What Makes Pre-Trained Visual Representations Successful for Robust Manipulation?

Kaylee Burns[1] **Zach Witzel**[1] **Jubayer Ibn Hamid**[1] **Tianhe Yu**[2]
Chelsea Finn[1] **Karol Hausman**[1,2]
[1]Stanford University [2]Google DeepMind

Abstract: Inspired by the success of transfer learning in computer vision, roboticists have investigated visual pre-training as a means to improve the learning efficiency and generalization ability of policies learned from pixels. To that end, past work has favored large object interaction datasets, such as first-person videos of humans completing diverse tasks, in pursuit of manipulation-relevant features. Although this approach improves the efficiency of policy learning, it remains unclear how reliable these representations are in the presence of distribution shifts that arise commonly in robotic applications. Surprisingly, we find that visual representations designed for control tasks do not necessarily generalize under subtle changes in lighting and scene texture or the introduction of distractor objects. To understand what properties *do* lead to robust representations, we compare the performance of 15 pre-trained vision models under different visual appearances. We find that emergent segmentation ability is a strong predictor of out-of-distribution generalization among ViT models. The rank order induced by this metric is more predictive than metrics that have previously guided generalization research within computer vision and machine learning, such as downstream ImageNet accuracy, in-domain accuracy, or shape-bias as evaluated by cue-conflict performance. We test this finding extensively on a suite of distribution shifts in ten tasks across two simulated manipulation environments. On the ALOHA setup, segmentation score predicts real-world performance after offline training with 50 demonstrations.

Keywords: representation learning, manipulation, visual features

1 Introduction

In spite of vast progress in computer vision, the question of how to learn a good visual representation for robotics remains open [1]. Elsewhere in computer vision, internet datasets are retrofit to new tasks with transfer learning, which promises both generalization and fast adaptation to downstream tasks in exchange for large-scale pre-training. But in the field of robotics, this promise has yet to be fulfilled even though policies learned from pixels struggle substantially with data efficiency [2] and especially generalization under visual changes in a scene [3].

Recent work [4, 5] posits that the missing piece is a large pre-training dataset of object interactions across diverse environments — the ImageNet [6] or CommonCrawl [7] of manipulation. That is, if we want to improve the visual generalization ability of pre-trained models we simply need to collect datasets of this kind at scale. Indeed, training on large datasets of first-person human interaction data increases policy performance and learning efficiency downstream [8, 9], but these evaluations occur in environments that are very similar to those used for policy learning. Robotic applications commonly contain environments with varying lighting conditions, scene textures, and background objects, and we want pre-trained representations to allow the robot to handle such variability. Yet we have few concrete measures of how well pre-trained representations generalize out-of-distribution. To take a step towards understanding these problems, our goal in this paper is to thoroughly answer the questions *"which models generalize?"* and *"how can we predict generalization ability?"*

8th Conference on Robot Learning (CoRL 2024), Munich, Germany.

Our first key finding is that, when evaluated under visual distribution shifts, models that are designed for manipulation and control do not outperform standard visual pre-training methods. This finding violates our intuitions about what is needed to scale up robot learning and brings into question what constitutes relevant data, how to quantify useful features, and the importance of design choices such as model architecture. In other words, we need more guiding principles to understand what representations are good for manipulation and make the problem of iterating on pre-training strategies much more straightforward. Currently, evaluating a pre-trained policy requires training and rolling out downstream policies across multiple environments and experimental conditions. Instead, we can take inspiration from computer vision, which has developed proxies for robust performance on out-of-distribution datasets [10].

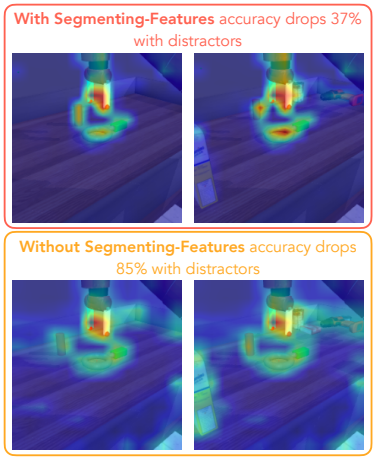

Figure 1: We find that the emergent segmentation ability of ViT attention heads (measured by Jaccard index) predicts performance under visual distribution shift. We refer to models with this property as having "segmenting-features." Notice how the attention of MVP shifts towards the sugar box distractor object in the bottom right image. The impact of this factor overshadows other design choices such as data relevance.

Our second key finding is that the emergent segmentation ability of a ViT model is a strong predictor of out-of-distribution generalization performance. We visualize this phenomenon, which we refer to as "segmenting-features," in Figure 1. Other metrics of model quality, such as linear probes on ImageNet [11], and out-of-distribution performance, such as in-domain accuracy [12] and shape-bias [13], are not predictive for this model class, despite their predictive power in other commonly-studied domains like image classification. This hints at the possibility that the transfer setting of manipulation differs from computer vision tasks typically studied within the robustness literature.

To reach the conclusions above, we run 9,000 different simulated evaluations. Our simulated environments are adapted from two different existing visual distribution shift benchmarks [14, 15] to capture the shifts that arise commonly in robotics applications: changes in lighting, background and object texture, and the appearance of distractors. More specifically, we train policies on top of 15 pre-trained models, including 4 models designed for manipulation or control: R3M [8], two MVP variants [9, 16], and VIP [17]. We further validate these findings by comparing a model designed for manipulation against a model with a similar parameter count on a real-world screwdriver pick-up task using the ACT training framework [18]. Through these experiments, we make two striking findings: (1) pre-trained visual models designed for control do not necessarily generalize better than models pre-trained on more standard vision datasets and (2) the emergent segmentation performance of a ViT model is a strong predictor of the out-of-distribution generalization of a down-stream policy.

2 Related Work

Representation learning for manipulation. The correct approach to visual representation learning for robotics is still an open question. There is evidence that separating visual representation learning from policy learning can further improve performance [19, 20]. Recent works have shown that models pre-trained on large manipulation-relevant datasets [21, 4, 22, 5] or learned with visual affordances from RGBD data [23] can improve the efficiency and performance of policy learning [24] in comparison to standard vision datasets such as ImageNet [6], but they do not focus on performance under visual distribution shift. We evaluate the performance of R3M [8], MVP [9, 16], and VIP [17]. Other work has studied generalization of pre-trained representations to new reinforcement learning tasks for manipulation [17] and navigation [25] where the agent is able to train on visual data from the new environment. Our objects of study are models that map RGB image data into single-vector feature representations. This excludes models such as LIV [26] and SUGAR [27], which operate on different data modalities, and Segment-Anything [28], which produces 3-dimensional feature maps and not compressed representations. Separate from the question of pre-training visual representa-

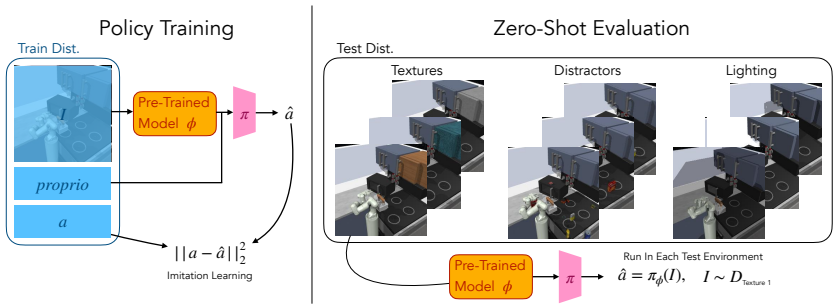

Figure 2: **Evaluation Scheme.** We begin our evaluation procedure by training a policy with behavior cloning on top of frozen features. In every experimental setting, we ablate the image observation encoder. The learned policy is then evaluated in each of the shift environments to attain a zero-shot success value.

tions is the question of how to best train policies on top of pixel observations [29, 30]. Majumdar et al. [31] benchmarks the performance of pre-trained visual representations on a handful of manipulation environments, but they focus on in-domain performance and also investigate navigation environments. Hu et al. [32] shows that model performance is highly sensitive to evaluation. We use imitation learning for our evaluation protocol, which they find to be a more stable measure of performance. Concurrently with our work, [33] demonstrates that the importance of proper data balancing supersedes the content of any one pre-training dataset. We focus on benchmarking visual generalization specifically and focus on advancing metrics that are predictive of generalization.

Robustness in computer vision. There is extensive work studying the impact of design choices, such as architecture, loss, and data, on the performance of visual models under distribution shift. See Geirhos et al. [10] for a comprehensive comparison. Most relevant to our paper are studies of shape-bias and architecture. While shape-biased models tend to be more robust than texture-biased ones [13], the impact of architecture on robustness is less straightforward. For example, vision transformers exhibit better robustness to universal adversarial attacks [34], but they are more susceptible to patch-level attacks [35]. When compared on natural distribution shifts [36, 37, 38], vision transformers and convolutional networks achieve comparable performance when provided with enough data [39]. But for occlusions specifically, vision transformers appear to have an edge [40]. Miller et al. [12] studies the predictive power of in-domain performance for out-of-distribution generalization. Unlike all of these prior works, we focus on how pre-trained representations affect robustness in downstream robotics tasks, instead of downstream vision tasks.

Learning robust policies. Unlike work that focuses on changes in dynamics or initial state distribution [41, 42, 43, 44, 45, 46], we focus exclusively on the setting of visual distribution shifts. Kirk et al. [47] and Zhao et al. [48] provide a comprehensive survey on non-visual distribution shifts in decision making problems. Policy adaptation approaches enable visual robustness specifically by leveraging insights from domain adaptation during policy training [49, 50, 51] or during deployment [52]. In the special case of closing the sim-to-real domain gap, a popular approach is to add randomized textures while training in simulation [53, 54, 55, 56]. By contrast, our work is interested in explaining properties of a robust visual model for control. Consequently, our insights can be leveraged with or without any task specific data.

3 Environments, Evaluation Protocol, and Pre-Trained Models

Our goal is to understand how robust existing representations for manipulation are to visual distribution shifts that are realistic in robotic applications. To that end, we learn policies parameterized by multi-layer perceptrons (MLPs) on top of frozen, pre-trained encoders and then evaluate these policies zero-shot under changes in lighting, object and scene texture, and the presence of distractors. We opt for MLP-based policy evaluations on frozen backbones because they the standard for probing visual representations [1, 11, 57], have strong theoretical underpinnings [58], and are the *de facto* benchmark for benchmarking representations for control [9, 8, 17]. The shifts tested are visualized in Appendix Figure 7 and a high level summary of our evaluation procedure is visualized

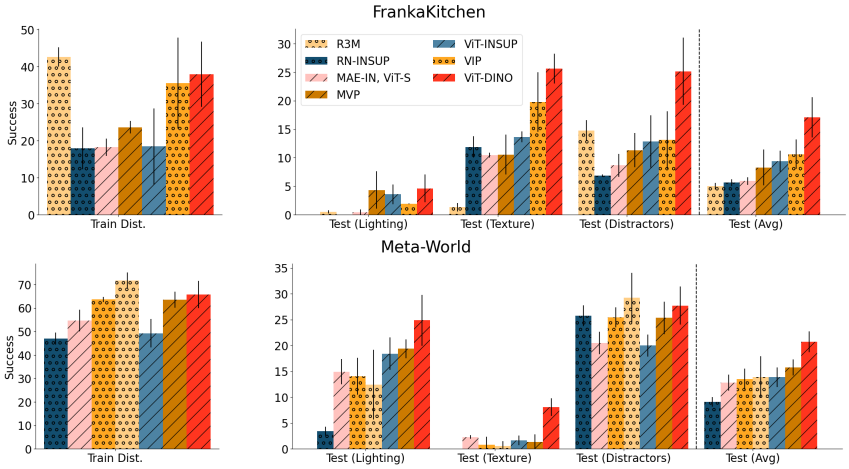

Figure 3: **Visual Generalization Performance.** Models trained with supervision on ImageNet are shades of blue. Models trained with self-supervision on ImageNet are in red. Models trained explicitly for control tasks are orange. Dotted bars denote ResNets and slashed bars denote ViTs. Surprisingly, the best performing models are not necessarily designed for manipulation. Each bar is an average over 30 experimental conditions.

in Figure 2. In this section, we describe the specifics of the manipulation environments, distribution shifts, and policy training setups.

Environments and tasks. We study ten tasks across two simulated manipulation environments, which are selected based on their popularity in studying learning-based approaches to manipulation. Within FrankaKitchen [59] we evaluate performance on opening a microwave, sliding a cabinet door open, pulling a cabinet open, turning a knob, and turning on a light. Within Meta-World [60] we study assembling a ring onto a peg, placing an object between two bins, pushing a button, opening a drawer, and hammering a nail.

Distribution shifts. We develop a benchmark for out-of-distribution generalization within FrankaKitchen and Meta-World. Within FrankaKitchen, we reimplement the texture and lighting changes from KitchenShift [14]. Within Meta-World we use texture changes from Xie* et al. [15] and reimplement the same lighting changes as in FrankaKitchen. In both environments we include three levels of distractors: one, three, and nine YCB objects [61]. More details about the implementation and parameterization of the distribution shifts are provided in Section A.3.

Policy training. Policy training is done in the same manner as R3M [8]. A summary of the evaluation scheme is provided in Figure 2. We train an MLP on top of the pre-trained embedding with imitation learning (IL), which, given actions sampled from expert trajectories, $a \sim \mathcal{D}_{train}$, minimizes the mean squared error objective, $||a - \hat{a}||_2^2$. Here $\hat{a}$ denotes the action predicted from a given policy. Details of the training procedure are provided in Section A.4. The embedding weights are frozen during policy learning, so the pre-trained models receive no task data. We train 3 different seeds within each task for each of two different camera angles. In total, we learn 60 policies for each model and perform 11 evaluations per policy, including on the train distribution.

Formally, for a pre-trained representation ϕ we learn policies, π_ϕ, each trained with a different seed, camera angle, and task. We average the performance of π_ϕ along each experimental condition and compute the mean performance and error across seeds.

Pre-trained Visual Representations. We categorize pre-trained models by loss type and data source: supervised ImageNet models, self-supervised ImageNet models, and models trained for manipulation and control tasks. Model specifics are provided in Appendix Section A.1.

4 Generalization of Models Pre-Trained for Manipulation

One factor motivating work in learning-based robotics is the hypothesis of scale: if we collect more high-quality manipulation data, we should see improvements in policy generalization. However, our understanding of what high-quality data looks like for manipulation and control tasks is still

imprecise. Past work on pre-training visual representations for manipulation and control tasks has focused on collecting large object interaction datasets and developing manipulation-relevant losses. But the generalization ability of such models in comparison to standard pre-training methods is still unknown. The goal of this section is to ask: *which models generalize?*

To focus our analysis, we compare models pre-trained for manipulation to two self-supervised ImageNet models and two supervised ImageNet models. Our main result is presented in Figure 3 where we plot the average success rate of the learned policies in the training environment distribution, within each class of visual shift, and across all types of visual shifts.

Models pre-trained for manipulation. Past work has trained visual representations for manipulation in two ways: by training with manipulation-specific losses or on data of human-object interactions. We focus on three recently introduced pre-trained models for manipulation that use different combinations of these approaches: Masked Visual Pretraining (MVP) [9], Reusable Representations for Robot Manipulation (R3M) [8], and Value-Implicit Pre-Training (VIP) [17]. We include important characteristics of these models, including dataset sizes, architecture sizes, and augmentations in Section A.1 and Table 1.

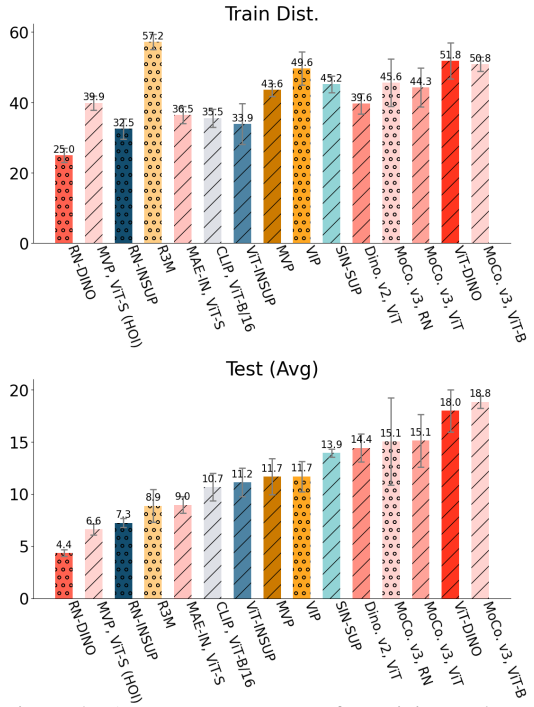

Figure 4: Average success rates for training and test distribution across both environments for every model in our evaluation suite. The best-performing model that was designed for manipulation ranks seventh out of all models evaluated.

These models perform strongly within the training distribution: R3M and VIP in particular comfortably beat standard pre-training baselines. This is expected, especially for R3M which was evaluated on the same training environment. However, under subtle distribution shifts, models designed for manipulation struggle to generalize as well as supervised or self-supervised training with ImageNet. This is surprising for a few reasons. First, each manipulation model is trained on a larger dataset than the pre-trained baselines. Ego4D alone is 4.5M frames while ImageNet is only 1.2M. By parameter count, MVP is also larger than the ViT-S baselines. Finally, we expect human-object interaction datasets such as Ego4D to be more similar to the distribution of images observed when training a manipulation policy. The viewpoints are more varied and the scenes are less curated than ImageNet. Although we expect this to improve the generalization of the learned policy, these results show that other factors may supersede the impact of data relevance or scale alone.

Supervised ImageNet models. Supervised training on ImageNet has long been a baseline for visual pre-training. Past work has found that features learned with supervised learning on ImageNet are also a strong baseline for control: even frozen features are competitive with ground-truth state information on a variety of simulated control tasks [20]. However, Parisi et al. [20] also find that self-supervised learning outperforms supervised learning. Our results contradict this finding. Figure 4 shows that supervised training on Stylized ImageNet achieves a higher success rate in the training distribution than self-supervised training on ImageNet with a masked auto-encoding loss. These models maintain the same rank out-of-domain as well. Even without stylization, in-domain performance of supervised ImageNet models are competitive with models trained with MAE on FrankaKitchen. From these results, we conclude that the presence of supervision is not as predictive of in-domain or out-of-domain performance as other factors. We also find that supervised ImageNet training is still a strong baseline for model generalization: ViT-INSUP outperforms R3M and MVP.

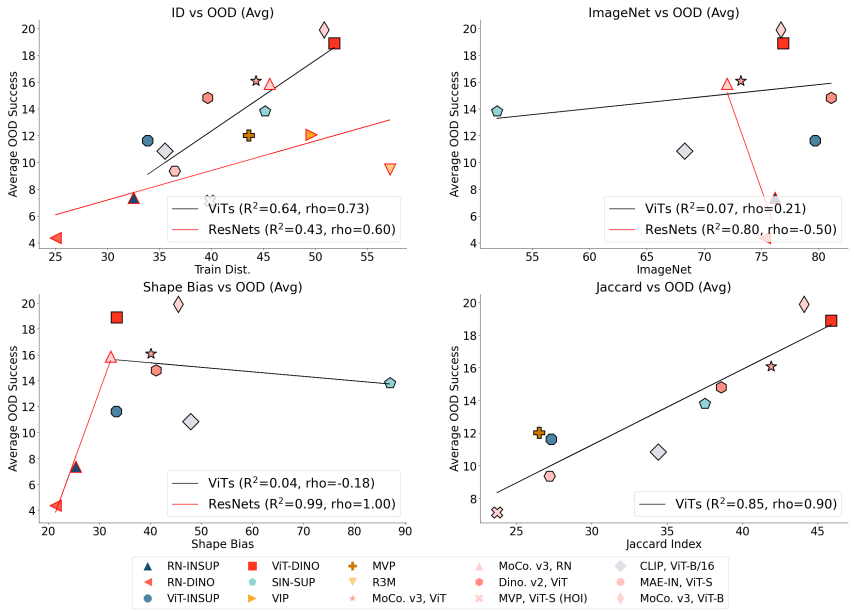

Figure 5: We plot the relationship between different metrics and out-of-distribution (OOD) generalization. There is a promising correlation between shape-bias and OOD performance for ResNets, but not ViTs. Instead, OOD performance for ViTs is strongly correlated with Jaccard index.

Self-Supervised ImageNet Models. In Figure 3 we include two self-supervised ViT-S models. Under visual distribution shifts, the model trained with the DINO objective outperforms all three models that are designed for manipulation. Moreover, this trend holds for every distribution shift except Meta-World with distractors. The distractors evaluation suite averages over different levels of distractions and therefore favors models with a high performance in training. In Appendix Section A.9 we plot model performance across different levels of distractors and find that several self-supervised ViTs experience a smaller drop in performance as more distractors are added compared to ResNet based pre-trained manipulation models like R3M and VIP.

Training with masked autoencoding performs well under distribution shifts in Meta-World, but is less strong under distribution shifts within FrankaKitchen. In Figure 4, we see that MoCo. v3, ViT-B also performs strongly out-of-distribution. When we compare MoCo and DINO against MAE-style training we see that MoCo and DINO use a more extensive set of augmentations. Taking this into account alongside the observation that a ViT trained with supervision on Stylized ImageNet performs well out-of-distribution we conclude that choice of augmentations outweighs the importance of supervision. This extends the findings of Geirhos et al. [10] to the setting of robust manipulation.

ViTs vs ResNets. One important design choice when selecting a pre-trained model is the choice of architecture. In all of our experiments, we use ResNet-50 [62] to be consistent with past work on visual pre-training [20, 8, 17]. Vision transformers (ViTs) [63] have seen widespread adoption within computer vision [64], but have only recently been used for learning representations for control [9]. We find that, on average, ViTs have a slight edge on out-of-distribution generalization compared to equivalently trained ResNets. In Figure 6, out of the seven pre-trained models that perform best out-of-distribution six are ViTs. Ablating architecture alone while holding dataset, training augmentations, and parameter count constant, we can compare the model pairs "MoCo. v3, RN" and "MoCo. v3, ViT", "RN-DINO" and "ViT-DINO", and "RN-INSUP" and "ViT-INSUP." In the latter two pairs, the ViT variant is much stronger out-of-distribution than the ResNet variant. For MoCo, the two variants achieve similar performance out-of-distribution.

5 Properties of Robust Visual Representations for Manipulation

Summary. This section identified which pre-trained models generalize, with several interesting findings. First, models designed for manipulaiton do not necessarily perform well under subtle distribution shifts in comparison to more standard pre-training methods. Second, the presence or

absence of supervision does not matter as much as other factors on both in- and out-of-distribution generalization. Finally, ViTs have a slight edge over ResNets in out-of-distribution generalization.

Our findings in the last section are both surprising and somewhat unsatisfying because they contradict many of our intuitions about scale and generalization. In our evaluation suite, we saw that better generalization is not cleanly explained by more data, bigger models, or more relevant data. The goal of this section is to identify the properties of pre-trained models that are predictive of generalization. To that end, we correlate out-of-distribution performance with three metrics that have been previously connected to generalization in the machine learning and computer vision literature—in-domain performance, accuracy of a linear probe trained on ImageNet, and shape-bias. We also include a fourth metric, which is specific to ViTs: the emergent segmentation accuracy of the output attention heads. We describe each metric in detail in Section 5.1, discuss our setup for correlating performance in Section 5.2, and analyze our results in Section 5.3.

5.1 Metrics

ID vs OOD. One of the goals of this paper is to understand how well the findings from existing evaluations of pre-trained models hold under the inevitable environment changes that we expect to see in real-world settings. If in-distribution performance is reasonably predictive of generalization, it is sufficient for researchers to continue developing pre-trained models with existing methods of evaluation. Past work has also shown that the in-distribution performance of a pre-trained model is positively correlated with out-of-distribution performance for a variety of computer vision tasks [12]. Concretely, we measure in-distribution performance as the success rate of the policy within the training distribution.

Imagenet vs OOD. Training linear probes on Imagenet is a common protocol for evaluating the quality of learned representations [65, 11]. Hu et al. [32] make the related finding that the ImageNet k-NN accuracy of a pre-trained model is predictive of performance on imitation learning with a visual reward function. We evaluate ImageNet validation accuracy for all models with linear probes.

Shape-Bias vs OOD. Shape bias is the extent to which a model makes prediction decisions based on shape. We calculate shape bias as the percent of shape (as opposed to texture) classification decisions on the Stylized-ImageNet validation set [13] using the same probes described above.

Jaccard vs OOD. Finally, for all of the ViT models, we look at the emergent segmentation performance. We evaluate the Jaccard index of an interpolated attention map averaged across heads in the last attention block at the [CLS] token.

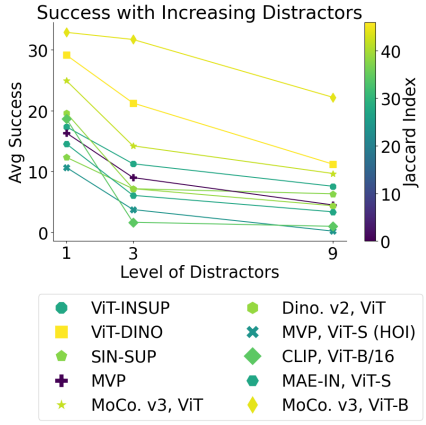

Figure 6: What happens to models with a high Jaccard index under an object-level distribution shift? Surprisingly, the models with the highest Jaccard index maintain the highest performance as the number of distractors increases.

5.2 Setup

We measure the coefficient of determination (R^2) and Spearman's rank correlation (ρ) for the correlation between the out-of-distribution success rate and each metric described above. Our goal is to find a metric that will result in high correlation between the metric and the OOD success, i.e. both coefficients being close to 1.0. We fit separate trend lines to ViTs and ResNets. Because of the lack of available probes, we exclude MVP, MVP ViT-S HOI, R3M, VIP, and MAE-IN ViT-S from the shape bias and ImageNet probe correlations. Each point represents one of the 15 pre-trained models we evaluated and represents the average of 6,000 evaluation runs.

5.3 Results

We visualize the correlation between each metric and the average out-of-distribution success rate in Figure 5. Although we see a positive relationship between in- and out-of distribution generalization, there are pre-trained models that notably deviate from this trend. Among ViT models one example is

MVP, ViT-S (HOI): the average success rate of this model drops to 6.63 from 39.86. By contrast, we find that ImageNet accuracy of a linear probe poorly predicts generalization performance for ViTs.

We also see little correlation between shape-bias and OOD performance for ViT models, but a promisingly strong correlation on the subset of ResNets evaluted. This is surprising because humans make highly shape-biased decisions and increasing shape-bias increases the robustness of imagenet trained CNNs [13, 10]. One explanation of this finding is that the ViT architecture obviates the need for shape-biased features. For example, a ResNet-50 trained with the DINO training scheme has a strong shape-bias, but not the equivalent ViT model.

Finally, we visualize the relationship between the Jaccard index and OOD performance on all ViT models in Figure 5. There is a strong positive correlation between Jaccard index and OOD performance both in terms of rank correlation and the coefficient of determination. These results suggest that while shape-bias may not be predictive of the OOD generalization ability of a pre-trained ViT, the segmentation ability is a predictive alternative.

One counter-argument to the use of Jaccard index as a metric for for OOD performance is that it would be less predictive for object-level distribution shift, which would occur any time a large distractor is placed in the background of the image. In Figure 6, we plot the success rates of each ViT model as the number of objects increases and verify that the models with the higher Jaccard index actually maintain the highest performance as the number of distractors increases.

5.4 Validating in the real world

In this section, we validate our finding on a real-world generalization scenario by comparing a ViT-B model designed for control (MVP) against a model not designed for control but with a high emergent segmentation score (MoCo-v3).

Setup. We learn policies for picking up a screwdriver on the ALOHA setup using the ACT training framework [18]. We follow the standard ACT training paradigm with further details listed in Appendix Table 3. From the training data to the test runs there is a distribution shift in both the placement of the target object (the screwdriver) and in the direction of the lighting. We calculate success on screw pick ups averaged over 10 rollouts in the test environment.

Results. We find that MoCo-v3, with a success rate of 40% is able to outperform MVP, with a success rate of 0%, even though it is not explicitly designed for manipulation. Qualitatively, the MVP model fails in localizing the object when attempting the grasp, whereas MoCo-v3 model reliably localizes the object, but experiences more failure in finding the right grasp point.

6 Conclusion

In this paper, we uncover a recipe for generalization: ViT models with a high emergent segmentation accuracy generalize well under visual distribution shifts. Emergent segmentation accuracy is not only a stronger predictor of generalization than many other metrics for robustness, but also requires no additional training to evaluate. This insight can guide the development of pre-trained vision models in future work: preferring architecture training algorithms that lead to strong emergent segmentation as opposed to only training on more manipulation-relevant data.

Limitations and future work. One limitation of this work is that our metric for predicting the robustness of visual representations is specific to ViT architectures. We believe that this is an artifact of the fundamental differences in the way representations are learned in ViT and ResNet architectures. One direction for future work is better understanding the conceptual connection between "shape-bias" in convolutional networks and the "segmenting features" in ViT models. The majority of our evaluation is in the theoretically well-understood, but still limited setting of training small multi-layer perceptrons on top of frozen encoders. An important direction for future work is to expand these findings with more complex decoders and with fine-tuning.

References

[1] X. Chen*, S. Xie*, and K. He. An empirical study of training self-supervised vision transformers. *arXiv preprint arXiv:2104.02057*, 2021.

[2] K. Cobbe, O. Klimov, C. Hesse, T. Kim, and J. Schulman. Quantifying generalization in reinforcement learning. *ArXiv*, abs/1812.02341, 2018.

[3] K. Cobbe, C. Hesse, J. Hilton, and J. Schulman. Leveraging procedural generation to benchmark reinforcement learning. *ArXiv*, abs/1912.01588, 2019.

[4] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray. Scaling egocentric vision: The epic-kitchens dataset. In *European Conference on Computer Vision (ECCV)*, 2018.

[5] K. Grauman, A. Westbury, E. Byrne, Z. Q. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu, M. Martin, T. Nagarajan, I. Radosavovic, S. K. Ramakrishnan, F. Ryan, J. Sharma, M. Wray, M. Xu, E. Z. Xu, C. Zhao, S. Bansal, D. Batra, V. Cartillier, S. Crane, T. Do, M. Doulaty, A. Erapalli, C. Feichtenhofer, A. Fragomeni, Q. Fu, C. Fuegen, A. Gebreselasie, C. González, J. M. Hillis, X. Huang, Y. Huang, W. Jia, W. Y. H. Khoo, J. Kolár, S. Kottur, A. Kumar, F. Landini, C. Li, Y. Li, Z. Li, K. Mangalam, R. Modhugu, J. Munro, T. Murrell, T. Nishiyasu, W. Price, P. R. Puentes, M. Ramazanova, L. Sari, K. K. Somasundaram, A. Southerland, Y. Sugano, R. Tao, M. Vo, Y. Wang, X. Wu, T. Yagi, Y. Zhu, P. Arbeláez, D. J. Crandall, D. Damen, G. M. Farinella, B. Ghanem, V. K. Ithapu, C. V. Jawahar, H. Joo, K. Kitani, H. Li, R. A. Newcombe, A. Oliva, H. S. Park, J. M. Rehg, Y. Sato, J. Shi, M. Z. Shou, A. Torralba, L. Torresani, M. Yan, and J. Malik. Ego4d: Around the world in 3,000 hours of egocentric video. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18973–18990, 2022.

[6] 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*, pages 248–255, 2009. doi:10.1109/CVPR.2009.5206848.

[7] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

[8] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta. R3m: A universal visual representation for robot manipulation. In *Conference on Robot Learning (CoRL)*, 2022.

[9] T. Xiao, I. Radosavovic, T. Darrell, and J. Malik. Masked visual pre-training for motor control. *arXiv:2203.06173*, 2022.

[10] R. Geirhos, K. Narayanappa, B. Mitzkus, T. Thieringer, M. Bethge, F. A. Wichmann, and W. Brendel. Partial success in closing the gap between human and machine vision. In *Neural Information Processing Systems*, 2021.

[11] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton. A simple framework for contrastive learning of visual representations. *ArXiv*, abs/2002.05709, 2020.

[12] J. Miller, R. Taori, A. Raghunathan, S. Sagawa, P. W. Koh, V. Shankar, P. Liang, Y. Carmon, and L. Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. *ArXiv*, abs/2107.04649, 2021.

[13] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bygh9j09KX.

[14] E. Xing, A. Gupta, S. Powers*, and V. Dean*. Kitchenshift: Evaluating zero-shot generalization of imitation-based policy learning under domain shifts. In *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021. URL https://openreview.net/forum?id=DdglKo8hBq0.

[15] A. Xie*, L. Lee*, and C. Finn. Benchmarking environment generalization in robotic imitation learning, 2023. URL https://github.com/RLAgent/factor-envs.

[16] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell. Real-world robot learning with masked visual pre-training. *CoRL*, 2022.

[17] 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.

[18] T. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware, 2023.

[19] J. Pari, N. M. M. Shafiullah, S. P. Arunachalam, and L. Pinto. The surprising effectiveness of representation learning for visual imitation. *ArXiv*, abs/2112.01511, 2022.

[20] S. Parisi, A. Rajeswaran, S. Purushwalkam, and A. K. Gupta. The unsurprising effectiveness of pre-trained vision models for control. In *International Conference on Machine Learning*, 2022.

[21] R. Goyal, S. E. Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fründ, P. N. 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 IEEE International Conference on Computer Vision (ICCV)*, pages 5843–5851, 2017.

[22] D. Shan, J. Geng, M. Shu, and D. F. Fouhey. Understanding human hands in contact at internet scale. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9866–9875, 2020.

[23] L. Yen-Chen, A. Zeng, S. Song, P. Isola, and T.-Y. Lin. Learning to see before learning to act: Visual pre-training for manipulation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2020. URL https://yenchenlin.me/vision2action/.

[24] S. Karamcheti, S. Nair, A. S. Chen, T. Kollar, C. Finn, D. Sadigh, and P. Liang. Language-driven representation learning for robotics. *arXiv preprint arXiv:2302.12766*, 2023.

[25] A. Sax, B. Emi, A. R. Zamir, L. J. Guibas, S. Savarese, and J. Malik. Mid-level visual representations improve generalization and sample efficiency for learning visuomotor policies. 2018.

[26] Y. J. Ma, W. Liang, V. Som, V. Kumar, A. Zhang, O. Bastani, and D. Jayaraman. Liv: Language-image representations and rewards for robotic control. *arXiv preprint arXiv:2306.00958*, 2023.

[27] S. Chen, R. Garcia, I. Laptev, and C. Schmid. Sugar: Pre-training 3d visual representations for robotics. In *CVPR*, 2024.

[28] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick. Segment anything. *arXiv:2304.02643*, 2023.

[29] M. Laskin, A. Srinivas, and P. Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. *Proceedings of the 37th International Conference on Machine Learning, Vienna, Austria, PMLR 119*, 2020. arXiv:2004.04136.

[30] D. Yarats, I. Kostrikov, and R. Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=GY6-6sTvGaf.

[31] A. Majumdar, K. Yadav, S. Arnaud, Y. J. Ma, C. Chen, S. Silwal, A. Jain, V.-P. Berges, P. Abbeel, J. Malik, D. Batra, Y. Lin, O. Maksymets, A. Rajeswaran, and F. Meier. Where are we in the search for an artificial visual cortex for embodied intelligence? 2023.

[32] Y. Hu, R. Wang, L. E. Li, and Y. Gao. For pre-trained vision models in motor control, not all policy learning methods are created equal, 2023.

[33] S. Dasari, M. K. Srirama, U. Jain, and A. Gupta. An unbiased look at datasets for visuo-motor pre-training. In *Conference on Robot Learning*. PMLR, 2023.

[34] R. Shao, Z. Shi, J. Yi, P.-Y. Chen, and C.-J. Hsieh. On the adversarial robustness of vision transformers. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=lE7K4n1Esk.

[35] Y. Fu, S. Zhang, S. Wu, C. Wan, and Y. Lin. Patch-fool: Are vision transformers always robust against adversarial perturbations? In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=28ib9tf6zhr.

[36] D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HJz6tiCqYm.

[37] D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo, D. Song, J. Steinhardt, and J. Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*, pages 8320–8329, 2021. URL https://doi.org/10.1109/ICCV48922.2021.00823.

[38] D. Hendrycks, K. Zhao, S. Basart, J. Steinhardt, and D. Song. Natural adversarial examples. *CVPR*, 2021.

[39] S. Bhojanapalli, A. Chakrabarti, D. Glasner, D. Li, T. Unterthiner, and A. Veit. Understanding robustness of transformers for image classification. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10211–10221, 2021. doi:10.1109/ICCV48922.2021.01007.

[40] M. Naseer, K. Ranasinghe, S. Khan, M. Hayat, F. Khan, and M.-H. Yang. Intriguing properties of vision transformers. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=o2mbl-Hmfgd.

[41] B. Huang, F. Feng, C. Lu, S. Magliacane, and K. Zhang. Adarl: What, where, and how to adapt in transfer reinforcement learning. *ArXiv*, abs/2107.02729, 2021.

[42] R. Raileanu, M. Goldstein, D. Yarats, I. Kostrikov, and R. Fergus. Automatic data augmentation for generalization in deep reinforcement learning. *ArXiv*, abs/2006.12862, 2020.

[43] M. Laskin, K. Lee, A. Stooke, L. Pinto, P. Abbeel, and A. Srinivas. Reinforcement learning with augmented data. *arXiv preprint arXiv:2004.14990*, 2020.

[44] K. Cobbe, C. Hesse, J. Hilton, and J. Schulman. Leveraging procedural generation to benchmark reinforcement learning. *arXiv preprint arXiv:1912.01588*, 2019.

[45] C. Packer, K. Gao, J. Kos, P. Krähenbühl, V. Koltun, and D. X. Song. Assessing generalization in deep reinforcement learning. *ArXiv*, abs/1810.12282, 2018.

[46] J. Farebrother, M. C. Machado, and M. H. Bowling. Generalization and regularization in dqn. *ArXiv*, abs/1810.00123, 2018.

[47] R. Kirk, A. Zhang, E. Grefenstette, and T. Rocktaschel. A survey of generalisation in deep reinforcement learning. *ArXiv*, abs/2111.09794, 2021.

[48] C. Zhao, O. Sigaud, F. Stulp, and T. M. Hospedales. Investigating generalisation in continuous deep reinforcement learning. *ArXiv*, abs/1902.07015, 2019.

[49] N. Hansen and X. Wang. Generalization in reinforcement learning by soft data augmentation. In *International Conference on Robotics and Automation*, 2021.

[50] L. Fan, G. Wang, D.-A. Huang, Z. Yu, L. Fei-Fei, Y. Zhu, and A. Anandkumar. Secant: Self-expert cloning for zero-shot generalization of visual policies. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 3088–3099. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/fan21c.html.

[51] T. Yoneda, G. Yang, M. R. Walter, and B. Stadie. Invariance through latent alignment, 2021.

[52] N. Hansen, R. Jangir, Y. Sun, G. Alenyà, P. Abbeel, A. A. Efros, L. Pinto, and X. Wang. Self-supervised policy adaptation during deployment. In *International Conference on Learning Representations*, 2021.

[53] F. Sadeghi and S. Levine. Cad2rl: Real single-image flight without a single real image. *ArXiv*, abs/1611.04201, 2017.

[54] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 23–30, 2017.

[55] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–8, 2018.

[56] S. James, P. Wohlhart, M. Kalakrishnan, D. Kalashnikov, A. Irpan, J. Ibarz, S. Levine, R. Hadsell, and K. Bousmalis. Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12619–12629, 2019.

[57] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020, 2021. URL https://arxiv.org/abs/2103.00020.

[58] A. Kumar, A. Raghunathan, R. M. Jones, T. Ma, and P. Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=UYneFzXSJWh.

[59] A. Gupta, V. Kumar, C. Lynch, S. Levine, and K. Hausman. Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. In *Conference on Robot Learning*, pages 1025–1037. PMLR, 2020.

[60] 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 (CoRL)*, 2019. URL https://arxiv.org/abs/1910.10897.

[61] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar. The ycb object and model set: Towards common benchmarks for manipulation research. In *2015 International Conference on Advanced Robotics (ICAR)*, pages 510–517, 2015. doi:10.1109/ICAR.2015.7251504.

[62] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[63] 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*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

[64] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah. Transformers in vision: A survey. *ACM Comput. Surv.*, 54(10s), sep 2022. ISSN 0360-0300. doi:10.1145/3505244. URL https://doi.org/10.1145/3505244.

[65] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick. Momentum contrast for unsupervised visual representation learning. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9726–9735, 2019.

[66] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[67] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou. Training data-efficient image transformers; distillation through attention. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10347–10357. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/touvron21a.html.

[68] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[69] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, and S. Levine. Time-contrastive networks: Self-supervised learning from video. *Proceedings of International Conference in Robotics and Automation (ICRA)*, 2018. URL http://arxiv.org/abs/1704.06888.

[70] K. He, X. Chen, S. Xie, Y. Li, P. Doll'ar, and R. B. Girshick. Masked autoencoders are scalable vision learners. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15979–15988, 2021.

[71] O. Nachum, B. Dai, I. Kostrikov, Y. Chow, L. Li, and D. Schuurmans. Algaedice: Policy gradient from arbitrary experience. *arXiv preprint arXiv:1912.02074*, 2019.

[72] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[73] T. Weyand, A. Araujo, B. Cao, and J. Sim. Google Landmarks Dataset v2 - A Large-Scale Benchmark for Instance-Level Recognition and Retrieval. In *Proc. CVPR*, 2020.

[74] A. Gupta, V. Kumar, C. Lynch, S. Levine, and K. Hausman. Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. *arXiv preprint arXiv:1910.11956*, 2019.

[75] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, E. Brynjolfsson, S. Buch, D. Card, R. Castellon, N. Chatterji, A. Chen, K. Creel, J. Q. Davis, D. Demszky, C. Donahue, M. Doumbouya, E. Durmus, S. Ermon, J. Etchemendy, K. Ethayarajh, L. Fei-Fei, C. Finn, T. Gale, L. Gillespie, K. Goel, N. Goodman, S. Grossman, N. Guha, T. Hashimoto, P. Henderson, J. Hewitt, D. E. Ho, J. Hong, K. Hsu, J. Huang, T. Icard, S. Jain, D. Jurafsky, P. Kalluri, S. Karamcheti, G. Keeling, F. Khani, O. Khattab, P. W. Koh, M. Krass, R. Krishna, R. Kuditipudi, A. Kumar, F. Ladhak, M. Lee, T. Lee, J. Leskovec, I. Levent, X. L. Li, X. Li, T. Ma, A. Malik, C. D. Manning, S. Mirchandani, E. Mitchell, Z. Munyikwa, S. Nair, A. Narayan, D. Narayanan, B. Newman, A. Nie, J. C. Niebles, H. Nilforoshan, J. Nyarko, G. Ogut, L. Orr, I. Papadimitriou, J. S. Park, C. Piech, E. Portelance, C. Potts, A. Raghunathan, R. Reich, H. Ren, F. Rong, Y. Roohani, C. Ruiz, J. Ryan, C. Ré, D. Sadigh, S. Sagawa, K. Santhanam, A. Shih, K. Srinivasan, A. Tamkin, R. Taori, A. W. Thomas, F. Tramèr, R. E. Wang, W. Wang, B. Wu, J. Wu, Y. Wu, S. M. Xie, M. Yasunaga, J. You, M. Zaharia, M. Zhang, T. Zhang, X. Zhang, Y. Zhang, L. Zheng, K. Zhou, and P. Liang. On the opportunities and risks of foundation models, 2022.

A Appendix

A.1 Pre-Trained Model Details

RN-INSUP [62] is a ResNet model trained on the ImageNet classificaiton task. We use the default weights and model provided by the Pytorch [66] library.

ViT-INSUP is a Vision Transformer [63] that has been distilled [67] from a larger network that was trained on the ImageNet classification task. In our experiments, we use the model weights and architecture provided in Naseer et al. [40] with a patch size of 16.

SIN-SUP [40] trains a vision transformer on Stylized Image-Net (SIN) [13]. The SIN dataset was constructed to increase the degree to which a model makes predictions on shape instead of texture. Our model weights come from Naseer et al. [40] and we use the non-distilled DeiT [67] training variant.

ViT-DINO [68] is trained with extensive augmentations and a self-supervised, contrastive loss that together lead to emergent segmentation within the self-attention heads of the ViT model. We use the model and weights provided by Caron et al. [68]. Interestingly, we don't find the DINO objective to lead to a high shape-bias. This suggests that there are other metrics that measure the degree to which a model is object-centric other than shape-bias.

ResNet50-DINO is learned with the same recipe as ViT-DINO. We use the model and weights from Caron et al. [68].

MoCo. v3, RN [1] leverages a contrastive loss with momentum encoding [65] of positive targets. It is trained with the same recipe as MoCo. v3, ViT-B.

MoCo. v3, ViT-B [1] are trained in a similar manner as the original MoCo [65], but with changes to improve the stability of training, which are specific to the ViT archiecture. We use the checkpoint after 300 epochs.

MoCo. v3, ViT-S [1] is trained in a similar manner as MoCo. v3, ViT-B. Even though the smaller model benefits from a longer training horizon, we use the checkpoint at 300 epochs for consistency.

MAE-IN, ViT-S follows the same training recipe as MVP, but on top of the ImageNet dataset. We use the weights provided by Radosavovic et al. [16].

R3M [8] trains a ResNet model with a combination of manipulation-specific losses–including a time-contrastive loss [69], video-language alignemnt loss, and L1-regularization–on the Ego4D [5] dataset.

MVP [16] trains a ViT-B for masked autoencoding (MAE) [70] on the Ego4D [5], Something-Something [21], YouTube 100 Days of Hands [22], EpicKitchens [4], and ImageNet [6] datasets. Unlike R3M, the model is not designed to be exclusive to manipulation.

MVP, ViT-S (HOI) [9] is a predecessor of the model described above that trains a ViT-S/16 with an MAE objective on Something-Something [21], YouTube 100 Days of Hands [22], EpicKitchens [4], and ImageNet [6].

VIP [17] uses an action-free dual of the Algaedice [71] objective to learn representations that are useful for trajectory optimization or reinforcement learning of control tasks unseen during representation pre-training. They train a ResNet-50 on Ego4D with this objective.

CLIP, ViT-B/16 [57] uses contrastive language-image pre-training to learn visual representations trained on an extensive internet datsaet. The learned models exhibit strong zero-shot performance for multiple tasks such as image classification.

DiNo v2, ViT [72] scales Caron et al. [68] to more parameters and a larger dataset. The full model is a 1B parameter ViT trained on LVD-142M, which is a 142M frame dataset composed of ImageNet-1k, ImageNet-22k, Google Landmarks [73], and a collection of other datasets spanning fine-grained classification, segmentation, depth estimation, and retrieval. The full model is distilled into smaller

models. We select the ViT-S distilled model for our experiments. In Table 1, we list the augmentations used on the teacher model. The training loop is only lightly modified during distillation. Suprisingly, the v2 model sees worse in- and out-of-domain performance on our evaluation suite in spite of being distilled from a ladrger model trained on a bigger dataset.

Name	Loss Function	Architecture	Datasets	Augmentations
RN-INSUP	BCE-Loss	ResNet-50 (23M params)	ImageNet (1.2M frames)	Random crop, Horizontal flip
ViT-INSUP	BCE-Loss	ViT-S/16 (22M params)	ImageNet (1.2M frames)	Random crop, Horizontal flip
SIN-SUP	BCE-Loss	ViT-S/16 (22M params)	Stylized-ImageNet (1.2M frames)	Random crop, Horizontal flip
ResNet50-DINO	Distillation	ResNet-50 (23M params)	ImageNet (1.2M frames)	Multi-crop, Color-jittering, Gaussian blur, Solarization
ViT-DINO	Distillation	ViT-S/16 (22M params)	ImageNet (1.2M frames)	Multi-crop, Color-jittering, Gaussian blur, Solarization
MoCo. v3, RN	Contrastive	ResNet50 (23M params)	ImageNet (1.2M frames)	Resize, Color-jittering, Horizontal flip, Grayscale, Gaussian blur, Solarization
MoCo. v3, ViT-S	Contrastive	ViT-S/16 (22M params)	ImageNet (1.2M frames)	Resize, Color-jittering, Horizontal flip, Grayscale, Gaussian blur, Solarization
MoCo. v3, ViT-B	Contrastive	ViT-B/16 (88M params)	ImageNet (1.2M frames)	Resize, Color-jittering, Horizontal flip, Grayscale, Gaussian blur, Solarization
MAE-IN, ViT-S	Masked auto-encoding	ViT-S (22M params)	ImageNet (1.2M frames)	Random resize, Random crop
R3M	Time-contrastive, L1-regularization, Video-lang alignment	ResNet-50 (23M params)	Ego4D (4.3M frames)	Random crop
MVP, ViT-S (HOI)	Masked auto-encoding	ViT-S (22M params)	EpicKitchens 100 Days of Hands, Something-Something (700k frames)	None
MVP	Masked auto-encoding	ViT-B (88M params)	Ego4D, ImageNet EpicKitchens, 100 Days of Hands, Something-Something (4.5M frames)	None
VIP	Algaedice Dual	ResNet-50 (23M params)	Ego4D (4.3M frames)	Random crop
CLIP, ViT-B/16	Contrastive	ViT-B/16 (88M params)	Internet data (400M pairs)	Random crop
DiNo v2, ViT	Distillation	ViT-S/14 (21M params)	LVD (142M frames)	Multi-crop, Color-jittering, Grayscale, Gaussian blur, Solarization

Table 1: List of pre-trained models with corresponding loss function, augmentations, and datasets used for pre-training. We color code by the data and loss type: ImageNet supervised, self-supervised, trained specifically for manipulation or control tasks, and other.

A.2 Details of the Environments

FrankaKitchen [74] is a simulated kitchen environment with a 9-DoF Franka robot. There a multiple household objects available for interaction. The environment is designed to compose tasks together hierarchically, but we focus on learning policies to successfully complete a single task. The episode length is 50 and we inherit the randomization scheme used in R3M, which randomizes the position of the kitchen at the start of each episode.

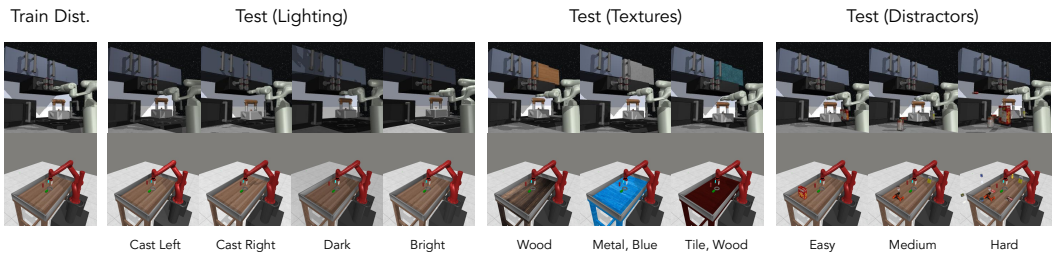

Figure 7: We visualize each distribution shift from the left camera angle on the FrankaKitchen (top) and Meta-World (bottom) environments.

Meta-World [60] is a simulated manipulation environment that consists of various table-top manipulation interactions. Unlike FrankaKitchen, the scene objects vary between different tasks. The positions of the objects are randomized at the start of each episode. The maximum episode length is 500.

A.3 Details of the Disribution Shifts

Each distribution shift is visualized from the left camera angle in Figure 7. We don't use the MuJoCo scanned object dataset that is used in [15] because of imperfections in the coloring of the textures.

A.4 Policy Training Details

Hyperparameter	Value
Loss type	MSE
Learning rate	0.001
Batch size	32
Train steps	20,000
Optimizer	Adam

Table 2: Hyperparameters for IL Policy Training

We learn a 2-layer MLP on top of the pre-trained, frozen features with 10 demonstrations. We use the same expert demonstrations as in R3M. We train policies independently over the 'left_cap2' and 'right_cap2' camera angles and show results averaged over both camera angles. We also provide proprioception to the policy. The final performance is averaged over the task settings for each seed. The hyperparamters for policy training are summarized in Table 2. Error bars are 95% confidence interval over seeds.

A.5 OOD Perf Details

To provide a more granular understanding of how the complete set of models performs on our evaluation suite, we break down performance by distribution shift type and environment in Figures 8 and 9.

A.6 ImageNet vs OOD Details

To evaluate ImageNet accuracy, we use all publicly available probes that have been trained on top of the frozen model features and evaluate them on the ImageNet validation set. The models with available probes are RN-INSUP, RN-DINO, MoCo. v3 RN, ViT-INSUP, ViT-DINO, MoCo. v3 ViT, Dino v2 ViT, MoCo. v3 ViT, SIN-SUP, and CLIP ViT-B/16 and we use the probes that are provided in the implementations cited in Section A.1.

A.7 Shape-Bias Details

We evaluate shape-bias using the 'model-vs-human' evaluation framework from Geirhos et al. [10] and use the same probes from Section A.6 to get classification results on the SIN validation dataset ($D_{cue-conflict}$).

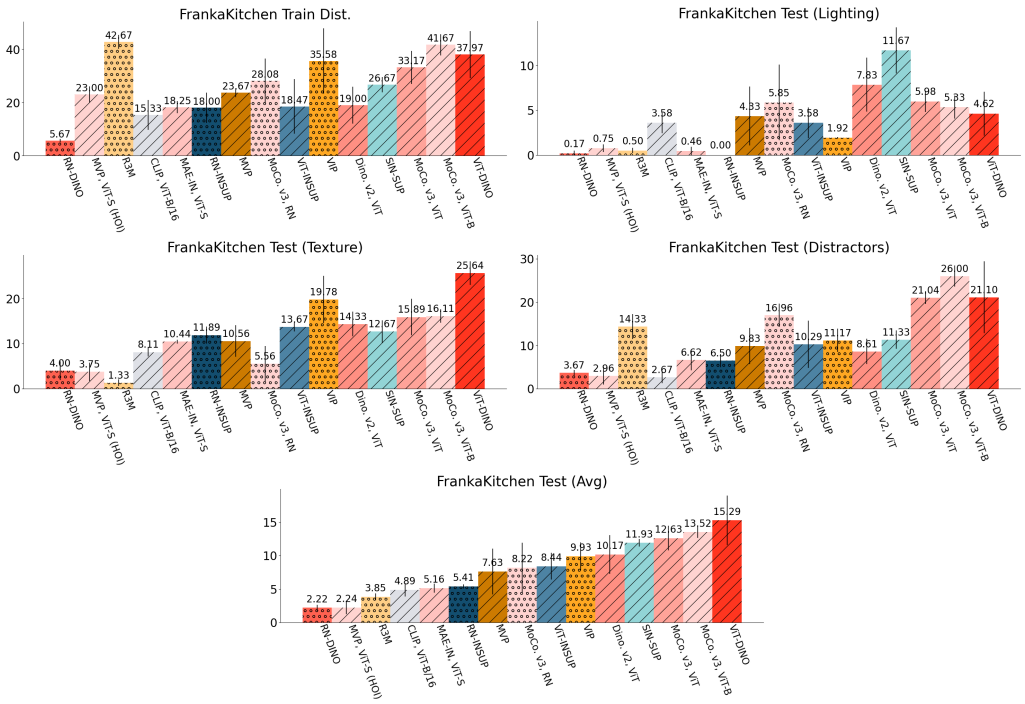

Figure 8: **Detailed OOD Performance on FrankaKitchen.**

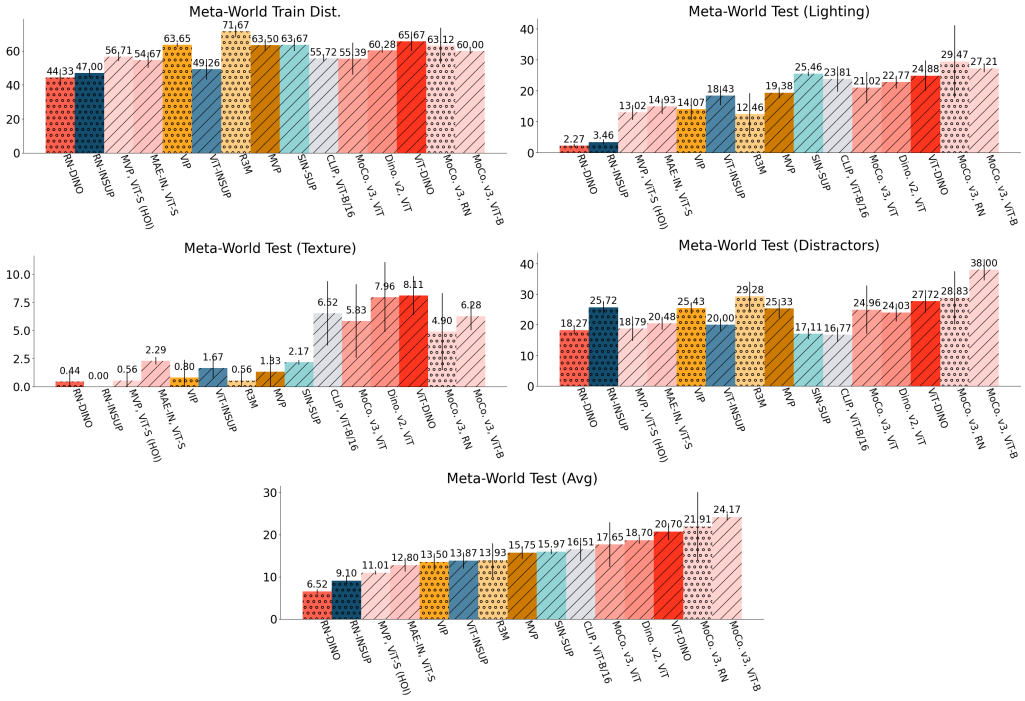

Figure 9: **Detailed OOD Performance on Meta-World.**

Notably, Naseer et al. [40] find that vision transformers are more shape-biased when making classification decisions than equivalently trained convolutional networks. In our results, we don't find vision transformers to be more strongly shape biased. Vision transformers and convolutional networks vary in how they handle spatial resolution: spatial resolution decreases in each layer of ResNet-50

but remains constant within a ViT. This could explain why we see the ViT architecture somewhat obviating the need for shape-bias in our results.

A.8 Jaccard Index Metric Details

We denote this nonlinear, deterministic transform as M. Formally, we compute the Jaccard index by calculating the mIoU on the PASCAL VOC validation set, D_{Pascal}:

$$J(x_i, x_j) = \mathbb{E}_{D_{Pascal}}\left[\frac{A \cap B}{A \cup B}\right]$$

Where A is a shorthand for positive classification for the target class by $M(\phi(\cdot))$ and B is a shorthand for positive label for the target class. J is evaluated pixel-wise over image indices x_i and x_j.

A.9 Different Levels of Distractors

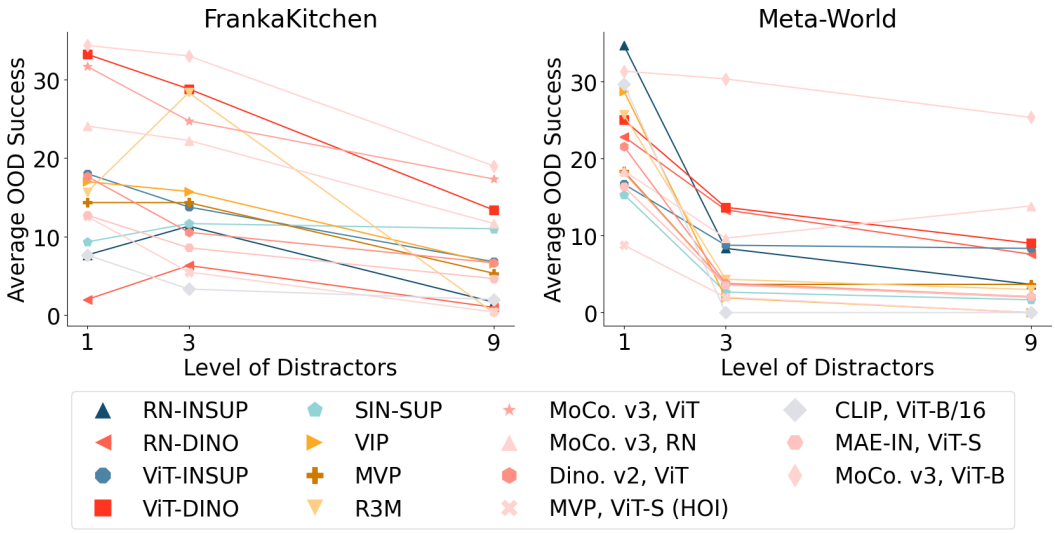

Figure 10: Different levels of distractors.

We extend Figure 6 by including results for ResNets in Figure 10. Models are color coded using the original color scheme in the paper.

A.10 Finetuning

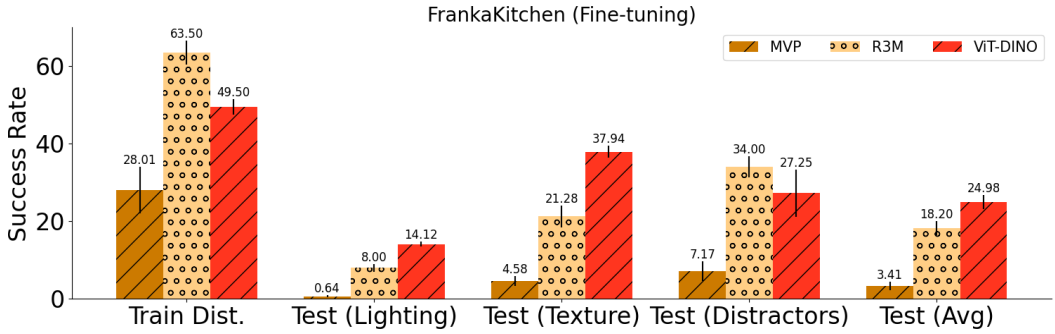

Figure 11: **Finetuning in FrankaKitchen.**

Because the goal of this paper is to probe the quality of learned representations, we follow the tradition of performing evaluation on top of frozen model features. This evaluation is also consistent

with the increasing view of pre-trained visual representations as "foundation models" [75, 72] that can be deployed without any gradient updates. Nonetheless, even in the fine-tuning regime, in Figure 11 we still see stronger performance from models that are not designed for manipulation. In this setting, we increased the number of demonstrations to 25 to allow for more data diversity when training the encoders.

A.11 Real-World Experiment Details

Our demonstration data contains two subtasks: an initial screwdriver pick-up and then a handover that happen in sequence. We only evaluate success on the subtask of picking up the screwdriver. The training dataset is comprised of 50 episodes collected by an expert human demonstrator. Images are collected from 4 camera view points (one on each wrist, one top camera, and one front camera). We replace the standard encoder with a ViT-B and change the initialization of the encoder based on the experimental condition (i.e., we select for a different pre-trained model).

Hyperparameter	Value
Chunk Size	100
KL Weight	10
Batch size	8
Epochs	10,000
Optimizer	Adam
Learning Rate	1e-5

Table 3: Hyperparameters for Policy Training

A.12 Additional Real-World Experiments

We also compare MVP and MoCo-v3 on an additional three real-world tasks and evaluate ten roll-outs across three different visual distribution shifts. For each task, we collect 30 demonstrations. Our training protocol is identical to Appendix Section A.11. However because these tasks are more challenging than the screw pick-pu task, we train for 20,000 total epochs with a batch size of 16. The real-world tasks are:

- **Corn on Plate.** In this task, one arm must pick up a plastic corn and stack it on top of a plate on the table.
- **Carrot on Plate.** In this task, one arm must pick up a plastic carrot and stack it on top of a plate on the table. This is slightly more difficult than the corn on plate task because the plastic carrot comes to a narrow point that is difficult to grasp.
- **Fork Hand-Over.** This is a long horizon task in which the right arm of the ALOHA robot must pick up a fork, hand it to the left arm, and the left arm must place the fork in a cup on the table.

We simulate the three classes of distribution shift as follows:

- **Lighting** We evaluate at a different time of day (evening) than the data was collected (day-time). This setting is the closest to the training setting.
- **Texture.** We cover the table with blue construction paper to simulate a change in table texture.
- **Distractors.** We place a single plastic piece of fruit to serve as a distracteor object in the center of the table.

To compute success, we assign a reward value of one to each of three subtasks—localizing the target object, completing a successful grasp or handover (depending on the task), and placing the target object in the correct location—and report the percent of reward achieved averaged over 10 trials.

Our results are reported in Table 4. We find that the high-segmentation scoring model (MoCo-v3) performs more robustly than the low-segmentation scoring model (MVP) on average. On easy tasks

Task/Distribution Shift	**MVP**	**MoCo-v3**
Corn on Plate (Lighting)	100%	100%
Corn on Plate (Distractor)	100%	100%
Corn on Plate (Blue Table)	96%	100%
Corn on Plate (Average)	99%	100%
Carrot on Plate (Lighting)	93%	90%
Carrot on Plate (Distractor)	100%	97%
Carrot on Plate (Blue Table)	33%	50%
Carrot on Plate (Average)	75%	80%
Fork Hand-Over (Lighting)	20%	100%
Fork Hand-Over (Distractor)	0%	96%
Fork Hand-Over (Blue Table)	0%	100%
Fork Hand-Over (Average)	7%	99%
All Tasks (Average)	**60%**	**93%**

Table 4: Performance under real-world distribution shifts

and with small distribution shifts, the difference between the two models is slight. However, on the long-horizon task, MVP struggles to maintain good performance. Interestingly, MoCo experiences a larger drop on the short-horizon Carrot on Plate task than on the Fork Hand-Over task. We believe this is because slight displacements in the gripper position can cause the plastic carrot to slip out from the grasp whereas the fork is more robust to grasp position.