Pre-Training for Manipulation: The Case for Shape Biased Vision Transformers

Kaylee Burns [1] **Tianhe Yu** [2] **Chelsea Finn** [2] **Karol Hausman** [2]

Abstract

Inspired by the success of transfer learning in computer vision, roboticists have investigated visual pre-training as a means to learn visually-robust policies from pixels. To that end, past work has favored large object interaction data, 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, we surprisingly find that it undermines robustness: policies fail under subtle changes in texture, lighting, and the introduction of distractor objects. Intrigued by this finding and inspired by pre-training effects on robustness in computer vision, we aim to find pre-training procedures that improve visual robustness of manipulation policies. In particular, we identify and analyze two key pre-training design decisions that maintain good performance under substantial visual changes in the environment: vision transformer architecture, and training with an inductive bias towards shape. We validate our findings on an extensive set of zero-shot visual distribution shifts in two simulated manipulation environments, improving over pre-trained models designed for manipulation by greater than 60%.

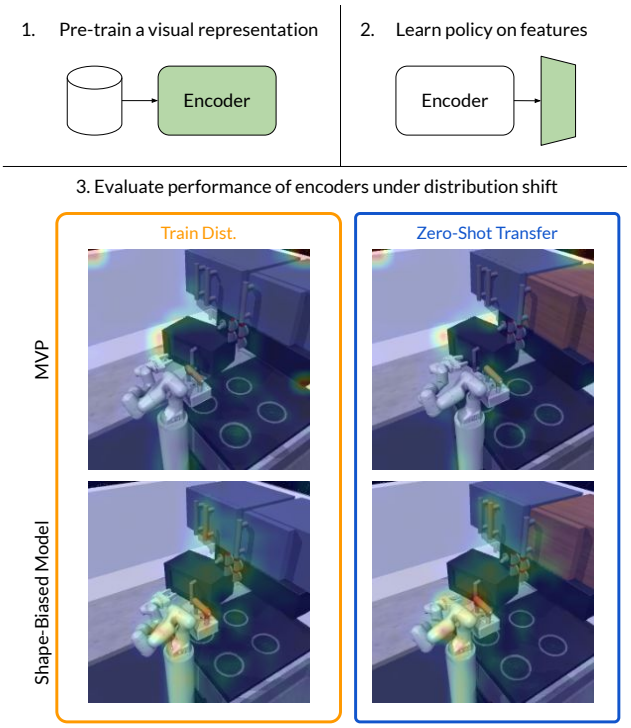

Figure 1. Our goal is to identify visual pre-training strategies whose features can be used for learning a manipulation policy that will remain performant under visual shifts in the environment. We find that vision transformers biased towards shape maintain the best performance underneath a variety of visual shifts.

1. Introduction

The promise of transfer learning is to enable efficient learning of downstream tasks by leveraging broad-scale pre-training. In particular, we expect this strategy to yield useful features in the presence of relatively little target domain data. Within the field of learning-based robotics, this promise has yet to be delivered even though there is a large need for it as policies learned directly from pixels struggle substantially with data efficiency and robustness (Cobbe et al., 2018; 2019a). Recent work (Damen et al., 2018; Grauman et al., 2022) posits that the missing piece is a large dataset of object interactions across diverse environments – the ImageNet (Deng et al., 2009) or CommonCrawl (Raffel et al., 2020) of manipulation. Indeed, training on large datasets of first person human interaction data increases policy performance and learning efficiency downstream (Nair et al., 2022; Xiao et al., 2022). However, the question of visual representations that improve *robustness* remains open.

Robotic task data differs substantially from common vision datasets in that large batches of images come from a relatively narrow visual environment, which makes the importance of robustness only more exaggerated. This differs from many computer vision tasks in that one episode may contain hundreds of images of the same lighting scenario, target object, and background. But for these representations to be practical, they need to maintain performance under visual changes that are inevitable in realistic settings. For example, someone could accidentally place a coffee cup in frame behind the robot or the sun could cast shadows

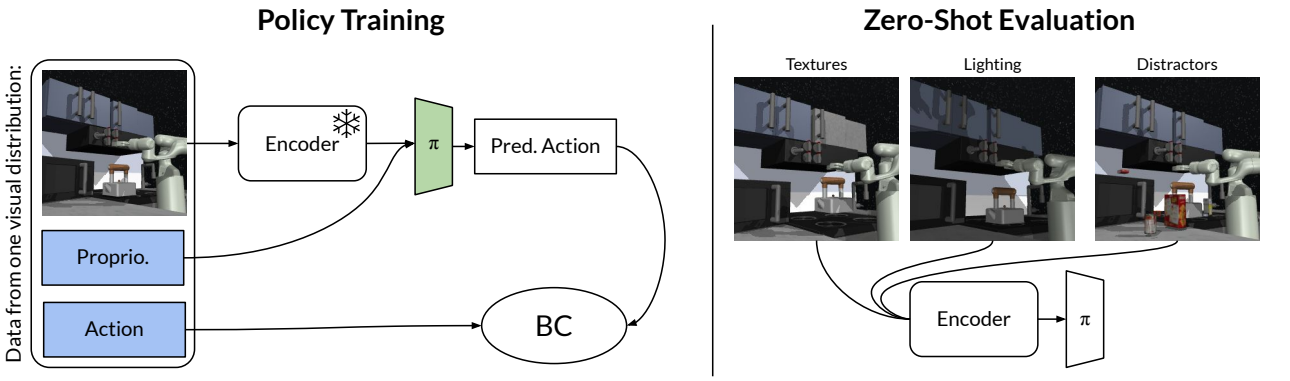

Figure 2. **Evaluation Scheme.** We begin our evaluation procedure by training a policy with behavior cloning on top of frozen features. In every experimental setting, we ablate the encoder used to extract features from the image observation. The learned policy is then evaluated in each of the visual shift environments to attain a zero-shot success value. Model components that are learned atg each stage are highlighted in green.

in new directions over the course of the day. Developing pre-trained visual representations that are resilient to these distribution shifts would make policies more robust and increase the likelihood that they can be deployed in diverse environments. Therefore, we seek a better understanding of the principles that govern the *robustness* of visual representations for manipulation tasks.

To identify strategies for visual pre-training, we take inspiration from work on texture and shape bias in computer vision. Some of the most common visual pre-training strategies result in models that rely heavily on local texture (Geirhos et al., 2019) and even though increasing shape-bias can improve robustness, these approaches have not yet been explored in the context of manipulation. We compare shape-biased models alongside a broad suite of visual pre-training approaches that compare different datasets, archiectures, and training paradigms to identify a recipe for robust visual pre-training.

The main contribution of this paper is the identification of two design choices for robust pre-trained models for manipulation: shape-biased training and vision transformer (Dosovitskiy et al., 2021) architectures. We validate our findings with an extensive evaluation of the zero-shot performance of policies trained on top of different visual encoders under a range of visual distribution shifts, which is visualized in Figure 3. Specifically, we look at changes in object texture, the presence of distractors, and changes in lighting on the Franka Kitchen and MetaWorld environments on a total of ten manipulation tasks. We run an extensive evaluation to verify our findings, where we conduct 465 training runs and 5115 evaluation runs. Surprisingly, we find that shape-biased vision transformers outperform both common pre-training strategies for computer vision as well as pre-trained models designed specifically for manipulation. We further analyze the relevance of shape bias by comparing two different ways of incorporating shape bias (specifically, introducing shape bias with two different loss functions) and verify that alternative ways of adding the shape bias recover the same result.

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 (Pari et al., 2022; Parisi et al., 2022). Recent works have shown that models pre-trained on large manipulation-relevant datasets (Goyal et al., 2017; Damen et al., 2018; Shan et al., 2020; Grauman et al., 2022) can improve the efficiency and performance of policy learning (Xiao et al., 2022) in comparison to standard vision datasets such as ImageNet (Deng et al., 2009), but they do not focus on performance under visual distribution shift. We compare directly against R3M Nair et al. (2022) and MVP Radosavovic et al. (2022). Other work has studied generalization of pre-trained representations to new reinforcement learning tasks for manipulation (Ma et al., 2022) and navigation (Sax et al., 2018) where the agent is able to train on visual data from the new environment. Other work has demonstrated visual robustness by learning visual affordances from RGBD data (Yen-Chen et al., 2020) as opposed to learning compressed features, which makes different assumptions about the structure of the features input to policies and the sensor data available. Separate from the question of pre-training visual representations is the question of how to best train policies on top of pixel observations

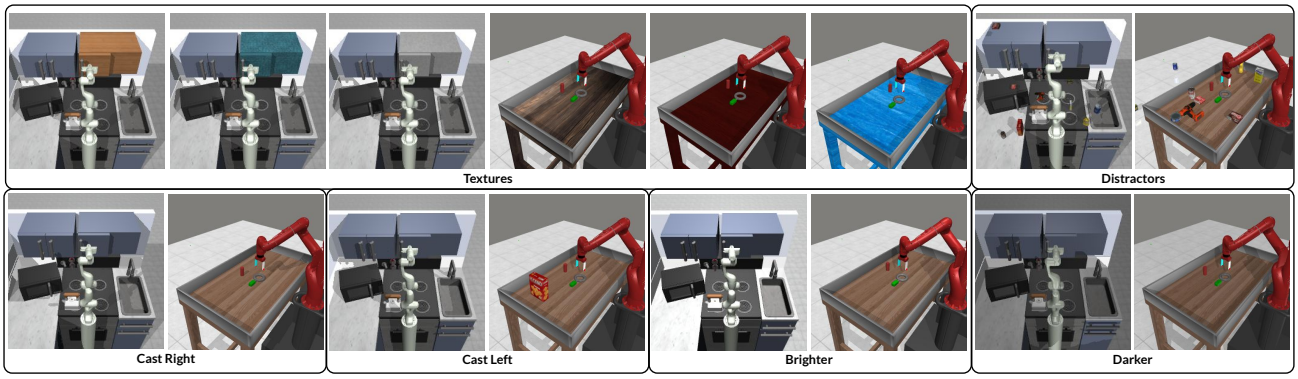

Figure 3. **Zero-Shot Evaluation Suite in FrankaKitchen and Meta-World.** We evaluate learned policies across a suite of visual distribution shifts, visualized here. We study visual changes in background texture, the presence of distractors, and different lighting conditions.

(Laskin et al., 2020b; Yarats et al., 2021).

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. (2021) 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 (Geirhos et al., 2019), the impact of architecture on robustness is less straightforward. For example, vision transformers exhibit better robustness to universal adversarial attacks (Shao et al., 2022), but they are more susceptible to patch-level attacks (Fu et al., 2022). When compared on natural distribution shifts (Hendrycks & Dietterich, 2019; Hendrycks et al., 2021a;b), vision transformers and convolutional networks achieve comparable performance when provided with enough data (Bhojanapalli et al., 2021). But for occlusions specifically, vision transformers appear to have an edge (Naseer et al., 2021). Unlike all of these prior works, we focus on how shape bias in pre-trained representations affects robustness in downstream robotics tasks, instead of downstream vision tasks.

Learning robust policies. Policy adaptation approaches focus on enabling robustness to visual distribution, often leveraging insights from domain adaptation during policy training (Hansen & Wang, 2021; Fan et al., 2021; Yoneda et al., 2021) or during deployment (Hansen et al., 2021). Our work differs from these in that our encoder weights not trained on any task data. Other work focuses on non-visual shifts in decision making problems, such as changes in dynamics or initial state distribution (Huang et al., 2021; Raileanu et al., 2020; Laskin et al., 2020a; Cobbe et al., 2019b; Packer et al., 2018; Farebrother et al., 2018). See Kirk et al. (2021) and Zhao et al. (2019) for a comprehensive survey. Past work has compensated for the special case of a sim-to-real domain gap adding randomized textures during training in simulation (Sadeghi & Levine, 2017; Tobin et al., 2017; Peng et al., 2018; James et al., 2019).

3. Experiment Setup

Our goal is to understand what properties of a pre-trained representation enable the best performance under visual changes on a manipulation task. To that end, we perform a thorough empirical analysis of current visual representation learning methods on robotic manipulation domains. In this section, we describe the details of our evaluation and the models we compare and our motivations for selecting them.

3.1. Evaluation Scheme

We are interested in the setting where we learn a policy on top of a frozen, pre-trained encoder and then evaluate the policy zero-shot under visual distribution shifts such as changes in lighting, object appearance, and the presence of distractors. These shifts are visualized in Figure 3 and a high level summary of our evaluation procedure is visualized in Figure 2. In this section, we describe the specifics of the manipulation environments, distribution shifts, and policy training setups that we use to understand this question.

Environments and tasks. We study five tasks in two simulated manipulation environments. Within FrankaKitchen (Gupta et al., 2020) we evaluate performance on opening a microwave, sliding a cabinet door open, pulling a cabinet door open, turning a knob, and turning on a light. Within Meta-World (Yu et al., 2019) we study assembling a ring onto a peg, picking and placing an object between two bins, pushing a button, opening a drawer, and hammering a nail.

Distribution shifts. We reimplement two benchmarks for policy generalization within FrankaKitchen and Meta-

World. Within FrankaKitchen, we use the texture and lighting changes from KitchenShift (Xing et al., 2021) and add three levels of distractions using YCB objects (Calli et al., 2015). We use similar visual changes within Meta-World, but instead change the texture of the workbench. More details about the specific implementation of the distribution shifts are provided in the appendix.

Policy training. Policy training is done in a similar manner as R3M (Nair et al., 2022). A summary of the evaluation scheme is provided in Figure 2 We train an MLP on top of the learned embedding with behavior cloning. The embedding weights are frozen during policy learning, so the pre-trained models receive no task data. We provide the policy with the image encoded by the pre-trained model of study and the proprioceptive observation. We train 3 different seeds across different combinations of tasks, demonstrations, and camera angles. In total, we learn 60 policies for each model and perform 480 total evaluations per model. More training details are available in the appendix.

3.2. Models.

In this section, we describe the pre-trained models that we compare in our experiments. Motivated by the finding that increasing shape bias can improve the robustness of image classifiers (Geirhos et al., 2019), we focus our analysis on understanding the impact of pre-training a visual model with shape-bias for robust manipulation policy learning downstream. We compare shape-biased models against pre-trained models designed for manipulation and models pre-trained on ImageNet classification (Deng et al., 2009).

3.2.1. Vision Transformers vs Convolutional Networks

One important design choice when selecting a pre-trained model is the choice of architecture. Convolutional networks are commonly used when learning control policies directly from pixels (Yarats et al., 2021; Espeholt et al., 2018; Mnih et al., 2013). In all of our experiments, we use ResNet-50 (He et al., 2016) to be consistent with past work on visual pre-training (Parisi et al., 2022; Nair et al., 2022; Ma et al., 2022). Vision transformers (ViT) (Dosovitskiy et al., 2021) have seen widespread adoption within computer vision (Khan et al., 2022), but have only recently been used for learning representations for control (Xiao et al., 2022). In our experiments, we present results with both the standard ViT architecture and a data efficient transformer variant (DeiT) (Touvron et al., 2021).

Notably, (Naseer et al., 2021) find that vision transformers are more shape-biased when making classification decisions than equivalently trained convolutional networks. Vision transformers and convolutional networks also vary in their spatial resolution: spatial resolution decreases in each layer of ResNet-50 but remains constant within a ViT.

3.2.2. Shape-Biased Models

Shape bias is the extent to which a model makes prediction decisions based on shape or texture. Formally, it is measured as the fraction of classification decisions made based on shape or texture information from a selection of images where these cues are sourced from conflicting classes (Geirhos et al., 2019). In the sections below we describe two models that have been measured to have a high shape bias in prior work (Naseer et al., 2021; Tartaglini et al., 2022).

Self-Distillation with No Labels (DiNo) is a self-supervised teacher-student training objective. In our experiments we use both a vision transformer and a residual network trained with this objective on ImageNet from Caron et al. (2021).

Supervised Learning with Stylized ImageNet (SIN). SIN is a version of ImageNet with local texture removed through AdaIn style transfer (Huang & Belongie, 2017) introduced by Geirhos et al. (2019). We use a vision transformer trained on SIN from Naseer et al. (2021).

3.2.3. Models Pre-trained for Manipulation

We focus on two recently introduced pre-trained models for manipulation as the main baselines for this work. Our goal is to develop representations that enable visually robust policies, so we compare against pre-trained models that have been designed for manipulation and control.

Masked Visual Pretraining (MVP) (Xiao et al., 2022) is an approach for learning a pre-trained Vision Transformer (ViT) (Dosovitskiy et al., 2021) for control that uses masked autoencoding (MAE) (He et al., 2021). We use the pre-trained model from Radosavovic et al. (2022), which is trained on the Ego4D (Grauman et al., 2022), Something Something (Goyal et al., 2017), YouTube 100 Days of Hands (Shan et al., 2020), EpicKitchens (Damen et al., 2018), and Imagenet (Deng et al., 2009) datasets.

Reusable Representations for Robot Manipulation (R3M) (Nair et al., 2022) trains a ResNet-50 (He et al., 2016) on Ego4D with a series of losses designed for manipulation. Specifically, they combine time-contrastive (Sermanet et al.) and video-language alignment losses with an L1 penalty to enforce sparsity in the resulting representation.

4. Evaluating Pre-Trained Models Under Visual Shifts

The goal of our analysis is to identify a recipe for visual pre-training that enables visually-robust policies. We focus specifically on the potential of shape-biased models to deliver more robust features in the zero-shot policy transfer

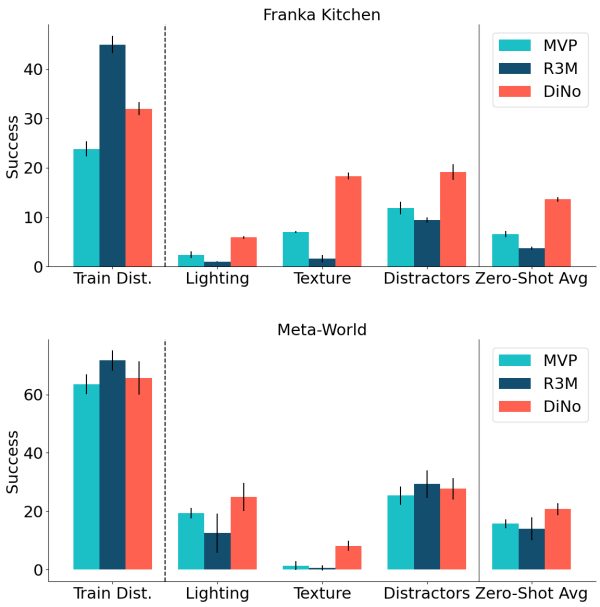

Figure 4. We report success results of learned models on the training distribution and zero-shot performance averaged over different visual change to the environment. The average result over all zero-shot changes (i.e., lighting, texture, distractors) is plotted on the far right. In both environments, a shape-biased vision transformer (DiNo) achieves the best zero-shot success rate.

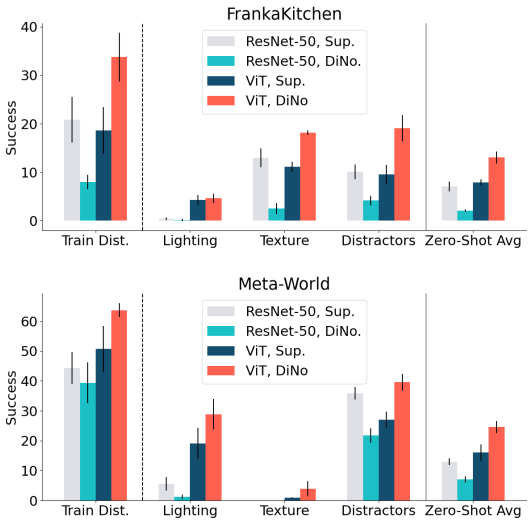

Figure 5. We present results for different combinations of architectures (namely, ResNet-50 and ViT) and training paradigms (supervised image classification and self-supervised learning with DiNo).

setting. To that end, we perform a thorough empirical analysis of current visual representation learning methods on robotic manipulation domains, focusing on the following perspectives:

1. In Section 4.1 we study how a shape-biased model (DiNo-ViT) compares to representations design for manipulation on policy transfer under visual distribution shifts.
2. Section 4.2 ablates the importance of model architectures and loss functions.
3. We analyse the effect of fine-tuning the pre-trained representations during policy training in Section 4.3.
4. We compare different instantiations of shape-bias in Section 4.4.
5. We visualize the attention heads of different pre-trained ViTs in Section 4.5.

4.1. How do pre-trained models designed for manipulation perform under visual distribution shift?

The averaged performance under each distribution type for both FrankaKitchen and Meta-World is shown in Figure 4.

Surprisingly, R3M achieves the lowest zero-shot performance of 3.7% on FrankaKitchen even though it achieves the highest success rate of 44.9% within the train distribution. This amounts to greater than an 91.8% perfomance drop. MVP style training achieves the lowest performance within the train distribution on the FrankaKitchen tasks, but achieves a transfer performance success rate of 6.6%. Averaged across environments, this amounts to a 72.2% performance drop from the training distribution. In spite of seeing the least data and less task-relevant data, DiNo performs best in the settings where there is a visual distribution shift. DiNo performance drops by 57.5% to 13.6% from 32.0%, so it also achieves a significantly lower percent drop than MVP or R3M.

On Meta-World, all models experience a larger performance drop. This is especially true for the texture distribution shift where MVP and R3M achieve an average success rate of 1.3% and 0.6% respectively. The texture shifts within Meta-World include two different wood table texture and one blue table texture, which is visualized in Figure 3. Interestingly, even though the blue table texture is the least realistic, MVP and R3M achieve a 0% success rate on not only the blue table but also the darker wood table. DiNo achieves greater than 6% on all 3 test-time table textures.

These results show that pre-trained models designed for manipulation perform poorly under visual distribution shifts. This is suprising because the distribution of data that MVP and R3M are trained on is much more similar to our evaluation environments and much larger than ImageNet. We

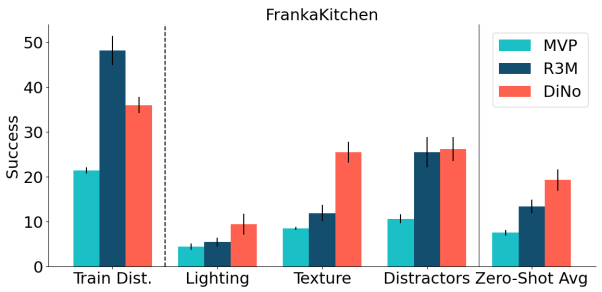

Figure 6. We evaluate zero-shot transfer after fine-tuning the pre-trained model during policy training.

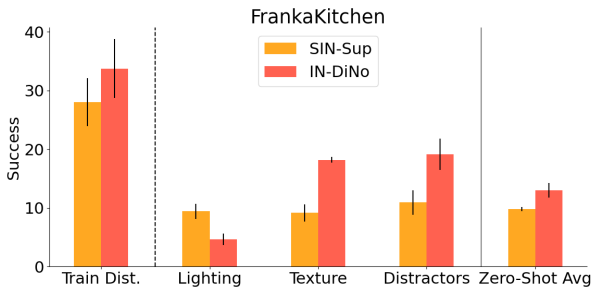

Figure 7. We explore multiple ways of inducing a shape bias within a ViT. We compare supervised pre-training with Stylized ImageNet (SIN-Sup) against DiNo trained on ImageNet. Both models are shape-biased and both achieve comparable zero-shot performance.

conclude that training with losses or large datasets that are designed for manipulation is insufficient for our goal. Instead, we find that DiNo provides more visually robust features, improving over the next best model by 106.0% in FrankaKitchen and 38.8% in Meta-World.

4.2. How do architecture and loss affect zero-shot performance?

In the last section, we saw that a shape-biased vision transformer in the form of DiNo can enable visually robust policies even when compared to models designed for manipulation. In this section, we ablate the choice of architecture and loss while holding the pre-training dataset constant. DiNo differs from MVP both in dataset and loss function and also differs from R3M in architecture. This analsyis allows us to get a more precise understanding of the components of DiNo's success. All models are trained on ImageNet and we separately train ResNet-50 and ViT models on the ImageNet classification and DiNo objectives.

We plot the success rate of each model in Figure 5. The best performing model under visual distribution shift in each task is ViT-DiNo. DiNo improves over the next best model by 64.6% in FrankaKitchen and 53.8% in Meta-World. However, unlike in Section 4.1, DiNo also achieves the highest success rate when evaluated in the training distribution. It improves by 62.0% over a supervised ResNet-50 in FrankaKitchen and by 25.7% over a supervised ViT in Meta-World. This suggests that pre-training a vision-transformer with a shape bias has a positive effect on overall policy performance when compared with supervised losses or residual networks.

Interestingly, even though the DiNo loss positively impacts performance for ViT models, it negatively affects performance when used with a ResNet-50. Caron et al. (2021) show that ViTs are more shape-biased in general and this result suggests a synergy between DiNo-style training and ViT architecture, which could further support that finding.

These findings not only show that DiNo is useful for training visually robust policies, but also hint at the possibility of improving overall policy performance. DiNo-style training could be combined with manipulation-specific losses proposed in R3M or applied to human-object interaction datasets such as Ego4D. In the spirit of this potential, we explore another shape-biased training strategy in the next section.

4.3. How does finetuning affect zero-shot performance?

In the previous sections, we do not fine-tune the visual representations during policy learning, so the encoders do not have a chance to see any task-data. From a practical perspective, freezing the encoder increases the speed of policy learning because it avoids computing gradients for and making updates to the encoder weights. This evaluation setting is also more similar to the prior work we compare against. However, there may be cases where fine-tuning makes sense, such as when we have a lot of data from one visual environment and we are not compute-constrained.

In Figure 6, we present fine-tuning on all models. Interestingly, R3M sees the largest benefit from fine-tuning while both vision transformer architectures see smaller performance improvements. This is true both especially under the visual shifts. In spite of this performance gain, shape-biased vision transformer still maintain the strongest performance on the zero-shot evaluation.

4.4. Can alternative loss functions enable good performance under shift?

We evaluate an alternative for training a shape-biased model. This is insightful not only because it provides further evidence that shape-biased vision transformers enable visually-robust policies, but also because it shows that visually robust pre-trained models can be trained with different loss func-

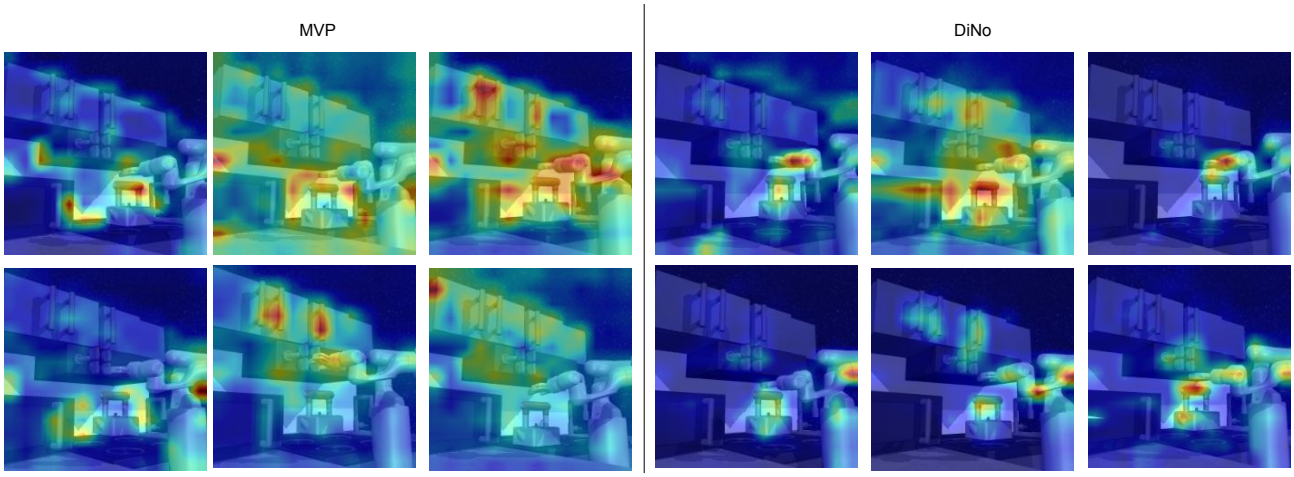

Figure 8. We visualize the first 6 attention heads from MVP and DiNo. Without seeing any environment data, DiNo segments task-relevant parts of the scene, such as the handle of the kettle, the joints of the robot, the drawer handles, and the knobs.

tions and data augmentation schemes. The DiNo model we use is trained with color jitter, Gaussian blur, mutli-crop, and horizontal flip augmentations during training. The assumptions underlying these augmentations may not be valid when performing representation learning for robotics. For example, in a manipulation task we may want to represent a horizontally flipped image differently from the original as left-ward actions in the original image become right-ward actions in the flipped image. Therefore, we compare DiNo, which is trained on ImageNet, against a vision transformer trained with supervision on Stylized ImageNet (SIN), which we describe in Section 3. Furthermore, we may want to pre-train models with labeled data when it is available, so identifying a strategy for leveraging supervision while maintaining zero-shot performance is critical.

Figure 7 compares the performance of DiNo againset supervised training on SIN on FrankaKitchen. SIN-Sup and DiNo achieves a 9.8% and 13.0% success rate, respectively, across all visual shifts. Both improve over MVP, which is the best pre-trained manipulation model with a success rate of 6.9%, and over ViT-Sup, which is the next best model overall with a success rate of 7.9%.

Within each visual shift, SIN-Sup performs much more consistently than DiNo, which is stronger in the presence of distractors and texture changes. Of all of the models evaluated, SIN-Sup is the most robust to lighting changes on the FrankaKitchen environment. These findings show that the benefits of shape bias are not limited to DiNo. However, within each kind of visual distribution shift, one strategy may be preferred over the other.

4.5. Visualizing Attention Heads of Different Training Strategies

An interesting artifact of training a Vision Transformer with DiNo is that the attention heads emergently learn segmentations of objects in the scene. In this section, we visualize the attention heads of two pre-trained vision transformer models. Using the same visualization procedure as (Caron et al., 2021), we show the first six attention heads of MVP and DiNo in Figure 8. In these examples, the attention with DiNo heads are much more concentrated than within the MVP heads and, interestingly, without seeing any task data, the head highlights task relevant objects like the robot arm or the door handles.

4.6. Operationalizing Shape-Bias

- train any ViT backbone with different percentages of shape bias
- correlate shape bias of each base model with transfer score
- use shape-token to modulate degree of shape bias

5. Conclusion

We show that shape-biased vision transformers produce representations that enable visually robust policies even when compared to pre-trained models that are designed for manipulation. We demonstrate that the success of shape biased models is not limited to a particular training scheme: using supervision with SIN gives comparable performance as training DiNo on ImageNet. This could be critical in settings where specific augmentations may impose invariances on the image representations that are not valid for robotics

(such as a horizontal flip) or where labeled data can be leveraged for supervised pre-training. Shape-biased vision transformers also improve overall policy performance when compared to other architecture and loss choices, opening the door for future work to scale such methods to larger and more task-relevant datasets such as Ego4D.

References

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

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

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

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

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

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

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

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Espeholt, L., Soyer, H., Munos, R., Simonyan, K., Mnih, V., Ward, T., Doron, Y., Firoiu, V., Harley, T., Dunning, I. R., Legg, S., and Kavukcuoglu, K. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. 2018. URL https://arxiv.org/abs/1802.01561.

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

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

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

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. 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.

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

Goyal, R., Kahou, S. E., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fründ, I., Yianilos, P. N., Mueller-Freitag, M., Hoppe, F., Thurau, C., Bax, I., and Memisevic, R. The "something something" video database for learning and evaluating visual common sense. *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5843–5851, 2017.

Grauman, K., Westbury, A., Byrne, E., Chavis, Z. Q., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., Martin, M., Nagarajan, T., Radosavovic, I., Ramakrishnan, S. K., Ryan, F., Sharma, J., Wray, M., Xu, M., Xu, E. Z., Zhao, C., Bansal, S., Batra, D., Cartillier, V., Crane, S., Do, T., Doulaty, M., Erapalli, A., Feichtenhofer, C., Fragomeni, A., Fu, Q., Fuegen, C.,

Gebreselasie, A., González, C., Hillis, J. M., Huang, X., Huang, Y., Jia, W., Khoo, W. Y. H., Kolár, J., Kottur, S., Kumar, A., Landini, F., Li, C., Li, Y., Li, Z., Mangalam, K., Modhugu, R., Munro, J., Murrell, T., Nishiyasu, T., Price, W., Puentes, P. R., Ramazanova, M., Sari, L., Somasundaram, K. K., Southerland, A., Sugano, Y., Tao, R., Vo, M., Wang, Y., Wu, X., Yagi, T., Zhu, Y., Arbeláez, P., Crandall, D. J., Damen, D., Farinella, G. M., Ghanem, B., Ithapu, V. K., Jawahar, C. V., Joo, H., Kitani, K., Li, H., Newcombe, R. A., Oliva, A., Park, H. S., Rehg, J. M., Sato, Y., Shi, J., Shou, M. Z., Torralba, A., Torresani, L., Yan, M., and Malik, J. Ego4d: Around the world in 3,000 hours of egocentric video. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18973–18990, 2022.

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

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

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

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

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

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

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

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

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

Huang, X. and Belongie, S. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017.

James, S., Wohlhart, P., Kalakrishnan, M., Kalashnikov, D., Irpan, A., Ibarz, J., Levine, S., Hadsell, R., and Bousmalis, K. 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)*, pp. 12619–12629, 2019.

Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F. S., and Shah, M. 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.

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

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

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

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

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. A. Playing atari with deep reinforcement learning. *ArXiv*, abs/1312.5602, 2013.

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

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

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

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

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

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

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

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. 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.

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

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

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

Sermanet, P., Lynch, C., Chebotar, Y., Hsu, J., Jang, E., Schaal, S., and Levine, S. Time-contrastive networks: Self-supervised learning from video. *Proceedings of International Conference in Robotics and Automation (ICRA)*.

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

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

Tartaglini, A. R., Vong, W. K., and Lake, B. M. A developmentally-inspired examination of shape versus texture bias in machines. *arXiv preprint arXiv:2202.08340*, 2022.

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

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

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

Xing, E., Gupta, A., Powers*, S., and Dean*, V. 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.

Yarats, D., Kostrikov, I., and Fergus, R. 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.

Yen-Chen, L., Zeng, A., Song, S., Isola, P., and Lin, T.-Y. 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/.

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

Yu, T., Quillen, D., He, Z., Julian, R., Hausman, K., Finn, C., and Levine, S. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2019.

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

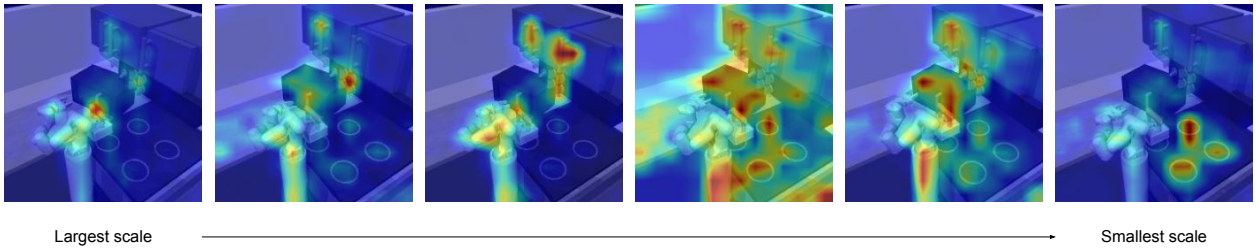

Figure 9. We scale each attention head in the final block of DiNo by a single parameter, which is learned during policy fine-tuning. The resulting weight allows us to qualitatively understand how the policy fits to different components of the output image feature. Interestingly, the policy does not strictly prefer attention heads that provide clean segmentation masks.

A. Appendix.

A.1. Details of the Policy Training and Experimental Conditions

We learn a 2-layer MLP on top of the pre-trained, frozen features. For MVP, DiNo, and R3M in Kitchen, we evaluate over 5, 10, and 25 demonstrations. For all other models we evaluate over 10 demonstrations. All reported changes in performance are calculated from all three levels of demonstrations in Section 4.1, but everywhere else compares model performance at 10 demonstrations. For both FrankaKitchen and Meta-World environments, we train policies independently over the left_cap2 and right_cap2 camera angles and show results averaged over both camera angles. For the Meta-World experiments in Section 4.2, we only show results on one camera angle due to time constraints. The final performance is averaged over the task settings for each seed. Error bars are 95% confidence interval over seeds.

A.2. More Details of Models

We use the ViT-B backbone for MVP. Both SIN and ImageNet supervised vision transformers use the more effiecient DeiT (Touvron et al., 2021) backbone.

A.3. Details of the Disribution Shifts

The R3M environment differs from the original KitchenShift environment in that the kitchen is randomly shifted at the start of each episode, making our evaluation suite more difficult. After sub-selecting for visual distribution shifts, we also add in easy, medium, and hard distractor settings, where the kitchen counter is cluttered with an increasing number of YCB objects (Calli et al., 2015). The difficulty of the setting corresponds to the number of distractors present. Within Meta-World we add in YCB distractor objects in the same way. We don't use the MuJoCo scanned object datasets because of imperfections in the coloring of the textures.

A.4. Learning Weights on Each Attention Head

To explore which attention heads are preferred during the downstream policy learning task, we re-run policy training while learning a single single scalar multiple for each of the six attention heads of the final block of DiNo. In Figure 9, we visualize each attention head sorted by the average scaling factor learned for that head across all kitchen tasks. Surprisingly, the more concentrated heads are not always preferred by the policy. We find that the attention head with the largest average scale is also preferred by the majority of tasks.

In Figure 10, we visualize attention heads sorted by the weight learned in our adapted fine-tuning setting over every task in FrankaKitchen from the right camera angle. Interestingly, four out of the five heads put the most weight on the same attention head even though their tasks require acting on different objects.

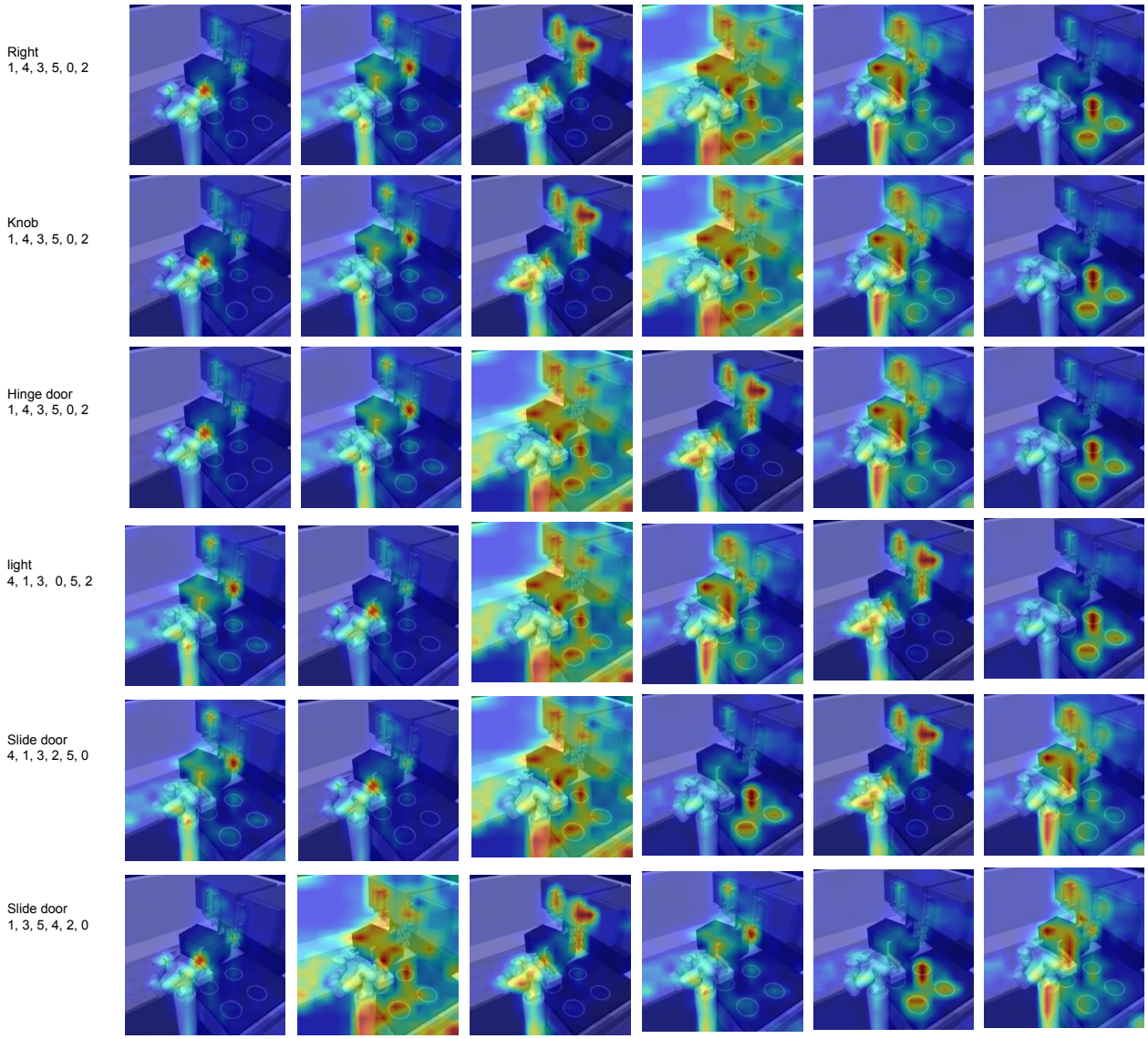

Figure 10. We order the attention head by the learned scale on each task. Images on the left are heads that received higher weight (i.e., the model leverages this head more for the task) and images on the right receive lower weight. The head index and task name are listed on the far left. The top row sorts the heads by average weight across tasks.