Efficient Adaptation for End-to-End Vision-Based Robotic Manipulation

Ryan Julian [1 2] Benjamin Swanson [1] Gaurav S. Sukhatme [2] Sergey Levine [1 3] Chelsea Finn [1 4] Karol Hausman [1]

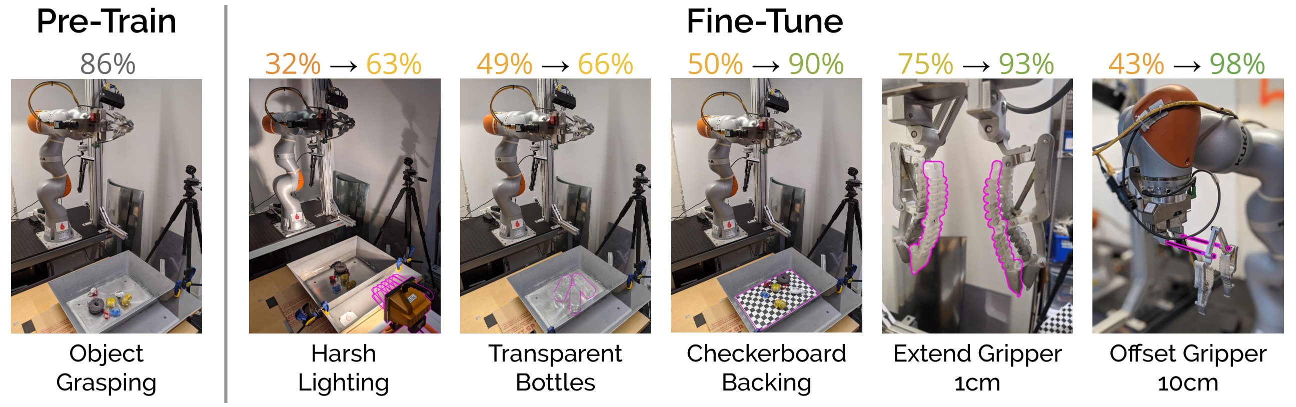

Figure 1. Original robot configuration used for pre-training (left), and adaptation challenges (highlighted in pink) studied in this work (right) with associated performance improvements (top) obtained using our fine-tuning method.

Abstract

One of the great promises of robot learning systems is that they will be able to learn from their mistakes and continuously adapt to ever-changing environments. Despite this potential, most of the robot learning systems today are deployed as a fixed policy and they are not being adapted after their deployment. We present empirical evidence towards a robot learning framework that facilitates continuous adaption. We demonstrate how to adapt vision-based robotic manipulation policies to new variations by fine-tuning via off-policy reinforcement learning. This adaptation uses less than 0.2% of the data necessary to learn the task from scratch. We find that pre-training via RL is essential: training from scratch or adapting from supervised ImageNet features are both unsuccessful with such small amounts of data. We also find that these positive results hold in a limited continual learning setting Our empirical conclusions are consistently supported by experiments on simulated manipulation tasks, and by 52 unique fine-tuning experiments on a real robotic grasping system pre-trained on 580,000 grasps. For video results, see the project website at https://ryanjulian.me/continual-fine-tuning.

1. Introduction

The ability to constantly learn, adapt, and evolve is arguably one of the most important properties of an intelligent agent prepared to exist in the real world. Similarly, our robots should be able to continuously learn and adapt throughout their lifetime to the ever-changing environments that they are deployed in. This is a widely recognized requirement. In fact, there is an entire academic sub-field of lifelong learning (Thrun, 1998) that is interested in the problem of agents that never stop learning. Despite the wide interest in this ability, most of the intelligent agents deployed today are not tested for their adaptation capabilities. Even though techniques such as reinforcement learning theoretically provide the ability to perpetually learn from trial and error, this is not how they are typically evaluated. Instead, the predominant method of acquiring a new task with reinforcement learning is to initialize a policy from scratch, collect entirely new data in a stationary environment, and evaluate a static policy that was trained with this data.

[*]Equal contribution [1]Google Research, Robotics at Google Team, Mountain View, California, USA [2]Department of Computer Science, University of Southern California, Los Angeles, California, USA [3]Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, USA [4]Department of Computer Science, Stanford University, California, USA. Correspondence to: Ryan Julian <ryanjulian@gmail.com>.

Proceedings of the 37^{th} International Conference on Machine Learning, Vienna, Austria, PMLR 119, 2020. Copyright 2020 by the author(s).

This static paradigm does not evaluate the robot's capability to adapt. It also traps robotic reinforcement learning in the worst-case regime for sample efficiency: the cost to acquire a new task is dominated by sample efficiency of the learning algorithm and the complexity of the task, as reflected in cost of acquiring diverse task data starting from naïve (e.g. random) exploration.

Most machine learning models successfully deployed in the real world, such as those used for computer vision and natural language processing (NLP) do not live in this regime. For instance, the predominant method of acquiring a new computer vision task is to start learning the new task with a pre-trained model for a related task, acquired from a pre-collected data set, and *fine-tune* that model to achieve the new task (Donahue et al., 2014; Howard & Ruder, 2018b; Devlin et al., 2018). This changes the sample efficiency regime of the learning process from one which is dominated by *task complexity* to one that is dominated by *task novelty*, i.e. the difference between the new task and the task on which the model was pre-trained. While a number of works have studied how to use pre-trained ImageNet (Deng et al., 2009) features for robotics (Yosinski et al., 2014; Huh et al., 2016; Kornblith et al., 2019), there are remarkably few works that study how to adapt motor skills themselves. Our work attempts to bridge this gap.

We adapt an image-based grasping policy to changes in background, object shape and appearance, lighting conditions, and robot morphology and kinematics, while using less than 0.2% of the data necessary to learn the same task from scratch (see Fig. 1). Our results, supported by simulation and extensive real-world experiments, indicate that a pre-adaptation policy acquired for a task using reinforcement learning can be used to acquire policies for nearby tasks using very little new data and a simple update procedure. Furthermore, we find that this approach of adapting pre-trained policies with off-policy reinforcement learning (RL) leads to substantial improvements over the course of fine-tuning, and that pre-training via RL is essential: it significantly outperforms conventional pre-training techniques using supervised learning on task-agnostic datasets. We believe this simple adaptation scheme provides a promising solution for creating a lifelong learning robotic agent, and show this potential using a simple continual learning experiment.

To our knowledge, this work is the first to demonstrate that simple fine-tuning of off-policy reinforcement learning can successfully adapt to substantial task, robot, and environment variations which were not present in the original training distribution (*i.e.* off-distribution).

2. Related Work

We consider how we might transfer knowledge for efficient learning in new conditions (Taylor & Stone, 2009; Pan & Yang, 2009; Tan et al., 2018a), a widely-studied problem particularly outside of the robotics domain (Donahue et al., 2014; Howard & Ruder, 2018b; Devlin et al., 2018; Dai et al., 2007; Raina et al., 2007). Prior works in robotics have considered how we might transfer information from models trained with supervised learning on ImageNet (Deng et al., 2009) by fine-tuning (Levine et al., 2016; Finn et al., 2016; Gupta et al., 2018; Pinto & Gupta, 2016) or other means (Sermanet et al., 2017; Hazara & Kyrki, 2019). Our experiments show that transfer from pre-trained conditions is significantly more successful than transfer from ImageNet. Other works have leveraged experience in simulation (Sadeghi & Levine, 2017; Tobin et al., 2017; Sadeghi et al., 2018; Tan et al., 2018b; OpenAI et al., 2019; Rusu et al., 2016; Peng et al., 2018; Higuera et al., 2017; Hämäläinen et al., 2019) or representations learned with auxiliary losses (Riedmiller et al., 2018; Mirowski et al., 2016; Sax et al., 2019) for effective transfer. While successful, these approaches either require significant engineering effort to construct an appropriate simulation or significant supervision. Most relevantly, recent work in model-based RL has used predictive models for fast transfer to new experimental set-ups (Chatzilygeroudis & Mouret, 2018; Ha & Schmidhuber, 2018), *i.e.* by fine-tuning predictive models (Dasari et al., 2019), via online search of a pre-learned representation of the space models, policies, or high-level skills (Chatzilygeroudis et al., 2018; Cully et al., 2015; Kaushik et al., 2020; Merel et al., 2019), or by learning physics simulation parameters from real data (Rastogi et al.; Jeong et al., 2019). We show how fine-tuning is successful with a model-free RL approach, and show how a state-of-the-art grasping system can be adapted to new conditions.

3. A Very Simple Fine-Tuning Method

We define then evaluate a simple technique for offline fine-tuning.

Our experiments model an "on the job" adaptation scenario, where a robot is initially trained to perform a general task (in our case, grasping diverse objects), and then the conditions of the task change in a drastic and substantial way as the robot performs the task, *e.g.* through the introduction of significantly brighter lighting, or a peculiar and unexpected type of object. The robot must adapt to this change quickly in order to recover a proficient policy. Handling these changes reflects what we expect to be a common requirement of reinforcement learning policies deployed in the real world: since an RL policy can learn from all of the experience that it has collected, there is no need to sepa-

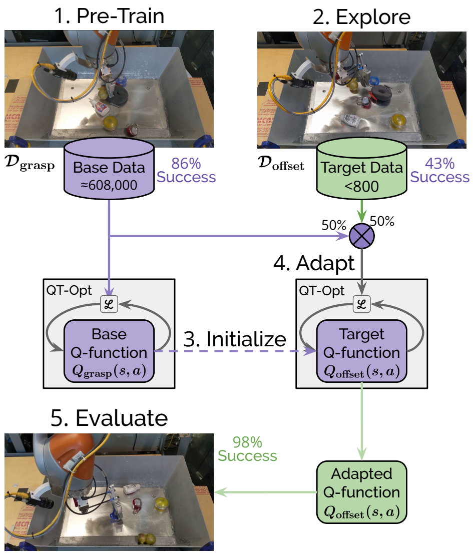
1. Pre-Train
2. Explore
$\mathcal{D}_{\text{grasp}}$
Base Data ≈608,000
86% Success
$\mathcal{D}_{\text{offset}}$
Target Data <800
43% Success
50%
50%
4. Adapt
QT-Opt
$\mathcal{L}$
Base Q-function $Q_{\text{grasp}}(s,a)$
3. Initialize
QT-Opt
$\mathcal{L}$
Target Q-function $Q_{\text{offset}}(s,a)$
5. Evaluate
98% Success
Adapted Q-function $Q_{\text{offset}}(s,a)$

Figure 2. Schematic of the simple method we test in Section 3, using the conceptual framework we discuss in Appendix B.1. We pre-train a policy using the old data from the pre-training task, which is then adapted using the new data from the fine-tuning task.

rate learning into clearly distinct training and deployment phases. Instead, it is likely desirable to allow the policy to simply continue learning "on the job" so as to adapt to these changes.

3.1. The Method

We define a very simple fine-tuning procedure for off-policy RL, as follows (Fig. 2).

First, we (1) pre-train a general grasping policy, as describe in Appendix A.1 and (Kalashnikov et al., 2018). To fine-tune a policy onto a new target task, we (2) use the pre-trained policy to collect an exploration dataset of attempts on the target task; then (3) initialize the same off-policy reinforcement learning algorithm which was used for pre-training (QT-Opt, in our case) with the parameters of the pre-trained policy, and both the target task and base task datasets[1] as the data sources (*e.g.* replay buffers); we then (4) update the policy with this training algorithm, using a reduced learning rate, and sampling training examples with equal probability from the base and target task datasets, for some number of update steps. Finally, we (5) evaluate the fine-tuned policy on the target task.

Our method is offline, *i.e.* it uses a single dataset of target task attempts, and requires no robot interaction after initial dataset collection to compute a fine-tuned policy, which may then be deployed onto a robot.

3.2. Evaluating offline fine-tuning for real-world grasping

We now turn our attention to how to evaluating this simple method's effectiveness as an adaptation procedure for end-to-end robot learning, and perhaps continual learning. Our goal is to determine whether the method is sample efficient, whether it works over a broad range of possible variations, and to determine whether it performs better than simpler ways of acquiring the target tasks.

With this goal in mind, we conduct a large panel of ablation experiments experiments on a real 7 DoF Kuka arm. These experiments evaluate the performance of our method across the diverse range of Challenge Tasks (See Appendix A) and a continuum of target task dataset sizes, and compare this performance to two comparison methods.

The experiments are very challenging. The Transparent Bottles task in particular presents a major challenge to most grasping systems: the transparent bottles generally confuse depth-based sensors and, especially in cluttered bins, require the robot to singulate individual items and position the gripper in the right orientation for grasping. Although our base policy uses only RGB images, it is still not able to grasp the glass bottles reliably, because they differ so much from the objects it observed during training. However, after fine-tuning with only 1 hour (100 grasp attempts) of experience, we observe that the transparent bottles can be picked up with a success rate of 66%, 20% better than the base policy. Figure 4 shows how the robot's view changes for each challenge task. Note the extreme glare and robot reflections visible in images from the Harsh Lighting challenge.

For videos of our experimental results, see the project website.[2]

Collect datasets First, we collect a dataset of 800 grasp attempts for each of our 5 challenge tasks (see Table 3) plus the base grasping task. We then partitioned each dataset into 6 tiers of difficulty by number of exploration grasps (25, 50, 100, 200, 400, and 800 grasp attempts), yielding 36 individual datasets.

Train fine-tuned policies We train a fine-tuned policy for each of these 36 datasets using the procedure described above. We execute the fine-tuning algorithm for 500,000 gradient steps (see Appendix C for more information on how we chose this number) and use a learning rate of 10^{-4}, which is 25% of learning rate used for pre-training. This yields 36 fine-tuned policies, each trained with a different combination of target task and target dataset size. This set of 36 policies includes 6 policies fine-tuned on data from

[1] We assume this dataset was saved during training of the base policy

[2] For video results, see `https://ryanjulian.me/continual-fine-tuning`

Challenge Task	Original Policy	Ours (exploration grasps)							Comparisons	
		25	50	100	200	400	800	**Best (Δ)**	Scratch	ImageNet
Checkerboard Backing	50%	67%	48%	71%	47%	89%	90%	**90% (+40)**	0%	0%
Harsh Lighting	32%	23%	16%	52%	44%	58%	63%	**63% (+31)**	4%	2%
Extend Gripper 1 cm	75%	93%	67%	80%	51%	90%	69%	**93% (+18)**	0%	14%
Offset Gripper 10 cm	43%	73%	50%	60%	56%	91%	98%	**98% (+55)**	37%	47%
Transparent Bottles	49%	46%	43%	65%	65%	58%	66%	**66% (+17)**	27%	20%
Baseline Grasping Task	86%	98%	81%	84%	78%	93%	89%	**98% (+12)**	0%	12%

Table 1. Summary of grasping success rates ($N \geq 50$) for the experiments by challenge task, fine-tuning method, and number of exploration grasps. The experiments “Scratch” and “ResNet 50 + ImageNet” both use 800 exploration grasps and the same update process as the other experiments. “Scratch” starts the grasping network with randomly-initialized parameters. “ResNet 50 + ImageNet” refers to training a grasping network with an equivalent architecture to the other experiments, but with its convolutional layers replaced with a ResNet 50 architecture and pre-loaded with ImageNet features; the non-CNN parts of the network (MLPs for the action inputs and the Q-value output) are randomly-initialized.

the base grasping task, for validation.

Train comparisons To provide points of comparison, we train two additional policies for each challenge task and the base grasping task, yielding 12 additional policies.

The first comparison (“Scratch”) is a policy trained using the aforementioned fine-tuning procedure and an 800-grasp data set, but using a randomly-initialized Q-function rather than the Q-function obtained from pre-training. The purpose of this comparison is to help us assess the contribution of the pre-trained parameters to the fine-tuning process’ performance.

The second comparison (“ImageNet”) is also trained using an identical fine-tuning procedure and the 800-grasp dataset, but uses a modified Q-function architecture in which we replace the convolutional trunk of the network with that of the popular ResNet50 architecture (He et al., 2016), initialized with the weights obtained by training the network to classify images from the ImageNet dataset (Deng et al., 2009). Refer to to Fig. 9 for a diagram of the unmodified architecture. We initialize the remaining fully-connected layers with random parameters, and concatenate the action input features at the end of the CNN (rather than the adding them in middle of the CNN, as in the original architecture). Note that in this comparison, the fine-tuning process still updates all parameters, including those of the ResNet50 sub-network. The purpose of this comparison is to provide a comparison to a strong alternative to end-to-end RL for obtaining pre-training parameters.

Evaluate performance Finally, we evaluate all 48 policies on their target task by deploying them to the robot and executing 50 or more grasp attempts to calculate the policy’s final performance. To reduce the variance of our evaluation statistics, we shuffle the contents of the bin between each trial by executing a randomly-generated sequence of sweeping movements with the end-effector.

The full experiment required more than 15,000 grasp attempts and 14 days of real robot time, and was conducted over approximately one month.

We present a full summary of our results in Table 1. Across the board, we observe substantial benefits arising from fine-tuning, suggesting that the robot can indeed adapt to drastically new condition with a modest amount of data: our most data-intensive experiment uses just 0.2% of the data used train the base grasping policy to similar performance. Our method consistently outperforms both the “ImageNet” and “Scratch” comparison methods. We provide more detailed analysis of this experiment in the next section.

The experiments are very challenging. For example, the “Transparent Bottles” task presents a major challenge to most grasping systems: the transparent bottles generally confuse depth-based sensors and, especially in cluttered bins, require the robot to singulate individual items and position the gripper in the right orientation for grasping. Although our base policy uses only RGB images, it is still not able to grasp the transparent bottles reliably, because they differ so much from the objects it observed during training. However, after fine-tuning with only 1 hour (100 grasp attempts) of experience, we observe that the transparent bottles can be picked up with a success rate of 66%, 20% better than the base policy. Similarly, the “Checkerboard Backing” challenge task asks the robot to differentiate edges associated with real objects from edges on an adversarial checkerboard pattern. It never needed this capability to succeed during pre-training, where the background is always featureless and grey, and all edges can be assumed to be associated with a graspable object. After 1 hour (100 grasp attempts) of experience, using our method the robot can grasp objects on the checkerboard background with a 71% success rate, 21% better than the base policy, and this success rate reaches 90% after 8 hours of experience (800 grasp attempts).

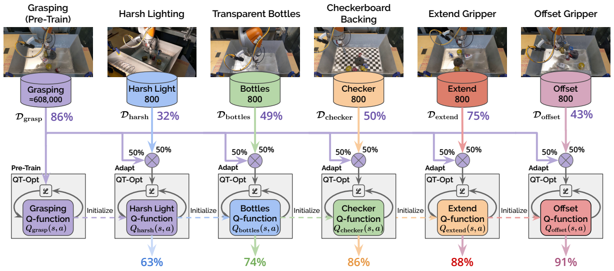
Grasping (Pre-Train)
Harsh Lighting
Transparent Bottles
Checkerboard Backing
Extend Gripper
Offset Gripper
Grasping ≈608,000
Harsh Light 800
Bottles 800
Checker 800
Extend 800
Offset 800
$\mathcal{D}_{\text{grasp}}$ 86%
$\mathcal{D}_{\text{harsh}}$ 32%
$\mathcal{D}_{\text{bottles}}$ 49%
$\mathcal{D}_{\text{checker}}$ 50%
$\mathcal{D}_{\text{extend}}$ 75%
$\mathcal{D}_{\text{offset}}$ 43%
50%
50%
Pre-Train
Adapt
QT-Opt
$\mathcal{L}$
Grasping Q-function $Q_{\text{grasp}}(s,a)$
Harsh Light Q-function $Q_{\text{harsh}}(s,a)$
Bottles Q-function $Q_{\text{bottles}}(s,a)$
Checker Q-function $Q_{\text{checker}}(s,a)$
Extend Q-function $Q_{\text{extend}}(s,a)$
Offset Q-function $Q_{\text{offset}}(s,a)$
Initialize
63%
74%
86%
88%
91%

Figure 3. Flow chart of the continual learning experiment, in which we fine-tune on a sequence of conditions. Every transition to a new scenario happens after 800 grasps.

3.3. Evaluating Offline Fine-Tuning for Continual Learning

Now that we have defined and evaluated a simple method for offline fine-tuning, we evaluate its suitability for use in continual learning, which could allow us to achieve the goal of an robot which adapts to ever-changing environments and tasks. To do so, we define a simple continual learning challenge as follows (Fig. 3).

As in the fine-tuning experiments, we begin with a base policy pre-trained for general object grasping. Likewise, we also use our fine-tuning method to adapt the base policy to a target task, in this case "Harsh Lighting." Not content to stop there, we use *this* adapted policy—*not* the base policy—as the initialization for another iteration of our fine-tuning algorithm, this time targeting "Transparent Bottles." We repeat this process until we have run out of new tasks, ending at the task "Offset Gripper 10cm," at which point we evaluate the policy on the last task.

We perform this experiment using 800 exploration-grasp datasets for each Challenge Task from our ablation study of online fine-tuning with real robots. We summarize the results in Table 2. Note that because it is the first step of the continual learning experiment, the policy for "Harsh Lighting" is identical to that of the 800-grasp variant of the single-step experiment.

Recall that our goal for this experiment is to determine whether continual fine-tuning incurs a significant performance penalty compared to the single-step variant, because we are interested in using this method as a building block for continual learning algorithms. We find that continual fine-tuning does not impose a drastic performance penalty compared to single-step fine-tuning. The continual fine-tuning policies for the "Checkerboard Backing," "Extend Gripper 1 cm," and "Offset Gripper 10 cm," challenges succeeded in grasping between 4% and 7% less often than their single-step fine-tuning counterparts, whereas the policy for the challenging "Transparent Bottles" case actually succeeded 8% more often. These small deltas are within the margin-of-error of our evaluation procedure, so we conclude that the effect of continual fine-tuning on the performance compared to single-step fine-tuning is very small. This experiment demonstrates that our method can perform continual adaptation, and may serve as the basis for a continual end-to-end robot learning method.

Challenge Task	Continual Learning	Δ	
		Base	Single
Harsh Lighting	63%	+32%	-
Transparent Bottles	74%	+25%	+8%
Checkerboard Backing	86%	+36%	−4%
Extend Gripper 1 cm	88%	+12%	−5%
Offset Gripper 10 cm	91%	+44%	−7%

Table 2. Summary of grasping success rates ($N \geq 50$) for the continual learning experiment by challenge task, and comparison to single-step fine-tuning. "Base" refers to the baseline grasping policy before fine-tuning, and "Single" refers to the best performance from the single-step fine-tuning experiment in Table 1. Note that because it is the first step of the continual learning experiment, the policy for "Harsh Lighting" is identical to that of the 800-grasp variant of the single-step experiment.

4. Conclusion

For robots to be able to operate in unconstrained environments, they must be able to continuously adapt to new situations. Our large-scale study shows that combining off-policy RL with a very simple fine-tuning procedure is an effective adaptation method, and this method is capable of achieving remarkable improvements in robot performance on new tasks with very little new data. Furthermore, our continual learning experiment shows that using this simple method in a continual setting imposes very little performance penalty compared to the single-step setting. This suggests that the combination of off-policy RL and fine-tuning can serve as a building block for future continual learning methods. Our results comparing supervised-learning-based initialization to those acquired with our RL-fine-tuning approach highlight a familiar truism about robotics: that robotic agents must do more than perceive the world, they must also act in it. The ability to learn the combination of these two capabilities is what makes RL well-suited for creating continually-learning robots.

Acknowledgements

The authors thank Noah Brown and Ivonne Fajardo for their superb and unyielding support with real robot experiments. We also thank Alex Irpan and Eric Jang for their help with robot learning software, Yevgen Chebotar for his advice on early revisions of this work and always-insightful discussions, Dmitry Kalashnikov and Jake Varley for their help with QT-Opt, and K.R. Zentner for her help with editing and artwork for this paper.

References

Chatzilygeroudis, K. and Mouret, J.-B. Using parameterized black-box priors to scale up model-based policy search for robotics. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–9. IEEE, 2018.

Chatzilygeroudis, K., Vassiliades, V., and Mouret, J.-B. Reset-free trial-and-error learning for robot damage recovery. *Robotics and Autonomous Systems*, 100:236–250, 2018.

Cully, A., Clune, J., Tarapore, D., and Mouret, J.-B. Robots that can adapt like animals. *Nature*, 521(7553):503–507, 2015.

Dai, W., Yang, Q., Xue, G.-R., and Yu, Y. Boosting for transfer learning. In *Proceedings of the 24th international conference on Machine learning*, pp. 193–200, 2007.

Dasari, S., Ebert, F., Tian, S., Nair, S., Bucher, B., Schmeckpeper, K., Singh, S., Levine, S., and Finn, C. Robonet: Large-scale multi-robot learning, 2019.

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

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pp. 647–655, 2014.

Finn, C., Tan, X. Y., Duan, Y., Darrell, T., Levine, S., and Abbeel, P. Deep spatial autoencoders for visuomotor learning. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 512–519. IEEE, 2016.

Gupta, A., Murali, A., Gandhi, D., and Pinto, L. Robot learning in homes: Improving generalization and reducing dataset bias, 2018.

Ha, D. and Schmidhuber, J. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*, pp. 2450–2462, 2018.

Hämäläinen, A., Arndt, K., Ghadirzadeh, A., and Kyrki, V. Affordance learning for end-to-end visuomotor robot control. *arXiv preprint arXiv:1903.04053*, 2019.

Hazara, M. and Kyrki, V. Transferring generalizable motor primitives from simulation to real world. *IEEE Robotics and Automation Letters*, 4(2):2172–2179, 2019.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Higuera, J. C. G., Meger, D., and Dudek, G. Adapting learned robotics behaviours through policy adjustment. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5837–5843. IEEE, 2017.

Howard, J. and Ruder, S. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*, 2018a.

Howard, J. and Ruder, S. Universal language model fine-tuning for text classification, 2018b.

Huh, M., Agrawal, P., and Efros, A. A. What makes imagenet good for transfer learning? *arXiv preprint arXiv:1608.08614*, 2016.

Irpan, A., Rao, K., Bousmalis, K., Harris, C., Ibarz, J., and Levine, S. Off-policy evaluation via off-policy classification. In *Advances in Neural Information Processing Systems*, pp. 5438–5449, 2019.

Jeong, R., Kay, J., Romano, F., Lampe, T., Rothorl, T., Abdolmaleki, A., Erez, T., Tassa, Y., and Nori, F. Modelling generalized forces with reinforcement learning for sim-to-real transfer. *arXiv preprint arXiv:1910.09471*, 2019.

Kalashnikov, D., Irpan, A., Pastor, P., Ibarz, J., Herzog, A., Jang, E., Quillen, D., Holly, E., Kalakrishnan, M., Vanhoucke, V., et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on Robot Learning*, pp. 651–673, 2018.

Kaushik, R., Desreumaux, P., and Mouret, J.-B. Adaptive prior selection for repertoire-based online adaptation in robotics. *Frontiers in Robotics and AI*, 6:151, 2020.

Kornblith, S., Shlens, J., and Le, Q. V. Do better imagenet models transfer better? In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2661–2671, 2019.

Levine, S., Finn, C., Darrell, T., and Abbeel, P. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

Merel, J., Tunyasuvunakool, S., Ahuja, A., Tassa, Y., Hasenclever, L., Pham, V., Erez, T., Wayne, G., and Heess, N. Reusable neural skill embeddings for vision-guided whole body movement and object manipulation. *arXiv preprint arXiv:1911.06636*, 2019.

Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A. J., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., et al. Learning to navigate in complex environments. *arXiv preprint arXiv:1611.03673*, 2016.

OpenAI, Akkaya, I., Andrychowicz, M., Chociej, M., Litwin, M., McGrew, B., Petron, A., Paino, A., Plappert, M., Powell, G., Ribas, R., Schneider, J., Tezak, N., Tworek, J., Welinder, P., Weng, L., Yuan, Q., Zaremba, W., and Zhang, L. Solving rubik's cube with a robot hand, 2019.

Pan, S. J. and Yang, Q. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10): 1345–1359, 2009.

Peng, X. B., Andrychowicz, M., Zaremba, W., and Abbeel, P. Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 1–8. IEEE, 2018.

Pinto, L. and Gupta, A. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *2016 IEEE international conference on robotics and automation (ICRA)*, pp. 3406–3413. IEEE, 2016.

Raina, R., Battle, A., Lee, H., Packer, B., and Ng, A. Y. Self-taught learning: transfer learning from unlabeled data. In *Proceedings of the 24th international conference on Machine learning*, pp. 759–766, 2007.

Rastogi, D., Koryakovskiy, I., and Kober, J. Sample-efficient reinforcement learning via difference models.

Riedmiller, M. A., Hafner, R., Lampe, T., Neunert, M., Degrave, J., de Wiele, T. V., Mnih, V., Heess, N. M. O., and Springenberg, J. T. Learning by playing solving sparse reward tasks from scratch. In *ICML*, 2018.

Rusu, A. A., Vecerík, M., Rothörl, T., Heess, N. M. O., Pascanu, R., and Hadsell, R. Sim-to-real robot learning from pixels with progressive nets. In *CoRL*, 2016.

Sadeghi, F. and Levine, S. Cad2rl: Real single-image flight without a single real image. *Robotics: Science and Systems XIII*, Jul 2017. doi: 10.15607/rss.2017.xiii.034. URL http://dx.doi.org/10.15607/RSS.2017.XIII.034.

Sadeghi, F., Toshev, A., Jang, E., and Levine, S. Sim2real viewpoint invariant visual servoing by recurrent control. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 4691–4699, 2018. doi: 10.1109/CVPR.2018.00493. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Sadeghi_Sim2Real_Viewpoint_Invariant_CVPR_2018_paper.html.

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. In *Conference on Robot Learning*, 2019.

Sermanet, P., Xu, K., and Levine, S. Unsupervised perceptual rewards for imitation learning. *Proceedings of Robotics: Science and Systems (RSS)*, 2017. URL http://arxiv.org/abs/1612.06699.

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., and Liu, C. A survey on deep transfer learning. *Lecture Notes in Computer Science*, pp. 270–279, 2018a. ISSN 1611-3349. doi: 10.1007/978-3-030-01424-7_27. URL http://dx.doi.org/10.1007/978-3-030-01424-7_27.

Tan, J., Zhang, T., Coumans, E., Iscen, A., Bai, Y., Hafner, D., Bohez, S., and Vanhoucke, V. Sim-to-real: Learning agile locomotion for quadruped robots. *Robotics: Science and Systems XIV*, Jun 2018b. doi: 10.15607/rss.2018.xiv.010. URL http://dx.doi.org/10.15607/RSS.2018.XIV.010.

Taylor, M. E. and Stone, P. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(Jul):1633–1685, 2009.

Thrun, S. *Lifelong Learning Algorithms*, pp. 181–209. Kluwer Academic Publishers, USA, 1998. ISBN 0792380479.

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)*, Sep 2017. doi: 10.1109/iros.2017.8202133. URL http://dx.doi.org/10.1109/IROS.2017.8202133.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pp. 3320–3328, 2014.