What Makes Representation Learning from Videos Hard for Control?

Tony Z. Zhao[1] **Siddharth Karamcheti**[1] **Thomas Kollar**[2] **Chelsea Finn**[1] **Percy Liang**[1]
[1] Stanford University [2] Toyota Research Institute

Abstract

A key goal in robotic control is building systems that can efficiently learn and generalize from minimal data. A promising approach for building such systems is by *pretraining visual state representations* from large datasets of videos. Unfortunately, in-distribution data for robot manipulation is scarce; to mitigate this, prior work has turned to pretraining on diverse data sources, such as egocentric videos of humans. Underlying this work is a fundamental question – how are these models able to transfer to downstream robotic control tasks effectively? A set of *distribution shifts* separates the pretraining and the downstream robotic control data distributions – for example, different tasks, camera configurations, visual features, behaviors, and morphologies. Understanding these shifts is key to understanding where existing methods fall short, and is also critical for developing better pretraining approaches moving forward. This understanding is our key contribution: we present a large-scale empirical study that identifies 5 types of distribution shifts, and uses simulation to generate precise, controlled pretraining data and target tasks that capture each shift. Given these datasets, we analyze how various pretraining methods perform as we vary the type and magnitude of each shift. Our experiments show that while distribution shifts impact performance, we can often *overcome* these shortfalls through a combination of diversity coupled with test-time adaptation. We also identify settings where surprisingly, the benefits of diversity and adaptation outweigh the cost of the distribution shift, where pushing models further out of distribution leads to improved downstream performance.

1 Introduction

Research in robotic manipulation is guided by an aspiration for systems that can learn from a handful of examples and generalize efficiently to novel scenes. One way towards this aspiration is by pretraining visual state representations, leveraging diverse datasets that encapsulate features of real-world manipulation – interactions such as simple pick and place and nuanced behaviors like opening the pull tab on cardboard cereal boxes. Pretraining has been effective in natural language processing [11, 50, 5] and computer vision [21, 7, 12], with pretrained models forming a strong base for downstream tasks [67, 66, 10]. While these models exhibit remarkable generalization potential, a key ingredient to their success has been *enormous* amounts of rich, source data (e.g., language on public forums, images in the wild) – data at a scale that we lack in robotic manipulation.

To mitigate this, recent work on pretraining visual state representations [46, 72, 42] has looked into in-the-wild datasets of human videos instead [17, 9, 41, 18], hoping to transfer enough knowledge to generalize to downstream manipulation tasks. Though this work has been effective in enabling sample-efficient learning on a small set of specific tasks, we have yet to understand *where* this generalization ability comes from, and on what types of downstream tasks these approaches may fall short. Transferring representations learned from human videos to robotic control involves navigating a set of different *distribution shifts* between the source pretraining data and downstream target task data. This work seeks to find an understanding of this by studying the following question: *what are*

Preprint. Under review.

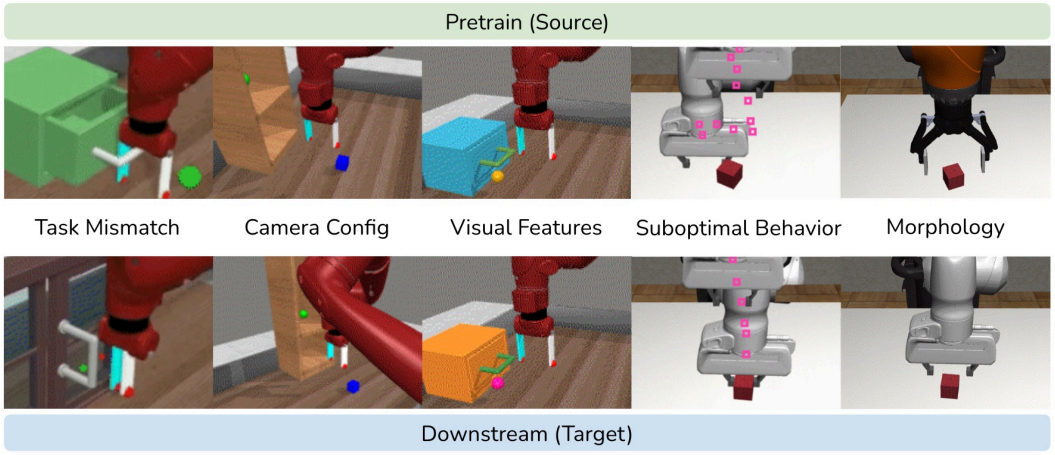


Figure 1: An overview of our empirical study on how pretrained state representations for robotic control transfer across distribution shifts. We evaluate on the 5 shifts pictured above.

the different types of distribution shifts underlying video representation learning for control, and what is their effect on downstream performance?

The aforementioned pretraining approaches [46, 72, 42] show improved sample efficiency and generalization relative to the dominant *tabula rasa* approaches in robot learning. To better understand *how* these methods that pretrain on human videos see success when adapting to downstream robot control tasks, we identify and measure the effect of different types of distribution shifts between pretraining and target task data. We seek to answer questions about when to expect efficient transfer across a set of 5 representative distribution shifts: **Task Mismatch** – if the pretraining data does not fully "cover" the full distribution of objects and interactions in the downstream tasks, are the learned representations still helpful? **Camera Configuration** – how does training on various camera angles transfer to new camera positions? **Visual Features** – how do shifts in color and texture affect downstream performance? **Suboptimal Behaviors** – how does the quality of behaviors in our source data, impact downstream performance? Finally, **Morphology** – how does pretraining affect downstream transfer to a disjoint embodiment (e.g., a different robot).

Our contribution is a large-scale empirical study of the effects of these 5 different types of distribution shifts between pretraining and target task data distributions. For each shift, we design precise "source" pretraining data and "target" control tasks. We evaluate a spectrum of representation learning approaches including methods that reconstruct visual inputs via a latent bottleneck [33, 13], contrastive methods [54], and forward predictive methods [68, 34]. To understand how data diversity affects downstream performance in the extreme, we include R3M [42], a model pretrained on 3,000 hours of human egocentric manipulation. We evaluate each pretraining method across different amounts of adaptation data, random seeds, and each shift.

We also perform a series of systematic ablations; many data sources for pretraining (e.g., human data) do not come with any action supervision (e.g., hand or end-effector pose); to understand the effect of this, we examine the impact of *action-conditional* and *state-only* data on pretrained representations. We also evaluate the impact of various downstream adaptation paradigms. Our takeaways are clear:

1. Unsurprisingly, large distribution shifts between source and target domains make frozen pretrained representations less effective on average.

2. However, fully finetuning pretrained representations on minimal target task data recovers much of the original performance, as long as the shift is not extreme.

3. Finally, behavioral shifts provide insight on what pretraining data matters; training on diverse behaviors (with different amounts of suboptimality), leads to flexible representations that can be adapted to *exceed* the performance pretraining directly on the target distribution.

This paper confirms much of the empirical knowledge around pretraining and representation learning; if there is certainty around the target domain, then pretraining on data from that distribution is often the right choice. But our analysis also shows that if the pretraining data is sufficiently diverse *and* high-coverage, one can learn flexible representations that accelerate downstream learning.

2 Related Work

We build on a long tradition of work spanning state representation learning for control and work that specifically builds frameworks to analyze and understand distribution shifts.

State Representation Learning for Control. Representation learning components are pervasive in high-dimensional control settings, especially when dealing with visual observation spaces. These methods learn to condense an observation down to a lower-dimensional space that captures only the features relevant for informing control. Typical methods for representation learning include reconstruction or forward modeling [32, 40, 16, 69, 13, 65, 28, 2, 15, 34, 20, 38], inverse modeling [47, 55, 1, 75], and contrastive learning [64, 57, 54, 42, 43], amongst others [36].

While these prior works are legion, the focus tends to be on learning representations *tabula rasa* updating representations online while learning to solve the target task. In contrast, this work focuses on pretrained state representations learned *offline*. Recent work [46, 72, 42] in this regime looks at different – often large – datasets that are quite different from the target tasks (e.g., egocentric videos of humans cooking), then finetuning these learned representations for downstream control. For example Parisi et al. [46] study several representations pretrained on standard computer vision datasets like ImageNet [10], evaluating on several control tasks, performing a study of when data augmentation during pretraining is helpful. Separately, Nair et al. [42] introduce R3M, a model that uses videos and language annotations from the large Ego4D dataset [18] in addition to language and time contrastive losses [54, 49] to learn state representations for imitation learning.

Other work performs representation learning from fixed datasets with different augmentations [6], or use data spanning multiple disjoint datasets collected in the same general environment [73]. Each work makes different assumptions when it comes to data – be it in-domain demonstrations, disjoint demos in the same environment, static images from large databases, or videos of humans. The goal of this work is to understand how different choices of data affects the downstream performance. Rather than propose new methods, we perform an empirical investigation that studies the distribution shift between the pretraining and downstream task distribution.

Understanding Distribution Shifts. A rich body of work studies distribution shifts in machine learning with broad coverage of problems such as out-of-distribution detection [37, 23, 24, 35] and generalization [61, 14, 63, 56, 58, 3]. Informed by this work, recent work has proposed concrete benchmarks to concretely measure model performance over various taxonomies of distribution shift [22, 3, 71, 30]. The most relevant work by Wiles et al. [70] provides a fine-grained analysis of various hand-crafted distribution shifts with a combination of real and synthetic datasets, in the context of image classification. The key difference with our work is our focus on taxonomizing and evaluating shifts in the context of pretraining visual state representation for transfer to downstream control tasks; unlike work in classification and regression, learning from videos, and evaluating on sequential decision making tasks present a series of completely new distribution shifts around behaviors, camera configuration, and morphologies, that we actively study in this work.

Within robotics, research into distribution shifts has focused on specific sub-problems such as domain adaptation [62, 19], sim-to-real transfer by learning inverse dynamics models [8], and domain randomization for cross-environment transfer [59, 52, 48, 44]. Each of these works focus on a *single* type of shift, and analyze a *method* for addressing the shift, rather than studying the underlying data distribution. In this work, we instead focus on the data directly, fixing the methods, and evaluating how pretrained models transfer given different amounts of adaptation data and under different adaptation schemes, under each of the 5 distribution shifts we identify.

3 Problem Statement and Experimental Setup

We study how the pretraining data distribution for visual state representation learning influences downstream control performance. Our experiments take the following form: We first learn a visual state representation by pretraining a given method on a fixed dataset. We then train a policy on top of the pretrained state representation via behavioral cloning on demonstrations from the target task distribution. We learn policies by either training a small network on top of the *frozen* pretrained representation, or by *finetuning* both the policy network and pretrained encoder together.

Formally, the pretraining dataset $\mathcal{D}$ consists of videos represented as T RGB frames $[I_0, I_1, \ldots, I_{T-1}]$ *without* action labels. We train an encoder q_ϕ that maps images I to a representation z, where $z = q_\phi(I)$. We step through the different encoding methods and their objectives below.

To adapt to downstream tasks after pretraining, we train policies on small datasets $\mathcal{D}_{demo}$ of trajectories $[[I_0, s_0], a_0, [I_1, s_1], a_1, ..., [I_{T-1}, s_{T-1}]]$ with behavioral cloning (BC). We concatenate the visual representation with the robot's proprioceptive state as input to the policy, resulting in $a_t = \pi_\psi([q_\phi(I_t), s_t])$. When training, we either freeze the representation and only update the policy, or finetune all parameters end-to-end. To avoid feature distortion when finetuning, we follow the two-step strategy of linear probe, then full finetune (LP-FT) [31]. We include an ablation study of other adaptation methods in §5.

Dataset Construction. In this work, we identify 5 shifts – task mismatch, camera configuration, visual features, suboptimal behaviors, and morphology – each of which is inspired by salient shifts that appear when transferring from a large offline dataset for pretraining (e.g., from human videos, videos of other robots performing tasks) to target control tasks (e.g., a different robot manipulator performing a pick-and-place task in a new environment). While these shifts are not exhaustive, they do cover different parts of an underlying Partially Observable Markov Decision Process (POMDP); task mismatch directly affects the reward function, camera configuration and visual features affect the observation model, suboptimality implicitly affects the reward and transition model, and morphology affects both transitions and observations.

To carefully study each of the 5 identified distribution shifts, we first identify a set of downstream tasks from which we compose our *fixed* target distribution. Subject to this fixed target distribution, we then generate *multiple pretraining datasets* that represent different types and magnitudes of distribution shift. Each shift has an "in-distribution" pretraining dataset $\mathcal{D}_{in\text{-}dist}$, where the videos are collected directly in the fixed target environment. Each dataset consists of 200 videos from a noise injected closed-loop policy. We follow a similar procedure for each of the different "distribution shifted" pretraining datasets.

Concretely, take the Camera Configuration as an example. We first start with a downstream target environment with a *single* camera position (first component of Fig. 5). To construct our first distribution shift dataset (with shift of small magnitude), we then collect our first pretraining dataset by repeatedly injecting a small amount of noise into the camera angle, obtaining dataset $\mathcal{D}_{shift1}$ (second component of Fig. 5). By injecting more and more noise, we can move the distribution further away from the target, resulting in a moderately shifted dataset $\mathcal{D}_{shift2}$ and further shifted dataset $\mathcal{D}_{shift3}$. We follow similar procedures for each of shifts, with more details in §4.

Pretraining Objectives. We consider three objectives for training: reconstruction, forward prediction, and contrastive learning. For reconstruction and forward prediction, we employ the β-VAE [25] which minimizes:

$$\mathcal{L}_{\text{vae}} = \mathbb{E}_{I_i \sim \mathcal{D}}[\mathbb{E}_{q_\phi(z|I_i)}[\log p_\theta(I_{target}|z)] - \beta D_{KL}(q_\phi(z|I_i)||p(z))]$$

where $p_\theta(I|z)$ is a decoder which is trained jointly with the encoder $q_\phi(z|I)$ to reconstruct the frame I from the latent bottleneck z. I_{target} is set to the current frame I_i for reconstruction and n-frames ahead (I_{i+n}) for forward prediction. We use $n = 5$ for all our forward prediction pretraining experiments.

For contrastive learning, we train Time Contrastive Networks (TCN) [54] by sampling a batch B of quadruples (I_i, I_j, I_k, I') from the dataset, where $j > i$, $k > j$ and I' sampled from a different video. We minimize an InfoNCE loss [64] of the following form:

$$\mathcal{L}_{\text{tcn}} = -\sum_{(I_i, I_j, I_k, I') \in B} \log \frac{e^{S(I_i, I_j)}}{e^{S(I_i, I_j)} + e^{S(I_i, I_k)} + e^{S(I_i, I')}}$$

Here, S measures the similarity between the representation of two images. In our experiments, we use the negative L2 distance $S(I_1, I_2) = -||q_\phi(I_1) - q_\phi(I_2)||_2^2$.

4 Experiments and Results

We leverage simulation to precisely control and generate data representative of the shifts we care about. Namely, we use tasks and features from MetaWorld [74] and RoboMimic [39], both of which are built with Mujoco [60]. MetaWorld provides a diverse set of configurable environments, making it ideal for studying shifts such as task mismatch, camera configuration, and visual features, whereas RoboMimic comes with out-of-the-box support for multiple robots and precollected human demonstration data of differing qualities, allowing us to directly study the remaining shifts we identified around suboptimal behaviors and morphology transfer.

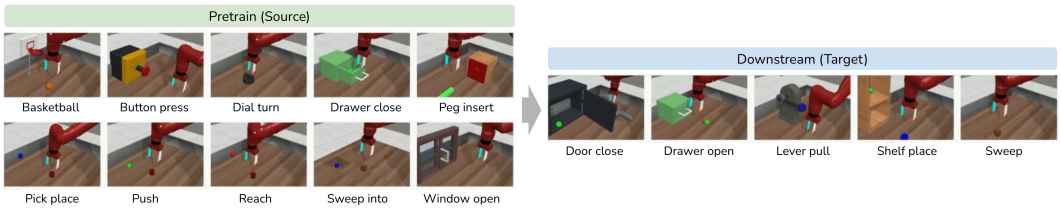


Figure 2: *Task Mismatch* – when the pretraining data does not fully "cover" the full distribution of objects and interactions in the downstream tasks. To study this, we use MetaWorld ML10 and collect pretraining data with tasks on the left, then evaluate the representation with tasks on the right.

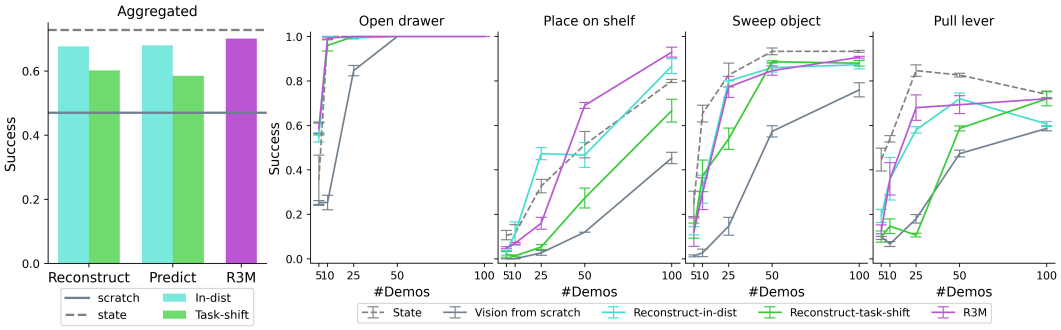


Figure 3: *Task Mismatch – Results:* **Left.** We plot the performance of pretraining on in-distribution (`in-dist`) data and Task Mismatch (`task-shift`) data for two methods, aggregated across 5 environments, 5 dataset sizes, and 3 seeds. **Right.** We look at pretraining with a Reconstruction-based approach, and evaluate downstream control performance against other methods.

While we pretrain all our models on 200 examples as described in §3, for adaptation for each task, we operate in a low-data regime; to study the effects of distribution shifts on sample-efficient transfer, we consider downstream task learning with $n = [5, 10, 25, 50, 100]$ demonstrations for each shift. To evaluate success, we compute average task success rate by rolling out our policy across 50 episodes. All tasks contain inherent variability, e.g. different initial object positions and robot start state.

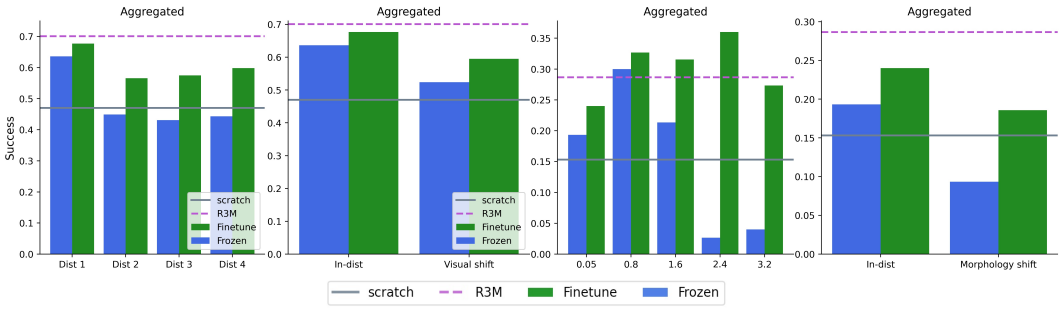
To provide two extra points of comparison for our various pretrained models, we first run behavioral cloning from scratch on the ground-truth simulation state, which provides a low-dimensional vector of absolute poses of objects in the scene; this is meant to represent a "full-information" upper-bound, though note that given a strong enough visual representation and enough finetuning data, it is possible for models to surpass this performance. As the second reference point, we run behavioral cloning from scratch on images, using a randomly initialized convolutional neural network encoder, to evaluate whether pretrained representations exhibit positive transfer.

4.1 Task Mismatch

The first shift we look at is task mismatch, where the downstream target task is not captured in the pretraining dataset. This is arguably the most common, natural shift that most agents will face when transferring pretrained representations from other datasets. Yet, despite its commonality, this type of transfer can be challenging, as it could involve perceiving new objects, scenes, and skills. Taking a workshop as an example, tools like a level could be absent during pretraining, and similarly, the robot would not necessarily have a skill for identifying the fulcrum location, and pull down on the lever appropriately. Despite this, the goal would be to learn to pull the lever from few demonstrations.

We use MetaWorld ML10 benchmark for this experiment, as depicted in Fig. 2, comprised of 10 pretraining environments, and 5 disjoint target environments. We fix the target environments as downstream tasks, and pretrain representation models either videos of the robot performing 10 training tasks combined (`task-shift`) or on each of the target tasks (`in-dist`).

In Fig. 3 (left), we plot the performance of reconstruction and prediction models, together with R3M. We aggregate the performance across all sizes of $\mathcal{D}_{demo}$, 5 environments, and 3 random seeds. We adapt all representations with LP-FT, and we refer readers to the appendix for detailed results with frozen representations and contrastive methods. All pretrained representations outperform training



(a) Camera configuration. (b) Visual Features (c) Suboptimal Behaviors (d) Morphology

Figure 4: *Results for the 4 core shifts, aggregated across environments, number of demos, and random seeds.*

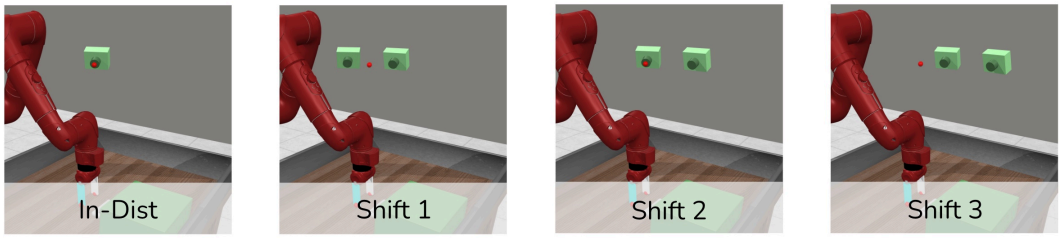


Figure 5: *Camera Configuration* – transfer to new camera positions unseen in pretraining data. The red dot marks the downstream camera, and each shifted distribution is uniform bounded by the two rendered cameras.

from scratch, with `in-dist` pretraining achieving performance close to oracle state. On `task-shift` however, the success rate drops by almost 10%. We also note that R3M, despite training on only human videos, is able to achieve close to `in-dist` data. Separately, in Fig. 3 (right), we focus only on the reconstruction pretraining method and plot success rate as we increase the size of $\mathcal{D}_{demo}$ for different target tasks. Despite the varying task difficulty, the ranking of different methods is relatively stable across environments.

4.2 Camera Configuration

The second shift we look at is camera configuration, where the location of the camera in the target task differs from the camera configurations seen during pretraining. These changes could be small (e.g., the camera gets jostled) or large (e.g., we deploy on new hardware, with new camera intrinsics/extrinsics). We focus our experiments on varying camera angles relative to the robot workspace, and illustrate shifts in Fig. 5. During downstream evaluation, the camera is fixed in the pose illustrated in the `in-dist` subfigure. The camera's position is marked with a red dot that is kept at the same pixel location in the image across all subfigures for reference.

In `shift-1`, the camera angle is sampled from a uniform distribution bounded by the two cameras visualized in the figure. They span $\pm 10°$ around the target camera's orientation. For `shift-2` and `shift-3` we translate move the distribution to the right, by $10°$ at a time (ensuring all angles do not occlude any relevant objects). Similarly to Task Mismatch, we then use the 5 target tasks from MetaWorld ML10 for evaluation. Since we only want to evaluate camera shift in this experiment, we use the same 5 tasks for pretraining as well. As we have 3 shifts for each task and 3 random seeds, this gives us $5 \times 3 \times 3$ models for each training objectives.

We plot the aggregated performance in Fig. 4 for both frozen representations and LP-FT representations. Camera shift drops the success rate of frozen representation by as much as 21% compared to pretraining directly on the target distribution, resulting in downstream performance worse than training from scratch. Despite the performance with frozen representations, the pretrained representations provide a surprisingly good basis for adaptation. After LP-FT, the gap between pretraining on the target and the pretraining on the shifted distribution shrinks to 8-12%.

4.3 Visual Features

While we want the pretraining data to cover a wide range of visual phenomenon, the robot may still encounter objects with new appearances at test time; as such the third shift we look at are shifts in visual features (e.g., color, texture) between the pretraining data and downstream task. We focus on

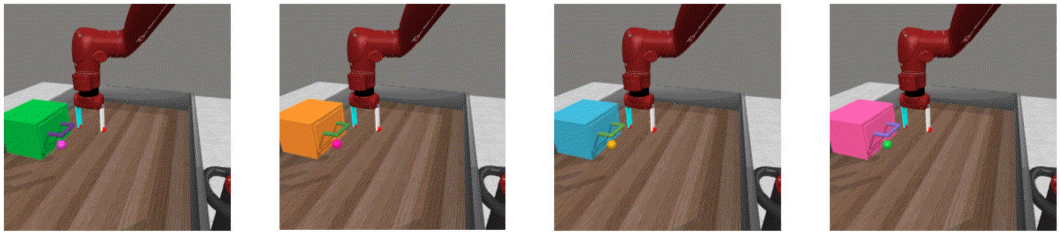


Figure 6: *Visual Features* – We randomize the colors and evaluate it's effect on downstream performance.

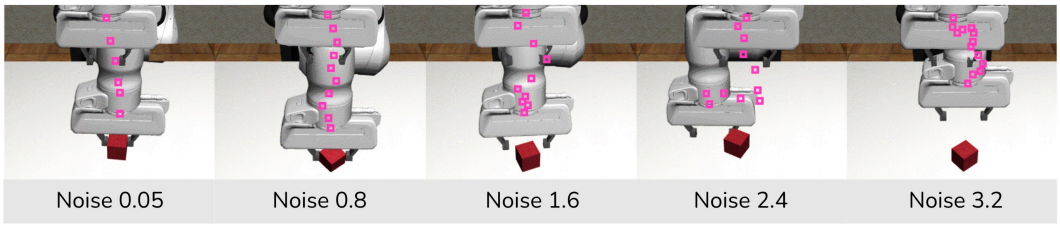


Figure 7: *Suboptimal Behaviors* - how does the quality of behaviors in source data impact downstream performance? We inject different amounts of noise to the policy to construct 5 datasets with gradual suboptimality.

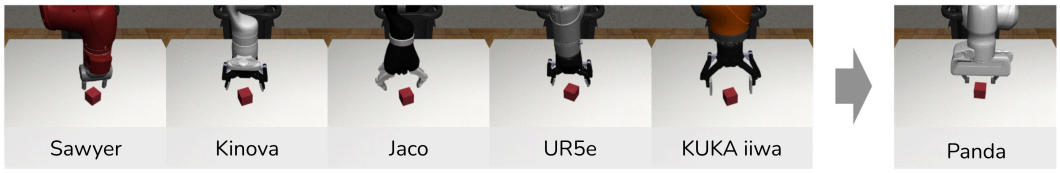


Figure 8: *Morphology* - we study the transfers across embodiments by pretraining representation on 5 different robot arms and evaluating downstream on a new robot.

the varying color of various objects in a scene, and conduct experiments with the 5 target tasks of MetaWorld ML10. Fig. 6 shows samples from the shifted pretraining dataset, where the drawer is assigned a random color for each video while the downstream distribution has a fixed color. The aggregated performance in Fig. 4 suggests that pretraining with randomized colors outperforms training from scratch. Similar to the camera configuration shift, adapting the representation closes the gap between in-distribution and visual shift pretraining from 11% to 8%.

4.4 Suboptimal Behaviors

The fourth shift we look at is regarding the agent behavior when solving tasks, especially transferring from suboptimal pretraining data to downstream "optimal" control. This shift is particularly prominent when data is coming from human, either through teleoperating a robot or with their own hands: human behavior can be non-deterministic and include lots of spurious data (e.g., pausing to brush ones hair back), while a scripted or learned robot policy would provide optimal and predictable demonstrations.

We focus our experiments on injecting different magnitudes of action noise into optimal demonstration data, to create different pretraining datasets. We illustrate some example trajectories in Fig. 7 traced with pink squares. We sample noise from a Gaussian with standard deviation in $[0.05, 0.8, 1.6, 2, 4, 3.2]$. This gives us 5 distinct data distributions, from near-optimal behavior gradually to random motion. The success rate in pretraining data drops from 100% to near-zero when noise magnitude exceeds 1.6.

We experiment with the Lift and Can task from RoboMimic, where the robot needs to either pick up a red cube on the table, or pick up and move a red can. After pretraining, we imitate the policy with 0.05 noise injected, and plot the aggregated performance of Lift across different number of demonstrations and random seeds in Fig. 4. We defer the plot of Can to the supplementary material. With a frozen pretrained representation, the performance peaks at noise-level 0.8, then quickly declines to less than 5% success for noise-levels of 2.4 and 3.2, which is worse than training from scratch. However, with adaptation, the performance continues to increase with more noise and peaks at noise-level 2.4, then remains high at 3.2 where the behavior are close to random. Notably, for both frozen and

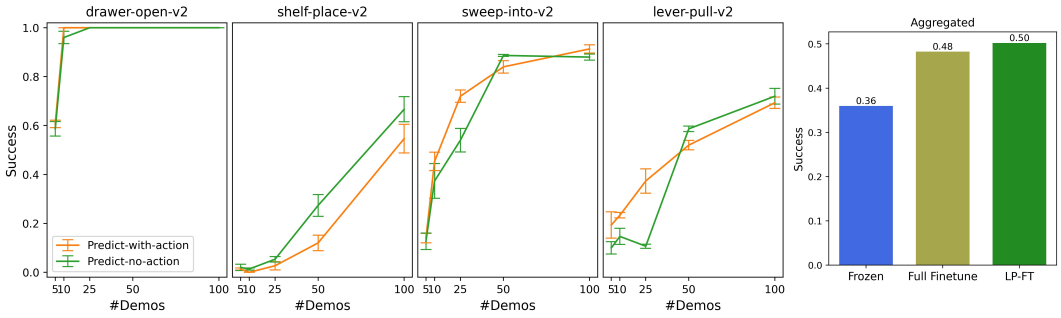


Figure 9: *Ablation Studies.* On the left, we compare the success rate of representation trained with and without action on downstream tasks. On the right, we plot the performance of three adaptation methods.

finetuned representations, the peak performance does not coincide with in-distribution pretraining (i.e. noise-level 0.05), unlike all prior shifts.

4.5 Morphology

Finally, morphology mismatch is when the robot we want to control at test time is not present in the pretraining data. This includes the scenarios when pretraining data is coming from humans or other types of robots. Overcoming such mismatch is challenging because the embodiment tends to form a large part of the observation, and affects how the robot interacts with the environment.

We use the same downstream tasks as in behavior shift, while collecting pretraining data from 5 robots: {Sawyer, IIWA, Jaco, Kinova3, UR5e}. We plot the aggregated performance in Fig. 7. Similar to the result from camera configuration shift, the frozen representation with morphology mismatch shows no positive transfer when compared to training from scratch, while still producing a representation amenable to adaptation. LP-FT doubles the frozen representation's success rate and reduces the gap from in-distribution pretraining to 11%.

5 Ablation Studies

5.1 Action vs. No Action

One caveat when using in-the-wild videos to pretrain visual state representations is the lack of action information. This prevents us from applying standard techniques like dynamics modeling. In this section, we investigate how action information influences the downstream performance of pretrained representation. We study this question against the backdrop of the task mismatch distribution shift, where we pretrain representations on data from 10 pretraining tasks from MetaWorld ML10, and downstream performance on the 5 downstream target tasks. We compare the performance of representations learned via standard dynamics modeling, i.e. predicting from (I_t, a_t) to I_{t+1}, against no-action prediction that maps I_t to I_{t+1} directly.

We plot the result for each environments in Fig. 9 (left). Across environments, representations learned without action information performs on-par with those from standard dynamics modeling.

5.2 Adapting Pretrained Representations

Identifying the best way to adapt pretrained state representations to downstream tasks remains an open problem [46, 45, 42]. Two typical approaches for adaptation are (1) training a policy (additional, added parameters) with the frozen representation as input and (2) full finetuning of all parameters (in the policy *and* in the representation learning model). There are pros and cons to each approach; frozen representations might not capture all the information we need from the image, and therefore lack flexibility in the presence of large distributino shifts. On the other hand, full finetuning schemes distort pretrained features, leading to poor out-of-distribution generalization [31, 53]. To mitigate this last problem though, Kumar et al. [31] show that the two-step strategy of linear probing then full fine-tuning (LP-FT) outperforms both fine-tuning and linear probing on a range of computer vision benchmarks, and also improves OOD performance. We conduct experiments comparing frozen representation, full finetuning, and LP-FT in the context of the task mismatch distribution shift on MetaWorld ML10. In Fig. 9 (right) we show that LP-FT outperforms both full finetuning and frozen representations across the 5 tasks and varying number of demonstrations.

6 Discussion

Each of the distribution shift experiments hint at a broad conclusion: as we increase the magnitude of the distribution shift between our pretraining data distribution and our target task data distribution, downstream policy performance seems to suffer. This is especially true for shifts like Task Mismatch (§4.1), Visual Features (§4.3), as well as Morphology (§4.5). However, not all of our shifts behave in this fashion; indeed, the Suboptimal Behaviors (§4.4) shift shows a key difference. As we increase the magnitude of the shift – in this case, the amount of noise in our behavioral data, leading to suboptimal videos where the robot takes seemingly random motions – we see a surprising divergence between the performance of *frozen* vs. *finetuned* downstream performance. Frozen performance degrades quite quickly as we increase the magnitude of the shift, yet finetuning follows a non-linear trend, where the finetuning downstream performance actually grows, eventually exceeding the performance of the representation that is pretrained directly on the target domain!

This seems to indicate something deeper about the role of diversity in creating *flexible representations*. Based on these results, it seems like the less diverse the pretraining data, the less amenable to finetuning the representation will be. This is demonstrated across most of our experiments by just looking at the delta between the frozen and finetuned representations when pretraining directly on target domain data. Instead, adding diverse data – and nonintuitively, even just random action data in these robotics contexts – produces arbitrarily worse frozen representations (as evidence by both the Camera Configuration and Suboptimal Behavior) results, but produce representations that can easily recover most of, if not *exceed*, the performance gap when finetuned.

This leaves us with the following takeaways:

Takeaway 1. For frozen representations, large gaps between pretraining and downstream task distributions lead to worse downstream performance.

Takeaway 2. Generally, across most of the shifts we study, finetuning representations pretrained on sufficiently diverse data can recover most of the performance lost by the shift in question.

Takeaway 3. Optimizing for *diversity* in pretraining data, even if one cannot cover or approximate the target distribution allows for flexible representations that are maximally amenable to finetuning and adaptation, allowing for sample-efficient transfer – transfer that can exceed the performance of training directly on the target distribution itself.

This final takeaway should inform future work on building pretraining datasets – coverage and scale alone are not enough. It is just as important to collect data captures diverse behaviors – including suboptimality – that cover multiple ways of performing a given task.

7 Conclusion, Limitations, and Future Work

This work presents a large-scale empirical study of 5 different distribution shifts that arise when transferring pretrained visual representations to downstream tasks in robotic control. At first glance, our results are not surprising; we confirm that when the target data distributions are known, pretraining on broad source datasets that cover the various targets is beneficial. However, we surprisingly find that in some cases, diversity of the source data – and in the case of robotics data, explicitly suboptimal data or data that encompasses multiple *modes* – can lead to improved performance of the pretrained representations when finetuned. Ensuring that future pretraining datasets are not only broad coverage, but capture diverse behaviors will be key to learning powerful visual state representations.

This work is only the beginning – as we develop more datasets, pretrained models, and different evaluation tasks, the need for targeted studies around this type of transfer will be necessary. The key limitations of our work are the scope of shifts we study and the construction of shift datasets. We work in simulation to generate controlled data ascribing to our shifts; capturing the effect of distribution shifts in the real world will be much harder, but is necessary for robust claims. A promising direction for future work would develop similar studies on real robots, as well as evaluate other types of representation learning. Robotics is inherently multi-modal, fusing inputs from different sensor modalities and even different types of human feedback; understanding shifts that go beyond visual state representations will become increasingly important in the years to come.

References

[1] Pulkit Agrawal, Ashvin Nair, P. Abbeel, Jitendra Malik, and Sergey Levine. Learning to poke by poking: Experiential learning of intuitive physics. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[2] Samuel Alvernaz and Julian Togelius. Autoencoder-augmented neuroevolution for visual doom playing. *2017 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–8, 2017.

[3] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[4] Mohammad Babaeizadeh, Mohammad Taghi Saffar, Suraj Nair, Sergey Levine, Chelsea Finn, and D. Erhan. Fitvid: Overfitting in pixel-level video prediction. *ArXiv*, abs/2106.13195, 2021.

[5] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[6] Cynthia Chen, Xin Chen, Sam Toyer, Cody Wild, Scott Emmons, Ian S. Fischer, Kuang-Huei Lee, Neel Alex, Steven H. Wang, Ping Luo, Stuart J. Russell, P. Abbeel, and Rohin Shah. An empirical investigation of representation learning for imitation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, pages 1597–1607, 2020.

[8] Paul Francis Christiano, Zain Shah, Igor Mordatch, Jonas Schneider, Trevor Blackwell, Joshua Tobin, Pieter Abbeel, and Wojciech Zaremba. Transfer from simulation to real world through learning deep inverse dynamics model. *arXiv preprint arXiv:1610.03518*, 2016.

[9] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The EPIC-KITCHENS dataset. In *European Conference on Computer Vision (ECCV)*, 2018.

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

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Association for Computational Linguistics (ACL)*, pages 4171–4186, 2019.

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

[13] Chelsea Finn, Xin Yu Tan, Yan Duan, Trevor Darrell, Sergey Levine, and P. Abbeel. Deep spatial autoencoders for visuomotor learning. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 512–519, 2016.

[14] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, Francois Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research (JMLR)*, 17, 2016.

[15] Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G. Bellemare. Deepmdp: Learning continuous latent space models for representation learning. In *International Conference on Machine Learning (ICML)*, 2019.

[16] Ross Goroshin, Michaël Mathieu, and Yann LeCun. Learning to linearize under uncertainty. In *NIPS*, 2015.

[17] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fründ, Peter N. Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland 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.

[18] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Q. Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh K. Ramakrishnan, F. Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Z. Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebreselasie, Cristina González, James M. Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jáchym Kolár, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran K. Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbeláez, David J. Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard A. Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the world in 3, 000 hours of egocentric video. *ArXiv*, abs/2110.07058, 2021.

[19] Abhishek Gupta, Coline Devin, Yuxuan Liu, P. Abbeel, and Sergey Levine. Learning invariant feature spaces to transfer skills with reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2017.

[20] Danijar Hafner, Timothy P. Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *ArXiv*, abs/1912.01603, 2020.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[22] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations (ICLR)*, 2019.

[23] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[24] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations (ICLR)*, 2019.

[25] Irina Higgins, Loïc Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

[26] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[27] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ArXiv*, abs/1502.03167, 2015.

[28] Maximilian Karl, Maximilian Sölch, Justin Bayer, and Patrick van der Smagt. Deep variational bayes filters: Unsupervised learning of state space models from raw data. *ArXiv*, abs/1605.06432, 2017.

[29] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

[30] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, 2021.

[31] Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. *ArXiv*, abs/2202.10054, 2022.

[32] Sascha Lange and Martin A. Riedmiller. Deep auto-encoder neural networks in reinforcement learning. In *International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2010.

[33] Sascha Lange, Martin A. Riedmiller, and Arne Voigtländer. Autonomous reinforcement learning on raw visual input data in a real world application. *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2012.

[34] Alex X. Lee, Anusha Nagabandi, Pieter Abbeel, and Sergey Levine. Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[35] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[36] Timothée Lesort, Natalia Díaz Rodríguez, Jean-François Goudou, and David Filliat. State representation learning for control: An overview. *Neural networks : the official journal of the International Neural Network Society*, 108:379–392, 2018.

[37] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[38] Hao Liu and P. Abbeel. Behavior from the void: Unsupervised active pre-training. In *NeurIPS*, 2021.

[39] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *Conference on Robot Learning (CoRL)*, 2021.

[40] Jan Mattner, Sascha Lange, and Martin A. Riedmiller. Learn to swing up and balance a real pole based on raw visual input data. In *ICONIP*, 2012.

[41] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2630–2640, 2019.

[42] Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

[43] Masashi Okada and Tadahiro Taniguchi. Dreaming: Model-based reinforcement learning by latent imagination without reconstruction. *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4209–4215, 2021.

[44] OpenAI, Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, Jonas Schneider, Nikolas A. Tezak, Jerry Tworek, Peter Welinder, Lilian Weng, Qiming Yuan, Wojciech Zaremba, and Lei M. Zhang. Solving rubik's cube with a robot hand. *ArXiv*, abs/1910.07113, 2019.

[45] Jyothish Pari, Nur Muhammad (Mahi) Shafiullah, Sridhar Pandian Arunachalam, and Lerrel Pinto. The surprising effectiveness of representation learning for visual imitation. In *Robotics: Science and Systems (RSS)*, 2022.

[46] Simone Parisi, Aravind Rajeswaran, Senthil Purushwalkam, and Abhinav Kumar Gupta. The unsurprising effectiveness of pre-trained vision models for control. *arXiv preprint arXiv:2203.03580*, 2022.

[47] Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017.

[48] Xue Bin Peng, Marcin Andrychowicz, Wojciech 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.

[49] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, volume 139, pages 8748–8763, 2021.

[50] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.

[51] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. Searching for activation functions. *ArXiv*, abs/1710.05941, 2018.

[52] Fereshteh Sadeghi and Sergey Levine. $(\text{CAD})^2\text{RL}$: Real single-image flight without a single real image. *ArXiv*, abs/1611.04201, 2017.

[53] Younggyo Seo, Kimin Lee, Stephen James, and P. Abbeel. Reinforcement learning with action-free pre-training from videos. *arXiv preprint arXiv:2203.13880*, 2022.

[54] Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, and Sergey Levine. Time-contrastive networks: Self-supervised learning from video. In *International Conference on Robotics and Automation (ICRA)*, pages 1134–1141, 2018.

[55] Evan Shelhamer, Parsa Mahmoudieh, Max Argus, and Trevor Darrell. Loss is its own reward: Self-supervision for reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2017.

[56] Rui Shu, Hung H. Bui, Hirokazu Narui, and Stefano Ermon. A DIRT-T approach to unsupervised domain adaptation. In *International Conference on Learning Representations (ICLR)*, 2018.

[57] A. Srinivas, Michael Laskin, and P. Abbeel. CURL: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2020.

[58] Yu Sun, Eric Tzeng, Trevor Darrell, and Alexei A. Efros. Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*, 2019.

[59] Joshua Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 23–30, 2017.

[60] E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 5026–5033, 2012.

[61] Eric Tzeng, Judy Hoffman, N. Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. In *Computer Vision and Pattern Recognition (CVPR)*, 2014.

[62] Eric Tzeng, Coline Devin, Judy Hoffman, Chelsea Finn, Xingchao Peng, Sergey Levine, Kate Saenko, and Trevor Darrell. Towards adapting deep visuomotor representations from simulated to real environments. *arXiv preprint arXiv:1511.07111*, 2015.

[63] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[64] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[65] Herke van Hoof, Nutan Chen, Maximilian Karl, Patrick van der Smagt, and Jan Peters. Stable reinforcement learning with autoencoders for tactile and visual data. *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3928–3934, 2016.

[66] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[67] Alex Wang, Amapreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations (ICLR)*, 2019.

[68] Manuel Watter, Jost Springenberg, Joschka Boedecker, and Martin Riedmiller. Embed to control: a locally linear latent dynamics model for control from raw images. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2746–2754, 2015.

[69] Manuel Watter, Jost Tobias Springenberg, Joschka Boedecker, and Martin A. Riedmiller. Embed to control: A locally linear latent dynamics model for control from raw images. In *NIPS*, 2015.

[70] Olivia Wiles, Sven Gowal, Florian Stimberg, Sylvestre-Alvise Rebuffi, Ira Ktena, Krishnamurthy Dvijotham, and Ali Taylan Cemgil. A fine-grained analysis on distribution shift. *arXiv preprint arXiv:2110.11328*, 2021.

[71] Kai Y. Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. Noise or signal: The role of image backgrounds in object recognition. In *International Conference on Learning Representations (ICLR)*, 2021.

[72] Tete Xiao, Ilija Radosavovic, Trevor Darrell, and Jitendra Malik. Masked visual pre-training for motor control. *arXiv preprint arXiv:2203.06173*, 2022.

[73] Mengjiao Yang and Ofir Nachum. Representation matters: Offline pretraining for sequential decision making. In *International Conference on Machine Learning (ICML)*, 2021.

[74] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan C. Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2019.

[75] Amy Zhang, Harsh Satija, and Joelle Pineau. Decoupling dynamics and reward for transfer learning. *ArXiv*, abs/1804.10689, 2018.

A Experiment Details

A.1 Architecture and Hyperparameters

We first describe the architecture of our representation learning models, as well as the policy network that we train on top for each of our evaluations. For all representation learning models, we use the same encoder architecture as FitVid [4], a state-of-the-art video prediction model that compresses histories down to a low-dimensional latent space, before reconstructing the next video frame. The FitVid model is comprised of "blocks," each of which consist of a sequence of a Convolutional layer followed by Batch Normalization [27], Swish activation [51], and a Squeeze & Excite unit [26]. In the original FitVid paper, the decoder architecture mirrors the encoder, with critical *skip connections* that connect the intermediate activations between the two components. While useful for video prediction, we find in our preliminary experiments that the skip connection actively *hurts* representation learning performance, as the model learns to rely on temporal redundancy embedded in the skip connection history. Removing the skip connections in this architecture for all our representation learning models leads to higher quality representations, and improved downstream performance. We train different encoders for Metaworld and RoboMimic, mostly due to the inherent observation resolution; for Metaworld, we learn an encoder that takes in an image with resolution 64×64, while for RoboMimic, we learn an encoder that takes in an image with resolution 84×84. Both encoders compress observations down to a representation with dimensionality 256.

Our policy is built atop this learned encoder, and is simple; we use a multi-layer perceptron (MLP) with 4 hidden layers and 2048 hidden units each. In our preliminary experiments, we found this architecture to outperform policies with fewer hidden units. Rather than directly outputting a continuous action, our policy outputs a k-dimensional mean and variance (where k is the dimensionality of the action space), which parameterizes a Gaussian distribution that is used to sample actions. In MetaWorld, $k = 4$, and in RoboMimic, $k = 7$.

We train the representation learning model with Adam [29] and a learning rate of 0.001. We set the batch size two times the episode length of the environment, which is around 100 steps. We train the policy with Adam [29] as well, adopting a larger learning rate of 0.001 when learning from directly from the ground-truth environment state or when using frozen representations, while using a smaller learning rate of 0.0001 when finetuning both the encoder and the policy. The batch size is set to eight times the episode length of the environment. We refer readers to the code in supplementary material for the exact implementations of each model and optimization procedure.

We stress that all architectures and hyperparameters are fixed throughout our empirical study.

A.2 Checkpoint Selection

Evaluating each model checkpoints during policy learning may incur formidable time cost in real robotic applications. Taking this into consideration, we minimize the amount of online interactions needed in both pretraining and downstream finetuning. Rather than evaluate every model checkpoint in the environment, we select the pretrained encoder checkpoint by evaluating its frozen representation on one fixed downstream task and demonstration dataset. The best encoder subject to this procedure is then used for **all** downstream tasks.

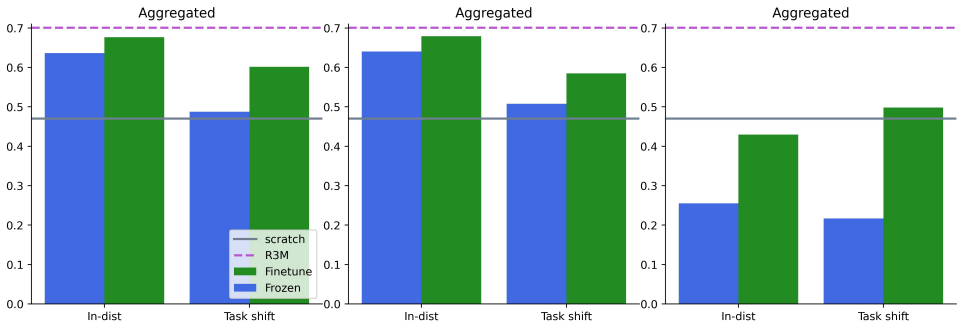
For policy checkpoint selection on the downstream task, we use a **fully offline** procedure where, for each training setting, we run 5 seeds with different policy initializations, then choose the 3 seeds with lowest mean squared error on a held-out set of validation demonstrations, and use their corresponding checkpoints. We report the mean and standard error across these 3 selected checkpoints, for all experiments in our study.

A.3 Compute Statistics

We use a combination of Nvidia RTX 3090 (24G) and Titan RTX (12G) GPUs. Because of the large batch size required by some of our experiments, we employ gradient accumulation to reduce memory usage. The whole study took around 30 days with 10 GPUs running in parallel. In total, we run more than 8000+ BC experiments from vision, pretrained 100+ representation models, and generated 100+ GB robot data.

B Additional Results

In the following subsections, we present additional results on each of our distribution shifts with separate representation learning models.



(a) Reconstruction (b) Prediction (c) Time Contrastive

Figure 10: Additional results for the Task Mismatch distribution shift (complements Fig. 3). We plot aggregated performance for Reconstruction, Prediction and Time Contrastive methods. We note that Reconstruction and Prediction achieves similar performance, while Time Contrastive models pretrained with Task Mismatch distribution shift outperforms in-distribution pretraining.

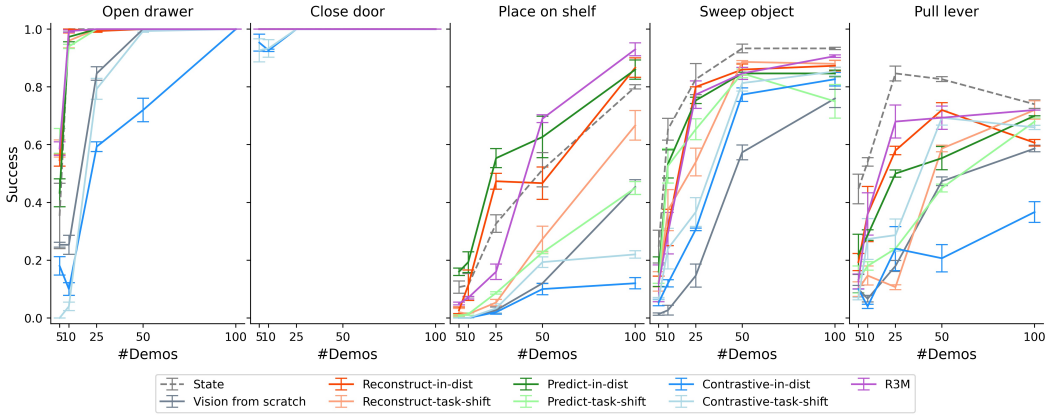


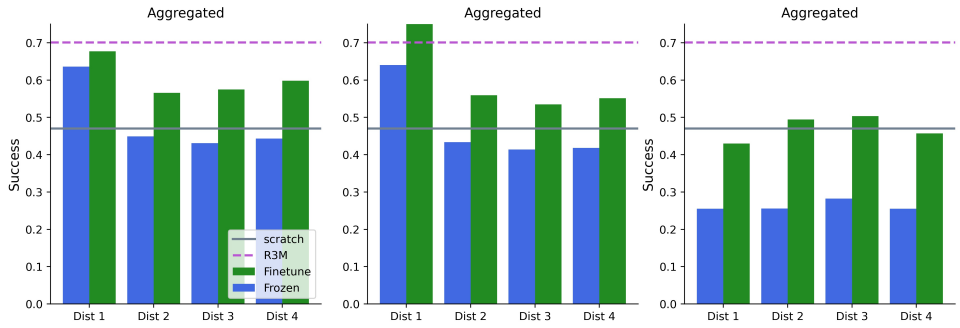
Figure 11: Detailed results for Task Mismatch across 5 environments, 9 methods and 5 demo dataset sizes.

B.1 Task, Camera and Color Shift

In this section, we include additional results on the Task Mismatch, Camera Configuration, and Visual Features distribution shifts, which we evaluated using targeted datasets grounded in the MetaWorld environments.

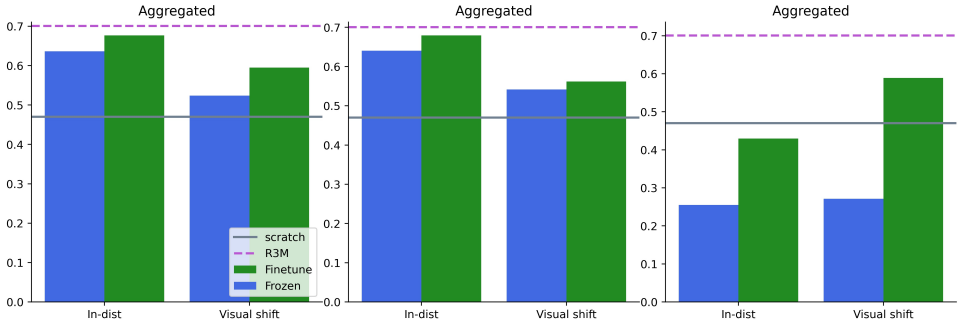
Starting with Task Mismatch, we plot the aggregated performance of all three representation learning methods in Fig. 10. We observe that representations learned with reconstruction objectives perform similarly to those learned with forward prediction objectives. Furthermore, for both methods, finetuning reduces the performance gap between in-distribution and distribution shift pretraining. Separately, representations learned with time contrastive learning reach worse performance than these other methods: after finetuning, contrastive representations only slightly outperforms training from scratch, while reconstruction representations and forward prediction representations surpass training from scratch by at least 10% to 20% success rate. We hypothesize that the effectiveness of time contrastive pretraining depends more heavily on the diversity of data. We will revisit this hypothesis as we observe similar trends in the next few distribution shifts. Wrapping up Task Mismatch, we provide a plot for all three representation methods tested on five downstream tasks in Fig. 11. We note that in Fig. 3, we omit plotting the "close door" environment as methods all perform similarly well (it is a quite easy task) and reach 100% success with less than 25 demos.

In Fig. 12, we provide results for the Camera Configuration distribution shift with both the forward prediction and time contrastive representation learning methods, in addition to the reconstruction method from the main body of the paper. Consistent with the trend observed for the Task Mismatch shift, both the reconstruction and forward prediction methods perform similarly, with both approaches significantly outperforming the representations learned via time contrastive learning. We also note that for the time contrastive representations, training on data with camera configurations (`Dist {2, 3, 4}`) outperform training on the in-distribution split (`Dist 1`). This is interesting, as one might



(a) Reconstruction (b) Prediction (c) Time Contrastive

Figure 12: Additional results for the Camera Configuration distribution shift (complements Fig. 4(a)). We again note similar performance from Reconstruction and Prediction, and added diversity from Dist {2, 3, 4} benefits finetuned performance for Time Contrastive model compared to Dist 1.



(a) Reconstruction (b) Prediction (c) Time Contrastive

Figure 13: Additional results for the Visual Features distribution shift (complements Fig. 4(b)), with similar trend as in previous shifts.

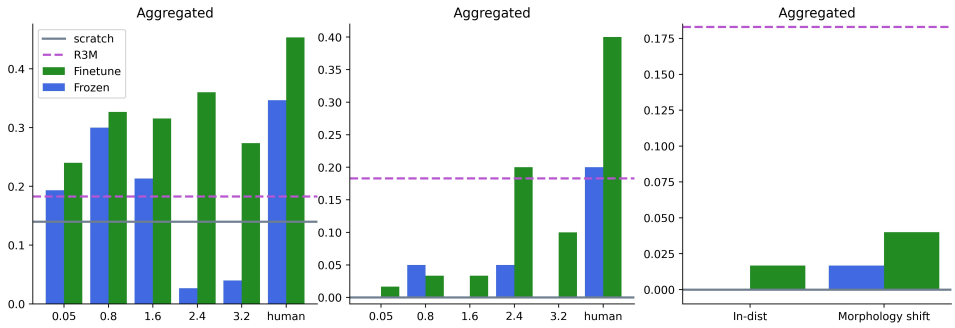
recall that the `Dist {2, 3, 4}` dataset splits contain *randomized* camera configurations while the in-distribution `Dist 1` split contains a single, fixed camera. It is our hypothesis that the time contrastive pretraining approach benefits heavily from this added diversity; indeed, it seems that despite the distribution shift, these methods perform better given more diverse, high coverage data.

Finally, we analyze the Visual Features distribution shift, and plot the aggregated performance across the three sets of methods – reconstruction, forward prediction, and time contrastive – in Fig. 13. We note an identical trend to both the Task Mismatch and Camera Configuration distribution shifts, where representations learned with the reconstruction objective perform similarly to those learned with the forward prediction objective. Similarly, both these approaches outperform the representations learned via time contrastive learning. The added diversity contained in the Visual Features (illustrated in Fig. 6) proves beneficial for time constrative learning, as we see a 16% absolute boost in success rate when training on the distribution shift data, vs. the in-distribution split. This is again consistent with our running hypothesis: time constrastive learning approaches benefit heavily from diverse datasets, and perform much worse when the dataset is restricted, and unimodal.

B.2 Behavior and Morphology Shift

In this section, we include additional results on our remaining two distribution shifts – Suboptimal Behaviors, and Morphology. Unlike the prior shifts, we evaluated these shifts on the richer RoboMimic environments. For these shifts, based on the results above, we only evaluate representations learned with the reconstruction objective, as reconstruction behaves very similarly to forward prediction objectives across the board, and consistently outperforms representations learned via time contrastive learning by a large margin.

For the Suboptimal Behaviors distribution shift, we additional plot downstream task success rate after pretraining our representation on *human teleoperation data* in Fig. 14(a, b). To obtain this human teleoperation data, we use the RoboMimic "multi-human" dataset [39], which is collected by 6 unique human teleoperators, each with a different amount of robot teleoperation proficiency.



(a) Behavior shift, Lift env (b) Behavior shift, Can env (c) Morphology shift, Can env

Figure 14: Additional results for the Behavior and Morphology distribution shift (complements Fig. 4(c), (d)). We note that pretraining on human teleoperated data greatly improves performance, while RoboMimic "Can" task is challenging to solve across the board.

Due to this diversity in proficiencies, and by the natural diversity of preferences across humans when teleoperating a robot to perform a task, this dataset contains *multiple modes* of behavior, unlike the synthetic generated datasets we create that just inject i.i.d. Gaussian noise at each timestep (to get a base level of difference across trajectories). For fair comparison, we subsample the "multi-human" so it is of the same size (200 episodes) as the rest of our pretraining datasets we use for all our other experiments. We observe that pretraining on human-teleoperated data outperforms all levels of injected Gaussian noise by as much as 15%, for both Lift and Can environments. This confirms our core takeaway that training on diverse behaviors (with different amounts of suboptimality), leads to flexible representations that can exceed the performance of pretraining directly on the target distribution.

Finally, we plot additional results for the Morphology distribution shift for the RoboMimic "Can" environment in Fig. 14(c). Despite outperforming training from scratch, which obtains 0% success rate, the performance is still low in general, this is because this is a long-horizon, difficult task with various "narrow" bottlenecks (grasping the can from the right angle to establish a solid grasp, and move towards the correct bin).