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

***Abstract*—We present the largest and most comprehensive empirical study of pre-trained visual representations (PVRs) or visual 'foundation models' for Embodied AI. First, we curate CortexBench, consisting of 17 different tasks spanning locomotion, navigation, dexterous, and mobile manipulation. Next, we systematically evaluate existing PVRs and find that none are universally dominant.**

To study the effect of pre-training data scale and diversity, we combine over 4,000 hours of egocentric videos from 7 different sources (over 5.6M images) and ImageNet to train different-sized vision transformers using Masked Auto-Encoding (MAE) on slices of this data. Contrary to inferences from prior work, we find that scaling dataset size and diversity does *not* improve performance universally (but does so on average).

Our largest model, named VC-1, outperforms all prior PVRs on average but does not universally dominate either. Finally, we show that task- or domain-specific adaptation of VC-1 leads to substantial gains, with VC-1 (adapted) achieving competitive or superior performance than the best known results on all of the benchmarks in CortexBench. These models required over 10,000 GPU-hours to train and can be found on our website for the benefit of the research community.

I. Introduction

Eyesight is considered one of the greatest inventions of biological evolution [1]. Out of the 38 known phyla in the animal kingdom, only 6 have evolved eyes yet they account for 95% of all species [1] – vision seems to confer an enormous advantage. 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 [2, 3] 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 egocentric-videos and web-images can substantially improve performance and learning efficiency for navigation [4–6] and manipulation tasks [7–10]. 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]

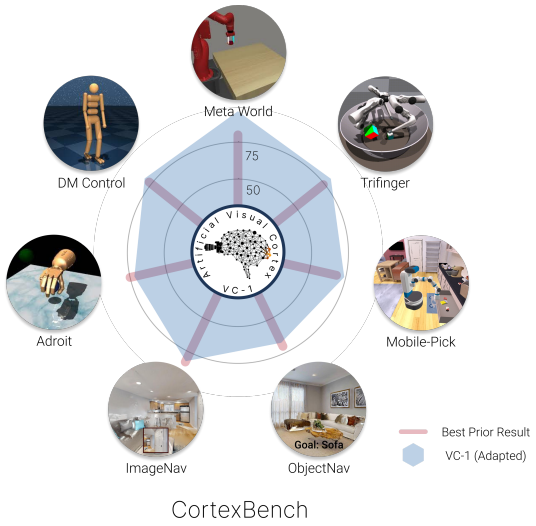


Fig. 1: An artificial visual cortex for embodied intelligence must support a diverse range of sensorimotor skills, environments, and embodiments; we curate CortexBench to systematically measure progress towards this ambitious goal. Our strongest model, denoted **VC-1** (adapted) above, is competitive with or outperforms the *best prior results* (success rates) on all benchmarks in CortexBench. Notice that this comparison is particularly unforgiving because the best prior results are benchmark-specific and not constrained to share any aspect of their design.

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 [11], table-top manipulation of rigid and articulated objects [12], dexterous manipulation [13], multi-finger coordinated manipulation [14], indoor visual navigation [15], and mobile manipulation [16]. 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

[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.

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*. 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.[3] However, curiously, the *kinds of PVRs* that are locally-dominant in CortexBench differ significantly in the size and type of pre-training datasets: CLIP [17] was pre-trained on 400M image-text pairs from the web; MVP [9] on 4.5M frames from web-images and many egocentric-video datasets; R3M [8] on ∼5M frames from Ego4D – yet, each performs best on some subset of tasks in CortexBench. This leads to a 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 of the diverse tasks in CortexBench?

To study these questions, we combine over 4,000 hours of egocentric videos from 7 sources containing humans manipulating objects and navigating indoor spaces, together with ImageNet. 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) [18] on these 4 datasets using Masked Auto-Encoding (MAE) [19], and systematically analyze their performance on CortexBench.

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 1.2% 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 data diversity – more is better on average, but not universally. For instance, the best performance on the `Mobile-Pick` task from Habitat 2.0 [16] is achieved by pre-training on the subset of video data focused on manipulation; presumably because the mobility involved in the task is fairly limited. 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 study *adapting* **VC-1** with either task-specific training losses or datasets (via MAE [19]) to specialize **VC-1** for each domain. We find that adapting **VC-1** results in it becoming competitive with or outperforming the *best prior results on all of the benchmarks* in CortexBench. We highlight that this comparison is particularly unforgiving, since best prior results are highly domain-specific and are not constrained to share any aspect of their design. To our knowledge, **VC-1** (adapted) is the first PVR that is competitive with (or outperforms) state-of-art results on such a diverse set of EAI tasks (Figure 1).

We will release code for CortexBench to enable the EAI, robotics, and CV communities to benchmark their own models, and share our pre-trained models (including **VC-1**) that we believe can serve as a starting point for all visuomotor tasks of interest today.

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

As shown in Figure 1, CortexBench includes 17 tasks drawn from 7 existing EAI benchmarks (tasks are detailed in Appendix C). 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 (Appendix D). 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 recommend two metrics to evaluate PVR 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.

A. Do we already have a foundation model?

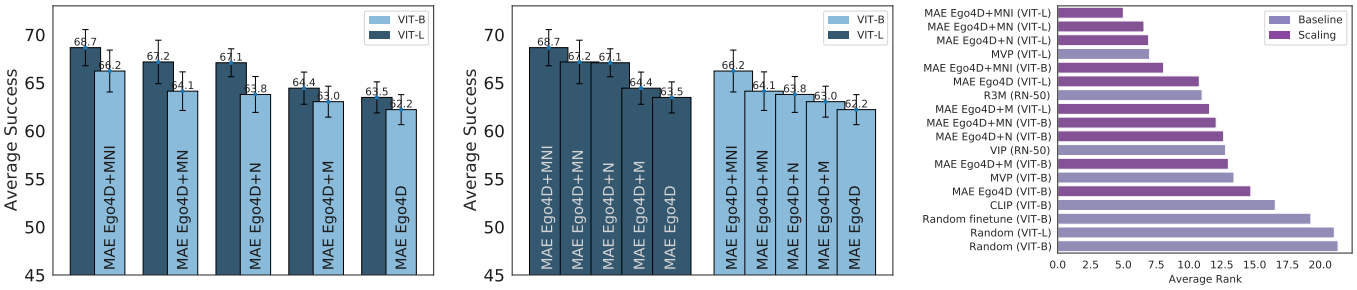
We use CortexBench to conduct the largest and most comprehensive empirical study to-date of PVRs from prior work. For all evaluations we consider frozen visual representations to disentangle the effect of learned representations from downstream task learning. Specifically, we include the following models:

- CLIP [17] Contrastive image-language pre-training objective; Trains on 400M images-text pairs from the internet (WIT); ViT-B backbone.
- R3M [8] Time-Contrastive video-language alignment pre-training objective; Trains on 5M images from a subset of Ego4D; ResNet-50 backbone.
- MVP [9]. MAE pre-training objective; Trains on 4.5M images from Egocentric videos and ImageNet; ViT-B and ViT-L backbones.
- VIP [10]. 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

#	Model	Imitation Learning					Reinforcement Learning		Mean	
		Adroit	MetaWorld	DMControl	Tri-Finger	ObjectNav	ImageNav	Mobile Pick	Rank	Success
1	Best prior result (any setting)	75	80	77	-	70.4	82.0	-		
2	Best prior result (Frozen PVR)	75	80	77	-	54.4	61.8	-		
3	Random (ViT-B) Frozen	2.0 ± 2.0	0.5 ± 0.5	10.1 ± 0.6	57.8 ± 0.5	19.2 ± 0.9	42.1 ± 0.8	10.8 ± 1.4	7.2	20.4
4	Random (ViT-L) Frozen	2.7 ± 1.8	0.5 ± 0.5	9.1 ± 0.2	57.2 ± 0.9	19.3 ± 0.9	45.2 ± 0.8	20.6 ± 1.8	6.9	22.1
5	Random (ViT-B) Fine-tuned	44.0 ± 2.0	49.9 ± 7.3	43.5 ± 2.4	56.1 ± 1.3	28.5 ± 1.0	62.5 ± 0.7	47.6 ± 2.2	5.3	47.4
6	MVP (ViT-B)	48.0 ± 3.3	91.2 ± 2.9	65.9 ± 2.4	59.7 ± 0.3	51.2 ± 1.1	64.7 ± 0.7	56.0 ± 2.2	3.1	62.4
7	MVP (ViT-L)	53.3 ± 4.1	87.5 ± 3.4	69.2 ± 1.5	74.1 ± 0.3	55.0 ± 1.1	68.1 ± 0.7	65.4 ± 2.1	2.1	67.5
8	CLIP (ViT-B)	47.3 ± 3.0	75.5 ± 3.4	55.5 ± 1.4	62.0 ± 0.5	56.6 ± 1.1	52.2 ± 0.8	49.8 ± 2.2	3.9	57.0
9	VIP (RN-50)	54.0 ± 4.8	90.1 ± 2.2	72.5 ± 2.7	66.7 ± 0.2	26.4 ± 1.0	48.8 ± 0.8	7.2 ± 1.2	4.0	52.3
10	R3M (RN-50)	73.3 ± 2.0	96.0 ± 1.1	81.1 ± 0.7	69.2 ± 0.8	22.7 ± 0.9	30.6 ± 0.7	33.2 ± 2.1	3.4	58.0

TABLE I: 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 [8], DMControl from [7], ImageNav from [5], ObjectNav from [20]. Frozen PVR Sources (row 2): Adroit, MetaWorld, and DMControl are the same as SOTA, ImageNav from [5], ObjectNav from [21].



(a) scaling model size (b) dataset diversity effect (c) ranking of all models

Fig. 2: Scaling experiments: Visualizing model performance averaged across all benchmarks in Table II. Overall, we demonstrate modest but positive scaling trends in both (a) scaling model size, and (b) dataset diversity. c) Average ranking across all benchmarks. We compare existing PVRs (baselines) (Table I) and scaling models (Table II) by showcasing their ranking across all benchmarks, **VC-1**: **Ego4D+MNI** (ViT-L) achieves the highest average rank.

fine-tuned weights to assess the necessity of pre-training and the limitations of pure end-to-end in-domain learning.

Table I shows the evaluation results aggregated by benchmark; no single model excels in all cases. Among all of the models, 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. These results indicate that we do not yet have one strong performing artificial visual cortex for embodied AI.

III. Analyzing the Scaling Hypothesis for EAI

In the previous section, we 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 (ObjectNav) 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 [19] – 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.

A. Constructing a Pre-training Dataset for EAI

To evaluate the impact of dataset size and diversity on our benchmark tasks, which involve various navigation and manipulation challenges, we employ a combination of nine datasets. These datasets include Ego4D [22], 100 Days of Hands (100DOH) [23], Something-Something v2 (SS-V2) [24], and Epic Kitchens [25]. This subset consists of videos showcasing people manipulating objects and are comparable to the datasets used in MVP [9]. Additionally, we use two egocentric indoor navigation datasets: the Real Estate 10K dataset [26] and the OpenHouse24 dataset (described in Appendix E1). Finally, we include ImageNet [27] as a representative static internet image dataset.

B. Scaling Hypothesis Findings

We use subsets of our pre-training dataset (described in Appendix E) to analyze the effect of increasing model size, dataset size, and dataset diversity. Results are shown in Figure 2 and Table II. Key takeaways are:

Model Size. We find that increasing model size positively impacts performance on CortexBench. Specifically, in Figure 2a, we find that with all pre-training datasets, switching from ViT-B to ViT-L improves average performance. However, in Table II, we find exceptions where

#	Benchmark	Adroit	Meta-World	DMControl	Trifinger	ObjectNav	ImageNav	Mobile Pick	Mean Rank	Mean Success
1	Best prior result (any setting)	75	80	77	-	70.4	82.0	-		
2	Rand (ViT-B) fine-tuned	44.0	49.9	34.2	55.0	28.5	65.0	47.6		
3	Best result Table I (Frozen PVR)	73.3	96.0	81.1	74.1	56.6	68.1	65.4		
4	Ego4D (VIT-B)	48.7 ± 1.3	86.1 ± 2.1	64.1 ± 2.3	68.3 ± 1.1	46.8 ± 1.1	64.0 ± 0.7	57.4 ± 2.2	8.6	62.2
5	Ego4D (VIT-L)	50.0 ± 1.2	92.9 ± 2.4	60.8 ± 3.3	69.7 ± 0.5	47.6 ± 1.1	55.8 ± 0.8	67.6 ± 2.1	5.9	63.5
6	Ego4D+N (VIT-B)	50.0 ± 2.4	86.4 ± 2.9	59.5 ± 2.4	67.8 ± 1.3	54.7 ± 1.1	68.7 ± 0.7	59.4 ± 2.2	7.2	63.8
7	Ego4D+N (VIT-L)	54.0 ± 1.2	89.1 ± 2.9	66.4 ± 1.7	66.9 ± 0.4	57.4 ± 1.1	70.5 ± 0.7	65.2 ± 2.1	3.5	67.1
8	Ego4D+M (VIT-B)	51.3 ± 2.4	83.5 ± 2.6	64.3 ± 1.8	69.1 ± 0.4	47.3 ± 1.1	65.8 ± 0.7	59.8 ± 2.2	7.0	63.0
9	Ego4D+M (VIT-L)	52.0 ± 1.3	88.3 ± 3.2	64.7 ± 2.4	64.7 ± 0.9	47.3 ± 1.1	65.5 ± 0.7	68.6 ± 2.1	6.0	64.4
10	Ego4D+MN (VIT-B)	48.7 ± 2.4	85.3 ± 5.2	64.2 ± 1.9	70.3 ± 0.5	52.8 ± 1.1	68.9 ± 0.7	58.6 ± 2.2	6.9	64.1
11	Ego4D+MN (VIT-L)	52.7 ± 4.2	86.7 ± 3.9	69.7 ± 3.3	72.4 ± 0.5	58.4 ± 1.1	69.1 ± 0.7	61.2 ± 2.2	3.1	67.2
12	Ego4D+MNI (VIT-B)	54.0 ± 4.0	89.6 ± 3.9	63.8 ± 2.7	72.2 ± 0.6	55.4 ± 1.1	67.9 ± 0.7	60.6 ± 2.2	4.4	66.2
11	**VC-1**: Ego4D + MNI (VIT-L)	59.3 ± 5.2	88.8 ± 2.2	66.9 ± 1.4	71.7 ± 0.4	60.3 ± 1.1	70.3 ± 0.7	63.2 ± 2.2	2.4	68.7

TABLE II: Average success per benchmark of scaling hypothesis models. On average the **VC-1 Ego4D+MNI** (VIT-L) performs best, but is not the best for each benchmark.

#	Method	Adroit	MetaWorld	DMControl	Tri-Finger	ObjectNav	ImageNav	Mobile Pick
1	Best prior result (any setting)	75	80	77	-	70.4	82.0	-
2	Best result from our experiments	73.3	96.0	81.1	74.1	60.3	70.5	68.6
3	In-domain MAE baseline	47.3	83.4	77.6	80.4	39.9	47.6	51.6
4	**VC-1**	59.3	88.8	66.9	71.7	60.3	70.3	63.2
5	**VC-1** E2E fine-tuning	15.9	22.7	6.7	70.9	67.7	81.6	74.0
6	**VC-1** MAE adaptation	72.0	96.0	80.9	80.6	57.4	67.0	62.4

TABLE III: Adapting **VC-1** with end-to-end fine-tuning or MAE adaptation improves performance.

this general trend does not hold. For instance, when pre-trained on **Ego4D+MNI**, the ViT-B model outperforms the ViT-L model on MetaWorld and Trifinger.

Dataset Size and Diversity. In Figure 2b, models are are ordered from right to left by increasing size and the diversity of their pre-training dataset. In general, we find that increasing dataset size and diversity mostly leads to improvements for both ViT-B and ViT-L.

Finally, on average, our largest model (ViT-L) pre-trained on all datasets (**Ego4D+MNI**), achieves the highest rank when averaged across all benchmark tasks (Table II row 11), with a mean rank of 2.4. This performance is superior to the second-best model (**Ego4D+MN** ViT-L, Table II row 9) that has an average rank of 3.1. We call this model **VC-1**, and will open-source it.

However, upon further dis-aggregation, we observe we find that while **VC-1** performs best on average, it is not the best for each benchmark. For example, the best model for Mobile Pick, a mobile manipulation task, is a ViT-L trained on **Ego4D+M** and the best model for ImageNav, an indoor navigation task, is the ViT-L trained on **Ego4D+N**. These findings suggest that task-specific pre-training datasets could enhance the performance of models on individual tasks. However, it is important to note that this approach would lead to multiple pre-trained models, each tailored to a specific task, and not a unified visual foundation model.

IV. Adapting VC-1

In prior sections, we focused on evaluating **VC-1** as a **frozen** PVR for EAI. We now study if *adapting* **VC-1** can improve results in downstream tasks. In the context of PVRs for EAI, adaptation can serve at least two purposes. The first is **task-specialization** in the feature extraction stage. Since **VC-1** was trained with MAE [19], it captures features that are generally useful for reconstructing images. Adaptation can specialize the visual backbone to extract features required for performing specific downstream tasks such as object rearrangement. Secondly, adaptation can also help **mitigate domain-gap** that might exist between pre-training and evaluation settings. In general, domain-gap can arise for several reasons such as poor coverage in pre-training data collection or deployment in novel conditions (e.g., on robots) not seen in the pre-training data (e.g., in human-centric video datasets). Domain gap is naturally instantiated in our setup, since **VC-1** was pre-trained on real-world, human video data while our downstream evaluation in CortexBench uses simulated EAI domains with different visual characteristics.

Table III studies two adaptation methods: end-to-end (E2E) fine-tuning and MAE adaptation in which we continue training **VC-1** with the MAE [19] pre-training objective on task-specific data. Overall, we find *adapting* **VC-1** results in competitive performance on all benchmarks. On MetaWorld, DMControl, and Tri-Finger **VC-1** with MAE adaptation (Table III row 6) is comparable with the best known results (SoTA) and the best results from previous sections (Table III rows 1 and 2). Similarly, on ImageNav and Mobile Pick, **VC-1** with E2E fine-tuning (Table III row 5) matches or exceeds the best results. Together, these results demonstrate that **adaptation** is a powerful paradigm for using PVRs for EAI.

V. 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.

References

[1] N. Lane, *Life Ascending: The Ten Great Inventions of Evolution*. W. W. Norton, 2010. [Online]. Available: https://books.google.com/books?id=zHyK5z3mkWwC 1

[2] K. Fukushima, "Cognitron: A self-organizing multilayered neural network," *Biological Cybernetics*, vol. 20, no. 3, pp. 121–136, Sep 1975. [Online]. Available: https://doi.org/10.1007/BF00342633 1

[3] ——, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, no. 4, pp. 193–202, Apr 1980. [Online]. Available: https://doi.org/10.1007/BF00344251 1

[4] A. Khandelwal, L. Weihs, R. Mottaghi, and A. Kembhavi, "Simple but effective: Clip embeddings for embodied ai," in *CVPR*, 2022. 1, 8

[5] K. Yadav, R. Ramrakhya, A. Majumdar, V.-P. Berges, S. Kuhar, D. Batra, A. Baevski, and O. Maksymets, "Offline visual representation learning for embodied navigation," in *arXiv preprint arXiv:2204.13226*, 2022. 3, 8

[6] K. Yadav, A. Majumdar, R. Ramrakhya, N. Yokoyama, A. Baevski, Z. Kira, O. Maksymets, and D. Batra, "Ovrl-v2: A simple state-of-art baseline for imagenav and objectnav," *arXiv preprint arXiv:2303.07798*, 2023. 1, 8

[7] S. Parisi, A. Rajeswaran, S. Purushwalkam, and A. K. Gupta, "The Unsurprising Effectiveness of Pre-Trained Vision Models for Control," *ICML*, 2022. 1, 3, 8, 9

[8] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3M: A Universal Visual Representation for Robot Manipulation," *CoRL*, 2022. 2, 3, 8, 9

[9] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell, "Real world robot learning with masked visual pre-training," in *6th Annual Conference on Robot Learning*, 2022. 2, 3, 8, 9

[10] Y. J. Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, and A. Zhang, "Vip: Towards universal visual reward and representation via value-implicit pre-training," *arXiv preprint arXiv:2210.00030*, 2022. 1, 2, 8

[11] Y. Tassa, Y. Doron, A. Muldal, T. Erez, Y. Li, D. de Las Casas, D. Budden, A. Abdolmaleki, J. Merel, A. Lefrancq, T. P. Lillicrap, and M. A. Riedmiller, "DeepMind Control Suite," arXiv:1801.00690, 2018. 1, 9

[12] T. Yu, D. Quillen, Z. He, R. Julian, K. Hausman, C. Finn, and S. Levine, "Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning," in *Conference on robot learning*. PMLR, 2020, pp. 1094–1100. 1, 9

[13] A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine, "Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations," in *Proceedings of Robotics: Science and Systems (R:SS)*, 2018. 1, 8

[14] M. Wüthrich, F. Widmaier, F. Grimminger, J. Akpo, S. Joshi, V. Agrawal, B. Hammoud, M. Khadiv, M. Bogdanovic, V. Berenz, J. Viereck, M. Naveau, L. Righetti, B. Schölkopf, and S. Bauer, "Trifinger: An open-source robot for learning dexterity," *CoRR*, vol. abs/2008.03596, 2020. [Online]. Available: https://arxiv.org/abs/2008.03596 1, 9

[15] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, D. Parikh, and D. Batra, "Habitat: A Platform for Embodied AI Research," in *International Conference on Computer Vision (ICCV)*, 2019. 1, 9

[16] A. Szot, A. Clegg, E. Undersander, E. Wijmans, Y. Zhao, J. Turner, N. Maestre, M. Mukadam, D. Chaplot, O. Maksymets, A. Gokaslan, V. Vondrus, S. Dharur, F. Meier, W. Galuba, A. Chang, Z. Kira, V. Koltun, J. Malik, M. Savva, and D. Batra, "Habitat 2.0: Training home assistants to rearrange their habitat," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 1, 2, 9, 11, 12

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

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

[19] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," arXiv:2111.06377, 2021. 2, 3, 4, 8, 12

[20] R. Ramrakhya, D. Batra, E. Wijmans, and A. Das, "Pirlnav: Pretraining with imitation and rl finetuning for objectnav," *arXiv preprint arXiv:2301.07302*, 2023. 3

[21] M. Deitke, E. VanderBilt, A. Herrasti, L. Weihs, J. Salvador, K. Ehsani, W. Han, E. Kolve, A. Farhadi, A. Kembhavi *et al.*, "Procthor: Large-scale embodied ai using procedural generation," *arXiv preprint arXiv:2206.06994*, 2022. 3

[22] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu *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*, 2022, pp. 18 995–19 012. 3, 9

[23] D. Shan, J. Geng, M. Shu, and D. F. Fouhey, "Understanding human hands in contact at internet scale," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9869–9878. 3

[24] R. Goyal, S. E. Kahou, V. Michalski, J. Materzyńska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax, and R. Memisevic, "The "something something" video database for learning and evaluating visual common sense," 2017. 3

[25] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price *et al.*, "Scaling egocentric vision: The epic-kitchens dataset," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 720–736. 3

[26] T. Zhou, R. Tucker, J. Flynn, G. Fyffe, and N. Snavely, "Stereo magnification: Learning view synthesis using multiplane images," *arXiv preprint arXiv:1805.09817*, 2018. 3

[27] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255. 3

[28] M. Deitke, D. Batra, Y. Bisk, T. Campari, A. X. Chang, D. S. Chaplot, C. Chen, C. P. D'Arpino, K. Ehsani, A. Farhadi, L. Fei-Fei, A. Francis, C. Gan, K. Grauman, D. Hall, W. Han, U. Jain, A. Kembhavi, J. Krantz, S. Lee, C. Li, S. Majumder, O. Maksymets, R. Martín-Martín, R. Mottaghi, S. Raychaudhuri, M. Roberts, S. Savarese, M. Savva, M. Shridhar, N. Sünderhauf, A. Szot, B. Talbot, J. B. Tenenbaum, J. Thomason, A. Toshev, J. Truong, L. Weihs, and J. Wu, "Retrospectives on the embodied ai workshop," *arXiv preprint arXiv:2210.06849*, 2022. 8

[29] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features

by contrasting cluster assignments," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9912–9924, 2020. 8

[30] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "Data2vec: A general framework for self-supervised learning in speech, vision and language," *arXiv preprint arXiv:2202.03555*, 2022. 8

[31] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," arXiv:2002.05709, 2020. 8

[32] X. Chen, S. Xie, and K. He, "An empirical study of training self-supervised vision transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 8

[33] H. Bao, L. Dong, and F. Wei, "Beit: Bert pre-training of image transformers," *ArXiv*, vol. abs/2106.08254, 2021. 8

[34] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations (ICLR)*, 2021. 8

[35] A. Baevski, A. Babu, W.-N. Hsu, and M. Auli, "Efficient self-supervised learning with contextualized target representations for vision, speech and language," *arXiv preprint arXiv:2212.07525*, 2022. 8

[36] Y. Yao, N. Desai, and M. Palaniswami, "Masked contrastive representation learning," *arXiv preprint arXiv:2211.06012*, 2022. 8

[37] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer, "Scaling vision transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 104–12 113. 8

[38] Y. Tian, O. J. Henaff, and A. van den Oord, "Divide and contrast: Self-supervised learning from uncurated data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 063–10 074.

[39] P. Goyal, M. Caron, B. Lefaudeux, M. Xu, P. Wang, V. Pai, M. Singh, V. Liptchinsky, I. Misra, A. Joulin *et al.*, "Self-supervised pretraining of visual features in the wild," *arXiv preprint arXiv:2103.01988*, 2021. 8

[40] A. Kumar, A. Raghunathan, R. Jones, T. Ma, and P. Liang, "Fine-tuning can distort pretrained features and underperform out-of-distribution," *arXiv preprint arXiv:2202.10054*, 2022. 8

[41] E. Wijmans, I. Essa, and D. Batra, "VER: Scaling on-policy rl leads to the emergence of navigation in embodied rearrangement," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 9

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

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

[44] S. Goyal, A. Kumar, S. Garg, Z. Kolter, and A. Raghunathan, "Finetune like you pretrain: Improved finetuning of zero-shot vision models," *arXiv preprint arXiv:2212.00638*, 2022. 8

[45] N. Hansen, Z. Yuan, Y. Ze, T. Mu, A. Rajeswaran, H. Su, H. Xu, and X. Wang, "On pre-training for visuo-motor control: Revisiting a learning-from-scratch baseline," *arXiv preprint arXiv:2212.05749*, 2022. 8

[46] J. Pari, N. M. Shafiullah, S. P. Arunachalam, and L. Pinto, "The surprising effectiveness of representation learning for visual imitation," *arXiv preprint arXiv:2112.01511*, 2021. 8

[47] "Real robot challenge 2020," https://real-robot-challenge.com/2020, 2020. 9

[48] A. Dittadi, F. Träuble, M. Wüthrich, F. Widmaier, P. Gehler, O. Winther, F. Locatello, O. Bachem, B. Schölkopf, and S. Bauer, "The role of pretrained representations for the ood generalization of reinforcement learning agents," 2021. [Online]. Available: https://arxiv.org/abs/2107.05686 9

[49] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. K. Gupta, L. Fei-Fei, and A. Farhadi, "Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning," *ICRA*, 2017. 9

[50] D. Batra, A. Gokaslan, A. Kembhavi, O. Maksymets, R. Mottaghi, M. Savva, A. Toshev, and E. Wijmans, "Objectnav revisited: On evaluation of embodied agents navigating to objects," *arXiv preprint arXiv:2006.13171*, 2020. 9, 12

[51] J. Gu, D. S. Chaplot, H. Su, and J. Malik, "Multi-skill mobile manipulation for object rearrangement," *arXiv preprint arXiv:2209.02778*, 2022. 9, 12

[52] K. Yadav, R. Ramrakhya, S. K. Ramakrishnan, T. Gervet, J. Turner, A. Gokaslan, N. Maestre, A. X. Chang, D. Batra, M. Savva *et al.*, "Habitat-matterport 3d semantics dataset," *arXiv preprint arXiv:2210.05633*, 2022. 9, 12

[53] R. Ramrakhya, E. Undersander, D. Batra, and A. Das, "Habitat-web: Learning embodied object-search from human demonstrations at scale," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 9

[54] E. Wijmans, A. Kadian, A. S. Morcos, S. Lee, I. Essa, D. Parikh, M. Savva, and D. Batra, "DD-PPO: Learning Near-Perfect PointGoal Navigators from 2.5 Billion Frames," in *International Conference on Learning Representations (ICLR)*, 2020. 9

[55] L. Mezghani, S. Sukhbaatar, T. Lavril, O. Maksymets, D. Batra, P. Bojanowski, and K. Alahari, "Memory-augmented reinforcement learning for image-goal navigation," *arXiv preprint arXiv:2101.05181*, 2021. 11

[56] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik *et al.*, "Habitat: A platform for embodied ai research," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 9339–9347. 11, 12

[57] F. Xia, A. R. Zamir, Z. He, A. Sax, J. Malik, and S. Savarese, "Gibson env: Real-world perception for embodied agents," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 9068–9079. 11

[58] Z. Al-Halah, S. K. Ramakrishnan, and K. Grauman, "Zero experience required: Plug & play modular transfer learning for semantic visual navigation," *arXiv preprint arXiv:2202.02440*, 2022. 11

[59] S. K. Ramakrishnan, A. Gokaslan, E. Wijmans, O. Maksymets, A. Clegg, J. M. Turner, E. Undersander, W. Galuba, A. Westbury, A. X. Chang *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. 12

[60] K. Yadav, S. K. Ramakrishnan, J. Turner, A. Gokaslan, O. Maksymets, R. Jain, R. Ramrakhya, A. X. Chang, A. Clegg, M. Savva *et al.*, "Habitat challenge 2022," 2022. 12

[61] A. Gupta, A. Murali, D. P. Gandhi, and L. Pinto, "Robot learning in homes: Improving generalization and reducing dataset bias," *Advances in neural information processing*

systems, vol. 31, 2018. 12

[62] R. Ramrakhya, E. Undersander, D. Batra, and A. 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*, 2022, pp. 5173–5183. 12

APPENDIX

A. 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 [28], 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.

B. Related Work

Pre-trained visual representations (PVRs). The last few years have seen increasing interest in the self-supervised learning (SSL) of visual representations [19, 29–32]. These algorithms use contrastive [31, 32], distillation-based [29, 30], or reconstructive [19, 33] objectives for training. Recently, a flurry of works have proposed using the vision transformers (ViTs) [34] with masked image modeling [19, 35, 36], which among other benefits reduces the computation time required for pre-training. In this work, we use one such pre-training algorithm (MAE [19]) 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 [4–10]. Specifically, Parisi et al. [7] evaluate several PVRs trained with supervised or self-supervised learning on a range of EAI tasks, demonstrating promising results under a few-shot imitation learning evaluation protocol. Nair et al. [8], Radosavovic et al. [9], Ma et al. [10] introduce new methods for pre-training visual representations using egocentric video data, targeting robotic manipulation tasks. Similarly, Khandelwal et al. [4], Yadav et al. [5, 6] use pre-trained visual representations to improve performance on multiple visual navigation tasks. Closely related, Radosavovic et al. [9] demonstrate that MAE pre-training on internet-scale video and image data can produce effective visual representations for robotic manipulation tasks. In contrast, our work studies a 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.

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

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

Scaling model and dataset size. Several works have showed that scaling model and dataset size improves performance on vision tasks like image classification [37–39]. In EAI, Radosavovic et al. [9] find that scaling model and data sizes improves downstream policy performances for robotic manipulation tasks. While such prior works have been confined to narrow domains like image classification and robotic manipulation, our work is the first to study if scaling can provide better models on a broad range of EAI tasks.

Adapting PVRs. When and how to adapt PVRs for downstream applications remains an open research question [40–44]. In the context of EAI, Parisi et al. [7] and Hansen et al. [45] show that naively fine-tuning PVRs with behavior cloning can reduce performance in simulation, and Radosavovic et al. [9] observe minimal gains in real-world tasks manipulation tasks. In large-scale RL settings, Yadav et al. [5, 6] show that end-to-end finetuning considerably improves performance for indoor visual navigation. By comparison, Pari et al. [46] 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.

C. Embodied AI Tasks in CortexBench

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

CortexBench includes tasks from 7 benchmarks illustrated in Figure 1 and described here:

Adroit (AD) [13] 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) [12] is a collection of tasks in which an agent commands a Sawyer robot arm to manipulate objects in a tabletop setting. We consider five tasks: `Assembly`, `Bin-Picking`, `Button-Press`, `Drawer-Open`, and `Hammer`, which follows the evaluations performed in [8].

DeepMind Control (DMC) [11] is a 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 [7].

TriFinger (TF) is a robot, introduced in [14], 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 [47]. We also consider the easier `Reach-Cube` task, which [48] 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 [15] 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`) [49] and object-goal navigation (`ObjectNav`) [50]. 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 [16] includes a set of mobile manipulation tasks in which an agent controls a Fetch robot with a 7-DoF arm, mobile base [51], 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 H.

D. 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. [7], Nair et al. [8]. 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 Nair et al. [8], Radosavovic et al. [9].

"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 `ObjectNav` policies with behavior cloning on 77k human demonstrations [52] collected by Habitat-Web [53], totaling 360M environment steps. For `ImageNav` and the Habitat 2.0 `Mobile-Pick` task, we use RL for 500M environment steps with DD-PPO [54] and VER [41]. We use patch representations for ViT-based PVRs and grid-features from last convolutional layer for ResNet models, passed through a compression layer [15] 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 training are in Appendix H.

E. Scaling Hypothesis Datasets

We strategically select combinations of these datasets (listed in Table V and below) to answer the following questions:

- What is the impact of scaling dataset size and diversity?
- How does the inclusion of *less-relevant* datasets influence the performance of PVRs on embodied AI tasks?

Ego4D [22] is our base pre-training dataset and encompasses a wide range of egocentric videos consisting of *daily life activities* such as home, leisure, transportation, and workplace activities.

Ego4D+M extends **Ego4D** with three object manipulation-centric datasets: 100DOH, SS-v2, and Epic Kitchens. This results in a dataset comprising 3.5 million frames that is primarily focused on manipulation scenarios.

Ego4D+N extends **Ego4D** with two egocentric indoor navigation datasets: OpenHouse24 and RealEstate10K. This results in a dataset with 3.5 million frames, which is similar in size to **Ego4D+M**, but is more diverse because it contains a larger proportion of navigation data than the manipulation-centric datasets **Ego4D** and **Ego4D+M**[3].

[3]While **Ego4D** does contain navigation data (e.g., people moving from location to another), the dataset is heavily skewed towards object manipulation activities.

Name	Contains	Total Frames	Frames used
Ego4D	Ego4D	418,578,043	2,790,520
Ego4D+M (Manipulation)	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+O (OpenHouse24)	Ego4D	418,578,043	2,790,520
	OpenHouse24	27,806,971	499,442
		Total	3,289,962
Ego4D+N (Navigation)	Ego4D	418,578,043	2,790,520
	OpenHouse24	27,806,971	499,442
	RealEstate10K	10,000,000	303,087
		Total	3,289,962
Ego4D+MN (Manipulation, Navigation)	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+MNI (Manipulation, Navigation, ImageNet)	Ego4D+MN	4,340,820	4,340,820
	ImageNet	1,281,167	1,281,167
		Total	5,621,987

TABLE V: Overview of the assembled datasets used for our scaling hypothesis experiments, using up to 5.6M frames.

Ego4D+MN combines **Ego4D** with both the three object manipulation-centric datasets and two indoor navigation dataset, resulting a dataset with 4.3 million frames. While larger than **Ego4D+M** and **Ego4D+N**, it does not include any new types of data beyond the manipulation and navigation videos in the previous subsets. Thus, it is no more diverse than **Ego4D+N** (which includes both types of data).

Ego4D+MNI includes **Ego4D**, all of the manipulation-centric and indoor navigation datasets, and ImageNet for a total of 5.6M frames. This dataset allows us to explore the impact of static internet images on our benchmark tasks.

1) 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.

*F. How does **VC-1** compare to existing PVRs?*

This section compares **VC-1** with existing PVRs from Section II-A. On average, **VC-1** ranks as the best model across all benchmarks Figure 2c. We focus on R3M, MVP, and CLIP, since they achieved the highest success in at least one benchmark; we also compare to fine-tuning from scratch to demonstrate the impact of end-to-end fine-tuning. In terms of mean success, **VC-1** (Table II row 11) outperforms MVP (ViT-L) by +1.2 points (67.5 → 68.7), R3M by +10.7 (58.0 → 68.7), CLIP by +11.7 (57.0 → 68.7), and end-to-end fine-tuning from scratch +19.6 (49.1 → 68.7).

Impressively, **VC-1** outperforms CLIP *on every benchmark* (Figure 3), despite training on a 70X smaller dataset, emphasizing the importance of egocentric interaction datasets. **VC-1** also outperforms fine-tuning from scratch on every benchmark, indicating that PVRs 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 3). 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*

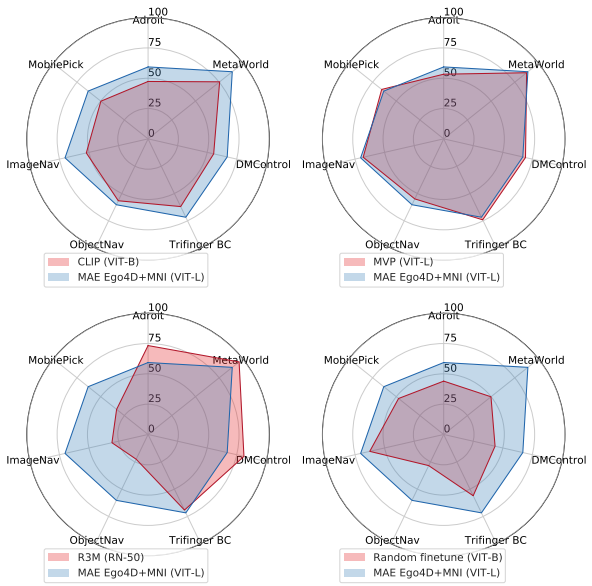


Fig. 3: Comparison of **VC-1** with existing PVRs. **VC-1** matches or exceeds existing PVRs on all benchmarks except R3M on AD, MW, and DMC, indicating an opportunity for model adaptation.

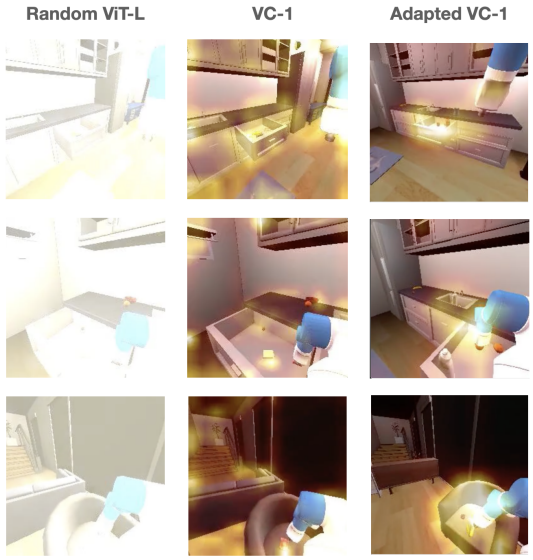


Fig. 4: Attention Visualization: We overlay the mean attention matrix in the last layer of the ViT encoder in one of our tasks -MobilePick-. We notice the effect of MAE pre-training on VC-1: The attention focuses in general features of the image; and of task-adaptation: the attention concentrates in task-specific regions of the image

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.

G. Attention Visualizations of VC-1

To visualize the attention we apply a mean pooling operation to the attention matrices of the ViT encoder's final layer during inference for downstream tasks. The resulting values are then overlaid onto the image.

We start by noticing the effect of MAE pre-training; frozen VC-1 attention maps appear to focus on the contours and general features of the image. We hypothesize that this results from the MAE reconstruction-based training objective, as contours provide essential information for reconstructing images.

Additionally, we study the attention maps after end-to-end fine-tuning of VC-1 on the downstream tasks. The attention appears to focus on regions of the image that are important for the task (e.g., the objects being manipulated). Thus, through adaptation (via E2E fine-tuning), the model learns to drop attention on areas irrelevant to the specific task.

H. CortexBench Tasks and Training Details

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

`ImageNav` Benchmark. Our study conducts `ImageNav` experiments using the standard dataset presented in [55]. This benchmark utilizes the Habitat simulator [16, 56] and is situated within the Gibson [57] 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 [58] 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 [52]. The dataset is comprised of 80 training, 20 validation, and 20 testing scenes and utilizes the Habitat simulator [16, 56] and HM3D [59] environments. Our results are reported on the v0.1 HM3D-SEM VAL split, which was used in the 2022 Habitat Challenge [60] `ObjectNav` benchmark. The agent in this evaluation is modeled after the LocoBot [61] with a height of 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 [50].

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 [52, 62] and Amazon Mechanical Turk, and consists of $77k$ demonstrations for 80 scenes from the HM3D-SEM dataset [60]. 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 ∼12.1 million steps of experience, with an average of ∼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 [16]. 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 [51]. 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. This relaxation places greater emphasis on visual input and makes the task significantly more challenging.

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.

I. Experiment Details of Training PVRs

To train the MAE models, we use the official codebase released by the authors on GitHub [19] 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 III 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 VI.

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+MNI (VIT-B)	182	5,621,987
VC-1 (**Ego4D+MNI** (VIT-L))	182	5,621,987

TABLE VI: Experiment Details of Training PVRs.