Where are we in the search for an Artificial Visual Cortex for Embodied Intelligence?

Anonymous Authors[1]

Abstract

We present the largest and most comprehensive empirical study of visual foundation models for Embodied AI (EAI). First, we curate CortexBench, consisting of 17 different EAI tasks spanning locomotion, navigation, dexterous and mobile manipulation. Next, we systematically evaluate existing visual foundation models and find that none is universally dominant.

To study the effect of pre-training data scale and diversity, we combine ImageNet with over 4,000 hours of egocentric videos from 7 different sources (over 5.6M images) and train different-sized vision transformers using Masked Auto-Encoding (MAE) on slices of this data. These models required over 10,000 GPU-hours to train and will be open-sourced to the community.

We find that scaling dataset size and diversity improves performance on average but not uniformly across all tasks. Finally, we show that adding a second pre-training step on a small in-domain dataset improves performance, matching or outperforming the best prior results in this setting.

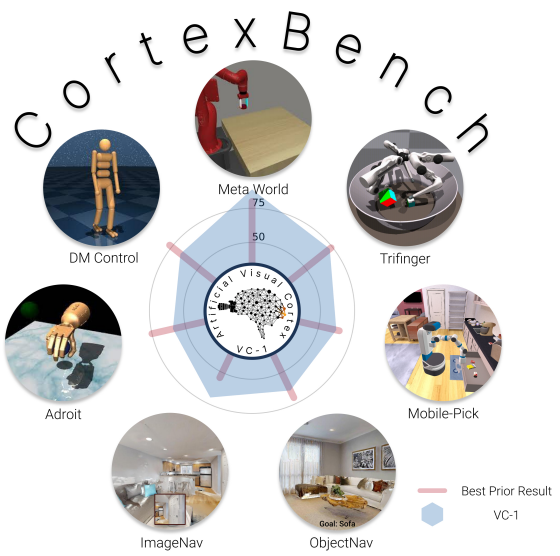


Figure 1. We assemble CortexBench from 7 benchmarks and systematically evaluate existing visual representation models. We then train a single new model **VC-1**, compare it to the *best prior result* on each benchmark (above), and adapt it to specific domains.

1. Introduction

Eyesight is considered one of the greatest inventions of biological evolution (Lane, 2010). Of course, the evolution of visual *sensing* via eyes progresses in concordance with visual *perception* – via a visual cortex, the region of the brain that (together with the motor cortex) enables an organism to convert sight into movement. In this work, we ask the same question Fukushima (Fukushima, 1975; 1980) did nearly 50 years ago – how do we design an *artificial visual cortex*, the module in a larger computational system that enables an artificial agent to convert camera input into actions?

In contemporary AI, this question has been operationalized as the design of pre-trained visual representations (PVRs) or visual 'foundation models' for embodied AI (EAI).[1] Indeed, recent work has shown that PVRs trained on large quantities of web-images and egocentric-videos can substantially improve performance and learning efficiency for navigation (Khandelwal et al., 2022; Yadav et al., 2022b) and manipulation tasks (Parisi et al., 2022; Nair et al., 2022; Radosavovic et al., 2022; Ma et al., 2022). Unfortunately, each study is fundamentally incommensurable, as each uses different self-supervised learning (SSL) algorithms on different pre-training datasets, designed for, and evaluated on different downstream EAI tasks. Naturally, one might ask: is there a universally-dominant configuration? Essentially, *does an artificial visual cortex already exist*?[2]

To answer this question, we conduct the largest and most comprehensive empirical study to-date of visual foundation models for EAI. First, we curate CortexBench, a new benchmark for evaluating PVRs, consisting of 17 tasks spanning low-level locomotion (Tassa et al., 2018), table-top manipulation of rigid and articulated objects (Yu et al., 2020),

[1] We use embodied AI (EAI) as an umbrella term for all communities studying visuomotor control such as robot learning, vision-based reinforcement learning, egocentric computer vision, etc.

[2] To the degree of our ability to measure it.

dexterous manipulation (Rajeswaran et al., 2018), multi-finger coordinated manipulation (Wüthrich et al., 2020), indoor visual navigation (Savva et al., 2019a), and mobile manipulation (Szot et al., 2021). The visual environments span from flat infinite planes to table-top settings to photorealistic 3D scans of real-world indoor spaces. The agent embodiments vary from stationary arms to dexterous hands to idealized cylindrical navigation agents to articulated mobile manipulators. The learning conditions vary from few-shot imitation learning to large-scale reinforcement learning. The exhaustiveness of this study enables us to draw conclusions with unprecedented scope and confidence.

Our first finding is a *negative result* about the universal applicability of prior work. We discover that while existing PVRs generally outperform learning-from-scratch baselines, none is universally dominant. Instead, we find that PVRs tend to work best in the domains (locomotion, manipulation, navigation) they were originally designed for. We note that no claims of universality were made in prior work, so this finding is illustrative rather than refutative. Overall, serendipity did not come to pass – an artificial visual cortex does not already exist.[2] However, curiously, the *kinds of PVRs* that are locally-dominant in CortexBench differ significantly in the size and type of pre-training datasets: CLIP (Radford et al., 2021) was pre-trained on 400M image-text pairs from the web; MVP (Radosavovic et al., 2022) on 4.5M frames from web-images and many egocentric-video datasets; R3M (Nair et al., 2022) on ∼5M frames from Ego4D – yet, each performs best on some subset of tasks in CortexBench. This leads to a next natural question: *how does scaling model size, dataset size, or diversity affect performance on* CortexBench*?* Can we use scaling as a means to learn a single PVR that works for all the diverse tasks in CortexBench?

To study these questions, we combine ImageNet with over 4,000 hours of egocentric videos from 7 different sources[3], which span videos of humans manipulating objects and navigating indoor spaces encountered in daily life. From this union, we create 4 pre-training datasets of varying size and diversity, with the largest containing over 5.6M images. We train vision transformers (ViT-B and ViT-L) (Dosovitskiy et al., 2020) on these 4 datasets using Masked Auto-Encoding (MAE) (He et al., 2021), and systematically analyze their performance on CortexBench. To benefit the EAI community, we will open-source these models, which required over 10,000 GPU hours to train.

We do find evidence supporting the scaling hypothesis, but the picture that emerges is more nuanced than what a superficial reading might suggest. Our largest model trained on all data, named **VC-1**, outperforms the best existing PVR by 1.3% on average. However, **VC-1** does *not* universally dominate either – i.e., there are PVRs trained on smaller amounts of data that outperform it on specific tasks. A similar trend emerges for diversity – more is better on average, but not universally. For instance, the best performance on the `Mobile-Pick` task from Habitat 2.0 (Szot et al., 2021) is achieved by pre-training on the subset of video data focused only on manipulation; presumably because the mobility involved in the task is fairly limited. Analogous trends hold for navigation-heavy tasks. Thus, our second key finding is: *Naively scaling dataset size and diversity does not improve performance uniformly across benchmarks*.

Our findings reveal a challenge and opportunity for the community – the search for a PVR that is universally dominant (or "foundational") for EAI calls for innovations in architecture, learning paradigm, data engineering, and more. As the final step in this paper, but as a first step towards this open problem, we show that adapting VC-1 with the same self-supervised learning algorithm (MAE) on a small amount of in-domain data can significantly improve performance and often becomes competitive with or outperforms the best PVRs in this setting. We note that this adaptation does not consume any additional in-domain data beyond what is already available for policy learning – for example, demonstrations in the case of imitation learning.

We will release code for CortexBench and pre-trained models, enabling the EAI, robotics, and CV communities to build on our work and benchmark their own models.

2. Related Work

Pretrained visual representations (PVRs). The last few years have seen increasing interest in the self-supervised learning (SSL) of visual representations (He et al., 2021; Caron et al., 2020; Baevski et al., 2022b; Chen et al., 2020; 2021). These algorithms use contrastive (Chen et al., 2020; 2021), distillation-based (Caron et al., 2020; Baevski et al., 2022b), or reconstructive (Bao et al., 2021; He et al., 2021) objectives for training. Recently, a flurry of works have proposed using the vision transformers (ViTs) (Dosovitskiy et al., 2021) with masked image modeling (He et al., 2021; Baevski et al., 2022a; Yao et al., 2022), which among other benefits reduces the computation time required for pre-training. In this work, we use one such pre-training algorithm (MAE (He et al., 2021)) to explore scaling and adapting pre-trained visual representations (PVRs).

PVRs for Embodied AI. Inspired by the advancements in self-supervised learning, recent work has incorporated visual representation learning into the training pipelines for EAI agents (Parisi et al., 2022; Nair et al., 2022; Radosavovic et al., 2022; Ma et al., 2022; Khandelwal et al., 2022; Yadav et al., 2022b). Specifically, Parisi et al. (2022) evaluate several PVRs trained with supervised or self-

[3] Ego4D, Epic Kitchens, Something-Something v2, 100 Days of Hands, HowTo100M, RealEstate10k, OpenHouse24k.

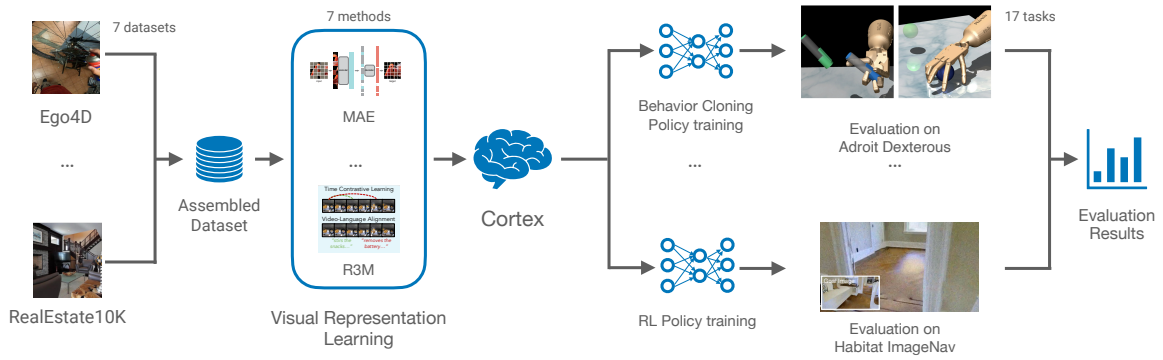


Figure 2. Overview of CORTEXBENCH. We assemble relevant datasets and visual representation learning algorithms to produce candidate Visual Cortex models, which are then evaluated using either reinforcement or imitation learning on a set of highly diverse tasks.

supervised learning on a range of EAI tasks, demonstrating promising results under a few-shot imitation learning evaluation protocol. Nair et al. (2022); Radosavovic et al. (2022); Ma et al. (2022) introduce new methods for pretraining visual representations using egocentric video data, targeting robotic manipulation tasks. Similarly, Khandelwal et al. (2022); Yadav et al. (2022b) use pretrained visual representations to improve performance on multiple visual navigation tasks. Closely related, Radosavovic et al. (2022) demonstrate that MAE pre-training on internet-scale video and image data can produce effective visual representations for robotic manipulation tasks. By contrast, this work studies a substantially larger range of embodied AI tasks (collected in CORTEXBENCH) to understand how PVRs can provide a general-purpose foundation for embodied agents and explores in-domain model adaptation for various tasks.

Scaling model and dataset size. Several works have showed that scaling model and dataset size improves performance on vision tasks like image classification (Zhai et al., 2022; Tian et al., 2021; Goyal et al., 2021). In EAI, Radosavovic et al. (2022) find that scaling model and data sizes improves downstream policy performances for robotic manipulation tasks. Crucially, they find that it is important to scale *both* the model and data sizes simultaneously. While these prior works have focused on narrow domains such as image classification and robotic manipulation, our work is the first to study if scaling can provide better models on a comprehensive collection of EAI tasks.

Adapting pre-trained visual representations. When and how to adapt PVRs for downstream applications remain open research questions (Kumar et al., 2022; Wijmans et al., 2022; Kirichenko et al., 2022; Lee et al., 2022; Goyal et al., 2022). In the context of EAI, Parisi et al. (2022) show that naively finetuning PVRs through behavior cloning reduces performance, whereas Radosavovic et al. (2022) finds the same strategy to slightly improve performance on real-world robot tasks. In an online RL setting, Yadav et al. (2022b) show that fine-tuning using policy loss improves visual navigation performance considerably. By comparison, Pari et al. (2021) find simple k-nearest-neighbor adaptation works well for real-world visual imitation tasks. Our work neither aims nor expects to be the final word on this fertile topic.

3. Benchmarking Progress Towards an Artificial Visual Cortex for Embodied AI

This section describes CORTEXBENCH, a curated set of EAI tasks designed to evaluate the ability of pre-trained visual representations (PVRs) to support a wide variety of EAI applications. Specifically, CORTEXBENCH includes 17 tasks drawn from 7 existing EAI benchmarks as shown in Figure 1. For each task, we delineate a downstream policy learning paradigm (e.g., few-shot imitation learning) and evaluation protocol that follows community standards in each domain (Section 3.2). By fixing the tasks and downstream learning methods, we are able to focus our evaluations on the contribution of PVRs, which allows us to measure progress towards the development of an artificial visual cortex for embodied intelligence. We use CORTEXBENCH to conduct the largest and most comprehensive empirical study to-date of PVRs from prior work (Section 4).

3.1. Embodied AI Tasks in CORTEXBENCH

CORTEXBENCH includes tasks from 7 benchmarks listed in Table 1, illustrated in Figure 1, and described here:

Adroit (AD) (Rajeswaran et al., 2018) is a suite of challenging dexterous manipulation tasks in which an agent must control a 28-DoF anthropomorphic hand to perform a variety of tasks. We study the two hardest tasks from Adroit: `Relocate` and `Reorient-Pen`. In these tasks, an agent must manipulate an object into a goal position and orientation, where the goal must be inferred from the scene.

MetaWorld (MW) (Yu et al., 2020) is a collection of tasks in which an agent commands a Sawyer robot arm to manipulate objects in a tabletop environment. We consider

Table 1. CortexBench includes tasks from 7 diverse benchmarks with different combinations of observations, actions, and goals as well as different standard policy learning paradigms.

Benchmark Suite	Observation Space	Action Space	Goal Specification	Policy Learning
Adroit (AD)	RGB + proprio.	Continuous	-	IL
Metaworld (MW)	RGB + proprio.	Continuous	-	IL
DMControl (DMC)	RGB + proprio.	Continuous	-	IL
Trifinger (TF)	RGB + proprio.	Continuous	Goal Image/Position	IL
ObjectNav (ON)	RGB + proprio.	Discrete	Object Category	IL
ImageNav (IN)	RGB	Discrete	Goal Image	RL
MobilePick (MP)	RGB + proprio.	Continuous	Goal Position	RL

five tasks from MetaWorld: `Assembly`, `Bin-Picking`, `Button-Press`, `Drawer-Open`, and `Hammer`, which follows the evaluations performed in (Nair et al., 2022).

DeepMind Control (DMC) (Tassa et al., 2018) is a widely studied benchmark for image-based continuous control in which an agent performs low-level locomotion and object manipulation tasks. We consider five tasks from DMC: `Finger-Spin`, `Reacher-Hard`, `Cheetah-Run`, `Walker-Stand`, and `Walker-Walk`, which follows the work in (Parisi et al., 2022).

TriFinger (TF) is a robot, introduced in (Wüthrich et al., 2020), that is composed of a three-finger hand with 3-DoF per finger. We consider two TriFinger tasks: `Reach-Cube` and `Push-Cube`. The `Push-Cube` task was part of the Real Robot Challenge 2020 (rrc, 2020). We also consider the easier `Reach-Cube` task, which (Dittadi et al., 2021) also studies. In these tasks, the agent must either touch the cube with one finger (`Reach-Cube`) or push the cube and move it to a goal location (`Push-Cube`).

Habitat (Savva et al., 2019a) is a simulation platform that includes several visual navigation tasks in which agents explore highly photo-realistic unseen 3D environments. We consider two semantic navigation tasks in Habitat: image-goal navigation (`ImageNav`) (Zhu et al., 2017) and object-goal navigation (`ObjectNav`) (Batra et al., 2020). In both tasks, the agent starts at a random location in an unknown 3D environment and must find a goal location – specified with an image taken from the goal location in `ImageNav` or with the name of an object (e.g., 'chair') in `ObjectNav`. Evaluation is conducted on unseen environments, thus testing the generalization capabilities of the visual encoder and policy.

Habitat 2.0 (Szot et al., 2021) includes a set of mobile manipulation tasks in which an agent controls a Fetch robot with a 7-DoF arm, mobile base (Gu et al., 2022), and suction gripper to rearrange objects in apartment scenes. We consider a challenging version of the `Mobile-Pick` (MP) task from Habitat 2.0, in which an agent must pick up a target object from a cluttered receptacle (e.g., a counter) while starting from a position in which the object is outside of the robot's reach (thus, requiring navigation). We relax the dense goal specification as described in Appendix A.5.

3.2. Downstream Policy Learning

Given a frozen PVR, an agent needs to learn a policy for each task. The EAI community has developed a range of policy learning algorithms from few-shot imitation learning (IL) to large-scale reinforcement learning (RL). For each task in CortexBench, we conform to the community standard for achieving state-of-art performance in that domain.

"MuJoCo Tasks" On the tasks from the Adroit, MetaWorld, and DMC suites we train policies using behavior cloning on a small number of expert demonstrations (100 for Adroit and DMC and 25 for MetaWorld), which follows Parisi et al. (2022); Nair et al. (2022). Specifically, we train policies for 100 epochs and report the average rollout performance on the test set for the best intermediate policy during training. For all tasks, the policy is a 3-layer MLP. When using vision transformers (ViT) based PVRs, we use the `[CLS]` token as input to the policy, and with ResNets we use features from the final convolutional layer after global average pooling. These design choices follow prior work such as Radosavovic et al. (2022); Nair et al. (2022).

"Trifinger Tasks" For TriFinger, we train policies using behavior cloning on 100 demonstrations per task. Specifically, we train a policy network composed of a 3-layer MLP for 100 epochs for `Reach-Cube` and 1,000 epochs for `Move-Cube`. We report the average score for the best checkpoint over the course of training. As in the "MuJoCo Tasks", the input to the policy is the `[CLS]` token for ViT-based PVRs and average pooled features from the last convolutional layer for ResNet-based models.

"Habitat Tasks" We train policies for `ObjectNav` using behavior cloning on 77k human demonstrations (Yadav et al., 2022c) collected by Habitat-Web (Ramrakhya et al., 2022b) for a total of 360M environment steps. For `ImageNav` and the Habitat 2.0 `Mobile-Pick` task, we use RL for 500M environment steps with DD-PPO (Wijmans et al., 2020) and VER (Wijmans et al., 2022). We use patch representations for ViT-based PVRs and grid-features from last convolutional layer for ResNet models, passed through a compression layer (Savva et al., 2019a) for a lower dimensional representation for use by the policy layers, which is a 2-layer LSTM for navigation and a 2-layer GRU for manipulation.

More details on tasks and downstream learning can be found in Appendix A.5.

4. Do we already have a foundation model?

First, we evaluate several existing pre-trained visual representations (PVRs) on CortexBench to study whether existing open-sourced visual backbones can consistently

		Imitation Learning					Reinforcement Learning		Mean	
#	Model	Adroit	MetaWorld	DMControl	Tri-Finger	ObjectNav	ImageNav	Mobile Pick	Rank	Success
1	Best prior result (any setting)	75	80	77	-	70.4	54.2	-		
2	Best prior result (Frozen PVR)	**75**	80	77	-	54.4	33.0	-		
3	Random (ViT-B) Frozen	2.0 ± 2.0	0.5 ± 0.5	10.1 ± 0.6	57.7 ± 0.7	21.1	42.1	10.8	7.1	20.6
4	Random (ViT-L) Frozen	2.7 ± 1.8	0.5 ± 0.5	9.1 ± 0.2	56.9 ± 0.1	20.5	45.1	20.6	7.1	22.2
5	Random (ViT-B) Fine-tuned	44.0 ± 2.0	49.9 ± 7.3	43.5 ± 2.4	53.4 ± 0.8	41.5	65.0	47.6	5.3	48.9
6	MVP (ViT-B)	48.0 ± 3.3	91.2 ± 2.9	65.9 ± 2.4	61.4 ± 1.0	52.4	64.7	56.0	3.1	63.1
7	MVP (ViT-L)	53.3 ± 4.1	87.5 ± 3.4	69.2 ± 1.5	$\mathbf{73.9 \pm 0.3}$	59.4	**68.1**	**65.4**	2.1	68.1
8	CLIP (ViT-B)	47.3 ± 3.0	75.5 ± 3.4	55.5 ± 1.4	64.4 ± 1.5	**61.3**	52.2	49.8	3.9	58.0
9	VIP (RN-50)	54.0 ± 4.8	90.1 ± 2.2	72.5 ± 2.7	72.0 ± 0.6	24.6	48.4	7.2	4.0	52.8
10	R3M (RN-50)	73.3 ± 2.0	$\mathbf{96.0} \pm 1.1$	$\mathbf{81.1} \pm 0.7$	69.7 ± 0.5	28.1	30.5	33.2	3.4	58.9

Table 2. Performance of different **frozen** pre-trained visual representations on a diverse suite of evaluation domains. Best prior results means that the results are the best reported in literature prior to this work. Overall, we find that no single PVR consistently performs the best across all benchmarks. However, we find that several of these pre-trained models often outperform a random training from scratch baseline. Best prior results sources (row 1): Adroit and MetaWorld approximated from (Nair et al., 2022), DMControl from (Parisi et al., 2022), ImageNav from (Yadav et al., 2022b), ObjectNav from (Ramrakhya et al., 2023). Frozen PVR Sources (row 2): Adroit, MetaWorld, and DMControl are the same as SOTA, ImageNav from (Yadav et al., 2022b), ObjectNav from (Deitke et al., 2022b).

perform well across all tasks. For all evaluations we consider frozen visual representations to disentangle the effect of learned representations from downstream task learning. Specifically, we include the following models:

- CLIP (Radford et al., 2021) Contrastive image-language pre-training objective; Trains on 400M images-text pairs from the internet (WIT); ViT-B backbone.
- R3M (Nair et al., 2022) Time-Contrastive video-language alignment pre-training objective; Trains on 5M images from a subset of Ego4D; ResNet-50 backbone.
- MVP (Radosavovic et al., 2022). Pre-trains with MAE; Trains on 4.5M images from Egocentric videos and ImageNet; ViT-B and ViT-L backbones.
- VIP (Ma et al., 2022). Goal-conditioned value function pre-training objective; Trains on 5M images from a subset of Ego4D; ResNet-50 backbone.

These models cover a wide range of architectures, pre-training objectives, and pre-training datasets, constituting a solid set for comparisons. Additionally, we include randomly initialized ViTs with both frozen weights and fine-tuned weights to assess the necessity of pre-training and the limitations of pure end-to-end in-domain learning.

We use two metrics to evaluate overall performance: Mean Success and Mean Rank. Mean Success: the average success rate across all benchmarks. Mean Rank: for each benchmark, we rank PVRs based on their success rate; then we average these rankings across all benchmarks.

Table 2 shows the evaluation results aggregated by benchmark; no single model excels in all cases. Among all of the models evaluated, R3M performs the best on Adroit, MetaWorld, and DMControl. While MVP (ViT-L) performs best on Trifinger, ImageNav, and Mobile Pick. CLIP, on the other hand, achieves the best results on ObjectNav.

The variance in performance of existing PVRs on CortexBench is further illustrated in Figure 5 (in Appendix A.3). Indeed, PVRs can be successful on some benchmarks but fail on others; for instance, while CLIP is the best model for ObjectNav (ranking first), its performance is poor on Adroit and MetaWorld (ranking fifth). This variance highlights that we do not yet have one strong performing artificial visual cortex for embodied AI yet.

5. Analyzing the Scaling Hypothesis for EAI

The previous section investigated models pre-trained on datasets of varying size and diversity. Interestingly, while the model pre-trained on the largest dataset (CLIP) performs well on one benchmark it does not perform well across all tasks. We now ask: how much does the relevance and diversity of the pre-training dataset and the model size matter? To study this, we fix the pre-training objective – MAE (He et al., 2021) – and then vary the composition of the pre-training dataset and the size of the visual backbone (ViT-B with 86M parameters and ViT-L with 307M parameters). We measure the corresponding changes in performance on CortexBench. MAE is selected for these experiments due to the strong performance on CortexBench of the MVP (Radosavovic et al., 2022) models (Table 2), which use the MAE pre-training objective.

5.1. Constructing a Pretraining Dataset for EAI

In order to evaluate the effect of dataset size and diversity on our benchmark tasks, which encompass a range of navigation and manipulation tasks, we select a combination of seven datasets for our study. These datasets include Ego4D (Grauman et al., 2022), 100 Days of Hands (100DOH) (Shan et al., 2020), Something-Something v2 (SS-V2) (Goyal et al., 2017), and Epic Kitchens (Damen

Name	Contains	Total Frames	Frames used
Ego4D+M	Ego4D	418,578,043	2,790,520
	100DOH	99,899	99,899
	SS-v2	25,209,271	315,115
	Epic Kitchens	19,965,439	332,757
		Total	3,538,291
Ego4D+N	Ego4D	418,578,043	2,790,520
	OpenHouse24	27,806,971	499,442
		Total	3,289,962
Ego4D+MN$^+$	Ego4D+M	3,538,291	3,538,291
	OpenHouse24	27,806,971	499,442
	RealEstate10K	10,000,000	303,087
		Total	4,340,820
Ego4D+MN$^+$I	Ego4D+MN$^+$	4,340,820	4,340,820
	ImageNet	1,281,167	1,281,167
		Total	5,621,987

Table 3. Overview of the assembled datasets used for our scaling hypothesis experiments, using up to 5.6M frames.

et al., 2018), which feature videos of people manipulating objects and is similar to datasets used in MVP (Radosavovic et al., 2022). In addition, we include two indoor navigation datasets: the Real Estate 10K dataset (Zhou et al., 2018) and OpenHouse24 dataset (see Appendix A.2 for details). Finally, we include ImageNet (Deng et al., 2009) as an out-of-domain dataset. It is infeasible to study the power set of dataset combinations. Therefore, we strategically select specific combinations of datasets as shown in Table 3 to answer specific questions about data diversity and size:

Ego4D+M is a combination of Ego4D, 100DOH, SS-v2 and Epic Kitchens totaling 3.5 million frames. The majority of theses datasets consist of manipulation settings, making this dataset primarily manipulation focused.

Ego4D+N combines the Ego4D dataset (manipulation focused) with OpenHouse24, an egocentric indoor navigation dataset, totaling 3.3 million frames. In comparison to **Ego4D+M**, **Ego4D+N** drops the additional manipulation datasets in favor of a navigation dataset that has less data. This allows us to investigate the question: *Is cross-domain data more useful than additional within-domain data?*

Ego4D+MN$^+$ combines Ego4D with 3 manipulation-focused datasets and 2 navigation-focused datasets, totalling 4.3M frames. In comparison to **Ego4D+N**, **Ego4D+MN$^+$** allows us ask: *Does scaling the data from same domains (manipulation and navigation) improve the representations?*

Ego4D+MN$^+$I: Finally, we combine all manipulation and indoor navigation datasets with ImageNet. This allows us the ask: *Does it help to add out-of-domain datasets?*

5.2. Findings

The full set of results is shown in Figure 3 and Table 4. We now now describe the key take-aways.

Model Size. We find that increasing model size does have a positive impact on performance on CortexBench. Figure 3 shows that scaling model size from ViT-B to ViT-L results in an improved performance averaged across all benchmarks (Figure 3a) and across all datasets (Figure 8 in the appendix). However, we find exceptions where this trend does not hold. For instance, we find that the ViT-B model pre-trained on **Ego4D+MN$^+$I** outperforms the larger ViT-L model on MetaWorld and Trifinger, as shown in Table 4.

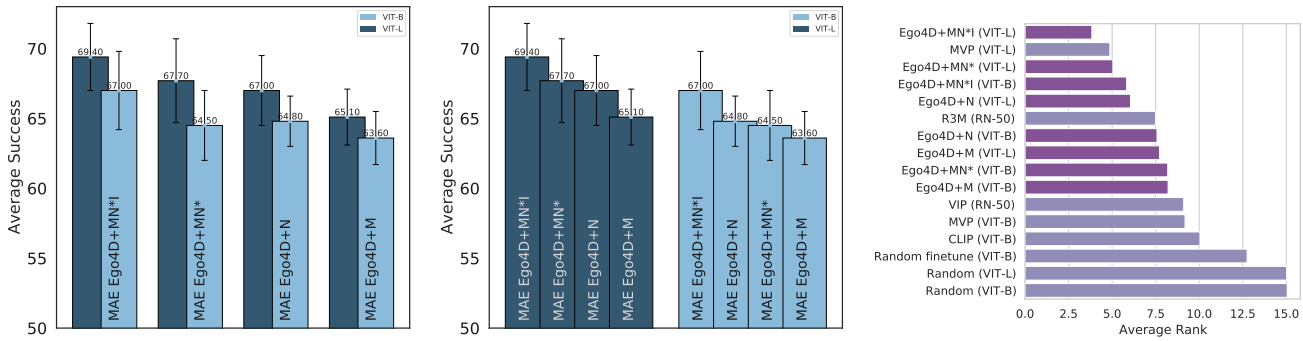
Dataset Diversity. Figure 3b shows that increasing the dataset diversity leads to improved performance – models that include both manipulation and navigation data outperform manipulation-only models on average across benchmarks. For instance, **Ego4D+MN$^+$** outperforms **Ego4D+M** by +0.9 points when using ViT-B (63.6 $\rightarrow$ 64.5) and by +2.6 when using ViT-L (65.1 $\rightarrow$ 67.7). This is also true when comparing **Ego4D+N** to **Ego4D+M**, where **Ego4D+N** outperforms **Ego4D+M** by +1.2 for ViT-B (63.6 $\rightarrow$ 64.8) and by +1.9 for ViT-L (65.1 $\rightarrow$ 67.0), which suggests that cross-domain diversity is more important than within-domain diversity, even at the cost of smaller dataset size.

Interestingly, pre-training on the **Ego4D+N** dataset outperforms pre-training on the larger **Ego4D+MN$^+$** dataset for ViT-B (Table 4 rows 4 & 8) indicating that increasing within-domain data isn't always better. Yet, for ViT-L (Table 4 rows 5 & 9) the additional data improves performance, which shows that larger models are needed to make use of extra data.

Finally, we observe that adding ImageNet data positively impacts average performance. For instance, models pre-trained on **Ego4D+MN$^+$I** outperform models pre-trained on **Ego4D+MN$^+$** by +2.5 points (64.5 $\rightarrow$ 67.0) on the ViT-B and +1.7 for the ViT-L (67.7 $\rightarrow$ 69.4), demonstrating that internet data can boost performance on EAI tasks.

When further dis-aggregating results in Table 4, we find that while we do generally observe that the model trained on the more diverse **Ego4D+MN$^+$I** dataset perform well, the *order* between **Ego4D+MN$^+$**, **Ego4D+N**, and **Ego4D+M** is mixed depending on the benchmark. For example, the best model for Mobile Pick is a ViT-L trained only on **Ego4D+M**– highlighting that naïvely scaling dataset diversity and size does not always improve performance.

While not dominant, the model pre-trained on the **Ego4D+MN$^+$I** dataset demonstrates effectiveness across all benchmarks. Our largest model (ViT-L) pre-trained on all data (**Ego4D+MN$^+$I**), which we call **VC-1**, consistently ranks either first or second across all tasks (Table 4 row 11), with a mean rank of 1.6. This performance is superior to the second-best model (**Ego4D+MN$^+$** ViT-L, Table 4 row 9) that has an average rank of 3.3.



(a) scaling model size (b) dataset diversity effect (c) Average ranking across all benchmarks.

Figure 3. Scaling experiments: Visualizing model performance averaged across all benchmarks in Table 4. Overall, we demonstrate modest but positive scaling trends in both (a) scaling model size, and (b) dataset diversity. Figure (c) shows the ranking of different PVRs and MAE models pre-trained on different datasets. Full experiments: we compare existing PVR models (Table 2) and scaling models (Table 4) by showcasing their ranking across all benchmarks, **VC-1**: **Ego4D+MN$^+$I** (VIT-L) achieves the highest average rank.

#	Benchmark	Adroit	Meta-World	DMControl	Trifinger	ObjectNav	ImageNav	Mobile Pick	Mean Rank	Mean Success
1	Best know result (any setting)	75	80	77	-	70.4	54.2	-		
2	Rand (ViT-B) fine-tuned	44.0	49.9	43.5	53.4	41.5	65.0	47.6		
3	Best result Table 2 (Frozen PVR)	75	96.0	81.1	73.9	61.3	68.1	65.4		
4	**Ego4D+N** (VIT-B)	54.0 ± 4.0	85.3 ± 5.2	63.0 ± 2.2	65.6 ± 0.3	51.1	66.6	59.6	5.9	64.8
5	**Ego4D+N** (VIT-L)	56.7 ± 3.5	88.5 ± 3.5	66.9 ± 2.4	71.7 ± 0.5	59.3	69.2	58.0	3.7	67.0
6	**Ego4D+M** (VIT-B)	51.3 ± 2.4	83.5 ± 2.6	64.3 ± 1.8	71.4 ± 0.8	48.9	65.8	59.8	6.4	63.6
7	**Ego4D+M** (VIT-L)	52.0 ± 1.3	88.3 ± 3.2	64.7 ± 2.4	65.8 ± 0.9	50.7	65.5	**68.6**	5.3	65.1
8	**Ego4D+MN$^+$** (VIT-B)	48.7 ± 2.4	85.3 ± 5.2	64.2 ± 1.9	69.9 ± 0.6	55.9	68.9	58.6	6.2	64.5
9	**Ego4D+MN$^+$** (VIT-L)	52.7 ± 4.2	86.7 ± 3.9	**69.7** ± 3.3	71.6 ± 0.8	62.8	69.1	61.2	3.3	67.7
10	**Ego4D+MN$^+$I** (VIT-B)	54.0 ± 4.0	**89.6** ± 3.9	63.8 ± 2.7	**73.5** ± 0.6	59.5	67.9	60.6	3.6	67.0
11	**VC-1**: **Ego4D+MN$^+$I** (VIT-L)	**59.3** ± 5.2	88.8 ± 2.2	66.9 ± 2.4	72.7 ± 0.8	**64.6**	**70.3**	63.2	**1.6**	**69.4**

Table 4. Average success per benchmark of scaling hypothesis models. We highlight the best model from the set of models trained to evaluate the scaling hypothesis in bold. We find that on average the **VC-1 EGO4D+MN$^+$I** (VIT-L) model performs best, but is not the best for each benchmark. Our best model outperforms in-domain from scratch learning on all benchmarks.

5.3. How does VC-1 compare to existing PVRs?

We now compare **VC-1** with existing PVRs from Section 4. On average it ranks as the best model across all benchmarks Figure 3c. We focus on R3M, MVP, and CLIP, since they achieved the highest success in at least one benchmark; we also compare to finetuning from scratch to demonstrate the impact of end-to-end fine-tuning. In terms of mean success, **VC-1** (Table 4 row 11) outperforms MVP (ViT-L) by +1.3 points (68.1 → 69.4), R3M by +10.5 (58.9 → 69.4), CLIP by +11.4 (58.0 → 69.4), and end-to-end fine-tuning +20.5 (48.9 → 69.4).

Impressively, **VC-1** outperforms CLIP *on every benchmark* (Figure 4), despite being trained on a 70X smaller dataset, emphasizing the importance of relevant datasets. **VC-1** also outperforms finetuning from scratch on every benchmark, indicating that visual representations trained with out-of-domain data can outperform end-to-end learning.

When compared to R3M, **VC-1** demonstrates superior performance on average and on 4 out of 7 benchmarks (Figure 4). It is outperformed by R3M on Adroit, MetaWorld and DMControl benchmarks. It is unclear whether this gap is caused by the different training objective, pre-training dataset, or backbone. This highlights the need for comparable evaluations on benchmarks like CORTEXBENCH.

The MVP model is the most similar in terms of results, architecture, and pre-training objective to **VC-1**, with the main difference being the addition of a *convolutional stem* in MVP. **VC-1** outperforms MVP VIT-L by 1.3 points on mean success and performs better on four out of seven benchmarks, likely due to the use of a more diverse dataset.

Overall, **VC-1** is an effective model across a broad set of tasks and thus a reasonable starting point for novel EAI problems. However, it is not always the best performing model for a specific task. This leads us to theorize that there is a domain gap that might be bridged with dataset engineering or adaptation of the PVR.

6. Addressing Domain Gap with Adaptation

This section explores *adapting* pre-trained visual representations (PVRs) for specific downstream application domains.

Motivation. Datasets used for pre-training such as those studied in Section 5, will generally be curated from data sources such as real-world videos collected by people.

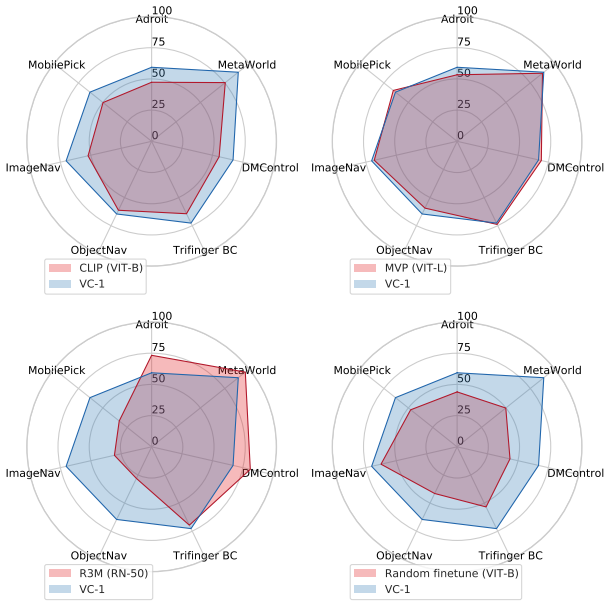


Figure 4. Performance of **VC-1** compared with existing PVRs. **VC-1** matches or exceeds existing PVRs on all benchmarks except R3M on AD, MW, and DMC, indicating an opportunity for model adaptation.

While these sources will capture diverse scenarios, we speculate that inevitably some downstream tasks or domains will have poor coverage. Indeed, significant research effort in computer vision has centered on understanding this challenge (e.g., Koh et al. (2021)). Here, we explore addressing domain gaps in the context of PVRs for embodied AI.

Specifically, we study a proxy for this challenge, by identifying a subset of tasks in CortexBench that have the largest visual difference to our training dataset. Since our training dataset comprises entirely of real-world images and video frames, we converge on 4 domains within CortexBench that do not have photorealistic rendering or textures: Adroit, MetaWorld, DMC, and Trifinger. We also observe that on this subset, **VC-1** underperforms PVRs from prior work (Table 4).

Approach. We evaluated adapting **VC-1**, which was trained on the **Ego4D+MN$^+$I** dataset, using a small amount of in-domain data. Specifically, we use the same data that would otherwise be used for learning behavior policies in that domain – i.e., frames from the expert demonstration in {Adroit, MetaWorld, DMC, Trifinger}. We start by initializing with pretrained weights from **VC-1**, and perform domain-specific adaptation of the PVR using MAE (He et al., 2021) – the same SSL algorithm used to pre-train VC-1. We perform this SSL-based adaptation for 100 epochs. To isolate the importance of the pre-trained VC-1 initialization, we also train 5 in-domain MAE models using the same in-domain data, but initialized with random weights. Subsequently, we evaluate all these adapted models in their respective domains, in frozen mode as described in Section 3.2.

#	Method	AD	MW	DMC	TF
1	Best prior result (Frozen PVR)	75	80	77	-
2	Best prior Table 4 (Frozen PVR)	75	96.0	81.1	73.9
3	In-Domain MAE Baseline	47.3	83.4	77.6	80.6
4	**VC-1**	59.3	88.8	66.9	72.7
5	**VC-1-Adapted**	72.0	96.0	80.9	80.1

Table 5. Results of adapting **VC-1** using MAE on in-domain data used for policy learning. We study 4 benchmarks in which **VC-1** underperforms the best scaling results in Table 4. We observe that adaptation leads to substantial improvements in all cases.

Proof-of-Concept Results. In Table 5, we observe that adaptation substantially improves the performance of **VC-1** in all 4 domains. On Adroit performance improves by +18.7 (59.3 $\rightarrow$ 72.0), MetaWorld by +7.2 (88.8 $\rightarrow$ 96.0), DMC by +14.0 (66.9 $\rightarrow$ 80.9), Trifinger by +7.4 (72.7 $\rightarrow$ 80.1). Interestingly, we find in DMC and Trifinger that the in-domain MAE baseline (Table 5 row 3) performs surprisingly well, highlighting the importance of in-domain data for representation learning. We observe that **VC-1-Adapted** surpasses the best performance from published work that evaluates PVRs in frozen mode (Table 5 row 1 vs. 5). Additionally, on 3 benchmarks, MetaWorld, DMC, and Trifinger is comparable with the frozen mode SoTA (Table 5 row 2 vs. 5) – established earlier in this paper.

7. Discussion

This work introduced CortexBench, which comprises of 17 different embodied AI (EAI) task spanning locomotion, indoor navigation, and dexterous and mobile manipulation. Enabled by CortexBench, we performed the most comprehensive study to-date of visual foundation models for EAI. Specifically, we evaluated state-of-art open-sourced foundation models and find that we do not yet have a strong backbone for all tasks. However, models trained via masked auto-encoders (MAEs) are the most promising. Furthermore, our study finds that naively scaling model size and pre-training data diversity does not improve performance universally across all tasks, but does so on average. Finally, we show by adding a second pre-training step, on a small in-domain dataset, we can be competitive with or outperform the best PVRs in this setting.

One of our primary contentions is that in order for the research community to make progress on foundation models for EAI, we need to develop strong benchmarks – for a PVR to be foundational, it must be broadly applicable. Furthermore, as a community we should converge on best practices and a rigorous reproducible experimental methodology; we hope CortexBench will help the community make progress towards that.

References

Real robot challenge 2020. https://real-robot-challenge.com/2020, 2020. 4

Al-Halah, Z., Ramakrishnan, S. K., and Grauman, K. Zero experience required: Plug & play modular transfer learning for semantic visual navigation. *arXiv preprint arXiv:2202.02440*, 2022. 13

Baevski, A., Babu, A., Hsu, W.-N., and Auli, M. Efficient self-supervised learning with contextualized target representations for vision, speech and language. *arXiv preprint arXiv:2212.07525*, 2022a. 2

Baevski, A., Hsu, W.-N., Xu, Q., Babu, A., Gu, J., and Auli, M. Data2vec: A general framework for self-supervised learning in speech, vision and language. *arXiv preprint arXiv:2202.03555*, 2022b. 2

Bao, H., Dong, L., and Wei, F. Beit: Bert pre-training of image transformers. *ArXiv*, abs/2106.08254, 2021. 2

Batra, D., Gokaslan, A., Kembhavi, A., Maksymets, O., Mottaghi, R., Savva, M., Toshev, A., and Wijmans, E. Objectnav revisited: On evaluation of embodied agents navigating to objects. *arXiv preprint arXiv:2006.13171*, 2020. 4, 14

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020. 2

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. E. A Simple Framework for Contrastive Learning of Visual Representations. arXiv:2002.05709, 2020. 2

Chen, X., Xie, S., and He, K. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 2

Damen, D., Doughty, H., Farinella, G. M., Fidler, S., Furnari, A., Kazakos, E., Moltisanti, D., Munro, J., Perrett, T., Price, W., et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 720–736, 2018. 5

Deitke, M., Batra, D., Bisk, Y., Campari, T., Chang, A. X., Chaplot, D. S., Chen, C., D'Arpino, C. P., Ehsani, K., Farhadi, A., Fei-Fei, L., Francis, A., Gan, C., Grauman, K., Hall, D., Han, W., Jain, U., Kembhavi, A., Krantz, J., Lee, S., Li, C., Majumder, S., Maksymets, O., Martín-Martín, R., Mottaghi, R., Raychaudhuri, S., Roberts, M., Savarese, S., Savva, M., Shridhar, M., Sünderhauf, N., Szot, A., Talbot, B., Tenenbaum, J. B., Thomason, J., Toshev, A., Truong, J., Weihs, L., and Wu, J. Retrospectives on the embodied ai workshop. *arXiv preprint arXiv:2210.06849*, 2022a. 12

Deitke, M., VanderBilt, E., Herrasti, A., Weihs, L., Salvador, J., Ehsani, K., Han, W., Kolve, E., Farhadi, A., Kembhavi, A., et al. Procthor: Large-scale embodied ai using procedural generation. *arXiv preprint arXiv:2206.06994*, 2022b. 5

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009. 6

Dittadi, A., Träuble, F., Wüthrich, M., Widmaier, F., Gehler, P., Winther, O., Locatello, F., Bachem, O., Schölkopf, B., and Bauer, S. The role of pretrained representations for the ood generalization of reinforcement learning agents, 2021. URL https://arxiv.org/abs/2107.05686. 4

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020. 2

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021. 2

Fukushima, K. Cognitron: A self-organizing multilayered neural network. *Biological Cybernetics*, 20(3): 121–136, Sep 1975. ISSN 1432-0770. doi: 10.1007/BF00342633. URL https://doi.org/10.1007/BF00342633. 1

Fukushima, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36 (4):193–202, Apr 1980. ISSN 1432-0770. doi: 10.1007/BF00344251. URL https://doi.org/10.1007/BF00344251. 1

Goyal, P., Caron, M., Lefaudeux, B., Xu, M., Wang, P., Pai, V., Singh, M., Liptchinsky, V., Misra, I., Joulin, A., et al. Self-supervised pretraining of visual features in the wild. *arXiv preprint arXiv:2103.01988*, 2021. 3

Goyal, R., Kahou, S. E., Michalski, V., Materzyńska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., Hoppe, F., Thurau, C.,

Bax, I., and Memisevic, R. The "something something" video database for learning and evaluating visual common sense, 2017. 5

Goyal, S., Kumar, A., Garg, S., Kolter, Z., and Raghunathan, A. Finetune like you pretrain: Improved finetuning of zero-shot vision models. *arXiv preprint arXiv:2212.00638*, 2022. 3

Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18995–19012, 2022. 5

Gu, J., Chaplot, D. S., Su, H., and Malik, J. Multi-skill mobile manipulation for object rearrangement. *arXiv preprint arXiv:2209.02778*, 2022. 4, 14

Gupta, A., Murali, A., Gandhi, D. P., and Pinto, L. Robot learning in homes: Improving generalization and reducing dataset bias. *Advances in neural information processing systems*, 31, 2018. 13

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. arXiv:2111.06377, 2021. 2, 5, 8, 15

Khandelwal, A., Weihs, L., Mottaghi, R., and Kembhavi, A. Simple but effective: Clip embeddings for embodied ai. In *CVPR*, 2022. 1, 2, 3

Kirichenko, P., Izmailov, P., and Wilson, A. G. Last layer re-training is sufficient for robustness to spurious correlations. *arXiv preprint arXiv:2204.02937*, 2022. 3

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., Lee, T., David, E., Stavness, I., Guo, W., Earnshaw, B., Haque, I., Beery, S. M., Leskovec, J., Kundaje, A., Pierson, E., Levine, S., Finn, C., and Liang, P. Wilds: A benchmark of in-the-wild distribution shifts. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5637–5664. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/koh21a.html. 8

Kumar, A., Raghunathan, A., Jones, R., Ma, T., and Liang, P. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022. 3

Lane, N. *Life Ascending: The Ten Great Inventions of Evolution*. W. W. Norton, 2010. ISBN 9780393071467. URL https://books.google.com/books?id=zHyK5z3mkWwC. 1

Lee, Y., Chen, A. S., Tajwar, F., Kumar, A., Yao, H., Liang, P., and Finn, C. Surgical fine-tuning improves adaptation to distribution shifts. *arXiv preprint arXiv:2210.11466*, 2022. 3

Ma, Y. J., Sodhani, S., Jayaraman, D., Bastani, O., Kumar, V., and Zhang, A. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022. 1, 2, 3, 5

Mezghani, L., Sukhbaatar, S., Lavril, T., Maksymets, O., Batra, D., Bojanowski, P., and Alahari, K. Memory-augmented reinforcement learning for image-goal navigation. *arXiv preprint arXiv:2101.05181*, 2021. 12

Nair, S., Rajeswaran, A., Kumar, V., Finn, C., and Gupta, A. R3M: A Universal Visual Representation for Robot Manipulation. *CoRL*, 2022. 1, 2, 3, 4, 5

Pari, J., Shafiullah, N. M., Arunachalam, S. P., and Pinto, L. The surprising effectiveness of representation learning for visual imitation. *arXiv preprint arXiv:2112.01511*, 2021. 3

Parisi, S., Rajeswaran, A., Purushwalkam, S., and Gupta, A. K. The Unsurprising Effectiveness of Pre-Trained Vision Models for Control. *ICML*, 2022. 1, 2, 3, 4, 5

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021. 2, 5

Radosavovic, I., Xiao, T., James, S., Abbeel, P., Malik, J., and Darrell, T. Real world robot learning with masked visual pre-training. In *6th Annual Conference on Robot Learning*, 2022. 1, 2, 3, 4, 5, 6

Rajeswaran, A., Kumar, V., Gupta, A., Vezzani, G., Schulman, J., Todorov, E., and Levine, S. Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations. In *Proceedings of Robotics: Science and Systems (R:SS)*, 2018. 2, 3

Ramakrishnan, S. K., Gokaslan, A., Wijmans, E., Maksymets, O., Clegg, A., Turner, J. M., Undersander, E., Galuba, W., Westbury, A., Chang, A. X., et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. 13

Ramrakhya, R., Undersander, E., Batra, D., and Das, A. 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*, pp. 5173–5183, 2022a. 14

Ramrakhya, R., Undersander, E., Batra, D., and Das, A. Habitat-web: Learning embodied object-search from human demonstrations at scale. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022b. 4

Ramrakhya, R., Batra, D., Wijmans, E., and Das, A. Pirlnav: Pretraining with imitation and rl finetuning for objectnav. *arXiv preprint arXiv:2301.07302*, 2023. 5

Savva, M., Kadian, A., Maksymets, O., Zhao, Y., Wijmans, E., Jain, B., Straub, J., Liu, J., Koltun, V., Malik, J., Parikh, D., and Batra, D. Habitat: A Platform for Embodied AI Research. In *International Conference on Computer Vision (ICCV)*, 2019a. 2, 4

Savva, M., Kadian, A., Maksymets, O., Zhao, Y., Wijmans, E., Jain, B., Straub, J., Liu, J., Koltun, V., Malik, J., et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9339–9347, 2019b. 13

Shan, D., Geng, J., Shu, M., and Fouhey, D. F. Understanding human hands in contact at internet scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9869–9878, 2020. 5

Szot, A., Clegg, A., Undersander, E., Wijmans, E., Zhao, Y., Turner, J., Maestre, N., Mukadam, M., Chaplot, D., Maksymets, O., Gokaslan, A., Vondrus, V., Dharur, S., Meier, F., Galuba, W., Chang, A., Kira, Z., Koltun, V., Malik, J., Savva, M., and Batra, D. Habitat 2.0: Training home assistants to rearrange their habitat. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 2, 4, 13, 14

Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., de Las Casas, D., Budden, D., Abdolmaleki, A., Merel, J., Lefrancq, A., Lillicrap, T. P., and Riedmiller, M. A. DeepMind Control Suite. arXiv:1801.00690, 2018. 1, 4

Tian, Y., Henaff, O. J., and van den Oord, A. Divide and contrast: Self-supervised learning from uncurated data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10063–10074, 2021. 3

Wijmans, E., Kadian, A., Morcos, A. S., Lee, S., Essa, I., Parikh, D., Savva, M., and Batra, D. DD-PPO: Learning Near-Perfect PointGoal Navigators from 2.5 Billion Frames. In *International Conference on Learning Representations (ICLR)*, 2020. 4

Wijmans, E., Essa, I., and Batra, D. VER: Scaling on-policy rl leads to the emergence of navigation in embodied rearrangement. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 3, 4

Wüthrich, M., Widmaier, F., Grimminger, F., Akpo, J., Joshi, S., Agrawal, V., Hammoud, B., Khadiv, M., Bogdanovic, M., Berenz, V., Viereck, J., Naveau, M., Righetti, L., Schölkopf, B., and Bauer, S. Trifinger: An open-source robot for learning dexterity. *CoRR*, abs/2008.03596, 2020. URL https://arxiv.org/abs/2008.03596. 2, 4

Xia, F., Zamir, A. R., He, Z., Sax, A., Malik, J., and Savarese, S. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9068–9079, 2018. 13

Yadav, K., Ramakrishnan, S. K., Turner, J., Gokaslan, A., Maksymets, O., Jain, R., Ramrakhya, R., Chang, A. X., Clegg, A., Savva, M., et al. Habitat challenge 2022, 2022a. 13, 14

Yadav, K., Ramrakhya, R., Majumdar, A., Berges, V.-P., Kuhar, S., Batra, D., Baevski, A., and Maksymets, O. Offline visual representation learning for embodied navigation. In *arXiv preprint arXiv:2204.13226*, 2022b. 1, 2, 3, 5

Yadav, K., Ramrakhya, R., Ramakrishnan, S. K., Gervet, T., Turner, J., Gokaslan, A., Maestre, N., Chang, A. X., Batra, D., Savva, M., et al. Habitat-matterport 3d semantics dataset. *arXiv preprint arXiv:2210.05633*, 2022c. 4, 13, 14

Yao, Y., Desai, N., and Palaniswami, M. Masked contrastive representation learning. *arXiv preprint arXiv:2211.06012*, 2022. 2

Yu, T., Quillen, D., He, Z., Julian, R., Hausman, K., Finn, C., and Levine, S. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pp. 1094–1100. PMLR, 2020. 1, 3

Zhai, X., Kolesnikov, A., Houlsby, N., and Beyer, L. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12104–12113, 2022. 3

Zhou, T., Tucker, R., Flynn, J., Fyffe, G., and Snavely, N. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018. 6

Zhu, Y., Mottaghi, R., Kolve, E., Lim, J. J., Gupta, A. K., Fei-Fei, L., and Farhadi, A. Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning. *ICRA*, 2017. 4

A. Appendix

A.1. Limitations

The study presents a thorough examination of visual foundation models but has several limitations. Firstly, in proposing the benchmark, we sought to find a balance between task diversity and the computational resources required for evaluation. However, new and challenging benchmarks in Embodied AI, such as those presented in (Deitke et al., 2022a), continue to emerge and may merit inclusion in future studies to track progress in this field. Additionally, while we have focused on masked auto-encoders as the pre-training objective and ViT as the architecture in our study, there may be other SSL algorithms that exhibit different scaling behaviors or superior performance on the proposed datasets in our benchmark. Lastly, the adaptation step of the PVR model necessitates separate training on in-domain datasets, as well as careful tuning of hyperparameters such as the number of training epochs and sampling ratio of the dataset. This results in a significant effort to produce a separate adapted PVR model for each benchmark evaluated on our benchmark, and the overall effort increases proportionately with the number of benchmarks included in the study.

In conclusion, it is important to note that although we utilize real-world images and videos for pre-training our visual representation models (PVRs), the evaluation benchmarks used in this study serve as proxies for actual robotic tasks, and thus, the performance of the PVR models on real robots may differ from the rankings established in this study. Further research is necessary to fully evaluate the effectiveness of these models in real-world scenarios.

A.2. OpenHouse24 description

The OpenHouse24 dataset (OH24) is a collection of video walk-throughs of furnished residential real estate properties. Over 1600 homes are represented in the dataset, totaling 139 hours of video footage. Each home is traversed in a continuous shot with a stable HD RGB camera by an operator that efficiently visits each room. The dataset represents a diverse set of properties, including (but not limited to) small and large suburban homes, high-rise apartments, ranch homes, and condos. The ensuing walk-throughs range from under a minute to 14 minutes in length, with the average taking 5 minutes and 12 seconds. The dataset will be open-sourced by a separate research project.

A.3. Do we already have a foundation model? Additional Plot

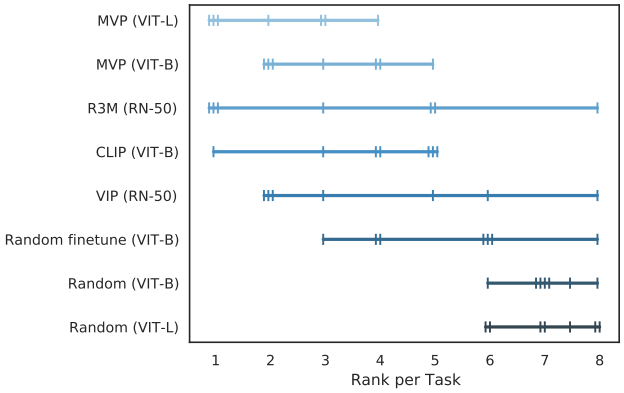


Figure 5. Rank distribution per model. For every model, we compute the ranks it achieved on each of the 7 benchmarks. We then visualize them as vertical lines, where each rank number x receives a tick if that model achieved such rank x. For instance, MVP (ViT-L) achieves ranks 1,1,1,2,3,3,4 across the 7 benchmarks. Significant variability exists in the performance of PVRs across benchmarks.

A.4. Does increasing model size improve performance on CortexBench? Additional Plot

A.5. CortexBench Tasks and Training Details

We discuss in more details task specification from Section 3.1 in this section.

`ImageNav` Benchmark. Our study conducts `ImageNav` experiments using the standard dataset presented in (Mezghani

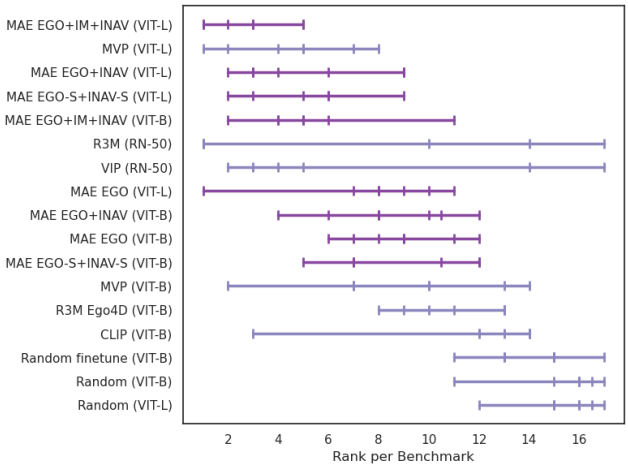


Figure 6. Rank distribution per model.

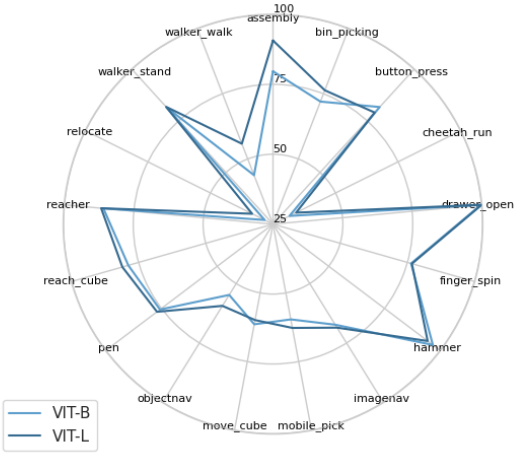


Figure 7. Average success per task.

et al., 2021). This benchmark utilizes the Habitat simulator (Savva et al., 2019b; Szot et al., 2021) and is situated within the Gibson (Xia et al., 2018) environments, which comprise 72 training scenes and 14 validation scenes. The validation set includes 300 episodes for each scene, for a total of 4,200 episodes. In this benchmark, agents are modeled as cylinders with a height of 1.5m, radius of 0.1m, and sensors located 1.25m above the center of the base. The RGB camera has a resolution of 128×128 and a 90° field-of-view. Agent is able to take up to 1000 steps within the environment and are deemed successful if they reach a location within 1m of the goal position and call STOPACTION.

To train the agents within the Gibson environments, we utilize 500M timesteps (25k updates) with 320 environments running in parallel. Each environment collects up to 64 frames of experience, which is followed by 2 PPO epochs utilizing 2 mini-batches. Unless otherwise specified, we use a learning rate of 2.5×10^{-4} for training the agents and update the parameters using the AdamW optimizer with a weight decay of 10^{-6}. We train agents with the reward functions presented in (Al-Halah et al., 2022) utilizing the following settings: success weighting $c_s = 5.0$, angle success weighting $c_a = 5.0$, goal radius $r_g = 1.0$, angle threshold $\theta_g = 25°$, and slack penalty $\gamma = 0.01$. We evaluate performance every 25M steps of training and report metrics based on the highest success rate (SR) achieved on the validation set.

`ObjectNav` Benchmark. We present an evaluation of object navigation (`ObjectNav`) using the HM3D-SEM dataset (Yadav et al., 2022c). The dataset is comprised of 80 training, 20 validation, and 20 testing scenes and utilizes the Habitat simulator (Savva et al., 2019b; Szot et al., 2021) and HM3D (Ramakrishnan et al., 2021) environments. Our results are reported on the v0.1 HM3D-SEM VAL split, which was used in the 2022 Habitat Challenge (Yadav et al., 2022a) `ObjectNav` benchmark. The agent in this evaluation is modeled after the LocoBot (Gupta et al., 2018) with a height of

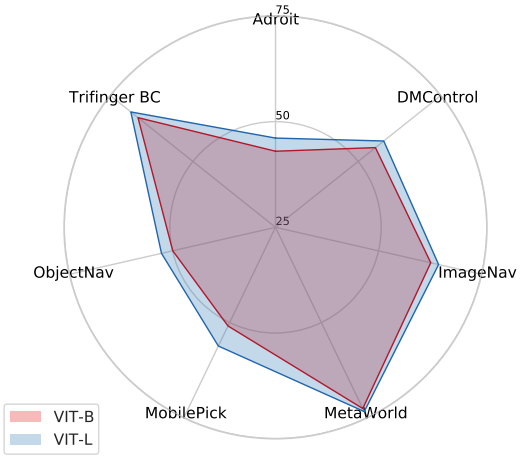


Figure 8. scaling model size per benchmark

0.88m, radius of 0.18m, and sensors placed at the top of the agent's head. The RGB camera has a 640×480 resolution and a 79° horizontal field of view. The task for the agent is to locate objects from one of 6 categories: *'chair'*, *'bed'*, *'plant'*, *'toilet'*, *'tv/monitor'*, and *'sofa'* within 500 steps. Successful episodes are determined by the agent stopping within 0.1m of a viewpoint that is (a) within 1m of any instance of the target object and (b) from which the object is visible, as outlined in the evaluation protocol of (Batra et al., 2020).

We utilize a dataset of human demonstrations for training our imitation learning agent in the task of `ObjectNav`. The dataset was collected using Habitat-Web (Ramrakhya et al., 2022a; Yadav et al., 2022c) and Amazon Mechanical Turk, and consists of $77k$ demonstrations for 80 scenes from the HM3D-SEM dataset (Yadav et al., 2022a). Each scene contains approximately 158 episodes, each with a unique goal object category and a randomly set start location, resulting in approximately 950 demonstrations per scene. The dataset includes a total of $\sim$12.1 million steps of experience, with an average of $\sim$159 steps per episode. By leveraging this human demonstration data, our imitation learning agent is able to learn a more effective policy for navigating to objects in complex environments.

We trained object navigation (`ObjectNav`) agent in the HM3D environment for an approximate total of 400 million steps, utilizing 25,000 updates and 512 parallel environments. Similar to our previous image-based navigation (`ImageNav`) experiments, we employed a weight decay of 10^{-6} and utilized different learning rates for the visual encoder and other elements of the model. Specifically, we used a learning rate of 10^{-4} for the visual encoder and 10^{-3} for all other elements, with the AdamW optimizer. To ensure the quality of our trained models, we evaluated checkpoints after every $10M$ steps and only reported metrics for the checkpoints with the highest validation success rate.

Habitat 2.0 Rearrangement We investigate the Habitat 2.0 Rearrangement task proposed by (Szot et al., 2021). This task involves a mobile manipulation scenario in which a Fetch robot navigates an ReplicaCAD apartment to pick up a target object from a cluttered receptacle using a mobile base (Gu et al., 2022). The robot starts from a non-trivial position and must utilize a variety of sensors, including an egocentric RGB camera, proprioceptive joint sensing, and an object grasping indicator. The action space for the robot includes continuous control over the robot's 7-DOF arm, base movement, and suction gripper. We relax the dense goal specification, where the relative position between the end-effector and the target object must be updated at each step, to a sparse goal specification, where this information is only provided at the start of the episode.

TriFinger Tasks The TriFinger tasks are implemented in Pybullet. For `Reach-Cube`, the state for the BC policy is $[x_t^{ft}, z_t]$, where x_t^{ft} is the current fingertip position and z_t is the latent visual state vector, obtained by passing the current image observation through the PVR. The success metric captures how close the fingertip is to the optimal distance from the center of the cube, accounting for the half=width of the cube. For `Move-Cube`, the state for the BC policy is $[x_t^{ft}, z_t, \Delta x_g^c]$, where Δx_g^c is the goal position for the cube, specified as a displacement from its initial position. Here the success is the distance of the center of the cube to the target goal position. We train a policy network with hidden layers of size 2000 and learning rate 10^{-4} for up to 100 epochs for the reach task and 1000 epochs for the move cube task.

A.6. Experiment Details of Training PVRs

To train the MAE models, we use the official codebase released by the authors on GitHub (He et al., 2021) and use the default hyperparameters provided by the repo to train the ViT-B and ViT-L models. We found the default values worked well on the CortexBench. However, we do vary the number of epochs we use to train the different models in Section 5 given the different dataset sizes. We choose the number of epochs per run such that the number of model updates remain constant across all runs and match the number of model updates taken by MAE on the ImageNet dataset. We provide details about the dataset sizes and the epochs calculated for the different runs in Table 6.

Dataset Name	Epochs	Frames used
Ego4D+N (VIT-B)	289	3,538,291
Ego4D+N (VIT-L)	289	3,538,291
Ego4D+M (VIT-B)	414	3,289,962
Ego4D+M (VIT-L))	414	3,289,962
Ego4D+MN$^+$ (VIT-B)	236	4,340,820
Ego4D+MN$^+$ (VIT-L)	236	4,340,820
Ego4D+MN$^+$I (VIT-B)	182	5,621,987
VC-1 (**Ego4D+MN$^+$I** (VIT-L))	182	5,621,987

Table 6. Experiment Details of Training PVRs.

task model	assembly	bin_picking	button_press	cheetah_run	drawer_open	finger_spin	hammer	imagenav	mobile_pick	move_cube	objectnav	pen	reach_cube	reacher	relocate	walker_stand	walker_walk
CLIP (VIT-B)	70.7	68.0	48.0	22.7	100.0	74.6	90.7	52.2	49.8	46.0	61.3	72.0	82.8	89.9	22.7	64.9	25.4
MAE Ego4D+M (VIT-B)	76.0	58.7	84.0	31.9	100.0	75.5	98.7	65.8	59.8	63.9	48.9	77.3	78.9	89.3	25.3	80.7	44.3
MAE Ego4D+M (VIT-L)	89.3	73.3	84.0	33.5	100.0	75.6	94.7	65.5	68.6	50.8	50.7	74.7	80.8	85.8	29.3	76.2	52.3
MAE Ego4D+MN* (VIT-B)	72.0	76.0	82.7	30.2	98.7	76.5	97.3	66.6	59.6	53.1	59.6	76.0	78.2	86.2	32.0	77.8	44.2
MAE Ego4D+MN* (VIT-L)	92.0	77.3	77.3	32.4	100.0	77.3	96.0	69.2	58.0	60.9	58.0	74.7	82.5	81.5	38.7	84.6	58.6
MAE Ego4D+MN*I (VIT-B)	88.0	78.7	82.7	32.3	100.0	76.0	98.7	67.9	60.6	65.0	59.5	76.0	82.1	82.6	32.0	83.5	44.7
MAE Ego4D+MN*I (VIT-L)	88.0	84.0	80.0	32.8	100.0	76.8	92.0	70.3	63.2	64.4	64.6	80.0	81.1	88.0	38.7	83.3	53.7
MAE Ego4D+N (VIT-B)	82.7	74.7	77.3	32.0	100.0	77.5	92.0	68.9	58.6	63.8	55.9	73.3	76.1	85.6	24.0	84.1	41.8
MAE Ego4D+N (VIT-L)	93.3	70.7	74.7	38.1	100.0	77.0	94.7	69.1	61.2	63.5	62.8	78.7	79.7	91.7	26.7	83.0	58.9
MVP (VIT-B)	92.0	73.3	92.0	33.9	100.0	76.9	98.7	64.7	56.0	48.6	54.6	69.3	74.2	86.3	26.7	84.7	47.9
MVP (VIT-L)	89.3	78.7	70.7	36.9	100.0	76.4	98.7	68.1	65.4	65.5	59.4	76.0	82.3	90.2	30.7	83.2	59.3
R3M (RN-50)	97.3	93.3	89.3	66.1	100.0	77.1	100.0	30.6	33.2	53.4	28.1	81.3	85.9	98.4	65.3	93.8	70.1
Random (VIT-B)	0.0	0.0	2.7	0.4	0.0	0.1	0.0	42.1	10.8	41.1	21.1	4.0	74.4	23.4	0.0	22.7	4.0
Random (VIT-L)	0.0	0.0	0.0	0.5	0.0	0.2	2.7	45.2	20.6	39.9	20.5	5.3	73.9	19.9	0.0	20.1	4.6
Random finetune (VIT-B)	61.3	34.7	20.0	10.2	40.0	48.6	93.3	62.5	47.6	34.3	41.5	73.3	72.6	26.8	14.7	73.6	58.1
VIP (RN-50)	93.3	76.0	88.0	53.2	100.0	76.1	93.3	48.8	7.2	58.0	24.6	81.3	86.1	83.2	26.7	86.6	63.4

Table 7. The success rate for each task and each model we evaluate during the study before being aggregated by benchmark.