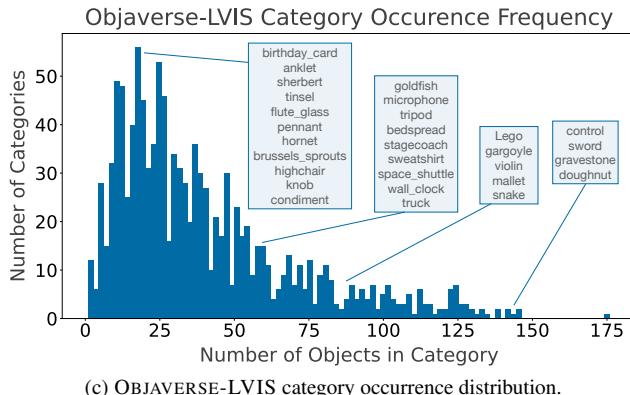
Model metadata. OBJAVERSE objects inherit a set of foundational annotations supplied by their creator when uploaded to Sketchfab. Figure 4 shows an example of the metadata available for each model. The metadata includes a name, assignments to a set of fixed categories, a set of unrestricted tags, and a natural language description.

OBJAVERSE-LVIS. While OBJAVERSE metadata contains a great deal of information about objects, Sketchfab’s existing categorization scheme covers only 18 categories, too coarse for most applications. Object names, categories, and tags provide multiple potential categorizations at varying levels of specificity and with some inherent noise. However, for many existing computer vision tasks, it is useful to assign objects to a single category drawn from a predetermined set of the right size and level of semantic granularity.

We choose the categories from the LVIS dataset [26] for categorizing a long-tail subset of objects in OBJAVERSE. We construct a 47K LVIS categorized object subset, called OBJAVERSE-LVIS, comprised of objects uniquely assigned to one of 1156 LVIS categories. We perform these assignments by first selecting 500 candidate objects per category using a combination of predictions from a CLIP classifi-



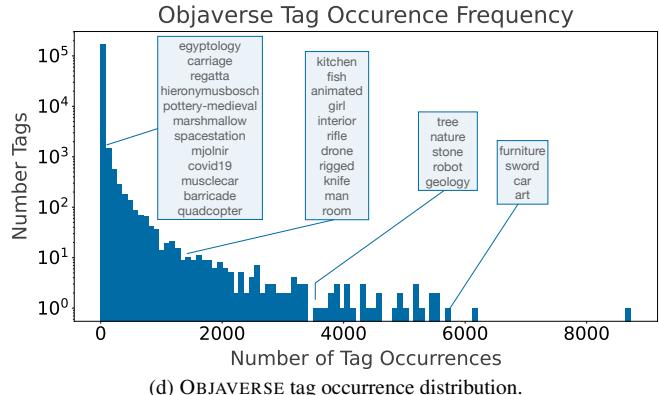
(a) Breakdown of high-level OBJAVERSE categories



(c) OBJAVERSE-LVIS category occurrence distribution



(b) Word cloud of OBJAVERSE metadata tags.



(d) OBJAVERSE tag occurrence distribution.

Figure 3. OBJAVERSE statistics. (a) All 18 high-level categories present in OBJAVERSE’s metadata with their corresponding number of occurrences. The relative share of most popular categories are evenly split, with a small number of less frequently categories. (b) A sample of several thousand popular object tags found in OBJAVERSE log-scaled by their frequency. (c) A histogram of fine-grained OBJAVERSE-LVIS categories with representative members from several bins highlighted. (d) A histogram of OBJAVERSE tags with representative members from several bins highlighted (note y-axis log scale). Tags from the low-occurrence side of the distribution correspond to unique objects that, taken individually, are rarely seen in the world. Frequently used tags like “furniture” and “car” reflect their real-world normalcy, but the high frequency of assets like “sword” diverge from their real-world counterparts.

cation model and candidates suggested by terms in their metadata. This combined pool contains objects visually resembling the target category (from the CLIP features of their thumbnail images) that might have missing metadata, as well as visually unusual instances of a category that are accurately named or tagged. These 250k candidate objects were then manually filtered and their assigned categories verified by crowdworkers. Since we only presented 500 object candidates per class, many popular categories, such as chair or car, have substantially more objects that could be included in OBJAVERSE-LVIS with future annotations.

Animated objects and rigged characters. OBJAVERSE includes 44K animated objects and over 63K objects self-categorized as characters. Examples of animations include fridge doors opening, animals running, and the hands on a clock moving. Rigged characters can be set up for animation and rendering, and may often come annotated with bone mappings. The vast scale of animations available in OBJAVERSE can support a wide range of research in temporal 3D learning, such as building text-based animation generative models [76], representing object changes over time

with NERFs [54, 59], and temporal self-supervised learning via. future frame prediction [30, 87].

Articulated objects. Decomposing 3D objects into parts has led to a flurry of research in the past few years, including work in learning robotic grasping policies [84, 86], 3D semantic segmentation [50], and shape generation [49]. Since many objects in OBJAVERSE were uploaded by artists, the objects often come separated into parts. Figure 5 shows an example, where a chair is separated by its backrest, wheels, and legs, among many smaller parts.

Exteriors. Photogrammetry and NERF advances have enabled the commercialization of capturing high-quality 3D objects of large exteriors by taking pictures [75, 85]. In OBJaverse, there are a large number of scanned buildings, cities, and stadiums. Figure 5 shows an example of a 3D object of NYC’s skyline captured through a scan.

OBJAVERSE-Interiors. There are 16K+ interior scenes in OBJAVERSE, including houses, classrooms, and offices. The scenes often have multiple floors, many types of rooms, and are densely populated with objects from human input. Objects in the scenes are separable into parts, which allows

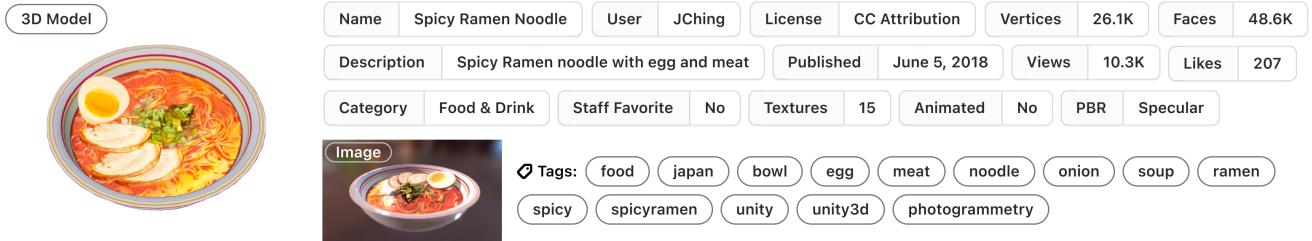


Figure 4. An example of metadata available for each object in OBJAVERSE. Each uploaded object has a 3D model, user-selected rendered thumbnail image, name, description, tags, category, and stats, among additional metadata.

them to be usable for interactive robotics, embodied AI, and scene synthesis. To put the scale of OBJAVERSE-Interiors in perspective, the number of scenes in OBJAVERSE-Interiors is significantly larger than the 400 or so existing hand-built interactive embodied AI scenes [23, 39, 42, 72].

Visual styles. Objects in the world can be constructed in many styles and often differ in style based on the time-period, geographic location, and artist’s style. OBJAVERSE objects cover a vast set of visual styles, including 3D scans, 3D modeled objects from virtually any platform, point clouds, and photo-realism via physically based rendering (PBR) [56]. Moreover, instances of objects often appear with many styles, which is critical for training and evaluating robust computer vision models [60]. Figure 5 shows examples of chairs in OBJAVERSE in many different styles, including Gothic, modern, Victorian, cartoon, and abstract.

Statistics. OBJAVERSE 1.0 includes 818K 3D objects, designed by 160K artists. There are >2.35M tags on the objects, with >170K of them being unique. We estimate that the objects have coverage for nearly 21K WordNet entities [48] (see appendix for details). Objects were uploaded between 2012 and 2022, with over 200K objects uploaded just in 2021. Figure 3 visualizes several statistics of the dataset, including the breakdown of objects into their

self-assigned Sketchfab categories, a word cloud over the tags, a frequency plot of the tags, and the number of objects in OBJAVERSE-LVIS categories.

4. Applications

In this section, we present 4 initial distinct applications of OBJAVERSE, including 3D generative modeling, instance segmentation with CP3D, open-vocabulary ObjectNav, and analyze robustness in computer vision models.

4.1. 3D Generative Modeling

3D generative modeling has shown much improvement recently with models such as GET3D [24] delivering impressive high quality results with rich geometric details. GET3D is trained to generate 3D textured meshes for a category and produces impressive 3D objects for categories like *Car*, *Chair*, and *Motorcycle* using data from ShapeNet [7]. OBJAVERSE contains 3D models for many diverse categories including tail categories which are not represented in other datasets. It also contains diverse and realistic object instances per category. This scale and diversity can be used to train large vocabulary and high quality 3D generative models. In this work, we showcase the potential of this data as follows. We choose three categories of ob-

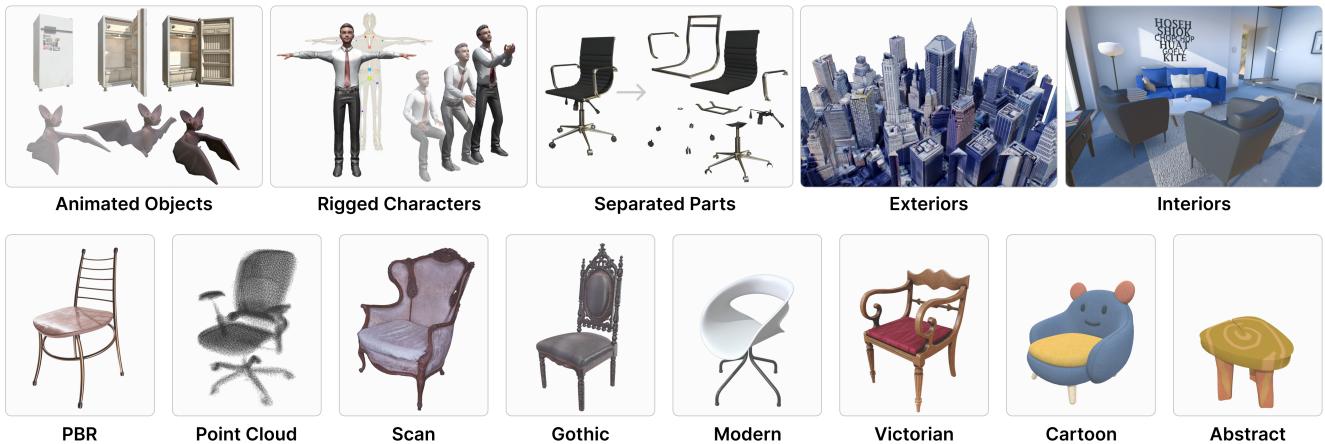


Figure 5. Highlights of the visual diversity of objects that appear in OBJAVERSE, including animated objects, rigged (body-part annotated) characters, models separable into parts, exterior environments, interior environments, and a wide range visual styles.

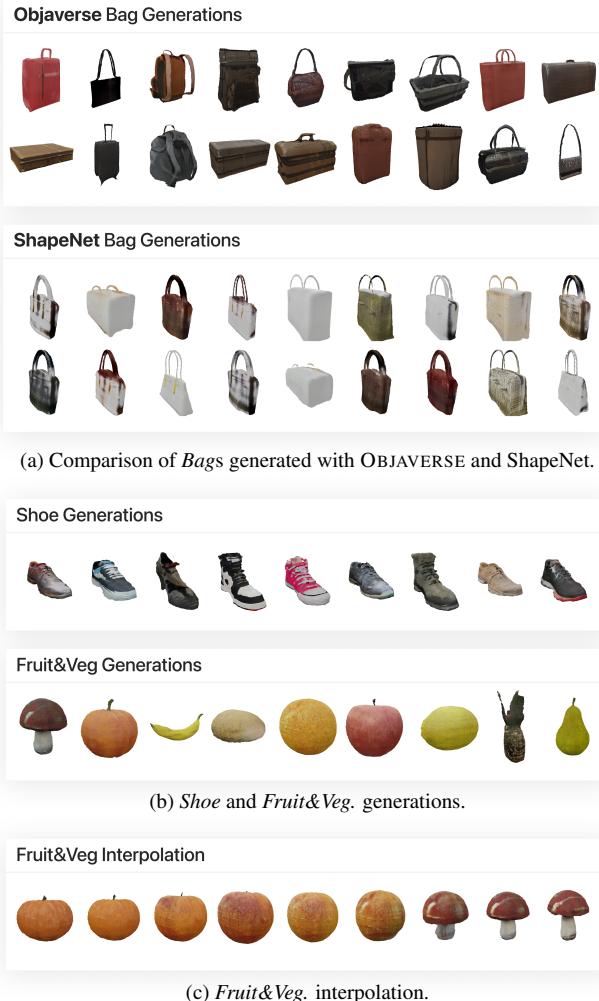


Figure 6. (a) Example GET3D *Bag* object generations using OBJAVERSE and ShapeNet models for training. (b) Additional *Shoe* and *Fruit&Veg* generations from OBJAVERSE models. (c) models generated when interpolating between two, randomly sampled, latent encodings with our trained *Fruit&Veg*. model; what appears to be a pumpkin smoothly transforms into a mushroom.

jects, *Shoe*, *Bag*, and *Fruit&Veg*, and subsample objects from OBJAVERSE to create three separate datasets containing, respectively, 143 shoes, 816 bags, and 571 fruits & vegetables (116 apples, 112 gourds, 92 mushrooms, 68 bananas, 52 oranges, 52 pears, 31 potatoes 24 lemons, and 24 pineapples). For comparison, we also train a GET3D model on the set of 83 bags from the ShapeNet dataset. Fig. 6 shows a collection of 3D objects generated by our trained GET3D models. Qualitatively, the 3D-meshes generated by the OBJAVERSE-trained models are high-quality and diverse, especially when compared to the generations from the ShapeNet-trained model. To quantify this observation, we asked crowdworkers to rate the diversity of *Bag* generations produced by the OBJAVERSE and ShapeNet trained models. When shown collections of nine randomly sampled

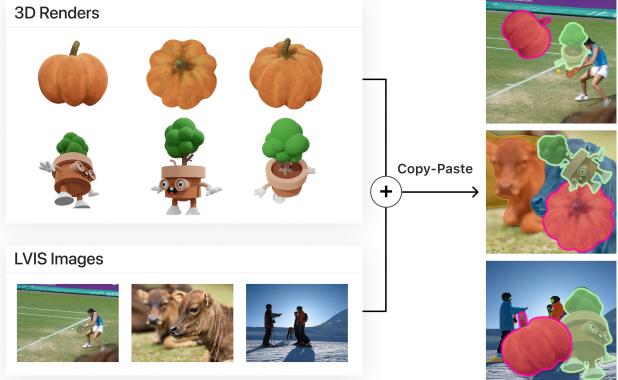


Figure 7. An illustration of 3DCP (3D copy-paste) for segmentation augmentation. We render 3D objects from multiple views and paste them over LVIS training images.

generations from both models, workers rated the collection generated from the OBJAVERSE trained model as more diverse in appearance 91% of the time.

Our fruits and vegetables, composed of 9 varieties produces perhaps the highest quality output, a promising signal that can inspire future work in text-to-3D generation.

4.2. Instance Segmentation with CP3D

A key advantage of using simulated data for computer vision is that it is much cheaper to obtain expert annotations. Annotated OBJAVERSE objects can be rendered into images, allowing them to serve as a rich source of additional data that can be used to enhance model performance on 2D computer vision tasks. As a proof-of-concept demonstrating the effectiveness of this approach, we use segmented data from OBJAVERSE objects as auxiliary labels for training models on the LVIS dataset for Large Scale Instance Segmentation [26]. The LVIS dataset contains instance segmentation masks for 1200 object categories that occur throughout a set of 164k images. Recognition is especially challenging in this task due to the long tail of the object category distribution in this dataset. LVIS categories only con-

Method	AP	APr	APc	APf
RFS [26]	23.7	13.3	23.0	29.0
EQLv2 [74]	25.5	17.7	24.3	30.2
LOCE [21]	26.6	18.5	26.2	30.7
NorCal with RFS [52]	25.2	19.3	24.2	29.0
Seesaw [78]	26.4	19.5	26.1	29.7
GOL [3]	27.7	21.4	27.7	30.4
GOL + 3DCP	28.3	21.8	28.3	31.1

Table 1. Comparison of our approach (GOL+3DCP) against SoTA Mask-RCNN ResNet-50 models on LVIS. We report results for APr, APc, and APf which measure AP for categories that are rare (appear in 1-10 images), common (appear in 11-100 images), and frequent (appear in >100 images), respectively



ProcTHOR

ProcTHOR + Objaverse

Figure 8. An existing ProcTHOR scene (left) and a semantically similar ProcTHOR generatable scene with OBJAVERSE objects (right).

tain an average 9 instances across the dataset, so training on simulated data is a promising approach for overcoming the challenges of learning in this low-sample regime.

Using the LVIS-annotated subset of OBJAVERSE, we introduce 3DCP: an enhancement to the simple, but effective, copy-and-paste technique of [25]. Figure 7 shows an example of the setup for 3DCP. Here, we render different views of 3D objects and paste them on-top of existing LVIS images. We render 5 distinct views of each object and cache them for use throughout training. During training, an image is selected for the copy-paste augmentation with 0.5 probability, and once selected, 1-3 images of randomly chosen LVIS-annotated OBJAVERSE objects are pasted onto the selected training image. The segmentation masks of the selected objects are added to the training image’s annotation as well. Object images and masks are randomly scaled and translated before being pasted. We use this strategy to finetune the pretrained ResNet-50 Mask-RCNN [27, 28] of [3]. As shown in Tab.1, simply finetuning this model for 24 epochs yields performance gains across several metrics.

4.3. Open-Vocabulary ObjectNav

In this section, we introduce open-vocabulary ObjectNav, a new task propelled by the vast diversity of objects that appear in OBJAVERSE. Here, an agent is placed at a random starting location inside of a home and tasked to navigate to a target object provided from a text description (*e.g.* “*Raspberry Pi Pico*”). To facilitate this task, we procedurally generate 10K new homes in ProcTHOR [17] fully populated with objects from OBJAVERSE-LVIS. Until now, ObjectNav tasks have focused on training agents to navigate to 20 or so target objects provided their category label [16, 17, 63], and existing interactive embodied AI simulations, including ProcTHOR, only include around 2K total objects across around 100 object types [17, 42, 72]. In this work, we take a large step to massively scale the number of target objects used in ObjectNav (**20 → OpenVocab**), the number of objects available in simulation (**2K → 36K**), and the number of object types of the objects (**100 → 1.1K**).

Object placement. To make the placement of objects in the houses more natural, we use the OBJAVERSE-LVIS subset and annotate placement constraints for each object

category. Specifically, we annotate if objects of a given category typically appears on the floor, on-top of a surface, or on a wall. If instances of the object category may appear on the floor, we also annotate whether it may appear in the middle of the scene (*e.g.* a clutter object like a basketball) or on the edge of the scene (*e.g.* a toilet or a fridge). For objects placed on the floor, we also automatically detect flat regions on top of the object’s mesh to place surface object types. The annotations are used by ProcTHOR for sampling objects to place in a scene. We also filter out OBJAVERSE-LVIS objects that do not appear inside of homes, such as a jet plane. Structural objects, like doors and windows, are inherited from ProcTHOR as they would require additional cleanup.

Object size correction. Objects in Sketchfab may be uploaded at unnatural scales (*e.g.* a plant being as large as a tower). We therefore scale the objects to be of a reasonable size for them to look natural in a house. Here, for each object category, we annotate the maximum bounding box dimension length that every instance of the object category should be scaled to. For example, we annotate the maximum bounding box dimension for bookcase to be 2 meters and fork to be 0.18 meters. If a 3D modeled bookcase then has a bounding box of $20m \times 6m \times 3m$, we shrink each side by a factor of $\max(20, 6, 3)/2 = 5$.

Preprocessing for AI2-THOR. We add support to AI2-THOR for loading objects on the fly at runtime. Previously, all objects had to be stored in a Unity build, but such an approach is impractical when working with orders of magnitude more object data. For each object, we compress it with Blender [13] by joining all of its meshes together, decimate the joined mesh such that it has at most 5K vertices, and bake all the UV texture maps into a single texture map. We then generate colliders using V-HACD [46] to support rigid-body interactions.

Approach. Given procedural houses populated with OBJAVERSE-LVIS, the task is to navigate to the proximity of a chosen target object and invoke a task-completion action when the target object is in sight, given an open-vocabulary description formed with the template “a {name} {category}”. The name is the object name given by its creator, which is often descriptive. We filter

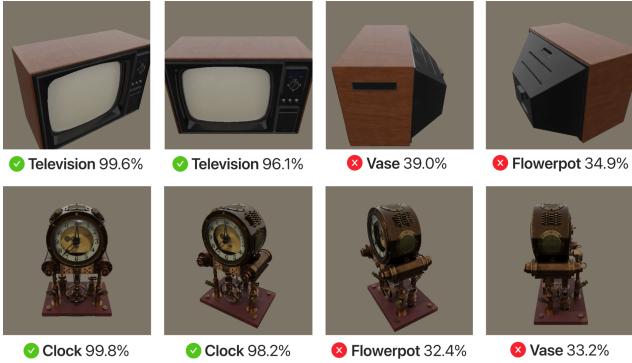


Figure 9. Examples of objects rendered from random orientations and their 0-shot classification categories with the CLIP ViT-B/32.

each by whether it is detected as being written in English by a language detector [33,34], and fall back to a class-only description for non-English name. Examples of the possible expressions include “*a victorian-monobike motorcycle*”, “*a unicorn pony*”, or “*a dino ghost lizard*”. The agent, similar to the ones in [36], observes an RGB egocentric view of the environment, pre-processed by the visual branch of a frozen ResNet-50 CLIP model [60] – the target description is pre-processed by the corresponding text branch. We train the agent with DD-PPO [81] and evaluate on houses with floor plans, objects, and descriptions unseen in training. We use the AllenAct [80] framework to train our agent. Our trained agent achieves a success rate of 19.9%, for a random policy success of 5.1%. For more details about the experiment refer to the appendix.

4.4. Analyzing Robustness

A persistent bias present in many image datasets, *e.g.* ImageNet [65], is that the subjects of interest are generally photographed from a forward-facing, canonical, orientation. When, for example, taking a photograph of a television, few would choose to take this photograph crouched on the floor behind the television. This impact of this bias was studied by Alcorn *et al.* [2] who find that modern computer vision systems are highly susceptible to deviations from canonical poses. This is more than a theoretical problem: computer vision systems deployed in the real world will frequently encounter objects in non-canonical orientations and in many applications, *e.g.* autonomous driving, it will be safety critical that they behave well.

Given the above, we adopt the experimental design of Alcorn *et al.* and design, using OBJAVERSE assets, a benchmark for evaluating the robustness of state-of-the-art computer vision classification models to orientation shifts. In particular, for each object in our OBJAVERSE-LVIS subset, we render 12 images of the object from random orientations rendered upon a background with RGB values equalling the mean RGB values from ImageNet; see Fig. 9 for examples. This ability to, at scale, render objects from random view-

Model	Random Rotation		Any Rotation		$\Delta\text{Top-1}$
	Top-1	Top-5	Top-1	Top-5	
OpenAI-400M [60]					
RN50	21.4%	45.0%	43.9%	70.8%	22.5%
ViT-L/14	29.1%	54.5%	52.3%	77.2%	23.2%
LAION-400M [67]					
ViT-B/32	24.1%	48.5%	46.9%	74.2%	22.8%
ViT-L/14	30.6%	56.8%	50.5%	77.0%	19.9%
LAION-2B [66]					
ViT-B/32	27.0%	51.8%	50.3%	76.1%	23.3%
ViT-L/14	32.9%	59.2%	52.1%	78.0%	19.2%
ViT-H/14	32.3%	58.8%	50.1%	77.3%	17.8%

Table 2. Evaluating 0-shot CLIP classification models on our rotational robustness benchmark. $\Delta\text{Top-1}$ denotes the difference between *Top-1 Any Rotation* and *Top-1 Random Rotation*. Models are strongly overfit to standard views of objects.

points is a practical impossibility in the real world but is made trivial when using 3D assets. We then evaluate several modern open-domain image-classification networks (constrained to the $\approx 1,200$ LVIS categories) on these images and report 4 metrics for each model. These metrics include:

- *Top-1 Random Rotation* – the frequency with which a model correctly classifies an image as belonging to the respective LVIS category.
- *Top-1 Any Rotation* – the frequency with which a model classifies an image correctly from at least one of the 12 random orientations.

This second metric is diagnostic and serves to represent a model’s performance when shown an object from a canonical pose. We also have *Top-5* variants of the above metric where the correct category need only be in the top 5 predictions from the model. We report our results in Tab. 2 in which we evaluate a variety of performant pretrained models. Comparing the gap in performance between the *Top-k Random Rotation* and *Top-k Any Rotation* metrics we find that model performance dramatically degrades when viewing objects from unusual orientations.

5. Conclusion

We present OBJAVERSE, a next-generation 3D asset library containing 818K high-quality, diverse, 3D models with paired text descriptions, titles, and tags. As a small glimpse of the potential uses of OBJAVERSE, we present four experimental studies showing how OBJAVERSE can be used to power (1) generative 3D models with clear future applications to text-to-3D generation, (2) improvements to classical computer vision tasks such as instance segmentation, (3) the creation of novel embodied AI tasks like Open Vocabulary Object Navigation, and (4) quantifying the rotational robustness of vision models on renderings of objects. We hope to see OBJAVERSE enable a new universe of new applications for computer vision.

References

- [1] Panos Achlioptas, Judy Fan, Robert Hawkins, Noah Goodman, and Leonidas J Guibas. Shapelog: Learning language for shape differentiation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8938–8947, 2019. 3
- [2] Michael A. Alcorn, Qi Li, Zhitao Gong, Chengfei Wang, Long Mai, Wei-Shinn Ku, and Anh Nguyen. Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 4845–4854. Computer Vision Foundation / IEEE, 2019. 8
- [3] Konstantinos Panagiotis Alexandridis, Jiankang Deng, Anh Nguyen, and Shan Luo. Long-tailed instance segmentation using gumbel optimized loss. *arXiv preprint arXiv:2207.10936*, 2022. 6, 7, 13
- [4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1
- [5] Berk Calli, Aaron Walsman, Arjun Singh, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols. *arXiv preprint arXiv:1502.03143*, 2015. 3
- [6] Carnegie Mellon University. Locobot: an open source low cost robot. <http://www.locobot.org/>. 13
- [7] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 3, 5
- [8] Kevin Chen, Christopher B Choy, Manolis Savva, Angel X Chang, Thomas Funkhouser, and Silvio Savarese. Text2shape: Generating shapes from natural language by learning joint embeddings. In *Asian conference on computer vision*, pages 100–116. Springer, 2018. 3
- [9] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022. 2
- [10] Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. Unifying vision-and-language tasks via text generation. In *International Conference on Machine Learning*, pages 1931–1942. PMLR, 2021. 2
- [11] Sungjoon Choi, Qian-Yi Zhou, Stephen Miller, and Vladlen Koltun. A large dataset of object scans. *arXiv preprint arXiv:1602.02481*, 2016. 3
- [12] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, et al. Abo: Dataset and benchmarks for real-world 3d object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21126–21136, 2022. 3
- [13] Blender Online Community. Blender - a 3d modelling and rendering package. <http://www.blender.org>, 2018. 7
- [14] Camille Couprie, Clément Farabet, Laurent Najman, and Yann LeCun. Indoor semantic segmentation using depth information. *arXiv preprint arXiv:1301.3572*, 2013. 3
- [15] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 3
- [16] Matt Deitke, Dhruv Batra, Yonatan Bisk, Tommaso Campari, Angel X Chang, Devendra Singh Chaplot, Changan Chen, Claudia Pérez D’Arpino, Kiana Ehsani, Ali Farhadi, et al. Retrospectives on the embodied ai workshop. *arXiv preprint arXiv:2210.06849*, 2022. 7
- [17] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Jordi Salvador, Kiana Ehsani, Winson Han, Eric Kolve, Ali Farhadi, Aniruddha Kembhavi, et al. Procthor: Large-scale embodied ai using procedural generation. *Conference on Neural Information Processing Systems*, 2022. 2, 7, 13
- [18] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2
- [19] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. *arXiv preprint arXiv:2204.11918*, 2022. 3
- [20] Christiane Fellbaum. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer, 2010. 14
- [21] Chengjian Feng, Yujie Zhong, and Weilin Huang. Exploring classification equilibrium in long-tailed object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3417–3426, 2021. 6
- [22] Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. *International Journal of Computer Vision*, 129(12):3313–3337, 2021. 3
- [23] Chuang Gan, Jeremy Schwartz, Seth Alter, Martin Schrimpf, James Traer, Julian De Freitas, Jonas Kubilius, Abhishek Bhandwaldar, Nick Haber, Megumi Sano, et al. Threed-world: A platform for interactive multi-modal physical simulation. *arXiv preprint arXiv:2007.04954*, 2020. 5
- [24] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. Get3d: A generative model of high quality 3d textured shapes learned from images. In *Advances In Neural Information Processing Systems*, 2022. 2, 5
- [25] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21126–21136, 2022. 3

- on Computer Vision and Pattern Recognition*, pages 2918–2928, 2021. 7
- [26] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5356–5364, 2019. 3, 6
- [27] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 7, 13
- [28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 7, 13
- [29] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Open-CLIP, July 2021. 2
- [30] Allan Jabri, Andrew Owens, and Alexei Efros. Space-time correspondence as a contrastive random walk. *Advances in neural information processing systems*, 33:19545–19560, 2020. 4
- [31] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 867–876, 2022. 3
- [32] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021. 2
- [33] Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hervé Jégou, and Tomas Mikolov. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016. 8
- [34] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016. 8
- [35] Alexander Kasper, Zhixing Xue, and Rüdiger Dillmann. The kit object models database: An object model database for object recognition, localization and manipulation in service robotics. *The International Journal of Robotics Research*, 31(8):927–934, 2012. 3
- [36] Apoorv Khandelwal, Luca Weihs, Roozbeh Mottaghi, and Aniruddha Kembhavi. Simple but effective: Clip embeddings for embodied ai. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14809–14818, 2022. 8, 13
- [37] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv*, 2014. 13
- [38] Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. Abc: A big cad model dataset for geometric deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2918–2928, 2021. 7
- [39] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, Aniruddha Kembhavi, Abhinav Kumar Gupta, and Ali Farhadi. Ai2-thor: An interactive 3d environment for visual ai. *arXiv e-prints*, pages arXiv-1712, 2017. 2, 5
- [40] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Mallocci, Alexander Kolesnikov, et al. The open images dataset v4. *International Journal of Computer Vision*, 128(7):1956–1981, 2020. 2
- [41] Marc Levoy, Kari Pulli, Brian Curless, Szymon Rusinkiewicz, David Koller, Lucas Pereira, Matt Ginzton, Sean Anderson, James Davis, Jeremy Ginsberg, et al. The digital michelangelo project: 3d scanning of large statues. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 131–144, 2000. 3
- [42] Chengshu Li, Fei Xia, Roberto Martín-Martín, Michael Lingelbach, Sanjana Srivastava, Bokui Shen, Kent Vainio, Cem Gokmen, Gokul Dharan, Tanish Jain, et al. igibson 2.0: Object-centric simulation for robot learning of everyday household tasks. *arXiv preprint arXiv:2108.03272*, 2021. 2, 5, 7
- [43] Joseph J Lim, Hamed Pirsiavash, and Antonio Torralba. Parsing ikea objects: Fine pose estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 2992–2999, 2013. 3
- [44] 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 *European conference on computer vision*, pages 740–755. Springer, 2014. 2
- [45] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019. 1
- [46] Khaled Mamou, E Lengyel, and A Peters. Volumetric hierarchical approximate convex decomposition. In *Game Engine Gems 3*, pages 141–158. AK Peters, 2016. 7
- [47] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2mesh: Text-driven neural stylization for meshes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13492–13502, 2022. 3
- [48] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 5
- [49] Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy Mitra, and Leonidas J Guibas. Structurenet: Hierarchical graph networks for 3d shape generation. *arXiv preprint arXiv:1908.00575*, 2019. 4
- [50] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9601–9611, 2019. 3

- conference on computer vision and pattern recognition*, pages 909–918, 2019. 4
- [51] D. Morrison, P. Corke, and J. Leitner. Egad! an evolved grasping analysis dataset for diversity and reproducibility in robotic manipulation. *IEEE Robotics and Automation Letters*, 5(3):4368–4375, 2020. 3
- [52] Tai-Yu Pan, Cheng Zhang, Yandong Li, Hexiang Hu, Dong Xuan, Soravit Changpinyo, Boqing Gong, and Wei-Lun Chao. On model calibration for long-tailed object detection and instance segmentation. *Advances in Neural Information Processing Systems*, 34:2529–2542, 2021. 6
- [53] Keunhong Park, Konstantinos Rematas, Ali Farhadi, and Steven M Seitz. Photoshape: Photorealistic materials for large-scale shape collections. *arXiv preprint arXiv:1809.09761*, 2018. 3
- [54] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021. 4
- [55] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014. 14
- [56] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016. 5
- [57] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 3
- [58] Sarah Pratt, Rosanne Liu, and Ali Farhadi. What does a platypus look like? generating customized prompts for zero-shot image classification. *arXiv preprint arXiv:2209.03320*, 2022. 15
- [59] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 4
- [60] 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 *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1, 2, 5, 8
- [61] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2
- [62] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 2
- [63] Ram Ramrakhyta, Eric Undersander, Dhruv Batra, and Abhishek Das. Habitat-web: Learning embodied object-search strategies from human demonstrations at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5173–5183, 2022. 2, 7
- [64] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1, 2
- [65] 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. *International journal of computer vision*, 115(3):211–252, 2015. 2, 8
- [66] 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. *arXiv preprint arXiv:2210.08402*, 2022. 1, 2, 8
- [67] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 1, 2, 8
- [68] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 1, 2
- [69] Arjun Singh, James Sha, Karthik S. Narayan, Tudor Achim, and P. Abbeel. Bigbird: A large-scale 3d database of object instances. *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 509–516, 2014. 3
- [70] Krishna Srinivasan, Karthik Raman, Jiecao Chen, Michael Bendersky, and Marc Najork. Wit: Wikipedia-based image text dataset for multimodal multilingual machine learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2443–2449, 2021. 1, 2
- [71] Xingyuan Sun, Jiajun Wu, Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Tianfan Xue, Joshua B Tenenbaum, and William T Freeman. Pix3d: Dataset and methods for single-image 3d shape modeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2974–2983, 2018. 3
- [72] Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijsmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. *Advances in Neural Information Processing Systems*, 34:251–266, 2021. 2, 5, 7
- [73] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 2
- [74] Jingru Tan, Xin Lu, Gang Zhang, Changqing Yin, and Quanquan Li. Equalization loss v2: A new gradient balance approach for long-tailed object detection. In *Proceedings of*

- the IEEE/CVF conference on computer vision and pattern recognition*, pages 1685–1694, 2021. 6
- [75] Matthew Tancik, Vincent Casser, Xincheng Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. Block-nerf: Scalable large scene neural view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8248–8258, 2022. 4
- [76] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Amit H Bermano, and Daniel Cohen-Or. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022. 4
- [77] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016. 2
- [78] Jiaqi Wang, Wenwei Zhang, Yuhang Zang, Yuhang Cao, Jiangmiao Pang, Tao Gong, Kai Chen, Ziwei Liu, Chen Change Loy, and Dahua Lin. Seesaw loss for long-tailed instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9695–9704, 2021. 6
- [79] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singh, Subhajit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022. 2
- [80] Luca Weihs, Jordi Salvador, Klemen Kotar, Unnat Jain, Kuo-Hao Zeng, Roozbeh Mottaghi, and Aniruddha Kembhavi. Allenact: A framework for embodied ai research. *arXiv preprint arXiv:2008.12760*, 2020. 8
- [81] Erik Wijmans, Abhishek Kadian, Ari S. Morcos, Stefan Lee, Irfan Essa, Devi Parikh, Manolis Savva, and Dhruv Batra. Dd-ppo: Learning near-perfect pointgoal navigators from 2.5 billion frames. In *ICLR*, 2020. 8, 13
- [82] Rundi Wu, Chang Xiao, and Changxi Zheng. Deepcad: A deep generative network for computer-aided design models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6772–6782, 2021. 3
- [83] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Lingshuang Zhang, Xiaou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015. 3
- [84] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11097–11107, 2020. 4
- [85] Yuanbo Xiangli, Lining Xu, Xingang Pan, Nanxuan Zhao, Anyi Rao, Christian Theobalt, Bo Dai, and Dahua Lin. Bungeenerf: Progressive neural radiance field for extreme multi-scale scene rendering. In *The European Conference on Computer Vision (ECCV)*, 2022. 4
- [86] Zhenjia Xu, Beichun Qi, Shubham Agrawal, and Shuran Song. Adagrasp: Learning an adaptive gripper-aware grasping policy. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4620–4626. IEEE, 2021. 4
- [87] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. Merlot reserve: Neural script knowledge through vision and language and sound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16375–16387, 2022. 1, 4

A. Instance Segmentation with CP3D

Model. We use the Mask-RCNN [27] model of [3] with a ResNet-50 backbone [28]; no additional changes to their model are made. Instead of a softmax activation, the model uses a Gumbel activation, given by the formula $\eta(q) = \exp(-\exp(-q))$, to transform logits into probabilities. More details about the model and activation can be found in [3].

Training. We take the pretrained ResNet-50 Mask-RCNN checkpoint of [3] and finetune the model for 24 epochs with the CP3D augmentation integrated into the training pipeline. We use a batch size of 64 and a learning rate of 0.002.

Additional Results Here we report detection metrics in addition to the segmentation results reported in the paper in Table 1. Notably, we see an impressive gain of two points on AP for rare categories.

Method	AP	APr	APc	APf
GOL [3]	27.5	19.8	27.2	31.2
GOL + 3DCP	28.9	21.8	28.7	32.2

Table 3. **Detection results for bounding box AP category metrics.** APr, APc, and APf measure AP for categories that are rare (appear in 1-10 images), common (appear in 11-100 images), and frequent (appear in >100 images), respectively.

B. Open-Vocabulary ObjectNav

Model. The agent’s embodiment is a simulated LoCoBot [6]. The action space consists of six actions: MOVEAHEAD, ROTATELEFT, ROTATERIGHT, END, LOOKUP, and LOOKDOWN. Given the excellent exploration capabilities of EmbCLIP [17, 36], we opt to keep the same overall architecture, just replacing the learned embedding for target types in prior work by a linear projection of the text branch output of CLIP for the target description, as shown in Fig. 10. Additionally, in order to provide more information about the target and the current visual input, we increase the respective internal representations for each modality from the original 32-D to 256-D. Note that our model does not employ the alternative zero-shot design described in [36], where the target description is not observed by the agent’s RNN. Given the scale of OBJaverse-LVIS, we can train agents with good generalization following a more standard design.

Training. For training, we use ProcTHOR to procedurally generate 10,080 houses. Each house has up to three rooms, entirely populated with OBJaverse-LVIS assets except for

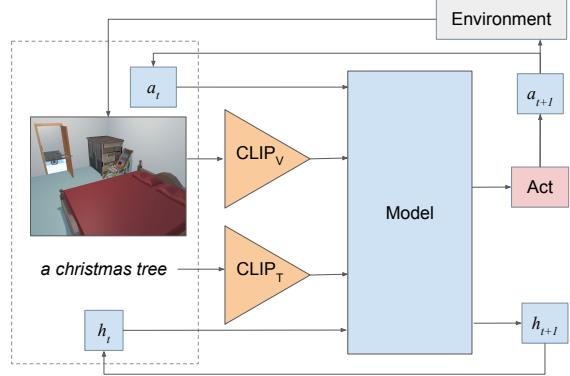


Figure 10. **Open-Vocabulary ObjNav Model overview.** The ObjectNav model (employing an RNN) uses the high-level architecture illustrated here, where it receives features from the visual and target object description encoders, besides previous hidden units and actions as input, and outputs the next action.

Hyperparameter	Value
Discount factor (γ)	0.99
GAE parameter (λ)	0.95
Value loss coefficient	0.5
Entropy loss coefficient	0.01
Clip parameter (ϵ)	0.1
Rollout horizons	32, 64, 128
Rollout timesteps	20
Rollouts per minibatch	1
Learning rate	$3 \cdot 10^{-4}$
Optimizer	Adam [37]
Gradient clip norm	0.5

Table 4. **Training hyperparameters for Open-Vocabulary ObjectNav.**

structural components like doors and windows, which are inherited from ProcTHOR [17]. We sample targets corresponding to LVIS categories for which a single instance is present in the scene, resulting in a total of 9,421 unique assets corresponding to 262 categories targeted during training. Training uses DD-PPO [81] and is distributed across 28 GPUs on 7 AWS g4dn.12xlarge machines, with each GPU hosting 360 houses and the subset of OBJaverse-LVIS assets populating them. The training hyperparameters, identical to the ones in [17], and the 262 training target categories are listed in Table 4 and Table 5, respectively.

Testing. For testing, we sample 150 episodes for each of 30 target categories, which are a subset of the training target categories. The resulting 4,500 episodes are sampled from 151 procedural houses not seen during training. The

Bible, Christmas tree, Rollerblade, alligator, ambulance, amplifier, arctic (type of shoe), armor, banner, barbell, barrel, barrow, baseball bat, basketball, bat (animal), bath mat, beachball, bear, bed, beetle, bench, beret, bicycle, binder, binoculars, bird, blackberry, bookcase, boot, bottle, bowling ball, bullhorn, bunk bed, bus (vehicle), butterfly, cab (taxi), cabinet, canoe, cape, car (automobile), card, cardigan, carnation, cart, cassette, cat, chair, chaise longue, chicken (animal), clothes hamper, coatrack, coffee table, cone, convertible (automobile), cornice, cow, cowboy hat, crab (animal), crate, crossbar, cube, cylinder, deck chair, deer, desk, dinghy, dirt bike, dog, dollhouse, doormat, dove, drawer, dresser, duckling, dumbbell, dumpster, easel, elephant, elk, fan, ferret, file cabinet, fireplace, fireplug, fishing rod, flag, flagpole, flamingo, flip-flop (sandal), flipper (footwear), foal, football (American), footstool, forklift, frog, futon, garbage, gargoyle, giant panda, giraffe, golf club, golfcart, gondola (boat), goose, gorilla, gravestone, grill, grizzly, grocery bag, guitar, handcart, hat, heater, hockey stick, hog, horse, horse carriage, jeep, kayak, keg, kennel, kitchen table, kitten, knee pad, ladder, ladybug, lamb (animal), lamp, lamppost, lawn mower, legging (clothing), lion, lizard, locker, log, loveseat, machine gun, mailbox (at home), manhole, mascot, mast, milk can, minivan, monkey, mop, motor, motor scooter, motor vehicle, motorcycle, mushroom, music stool, nut, ostrich, owl, pajamas, parasail (sports), parka, penguin, person, pet, pew (church bench), piano, pickup truck, pinecone, ping-pong ball, playpen, pole, polo shirt, pony, pool table, power shovel, propeller, pug-dog, pumpkin, rabbit, radiator, raincoat, ram (animal), rat, recliner, refrigerator, rhinoceros, rifle, road map, rocking chair, router (computer equipment), runner (carpet), saddle (on an animal), saddle blanket, saddlebag, sandal (type of shoe), scarecrow, scarf, sculpture, seabird, shark, shepherd dog, shield, shirt, shoe, sink, skateboard, ski parka, skullcap, snake, snowmobile, soccer ball, sock, sofa, sofa bed, solar array, sparkler (fireworks), speaker (stereo equipment), spear, spider, sportswear, statue (sculpture), step stool, stepladder, stool, subwoofer, sugarcane (plant), suit (clothing), suitcase, sunhat, surfboard, sweat pants, sweater, swimsuit, table, tape measure, tarp, telephone pole, television camera, tennis ball, tennis racket, tights (clothing), toolbox, tote bag, towel, trailer truck, trampoline, trash can, tricycle, trousers, truck, trunk, turtle, tux, underdrawers, vacuum cleaner, vending machine, vest, wagon wheel, water ski, watering can, wet suit, wheel, window box (for plants), wok, wolf, and wooden leg.

Table 5. Training target types for Open-Vocabulary ObjectNav.

30 testing target categories are listed in Table 6. For the results provided in the main paper, the agent is trained for just 18 million simulation steps, but the resulting policy already shows reasonable performance given the variety of targets and scenes. Improved performance can be achieved with extended training (e.g., after approx. 460 million steps, the success rate is 33.0%).

C. Composition

Human subjects data. A portion of the data included in OBJAVERSE is generated by human subjects (*i.e.* crowdworkers recruited through Amazon’s Mechanical Turk platform) as outlined in Section 3 and detailed below. The collection process has been reviewed and approved for release by an Institutional Review Board.

Data collection interfaces. Human annotators were used to provide the category labels for OBJAVERSE-LVIS as described in Section 3. This task was accomplished by first creating sets of 500 candidate objects for each LVIS category. These candidate sets included objects visually resembling the target category (as ranked by the CLIP features of

their thumbnail images), as well as instances whose metadata contained terms with a high similarity to the target category (as ranked by their GloVe vector similarity [55]). Candidate objects were shown to crowdworkers nine at a time, and they were asked to mark objects that were members of the category, as shown in Figure 11 a. In addition to the visual reference for each object, annotators also had access to the object’s name and were encouraged to use this when helpful. Human annotators were also used to rate the relative diversity of 3D objects generated by models trained using OBJAVERSE and ShapeNet. The user interface and instructions for this task are shown in Figure 11 b. Two sets of nine objects generated by each model were shown with random left-right orientations, and workers were asked to choose the set exhibiting the greater variety in appearance.

D. Estimating Coverage

We use OpenAI’s CLIP ViT-B/32 model to estimate the categorical coverage of the objects in OBJAVERSE. Specifically, for each object, we compute the CLIP image embedding from the thumbnail and the cosine similarity between an text embedding of each WordNet entity [20]. The entity

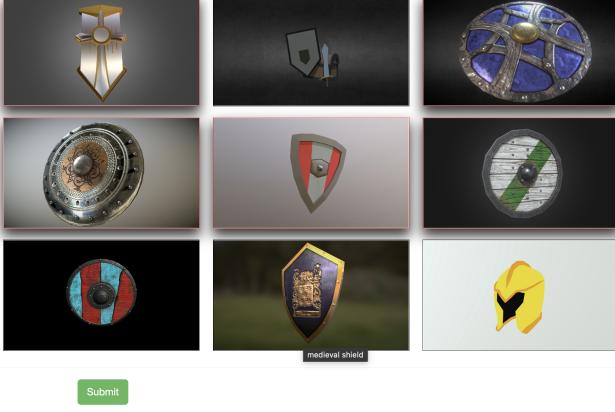
Christmas tree, bed, bench, blackberry, chair, chicken (animal), dog, easel, elk, fireplug, forklift, garbage, gargoyle, guitar, mascot, motor, penguin, pony, pool table, radiator, rifle, scarf, sock, speaker (stereo equipment), sportswear, sweat pants, trash can, trunk, wet suit, and wheel.

Table 6. Testing target types for Open-Vocabulary ObjectNav.

Select every image below that shows a target object type by clicking on the image.
When you've clicked on and selected an image, its border will change to red to indicate this.
Some sets of images might all contain this object, and other sets might contain none.
If you want to change your selections, just refresh the page and reselect the correct images.
Only select images you're sure show the kind of object you're looking for.
Don't select an object if the target category is only one piece of a larger whole (for example, don't select an entire car if the target is tire.)
If you're having trouble telling what an object is, you can hover over the image with your mouse and it will show the name of the object. This is often helpful, but can sometimes be missing or hard to interpret.

This is the type of object to select for this HIT:

shield

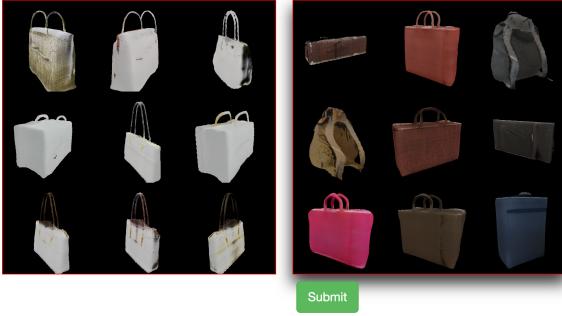


(a) Screenshot of OBJAVERSE-LVIS categorization task.

- There are two images showing groups of nine different versions of a given object type.
- We would like you to tell us which of the two groups has the greater variety of this object.
- The more diverse group could contain different types of the object, a greater variety of colors, materials, or other differences in appearance.
- You can select an image by clicking on it, and then clicking the submit button.
- You can change your mind by clicking on the other image before submitting.

Object type:

bag



(b) Screenshot of relative diversity rating task.

Figure 11. Data collection interfaces.

with the maximum cosine similarity is then assigned as the object's entity. The WordNet entities are textually encoded in the form, “a {entity} is a {definition}”, which is loosely inspired by CuPL [58]. For instance, we might have “a *bat*

is a nocturnal mouselike mammal with forelimbs modified to form membranous wings and anatomical adaptations for echolocation by which they navigate” or “a *bat* is a club used for hitting a ball in various games”. Computing the nearest WordNet entity for each object gave us an estimated coverage of 20.8K entities.