Habitat Synthetic Scenes Dataset (HSSD-200):
An Analysis of 3D Scene Scale and Realism Tradeoffs for ObjectGoal Navigation

Mukul Khanna[1]*, **Yongsen Mao**[2]*, **Hanxiao Jiang**[2] , **Sanjay Haresh**[2] , **Brennan Shacklett**[3] ,
Dhruv Batra[1,4] , **Alexander Clegg**[4] , **Eric Undersander**[4] , **Angel X. Chang**[2] , **Manolis Savva**[2]

[1]Georgia Tech , [2]Simon Fraser University , [3]Stanford University , [4]Meta AI

3dlg-hcvc.github.io/hssd

Figure 1. **Left**: we contribute the Habitat Synthetic Scenes Dataset (HSSD-200), a new dataset of high-quality, human-authored synthetic 3D scenes. **Right**: zero-shot ObjectNav performance on HM3DSem [41] for agents pretained on synthetic 3D scene datasets of different scale and quality. Through a systematic analysis of scene dataset scale and realism our experiments show that the benefit of dataset scale saturates quickly, and scene realism and quality become the bottleneck for improved ObjectNav agent generalization to realistic scenes. Concretely, we find that agents trained on 122 scenes from HSSD outperform agents trained on two orders of magnitude more scenes from the ProcTHOR [14] dataset (19.2 vs 12.5 success rate).

Abstract

We contribute the Habitat Synthetic Scenes Dataset (HSSD-200), a dataset of 211 high-quality 3D scenes, and use it to test navigation agent generalization to realistic 3D environments. Our dataset represents real interiors and contains a diverse set of 18,656 models of real-world objects. We investigate the impact of synthetic 3D scene dataset scale and realism on the task of training embodied agents to find and navigate to objects (ObjectGoal navigation). By comparing to synthetic 3D scene datasets from prior work, we find that scale helps in generalization, but the benefits quickly saturate, making visual fidelity and correlation to real-world scenes more important. Our experiments show that agents trained on our smaller-scale dataset can outperform agents trained on much larger datasets. Surprisingly, we observe that agents trained on just 122 scenes from our dataset outperform agents trained on 10,000 scenes from the ProcTHOR-10K dataset in terms of zero-shot generalization in real-world scanned environments.

1. Introduction

Recent years have brought considerable progress in embodied AI agents that navigate in realistic scenes, follow language instructions [2], find and rearrange objects [3, 9, 33, 36], and perform other tasks involving embodied sensing, planning, and acting [10, 11, 26]. This progress is supported by simulation platforms that enable systematic, safe, and scalable training and evaluation of embodied AI agents before deployment to the real world [18, 20, 32, 34, 35].

Much of the success of simulation for embodied AI relies on 3D scene datasets that mimic the real world. To this end, the community has leveraged 3D reconstructions and synthetic datasets composed of arrangements of human-designed 3D objects. Though reconstruction datasets [7, 29, 39, 41] capture the diversity and complexity of real-world arrangements, the reconstructed scenes are often noisy with missing geometry for thin structures or shiny surfaces. Other prevalent artifacts include "holes" on walls and other surfaces, as well as partially reconstructed object, which can adversely impact the training of agents and lead

Figure 2. **Example scenes from HSSD-200.** Our dataset provides high-quality 3D interiors that are fully human-authored. The scenes are densely annotated with object semantic information and assets are prepared to enable performant embodied AI experiments. This dataset will be open-sourced and distributed under a permissive academic research license (free of cost).

to overt specialization tailored to artifacts. In addition, the acquisition and annotation of reconstructions is a significant undertaking and is hard to scale. Furthermore, 3D reconstructions are "monolithic" scene representations and do not easily allow manipulations such as addition, removal, or state changes of constituent objects (e.g., opening drawers). Such manipulations are critical in tasks that require interaction with the environment [3]. This has led to a recent trend in embodied AI to lean more on synthetic 3D scenes which represent real-world environments through composition of human-authored 3D objects [14, 22, 33].

Despite the ubiquitous use of synthetic 3D scene datasets in embodied AI experiments, there has been no systematic analysis of the tradeoffs between dataset scale (number of scenes and total scene physical size) and dataset realism (visual fidelity and correlation to real-world statistics). Prior work has largely treated extant datasets as "black boxes" for training and evaluation, even in settings where generalization to real-world setups is important. Moreover, procedural scene generation [14] has enabled near-infinite dataset scale but the value of such scale to task performance has not been investigated in a focused manner.

In this paper, we contribute Habitat Synthetic Scenes Dataset (HSSD-200): a human-authored 3D scene dataset that more closely mirrors real scenes than prior datasets. Our

dataset consists of recreations of real houses modeled using a diverse set of 18,656 unique, high-quality 3D models of real objects. This dataset will be open-sourced and distributed under a permissive academic research license free of charge. We compare HSSD with prior synthetic datasets to show it is closer to real-world scenes in terms of visual fidelity, scene dimensions, and object occurrence statistics. We then perform a systematic study of scale vs realism with this dataset and other synthetic scene datasets from prior work. Our experiments show that the smaller-scale but higher-quality HSSD dataset leads to ObjectGoal navigation (ObjectNav) agents that outperform agents trained on significantly larger datasets. In ObjectNav, an embodied agent is spawned at a random location and orientation and has to efficiently navigate to an instance of a goal category (bed, tv, chair, etc.) [4, 40]. Surprisingly, we find that we can train navigation agents with better generalization to real-world 3D reconstructed scenes using two orders of magnitude fewer scenes from HSSD than from prior datasets. Figure 1 shows that training on 122 HSSD scenes leads to agents that generalize better to HM3DSem [41] and MP3D [7] real-world scenes than agents trained on 10,000 ProcTHOR [14] scenes. Beyond training navigation agents, HSSD enables work in object manipulation and rearrangement [28, 43, 44]. This is possible due to the compositionality of synthetic scenes

allowing easy removal or addition of objects, operations that are challenging in reconstructions as they require 3D object segmentation and infilling of holes left by moved objects.

2. Related work

Progress in embodied AI has been driven by the availability of 3D scene datasets that can be used with simulation platforms to train and evaluate agents. We focus on analyzing the transfer performance of ObjectNav agents trained on synthetic data to real-world 3D scene scans.

ObjectGoal navigation. There are three families of approaches to this task: end-to-end reinforcement learning (RL), imitation learning (IL), and modular learning (ML). RL methods learn policies that map visual observations directly to action probabilities through hidden states of minimal recurrent neural networks (RNNs) that serve as memory [23, 27, 42]. These methods learn from taking actions and getting rewards based on the progress they make towards the goal object. On the other hand, IL methods learn navigation from large-scale human demonstrations [30]. Modular approaches typically build and leverage scene semantic maps to navigate to the goal object [9, 18]. Recent work has tackled ObjectNav using modular approaches [1] and using self-supervised training [26]. We choose ObjectNav as the focus of our investigation since it can serve as a building block for more complex or longer-horizon tasks, including rearrangement [3] where the agent must first navigate in order to manipulate an object or move it to another location. Moreover, the ObjectNav task requires semantic information about the objects present in the scene and involves reasoning about spatial arrangement patterns of common object categories. Thus, the characteristics of the 3D scene datasets in terms of visual fidelity and correlation to real-world object occurrence and spatial arrangement patterns are important.

3D scene datasets. ObjectNav experiments have been conducted in both scanned real-world environments [7, 29, 39, 41] and synthetic datasets [12, 14, 22, 33]. The performance of ObjectNav agents depends on the scale of the training data as well as the complexity of the environments used for training and testing. There is a trend of training with increasing number of environments, from small single room environments [22] to ever larger multi-room environments [7, 39, 41]. Despite the desire to train on larger datasets, the number of available annotated semantically annotated real-world scans remains limited. While HM3D [29] has 1000 scenes, only 216 have been semantically annotated and can be used to train semantically-aware agents. In addition, Ramakrishnan et al. [29] pointed out issues with training ObjectNav agents on these scanned environments due to holes in walls, ceilings, and floors. Most recently,

Deitke et al. [14] demonstrated improved performance on ObjectNav and transfer to real-world scans by leveraging procedurally generated scenes.

Despite this, there has been limited work investigating the scale vs quality tradeoff of synthetic scenes for training ObjectNav agents. This is partly due to the limited number of synthetic datasets available. Existing synthetic scene datasets tend to be either limited to single rooms [22], number of scenes [12, 33], or incomplete [16]. For instance, 3D-FRONT [16], while large, does not have populated kitchens or bathrooms, and has interpenetration issues due to use of algorithmic object asset replacement. In this work, we investigate aspects of synthetic 3D scene datasets that are important for transfer to real-world scans. Concretely, we ask the following question for synthetic 3D scene datasets: does large scale suffice, or is realism (match to real-world scenes) also important? To do so, we contribute HSSD-200, a high-quality 3D scene dataset that better matches real-world scenes than prior synthetic datasets. While the number of scenes is small compared to ProcTHOR, we show that the high quality of this dataset allows for better transfer to navigation in real-world environments.

3. HSSD-200: Habitat Synthetic Scenes Dataset

To enable our investigation, we develop and contribute a new dataset providing high-quality synthetic 3D interiors. The Habitat Synthetic Scenes Dataset (HSSD-200) consists of 211 houses containing 18,656 objects across 466 semantic categories. These scenes were designed using the Floorplanner[1] web interior design interface. The layouts are predominantly re-creations of real houses by realtors. Individual objects are created by professional 3D artists and in most cases match specific brands of real-world furniture and appliances. HSSD is distinguished from prior work along several axes: i) high-quality, fully human-authored 3D interiors; ii) fine-grained semantic categorization corresponding to WordNet ontology; iii) asset compression to enable high-performance embodied AI simulation. See Figure 2 for example scenes from the dataset, and Table 1 for a comparison against other synthetic scene datasets in terms of overall statistics.

The preparation of this dataset involved several stages: object extraction, decomposition, alignment, semantic categorization, and asset compression. We describe each here.
Object extraction. The original scenes are exported as a single glTF asset from the Floorplanner database. We decompose each of these scene assets into constituent objects, as well as architectural elements (walls, floors, ceilings) and openings (doors or windows). We extract all objects and deduplicate into unique glTF assets to create a shared 3D object model database. This database of 18,656 objects includes a variety of objects which we semantically annotate.

---

[1] https://floorplanner.com – data licensed from Floorplanner and made available for academic use.

Dataset	Total				Average per scene				
	Scenes	Objects	Categories	Nav area	Nav area	Nav comp	Clutter	Categories	Instances
HM3DSem (scans) [41]	181	59,269	1,533	20.7K	114.4	9.7	4.0	103.9	327.5
MP3D (scans) [7]	90	50,851	1,658	29.6K	328.4	12.8	2.8	95.5	565.0
Gibson tiny (scans) [39]	35	2,397	35	4.9K	139.4	8.8	4.2	15.5	68.5
ReplicaCAD [33]	90	92	39	4.5K	49.8	7.2	4.8	14.3	25.5
iTHOR [22]	120	3,748	112	2.0K	17.1	2.4	9.6	28.2	46.2
RoboTHOR [12]	75	652	47	1.9K	25.9	8.1	5.8	28.8	38.4
ProcTHOR [14]	12K	1,547	95	808.4K	67.4	10.0	5.2	40.5	74.7
HSSD-200 (ours)	211	18,656	466	53.2K	252.2	13.7	5.9	61.5	329.7

Table 1. **Scene dataset statistics.** From left: number of scenes, number of unique objects, number of object categories, total navigable area in m^2, average navigable area per scene, navigation complexity as defined by Ramakrishnan et al. [29] with threshold of points 1m or more apart, mean categories per scene, clutter as defined by Ramakrishnan et al. [29], object categories per scene, and instances per scene. Note that numbers for RoboTHOR and HM3DSem do not include the hidden test sets, and we exclude architectural objects (e.g., walls).

Object decomposition. In some cases, a source 3D model of an object represented multiple semantically-distinct objects (e.g., a dining table with chairs, plates, and silverware). Four graduate students annotated all 3D models into single object or "multiple object" classes. A model has multiple objects if there are distinct nameable components that can be easily detached. We identified 1,791 such 3D models. We used an interface based on Mao et al. [24] web UI to segment these models. First, we use connected component analysis on the 3D mesh topology to obtain an initial segmentation. Then, the four annotators decomposed 1,662 models resulting in 11,153 object parts. We then extracted submeshes, identified duplicate geometry and computed alignment transforms to correspond instances of the same object (e.g., chairs around a table). We did this by fitting an oriented bounding box (OBB) on each extracted part, initializing alignment using the OBB parameters and then running ICP [5] for one iteration.

Object alignment. All objects including single objects are then aligned to have semantically consistent orientations. We aligned the objects to have a consistent up and front orientation based on the interface of Mao et al. [24]. A total of 2,883 object models required such manual alignment.

Semantic categorization. Each object was then annotated with a semantic category label. The labels are from an augmented set of WordNet [25] synsets that we call WordNetCo (WordNet common objects). WordNet is a popular taxonomy used to organize popular datasets such as ImageNet [15] and ShapeNet [8]) but it is lacking in granularity for common objects and modern devices (e.g., iPad, USB stick). We augmented WordNet with synsets for several common objects (e.g., potted plant, wall lamp) and initialize the object labels by mapping internal tags provided by Floorplanner to WordNetCo synsets. We then asked the annotators to manually check, correct, and refine the linked WordNet synsets. The annotators also specified in what room category a particular object is typically found (e.g., bed in bedrooms). Unlike

Simulator	ProcTHOR-S (FPS ↑)			ProcTHOR-L (FPS ↑)		
	1 Proc	1 GPU	8 GPU	1 Proc	1 GPU	8 GPU
AI2-THOR [22]	240	1,427	8,599	115	6,280*	3,208
	±69	±74	±359	±19	±40	±127
Habitat [31, 33]	2,297	6,374	57,160	1,007	5,237	39,510†
(uncompressed)	±447	±798	±4,917	±187	±1,130	±6,345
Habitat [31, 33]	**2,523**	**7,363**	**58,947**	**1,233**	**6,508**	**46,674**
(compressed)	±300	±394	±1,804	±224	±495	±5,640

Table 2. **Benchmark of navigation FPS.** We optimize ProcTHOR scenes similarly to HSSD and compare performance when the assets are loaded in Habitat to the original scenes in AI2-THOR. We observe close to an order of magnitude improvement in simulation speed across most setups. *After correspondence with Deitke et al. [14], the authors confirmed this number is a typo and the true number is unknown. †ProcTHOR-L 8-GPU benchmark conducted with 7 processes per GPU instead of 15 due to memory bottlenecks.

other datasets that use heuristics to estimate the real-world size of the objects [13], our objects are already modeled with real-world dimensions and consistently scaled.

Asset compression. After the above annotation was performed, all object assets were compressed to reduce GPU memory requirements during experimentation. We performed quadric mesh simplification [17] to an error threshold of $1e^{-3}$, reduced texture resolution to a maximum dimension of 256, and compressed all textures using the Basis[2] supercompression algorithm. This compression resulted in a 12.4x reduction of on-disk size and comparable on-GPU memory consumption reduction.

Comparison with Habitat-optimized ProcTHOR assets. To demonstrate the performance benefits of this pipeline we converted and compressed all ProcTHOR [14] scenes in the same way as HSSD. Table 2 benchmarks these ProcTHOR assets in Habitat and compares with performance

---
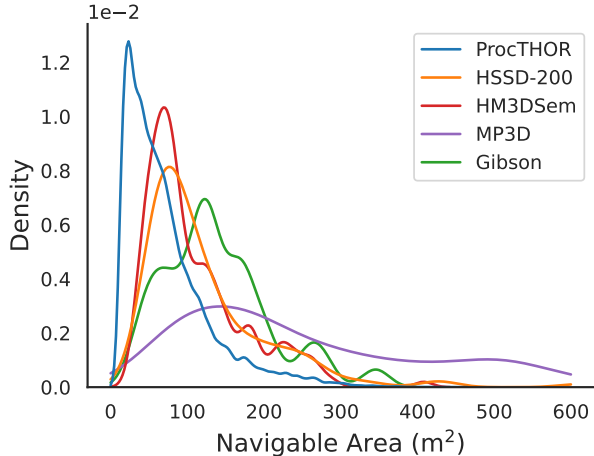
[2]https://github.com/BinomialLLC/basis_universal

Figure 3. **Per-scene navigable area distribution in ProcTHOR, HSSD and HM3DSem.** HSSD (orange) more closely matches real-world scans in HM3DSem, Gibson, and MP3D than ProcTHOR (blue). Note that the MP3D area distribution is broader as it contains larger-scale public and commercial spaces in addition to residences.

Dataset	HM3D		Gibson		MP3D	
	FID $\downarrow$	KID $\downarrow$	FID $\downarrow$	KID $\downarrow$	FID $\downarrow$	KID $\downarrow$
ProcTHOR	88.5	78.0 ± 1.6	95.6	82.6 ± 1.8	87.2	63.7 ±1.3
HSSD	**65.2**	**57.0** ±1.4	**73.0**	**61.7** ± 1.6	**61.6**	**43.3** ±1.3
HM3D	—	—	18.4	12.7 ± 0.8	25.2	18.3 ±0.7
Gibson	18.4	12.8 ± 0.8	—	—	38.4	26.2 ±1.2
MP3D	25.3	18.3±0.7	38.5	26.2 ± 1.2	—	—

Table 3. **Visual fidelity against real images from HM3D, Gibson, and MP3D.** We render images from ProcTHOR and HSSD using Habitat, and compute FID and KID to HM3D and Gibson images. HSSD is closer to the real-world image datasets. See the supplement for qualitative visuals.

of the original assets in AI2-THOR. We use both the small and large ProcTHOR sets from Deitke et al. [14], randomly sample 50 scenes from each and take 2K steps in each scene with a random navigation agent, rendering $224 \times 224 \times 3$ RGB images. The 1 GPU benchmark uses 15 simulation processes and the 8 GPU benchmark uses 15 processes per GPU. Benchmarking is done on a server with 8 NVIDIA RTX Quadro 4000 GPUs. Note that the AI2-THOR benchmark numbers were using RTX Quadro 8000 GPUs which have higher memory and more CUDA cores. As we can see, the asset compression and the use of the Habitat simulation platform enable much higher performance. These optimized ProcTHOR assets are independently valuable by enabling faster training and evaluation on other embodied AI tasks.

4. Dataset analysis

Before carrying out experiments with HSSD, we characterize its properties by comparison against datasets from prior work. We compare mainly against iTHOR [22] and Proc-THOR [14], and use the HM3DSem [41] and MP3D [7] real scan datasets as references since they are the basis of our ObjectNav transfer experiments. We characterize the synthetic datasets and HSSD along three axes: *scale*, *realism*, and *complexity*.

Scale. The volume of scene data available for experiments is naturally an important characteristic. We measure scale in terms of scene and object counts, as well as navigable area since that is a key attribute for the ObjectNav task. Table 1 summarizes these statistics. HSSD has a relatively small number of scenes but a high number of unique objects

(more than 18K), and significantly higher average navigable area per scene (252.2 m^2) than prior datasets. The average number of object categories and object instances per scene is also significantly higher compared to prior synthetic 3D scene datasets, and closer to the real-world scenes in HM3DSem [41], Gibson [39], and MP3D [7]. In Figure 3 we plot the distribution of navigable area per scene for Proc-THOR and HSSD, and compare against the distributions for these scan datasets. We see that ProcTHOR has a high peak and narrow distribution with scenes that have relatively small total navigable areas. In contrast, HSSD matches the per-scene navigable area distribution of the scanned real-world environments more closely.

Realism. The realism of scenes impacts agent perception. Following Ramakrishnan et al. [29], we measure visual realism using the FID [19] and KID [6] metrics. Both metrics measure the perceptual similarity between two distributions of images. We compare rendered images from the synthetic datasets against images from real scenes. Table 3 shows that HSSD is closer to real-world images from HM3D, Gibson [39], and MP3D [7]. Note that these metrics measure realism of the dataset–renderer combination, and we are using a simple, efficient rasterization-based renderer in Habitat. Future improvements to rendering quality can lead to improvements in visual fidelity. Moreover, this measure of visual fidelity does not directly capture the higher-level "semantic" realism of the scenes.

To measure semantic realism, we compute object co-occurrence statistics. We select 28 common object categories found in all datasets (see supplement). These objects span a range of sizes and vary in their placement on different surfaces (e.g., floor, countertop, tabletop, shelf). We consider two objects to co-occur if the Euclidean distance of the object centroids is less than 1m. Figure 4 shows max-normalized co-occurrence heatmaps for the 28 object categories. We see that the HSSD co-occurrence patterns are qualitatively more similar to real-world scenes from HM3DSem and MP3D than ProcTHOR. To quantify these trends we also compute overall co-occurrence similarity met-

ProcTHOR	HSSD	HM3DSem	MP3D

$s_{\text{THOR-HM3D}} = 0.14$ $s_{\text{HSSD-HM3D}} = 0.40$ $s_{\text{HM3D-HM3D}} = 1.00$ $s_{\text{MP3D-HM3D}} = 0.72$

$s_{\text{THOR-MP3D}} = 0.07$ $s_{\text{HSSD-MP3D}} = 0.28$ $s_{\text{HM3D-MP3D}} = 0.45$ $s_{\text{MP3D-MP3D}} = 1.00$
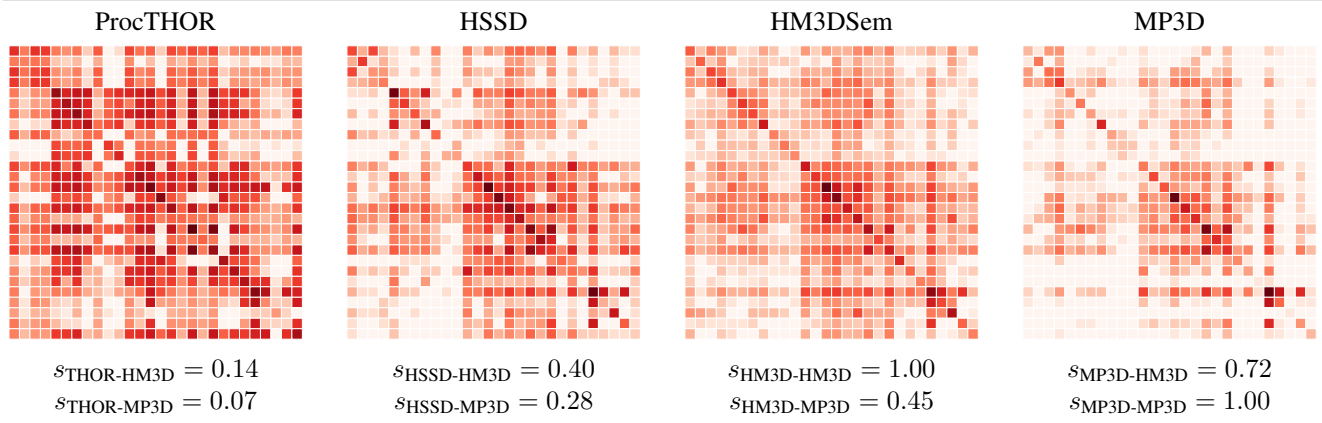
Figure 4. **Object category co-occurrence heatmap visualizations for 28 object categories in ProcTHOR, HSSD, and HM3DSem.** We observe that object co-occurrences in HSSD are closer to the real-world HM3DSem scenes relative to ProcTHOR scenes. We also quantify this observation using a similarity metric measuring the mean Jaccard Index between cluster pairs of hierarchically clustered co-occurrences (higher is better). The bottom two rows of values report the value of this metric with HM3DSem and MP3D as the reference point.

rics. The Spearman's rank correlation coefficient for the pairwise co-occurrences between HSSD and HM3DSem is $\rho_{\text{HSSD-HM3D}} = 0.219$ compared to $\rho_{\text{THOR-HM3D}} = 0.083$ (higher is better). The correlation coefficient with MP3D is also higher for our dataset ($\rho_{\text{HSSD-MP3D}} = 0.103$) than ProcTHOR ($\rho_{\text{THOR-MP3D}} = 0.017$). To further quantify the co-occurrences we also compute a hierarchical clustering on the co-occurrence matrices and extract clusters at threshold 0.8. We then compute the mean Jaccard Index score between cluster pairs (maximum score pairs chosen with the Hungarian algorithm). This gives a value of $s_{\text{HSSD-HM3D}} = 0.40$ compared to $s_{\text{THOR-HM3D}} = 0.14$ for HM3D, and value of $s_{\text{HSSD-MP3D}} = 0.28$ compared to $s_{\text{THOR-MP3D}} = 0.07$ for MP3D (higher is better).

Complexity. Most real-world houses are cluttered with furniture items and other objects packed in relatively small spaces. We characterize this aspect of real-world scene complexity using object and scene statistics. In Table 1 we see that HSSD has more than 4x object instances per scene than the next highest dataset (329.7 vs 74.7 for ProcTHOR), and about 4x more object categories in total (466 vs 112 for iTHOR). These statistics are much closer to those of real-world scenes from HM3DSem (327.5 object instances on average and 1,533 categories total) and MP3D (565 object instances on average and 1,658 categories total).

5. Experimental setup

We investigate the role of dataset scale and realism on ObjectNav agent performance, focusing on zero-shot transfer.
Task. We adopt the Habitat ObjectNav 2022 challenge setup [40], which we briefly summarize here. There are 6 goal categories: 'bed', 'chair', 'sofa', 'tv', 'plant', 'toilet'. The agent is successful if it predicts STOP action within 0.1m
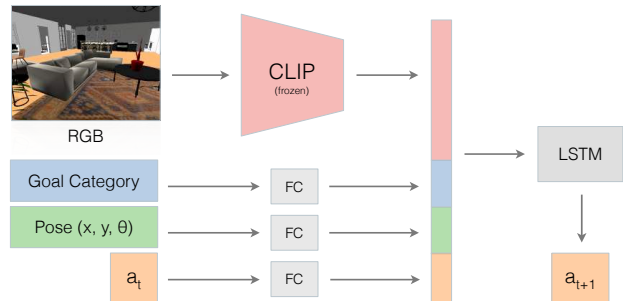

Figure 5. **Architecture of the ObjectNav agent.** The RGB observation is passed into a frozen CLIP backbone to produce a feature embedding. The goal category, agent pose and previous action are embedded into 32-dim vectors. All embeddings are fed to an LSTM to predict the next action.

of a "viewpoint" around each goal. The agent is a LoCoBot with base radius of 0.18m and height of 0.88m and possesses an RGB sensor plus a compass and GPS sensor. The agent action space consists of: STOP, MOVE_FORWARD, TURN_LEFT, TURN_RIGHT, LOOK_UP, LOOK_DOWN, with forward step of 0.25m and turn angle of 30 degrees.

Agent architecture. At each step, the agent has access to RGB observations, GPS and compass sensors, the object category of the goal to navigate to as input. The RGB frames are of resolution 224×224. The GPS and compass sensors are specified as in Yadav et al. [41]. The input goal is an integer o corresponding to the object category and used to index into an embedding matrix O to produce a 32-dimensional embedding. We use an architecture similar to Khandelwal et al. [21] (see Figure 5). The RGB frames are passed through a frozen CLIP pre-trained ResNet 50 visual encoder to get a

Dataset	train			val		
	S	E/S	T	S	E/S	T
iTHOR [22]	64	2K	128K	13	30	390
ProcTHOR [14]	10K	1K	10M	1K	10	10K
HSSD	122	1K	122K	40	30	1.2K
MP3D [7]	56	16.5K	925K	10	65.8	658
HM3DSem [41]	145	50K	7.25M	36	30	1.08K

Table 4. **Train/val scenes (S), episodes-per-scene (E/S) and total (T) episode statistics.** We generate ObjectNav episodes for the 6 object categories used in our experiments across available train and val set scenes in each dataset.

2048 dimension feature vector. The agent also takes the previous action as input. Both the goal and action embedding matrices (O, A) are learned during training. These embeddings are concatenated along with the outputs from the GPS and compass sensors and passed through a 2-layer LSTM network. The LSTM outputs are then passed through a linear layer to produce action probabilities for the next step. We train agents using DDPPO [37] with VER [38] on 4 NVIDIA A40 GPUs and 24 environment workers per GPU.

Episode dataset generation. We split HSSD into train/val/test following a ratio 60/20/20 ratio. We use the standard splits for ProcTHOR [14], and HM3DSem [41]. For iTHOR [22], we start with the standard split (80 train, 20 val scenes) but filter out scenes that do not contain any of the 6 goal categories or are too small for navigation. Following prior work, we generate 2K training episodes per scene for this dataset [22]. For ProcTHOR and HSSD, we use 1K training episodes per scene. For HM3DSem, we follow prior work and use 50K training episodes per scene for the 145 HM3DSem train scenes. For MP3D, we again follow prior work but remove episodes with target objects that do not belong to the 6 goal categories. This results in a dataset of 56 scenes with around 16.5K episodes each for training and 10 scenes with 658 total episodes for validation.

For the val set we use 30 episodes per scene in HSSD, iTHOR, and HM3DSem scenes, and 10 episodes per scene for ProcTHOR (see Table 4). The total number of episodes per scene are uniformly divided across all object categories and object instances within each category in each scene. See the supplement for details on episode generation.

6. Results

We compare agents in terms of generalization performance to HM3DSem and MP3D scenes when trained on synthetic 3D scene datasets of different scale and realism. We evaluate generalization of agents trained on iTHOR [22], ProcTHOR-10k [14], and HSSD. We also train agents on HM3DSem and MP3D to provide a comparison point of agents trained on reconstruction datasets. The trained agents for each dataset

are then evaluated on the validation sets of all datasets.

Zero-shot generalization. We train agents until convergence on the train set and report the performance of the checkpoints with highest val set SPL averaged across three training runs in Table 5. The supplement provides training plots for these experiments. As expected the best performance in most cases is achieved by the agent trained on the same dataset. Note that iTHOR and ProcTHOR scenes use the same object assets and therefore agents tend to transfer relatively well between them. Despite the much smaller overall dataset size of HSSD compared with ProcTHOR-10K, zero-shot evaluation on HM3DSem results in higher success (19.15% vs 12.53%) and SPL (7.71 vs 5.26). Similarly, zero-shot evaluation on MP3D also results in higher success (12.56% vs 8.26%) and SPL (4.56 vs 2.96). This indicates that the much higher scale of ProcTHOR-10K does not translate to improved zero-shot ObjectNav agent generalization.

Fine-tuned agents. In addition to the zero-shot generalization performance of agents, we carried out a number of experiments where agents were fine-tuned on the target dataset's train split prior to evaluation on that dataset's val split. As expected, after finetuning the differences between agents initialized from different training sets are reduced, and agents converge to more similar levels of performance on the target dataset. After finetuning, agents trained on HSSD-122 achieve 48.23% success and 23.1 SPL compared to ProcTHOR-10K agents achieving 48.32% success and 21.8 SPL. The gap in combined efficiency and success (as measured by SPL) remains but is smaller compared to the zero shot setting. This finding stands in contrast to the appreciable gap in the zero shot generalization performance which is more representative of real world deployment to previously unseen environments. See the supplement for a more complete summary of these fine-tuning experiments.

Disentangling scene dataset scale and realism. To disentangle scene dataset scale and scene dataset realism in agent generalization we create scale-matched datasets for HSSD and ProcTHOR by varying the number of scenes and total navigable area. We consider different total dataset scales for HSSD (60 scenes, and all 122 training scenes) and Proc-THOR (60, 122, and all 10K scenes). The 60 and 122 Proc-THOR scene subsets match the navigable area distribution in the HSSD counterparts. All the aforementioned datasets have 1K episodes per scene. We refer to these subsets as ProcTHOR-60 and ProcTHOR-122 (see supplement for details). Then, we compare zero-shot generalization of agents on the MP3D and HM3DSem val sets. All agents are trained until convergence (i.e. until training and validation success saturate). The results are plotted in Figure 6. Agents trained on HSSD-60 and HSSD-122 outperform agents trained on the scale-matched ProcTHOR-60 and ProcTHOR-122 in terms of both success rate and SPL. The difference in the 122 scene setting is particularly pronounced with HSSD-122 vs

Eval dataset →	iTHOR		ProcTHOR		HSSD		MP3D		HM3DSem	
Train dataset ↓	Success ↑	SPL ↑	Success ↑	SPL ↑	Success ↑	SPL ↑	Success ↑	SPL ↑	Success ↑	SPL ↑
iTHOR	78.06 ± 1.31	53.05 ± 0.59	29.8 ± 0.08	15.61 ± 0.08	13.43 ± 0.29	4.75 ± 0.11	6.18 ± 0.41	2.13 ± 0.27	6.16 ± 0.19	2.38 ± 0.22
ProcTHOR-10K	67.04 ± 4.31	36.32 ± 4.24	80.72 ± 0.43	46.44 ± 0.33	31.54 ± 1.42	12.94 ± 0.7	8.26 ± 0.73	2.96 ± 0.46	12.53 ± 0.49	5.26 ± 0.2
HSSD-122	27.5 ± 3.36	12.34 ± 2.08	9.57 ± 0.5	3.86 ± 0.24	54.81 ± 0.28	24.12 ± 0.14	**14.44 ± 1.29**	**5.17 ± 0.48**	**19.15 ± 0.95**	**7.71 ± 0.47**
MP3D							32.07 ± 1.22	14.73 ± 0.99	31.95 ± 1.17	13.13 ± 0.45
HM3DSem							30.8 ± 1.82	13.99 ± 0.89	48.1 ± 1.54	22.16 ± 0.05

Table 5. **ObjectNav zero-shot generalization across datasets.** Agents are trained on the training set of the dataset indicated in each row and evaluated on the val set of the datasets in the columns. We report the average across three independent training runs, and the standard error on this average. As expected, agents perform well when evaluated on the dataset on which they were trained. When looking at the generalization trends, we observe that surprisingly HSSD-122 achieves better generalization performance on both MP3D and HM3DSem compared to agents trained with the much larger scale ProcTHOR-10K.
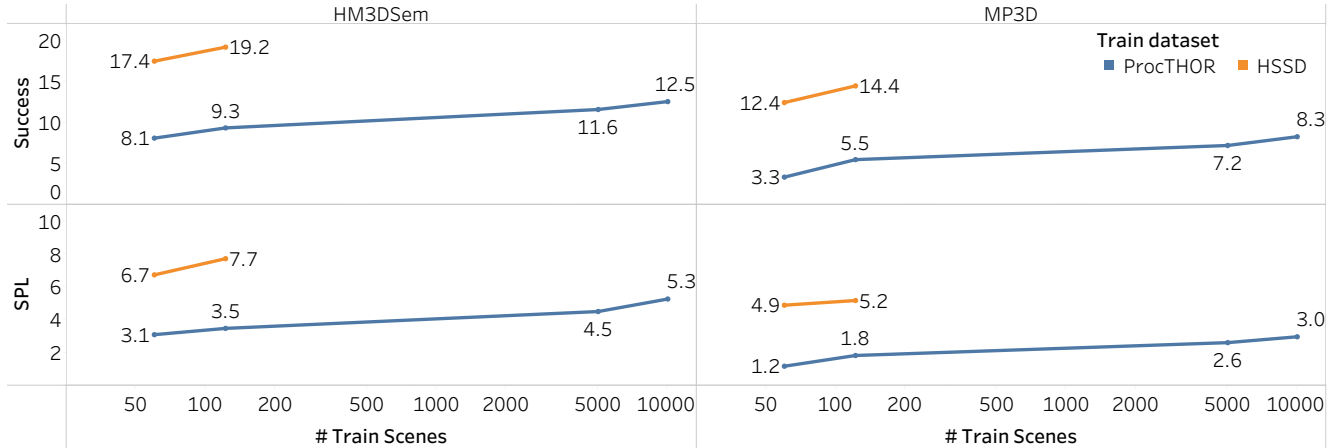

Figure 6. **ObjectNav zero-shot generalization performance for agents trained with different dataset scales.** The plots show zero-shot performance of agents on the HM3DSem and MP3D val set. We see that agents trained on HSSD-60 and HSSD-122 outperform agents trained on ProcTHOR-60 and ProcTHOR-122. Moreover, the much larger ProcTHOR-10K dataset does not lead to significant improvements in ObjectNav generalization performance, with agents trained on it generalizing less well than ones trained with far fewer HSSD scenes.

ProcTHOR-122 agents achieving 19.15 vs 9.33 success and 7.71 vs 3.47 SPL, respectively on the HM3DSem val set (see Figure 6 left column). The trend is even more pronounced when evaluating on MP3D val: 14.44 vs 5.47 success and 5.17 vs 1.83 SPL, for HSSD-122 vs ProcTHOR-122 agents respectively (see Figure 6 right column). Agents trained on ProcTHOR-10K are only marginally better than agents trained on the much smaller ProcTHOR-122 indicating the limited value of larger scale by itself.

Limitations. Our investigation is limited to one type of agent: monolithic pixels-to-actions agents trained end-to-end in an RL fashion. A broader investigation including agents designed using modular approaches [18] or imitation learning [30] would offer insights into the comparative trends between these families of approaches.

7. Conclusion

Our goal was to investigate the impact of scene scale and realism on the generalization ability of ObjectGoal navigation agents. To this end, we constructed HSSD: a high-quality, human-authored synthetic 3D scene dataset. We carried out a systematic analysis of how agents trained in this scene dataset and other synthetic 3D scene datasets from prior work generalize to realistic 3D scenes. We found that a smaller number of higher-quality synthetic 3D scenes leads to better generalization compared to larger numbers of procedurally generated 3D scenes or lower-quality scenes. We hope our dataset and our findings enable future work on Embodied AI agents for visual navigation and related tasks.

References

[1] Ziad Al-Halah, Santhosh Kumar Ramakrishnan, and Kristen Grauman. Zero experience required: Plug & play modular transfer learning for semantic visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17031–17041, 2022. 3

[2] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683, 2018. 1

[3] Dhruv Batra, Angel X Chang, Sonia Chernova, Andrew J Davison, Jia Deng, Vladlen Koltun, Sergey Levine, Jitendra Malik, Igor Mordatch, Roozbeh Mottaghi, et al. Rearrangement: A challenge for embodied ai. *arXiv preprint arXiv:2011.01975*, 2020. 1, 2, 3

[4] Dhruv Batra, Aaron Gokaslan, Aniruddha Kembhavi, Oleksandr Maksymets, Roozbeh Mottaghi, Manolis Savva, Alexander Toshev, and Erik Wijmans. ObjectNav revisited: On evaluation of embodied agents navigating to objects. *arXiv preprint arXiv:2006.13171*, 2020. 2

[5] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. Spie, 1992. 4

[6] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2018. 5

[7] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3D: Learning from RGB-D data in indoor environments. In *Proc. of the International Conference on 3D Vision (3DV)*, 2017. 1, 2, 3, 4, 5, 7

[8] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015. 4

[9] Devendra Singh Chaplot, Dhiraj Prakashchand Gandhi, Abhinav Gupta, and Russ R Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. *Advances in Neural Information Processing Systems*, 33:4247–4258, 2020. 1, 3

[10] Devendra Singh Chaplot, Ruslan Salakhutdinov, Abhinav Gupta, and Saurabh Gupta. Neural topological slam for visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12875–12884, 2020. 1

[11] Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. Soundspaces: Audio-visual navigation in 3d environments. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 17–36. Springer, 2020. 1

[12] Matt Deitke, Winson Han, Alvaro Herrasti, Aniruddha Kembhavi, Eric Kolve, Roozbeh Mottaghi, Jordi Salvador, Dustin Schwenk, Eli VanderBilt, Matthew Wallingford, Luca Weihs, Mark Yatskar, and Ali Farhadi. RoboTHOR: An open simulation-to-real embodied AI platform. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3164–3174, 2020. 3, 4

[13] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3D objects. *arXiv preprint arXiv:2212.08051*, 2022. 4

[14] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. ProcTHOR: Large-scale embodied AI using procedural generation. In *Advances in Neural Information Processing Systems*, 2022. 1, 2, 3, 4, 5, 7

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. IEEE, 2009. 4

[16] Huan Fu, Bowen Cai, Lin Gao, Lingxiao Zhang, Cao Li, Zengqi Xun, Chengyue Sun, Yiyun Fei, Yu Zheng, Ying Li, et al. 3D-FRONT: 3D Furnished Rooms with layOuts and semaNTics. *arXiv preprint arXiv:2011.09127*, 2020. 3

[17] Michael Garland and Paul S Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 209–216, 1997. 4

[18] Theophile Gervet, Soumith Chintala, Dhruv Batra, Jitendra Malik, and Devendra Singh Chaplot. Navigating to objects in the real world. *arXiv preprint arXiv:2212.00922*, 2022. 1, 3, 8

[19] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 5

[20] Abhishek Kadian, Joanne Truong, Aaron Gokaslan, Alexander Clegg, Erik Wijmans, Stefan Lee, Manolis Savva, Sonia Chernova, and Dhruv Batra. Sim2real predictivity: Does evaluation in simulation predict real-world performance? *IEEE Robotics and Automation Letters*, 5(4):6670–6677, 2020. 1

[21] Apoorv Khandelwal, Luca Weihs, Roozbeh Mottaghi, and Aniruddha Kembhavi. Simple but effective: CLIP embeddings for embodied AI. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14829–14838, 2022. 6

[22] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. AI2-THOR: An interactive 3D environment for visual AI. *arXiv preprint arXiv:1712.05474*, 2017. 2, 3, 4, 5, 7

[23] Oleksandr Maksymets, Vincent Cartillier, Aaron Gokaslan, Erik Wijmans, Wojciech Galuba, Stefan Lee, and Dhruv Batra. Thda: Treasure hunt data augmentation for semantic

navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15374–15383, 2021. 3

[24] Yongsen Mao, Yiming Zhang, Hanxiao Jiang, Angel X Chang, and Manolis Savva. MultiScan: Scalable RGBD scanning for 3D environments with articulated objects. In *Advances in Neural Information Processing Systems*, 2022. 4

[25] George A Miller. WordNet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 4

[26] So Yeon Min, Yao-Hung Hubert Tsai, Wei Ding, Ali Farhadi, Ruslan Salakhutdinov, Yonatan Bisk, and Jian Zhang. Object goal navigation with end-to-end self-supervision. *arXiv preprint arXiv:2212.05923*, 2022. 1, 3

[27] Arsalan Mousavian, Alexander Toshev, Marek Fišer, Jana Košecká, Ayzaan Wahid, and James Davidson. Visual representations for semantic target driven navigation. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8846–8852. IEEE, 2019. 3

[28] Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander William Clegg, Michal Hlavac, So Yeon Min, et al. Habitat 3.0: A co-habitat for humans, avatars and robots. *arXiv preprint arXiv:2310.13724*, 2023. 2

[29] Santhosh K Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alex Clegg, John Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-Matterport 3D dataset (hm3d): 1000 large-scale 3D environments for embodied AI. *arXiv preprint arXiv:2109.08238*, 2021. 1, 3, 4, 5

[30] Ram Ramrakhya, 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. 3, 8

[31] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied AI research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347, 2019. 4

[32] Davide Scaramuzza and Elia Kaufmann. Learning agile, vision-based drone flight: From simulation to reality. In Aude Billard, Tamim Asfour, and Oussama Khatib, editors, *Robotics Research*, pages 11–18, Cham, 2023. Springer Nature Switzerland. ISBN 978-3-031-25555-7. 1

[33] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their Habitat. *Advances in neural information processing systems*, 2021. 1, 2, 3, 4

[34] Joanne Truong, Sonia Chernova, and Dhruv Batra. Bidirectional domain adaptation for sim2real transfer of embodied navigation agents. *IEEE Robotics and Automation Letters*, 6(2):2634–2641, 2021. 1

[35] Joanne Truong, Max Rudolph, Naoki Yokoyama, Sonia Cher-

nova, Dhruv Batra, and Akshara Rai. Rethinking sim2real: Lower fidelity simulation leads to higher sim2real transfer in navigation. *arXiv preprint arXiv:2207.10821*, 2022. 1

[36] Luca Weihs, Matt Deitke, Aniruddha Kembhavi, and Roozbeh Mottaghi. Visual room rearrangement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5922–5931, 2021. 1

[37] Erik Wijmans, Abhishek Kadian, Ari 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 *Proc. of the International Conference on Learning Representations (ICLR)*, 2020. 7

[38] Erik Wijmans, Irfan Essa, and Dhruv Batra. VER: Scaling on-policy RL leads to the emergence of navigation in embodied rearrangement. *Advances in Neural Information Processing Systems*, 2022. 7

[39] Fei Xia, Amir R. Zamir, Zhi-Yang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson Env: real-world perception for embodied agents. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018. 1, 3, 4, 5

[40] Karmesh Yadav, Santhosh Kumar Ramakrishnan, John Turner, Aaron Gokaslan, Oleksandr Maksymets, Rishabh Jain, Ram Ramrakhya, Angel X Chang, Alexander Clegg, Manolis Savva, Eric Undersander, Devendra Singh Chaplot, and Dhruv Batra. Habitat challenge 2022. https://aihabitat.org/challenge/2022/, 2022. 2, 6

[41] Karmesh Yadav, Ram Ramrakhya, Santhosh Kumar Ramakrishnan, Theo Gervet, John Turner, Aaron Gokaslan, Noah Maestre, Angel Xuan Chang, Dhruv Batra, Manolis Savva, et al. Habitat-Matterport 3D semantics dataset. *arXiv preprint arXiv:2210.05633*, 2022. 1, 2, 3, 4, 5, 6, 7

[42] Joel Ye, Dhruv Batra, Abhishek Das, and Erik Wijmans. Auxiliary tasks and exploration enable objectgoal navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16117–16126, 2021. 3

[43] Sriram Yenamandra, Arun Ramachandran, Mukul Khanna, Karmesh Yadav, Devendra Singh Chaplot, Gunjan Chhablani, Alexander Clegg, Theophile Gervet, Vidhi Jain, Ruslan Partsey, Ram Ramrakhya, Andrew Szot, Tsung-Yen Yang, Aaron Edsinger, Charlie Kemp, Binit Shah, Zsolt Kira, Dhruv Batra, Roozbeh Mottaghi, Yonatan Bisk, and Chris Paxton. The homerobot open vocab mobile manipulation challenge. In *Thirty-seventh Conference on Neural Information Processing Systems: Competition Track*, 2023. 2

[44] Sriram Yenamandra, Arun Ramachandran, Karmesh Yadav, Austin Wang, Mukul Khanna, Theophile Gervet, Tsung-Yen Yang, Vidhi Jain, Alexander William Clegg, John Turner, Zsolt Kira, Manolis Savva, Angel Chang, Devendra Singh Chaplot, Dhruv Batra, Roozbeh Mottaghi, Yonatan Bisk, and Chris Paxton. Homerobot: Open-vocabulary mobile manipulation. In *Proc. of the Conference on Robot Learning (CoRL)*, 2023. 2