Learning Deformable Manipulation from Expert Demonstrations

Gautam Salhotra*, I-Chun Arthur Liu*, Marcus Dominguez-Kuhne, & Gaurav S. Sukhatme‡
University of Southern California

Abstract— **We present a novel Learning from Demonstration (LfD) method, Deformable Manipulation from Demonstrations (DMfD), to solve deformable manipulation tasks using states or images as inputs, given expert demonstrations. Our method uses demonstrations in three different ways, and balances the trade-off between exploring the environment online and using guidance from experts to explore high dimensional spaces effectively. We test DMfD on a set of representative manipulation tasks for a 1-dimensional rope and a 2-dimensional cloth from the SoftGym suite of tasks, each with state and image observations. Our method exceeds baseline performance by up to 12.9% for state-based tasks and up to 33.44% on image-based tasks, with comparable or better robustness to randomness. Also, we create two challenging environments for folding a 2D cloth using image-based observations, and set a performance benchmark for them. We deploy DMfD on a real robot (sim2real gap $\sim$6%).**

I. Introduction

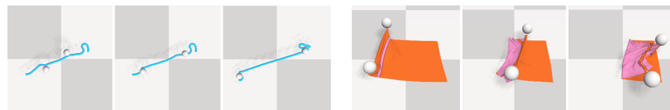
Autonomous dexterous robotic manipulation is challenging. For rigid objects, challenges include estimating pose and mass distribution, grasp prediction, and real world grasp planning. Obtaining the state and dynamics for deformable objects is much harder than for rigid objects. Even with 'full' state information, deformable manipulation is very high dimensional, making it more challenging than rigid manipulation [1].

Our method, Deformable Manipulation from Demonstrations (DMfD), is a learned agent for deformable manipulation using expert data three ways. We leverage an advantage-weighted formulation [2], [3] in the loss function, with expert samples (pre-populated in the replay buffer) appropriately weighted to encourage the policy to mimic expert actions. Finally, during experience collection, we use reference state initialization [4], where the agent is reset along an expert trajectory with some probability. We then compare the state trajectories of the expert and agent, helping exploration in difficult to reach states. Fig. 1 shows rollouts of our method for challenging image-based manipulation tasks.

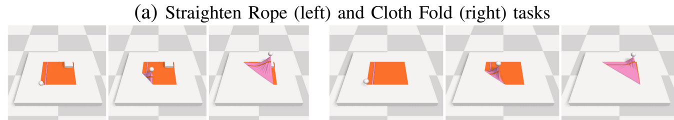
Contributions: We propose a novel method (DMfD) for a learning agent to absorb expert guidance (from human execution or hand-engineered methods), while learning to solve challenging deformable manipulation tasks online.

- DMfD solves deformable manipulation tasks for state and image based observations using expert data in three ways. Our online training loss formulation balances exploring online and mimicking experts.

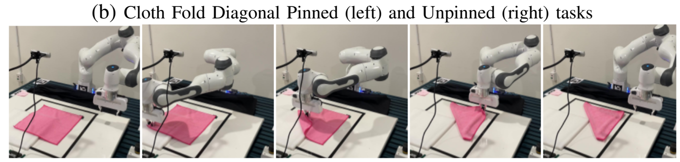
* Equal contribution.
`salhotra,ichunliu,marcusdo,gaurav@usc.edu`
‡ G.S. Sukhatme holds concurrent appointments as a Professor at USC and as an Amazon Scholar. This paper describes work performed at USC and is not associated with Amazon.



(a) Straighten Rope (left) and Cloth Fold (right) tasks



(b) Cloth Fold Diagonal Pinned (left) and Unpinned (right) tasks



(c) Real world execution of Cloth Fold Diagonal Unpinned Task

Fig. 1: **Learning Deformable Manipulation.** DMfD is a learned agent achieving state-of-the-art performance (among methods that use expert demonstrations) for deformable manipulation tasks. We set a new performance benchmark on the Straighten Rope and Cloth Fold tasks (Fig. 1a) from SoftGym [5], both with two end effectors, shown as white spheres. We also introduce tasks for *one* end effector - the Cloth Fold Diagonal tasks, where a square cloth is folded diagonally, with pinned and unpinned (Fig. 1b) varieties. Fig. 1c shows the real world execution of the unpinned variety.

- DMfD outperforms baselines on state- and image-based environments (by up to 12.9% and 33.44% respectively). It outperforms experts it was trained on for some tasks.
- We create two challenging variants of a new folding task and deploy our system on a real world for the unpinned variant.

II. Background

Autonomous deformable manipulation is a challenge with many real-world applications such as folding clothes, cooking, or assisting humans [6]–[9]. Analytical methods such as Finite Element Method [10] and Material-Point Methods[11] are used to model object dynamics. Control methods such as trajectory optimization [12]–[15] and model predictive control [16] are used to manipulate objects. However, they might not generalize to environment variations. Data-driven methods are popular for manipulation tasks [17], including Imitation Learning (IL) [18]–[23], Reinforcement Learning (RL) [24]–[28], and their combination [29]–[32]. However, most successes have been in rigid body manipulation.

Here, we focus on deformable object manipulation using expert-guided RL. Learning from expert Demonstrations (LfD) has been applied to deformable manipulation tasks like bed making [33] and manipulating beads, cloths, and bags [34]. Reinforcement learning has been applied to manipulation of ropes, cloths, and liquids [5], [35], sometimes with vision [7], [36]–[40]. Combining RL with LfD can balance expert guidance with online exploration [29]–[31]. Deep Mimic [4] uses Reference State Initialization (RSI) to initialize from high-value states, mitigating such exploration costs. Advantage

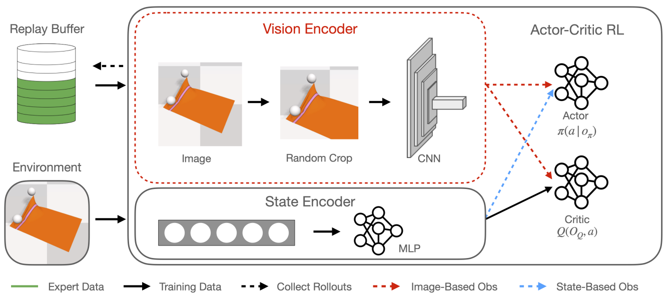


Fig. 2: **Method schematic.** The environment (replay buffer $\mathcal{B}$) gives observations during experience collection (training). Pre-populated expert demonstrations are **Green**. Training works with state-based or image-based observations. For state observations, the actor and critic get a state encoding ($o_Q = o_\pi = o_{\boldsymbol{s}}$), shown as **Black** and **Blue** arrows. For image observations, the actor gets an image encoding and the critic gets image & state encodings ($o_\pi = o_{img}, o_Q = o_{\boldsymbol{s}} \cup o_{img}$), shown as **Black** and **Red** arrows.

Weighted Actor Critic (AWAC) [3] uses implicit policy constraints to learn from experts offline, followed by online finetuning. We show that RL, with intelligent use of expert data, significantly improves deformable manipulation performance.

III. Formulation and Approach

We formulate deformable manipulation as a partially observable Markov decision process (POMDP) with state space $\mathcal{S}$, action space $\mathcal{A}$, observation space $\mathcal{O}$, discount factor γ, horizon H, dynamics $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ and reward $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. The discounted reward at time t is $R_t = \sum_{i=t}^{H} \gamma^i r(\boldsymbol{s}_i, \boldsymbol{a}_i)$ where $\boldsymbol{s}_i \in \mathcal{S}, \boldsymbol{a}_i \in \mathcal{A}$. Additionally, tasks are generalized over variants $\mathcal{V}$ indicating object properties. The initial state is a function of the variant, $\boldsymbol{s}_0(v), v \sim \mathcal{V}$.

The problem is to find the policy $\pi \in \Pi$, that maximises the expected discounted reward $\mathbb{E}_{\tau\sim\pi(\tau),v\sim\mathcal{V}}[R_0]$ of an episode over variants and distribution induced by policy, subject to $\boldsymbol{s}_{t+1} = \mathcal{T}(\boldsymbol{s}_t, \boldsymbol{a}_t)$, and initial state $\boldsymbol{s}_0(v)$. $\pi(\tau)$ is the likelihood of trajectory τ under π and $\boldsymbol{s}_0(v)$. Additionally, we assume expert trajectories $\mathcal{E}$ are readily available through a demonstration dataset.

Our objective is maximizing expected improvement over sampled transitions from a replay buffer $\mathcal{B}$. This formulation is similar to Advantage-Weighted Regression (AWR) [2] with experience replay over a mixture of policies. We create an advantage-weighted objective for a policy with parameters θ,

$$\mathcal{L}_A = \mathop{\mathbb{E}}_{\boldsymbol{s},\boldsymbol{a}\sim\mathcal{B}} \left[\log \pi_\theta(\boldsymbol{a}|\boldsymbol{s}) \exp\left(\frac{1}{\lambda} A^\pi(\boldsymbol{s}, \boldsymbol{a})\right)\right] \tag{1}$$

λ is a temperature parameter (see [2] for a complete derivation). We use the standard critic loss $\mathcal{L}_Q = \mathbb{E}_\mathcal{B}[||q_{\phi,\mathcal{B}} - b||^2]$ to minimize error between the Q-estimate and Bellman update.

A. Deformable Manipulation from Demonstrations (DMfD)

Since state-estimation is hard for deformables, we extend the problem to make the policy act on an observation, $\pi_\theta(\boldsymbol{a}|\boldsymbol{o})$.

DMfD learns both from a pre-populated replay buffer $\mathcal{B}$ (with expert trajectories $\mathcal{E}$) and online environment interaction. Online interaction allows DMfD to find better trajectories than $\mathcal{E}$, enabling it to exceed the expert.

Secondly, we balance exploration with exploitation [41] by requiring the policy to minimize an entropy loss term,

$$\mathcal{L}_E = \mathop{\mathbb{E}}_{\boldsymbol{s},\boldsymbol{a},\boldsymbol{o}\sim\mathcal{B}} [\alpha \log \pi_\theta(\boldsymbol{a}|\boldsymbol{o}) - Q(\boldsymbol{s}, \boldsymbol{a})] \tag{2}$$

Our policy loss is a w_E-weighted combination,

$$\mathcal{L}_\pi = (1 - w_E)\mathcal{L}_A + w_E\mathcal{L}_E, \ 0 \le w_E \le 1 \tag{3}$$

While collecting experience, we reset the robot to an expert's state with probability p_η, and compare the agent's generated trajectory to the expert's, giving an imitation reward. This reference state initialisation (RSI) was introduced in DeepMimic [4] to help explore hard to reach high-dimensional states. Our method uses this idea to help mimic the expert during the initial stages of training.

Our actor and critic networks have hidden layers with $\tanh$ activation. A Convolutional Neural Network (CNN) encoder and random image crops [37] are added for image-based training. Fig. 2 shows these architectures. Note the critic also gets state input in addition to the observation. This privileged information helps stabilize it [42].

IV. Experiments

A. Tasks and Experimental Setup

We test on the tasks below with state or image observations as applicable. Object states are encoded with their object-specific reduced-state. Image observations are 32x32 RGB images showing the object and robot end-effector. Each task has a set of variants, where the deformable object's properties vary for effective domain randomization.

1) **Straighten Rope:** Stretch the rope a fixed distance apart, to straighten it. The reduced state is the (x, y, z) coordinates of 10 equidistant points including rope ends.
2) **Cloth Fold:** Fold a flattened cloth into half, along an edge, using two end-effectors. The reduced state is the (x, y, z) coordinates of each corner.
3) **Cloth Fold Diagonal Pinned (Unpinned):** Fold the square cloth along a specified diagonal, with a single end-effector, and one corner pinned (unpinned). The reduced state is the (x, y, z) coordinates of each corner. These are two new tasks we introduce.

Image-based environments are more difficult to solve than state-based environments. Hence, we focused on image inputs for the two novel Cloth Fold Diagonal tasks. This gives 6 test environments: 4 from SoftGym (state and image inputs for Straighten Rope and Cloth Fold) and 2 new tasks (image inputs for Cloth Fold Diagonal). Demonstrations are hand-coded for variants $v \subseteq \mathcal{V}$, using full state and dynamics.

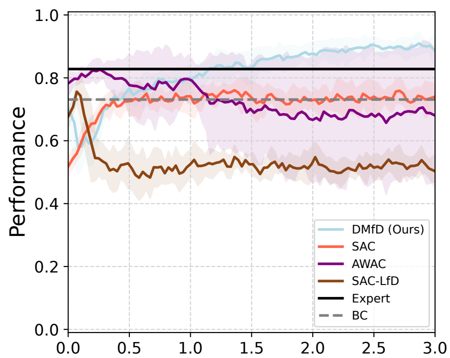
We use normalized performance in $[0, 1]$ from SoftGym,

$$\hat{p}(t) = \frac{p(\boldsymbol{s}_t) - p(\boldsymbol{s}_0)}{p_{opt} - p(\boldsymbol{s}_0)} \tag{4}$$

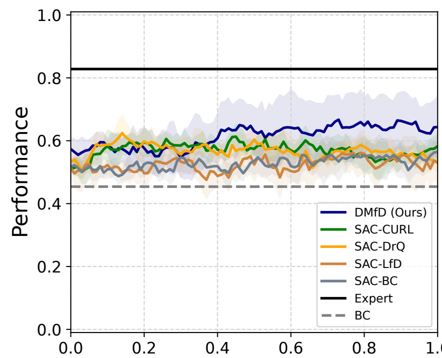
where $p(\boldsymbol{s}_t)$ is the env-specific performance at state $\boldsymbol{s}_t$ at time t, and p_{opt} is the best possible performance. As in SoftGym, we compare performance at the end of the episode, $\hat{p}(H)$.

B. Performance Comparisons

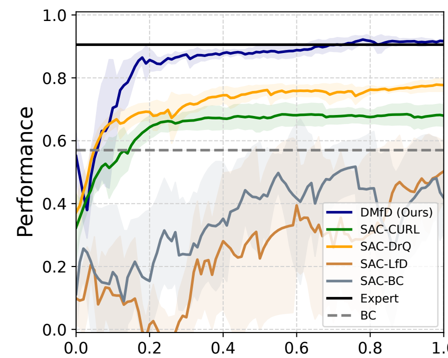
We compare our method with LfD baselines AWAC [3], BC [20], SAC-LfD (SAC with pre-populated expert data in the replay buffer), and SAC-BC (SAC with initialized actor networks from pre-trained BC-Image on expert demonstrations). We also compare with non-LfD baselines SAC, SAC-CURL [36], and DrQ [37].



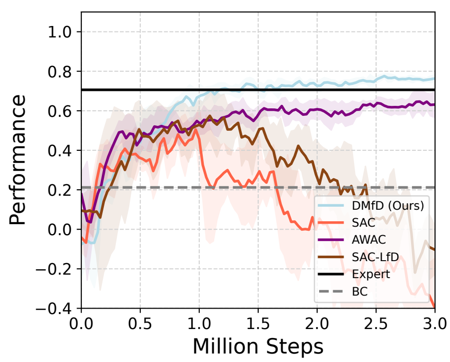
(a) Straighten Rope State



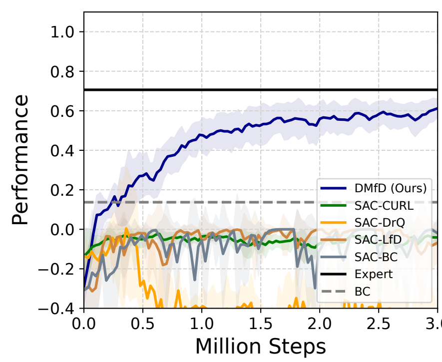
(b) Straighten Rope Image



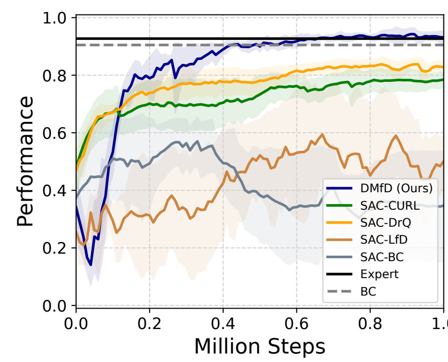
(c) Cloth Fold Diagonal Pinned Image



(d) Cloth Fold State



(e) Cloth Fold Image



(f) Cloth Fold Diagonal Unpinned Image

Fig. 3: **State-of-the-art comparisons.** Learning curves of normalized performance $\hat{p}(H)$ for all environments during training, until convergence. The first column(3a & 3d) shows SoftGym state-based environments. The second column(3b & 3e) show SoftGym image-based environments, and the third column (3c & 3f) show our new Cloth Fold Diagonal environments. State-based DMfD is light blue, image-based agent is dark blue, and expert performance is black. Behavioural Cloning does not train online; its results are shown as constant gray line. The means μ are plotted as solid lines, and one standard deviation ($\mu \pm \sigma$) is the shaded region. We find that DMfD consistently beats the baselines, with comparable or better variance.

Fig. 3 shows the training curves of our method against baselines for all environments. As the environments increase in difficulty, our method outperforms baselines by increased margins for state-base and image-based environments.

C. Discussion

When comparing with experts, Fig. 3 shows that our state-based agents beat the oracle in both state environments. However, the image-based agents are, at best, comparable to the expert, since they do not have privileged state information.

We see the performance gap between DMfD and baselines increase with task difficulty. In a hard task like Cloth Fold Image (Fig. 3e), baselines perform at or below 0 after training.

State-based environments: DMfD's multiple uses of expert data is one main reason why it performs better than SAC, which does not use expert data. This significantly affects performance on hard state-based tasks like Cloth Fold.

Conversely, AWAC can achieve better performance on difficult tasks with expert data. However, no entropy regularization causes AWAC's vulnerability to reach local optimums in training, causing its higher variance than DMfD and lower robustness to randomness. As seen in Fig. 3a, high variance after 1M steps leads to its performance deteriorating.

Image-based environments are harder to solve, and our method outperforms the baselines even further. Our critic is privileged with state data, helping it estimate the value function better. The use of expert data, with the exploration due to entropy regularization, helps our method outperform baselines with comparable or better variance.

Although the baselines have state-of-the-art methods for learning with vision, only the LfD baselines incorporate expert demonstrations, such as BC. In fact, BC outperforms CuRL and DrQ in some environments despite training offline. BC however has drawbacks such as covariate shift and sensitivity to environmental changes. Fig. 3c and Fig. 3f show very different BC performance between the Pinned and Unpinned Cloth Fold Diagonal tasks, even though they are similar. Additionally, BC cannot exceed the expert performance.

Our experiments show DMfD matches or outperforms baselines across environments, and is robust to noise.

V. Conclusion

Deformable Manipulation from Demonstrations (DMfD) is a novel method leveraging expert demonstrations and outperforms state-of-the-art LfD methods for deformable manipulation tasks. We demonstrate the effectiveness of our method on six tasks, including two new challenging cloth folding tasks we created. We show a consistent and significant performance improvement over baselines in state-based environments (up to 12.9%) and an even higher improvement on tougher image-based environments (up to 33.44%). We observe comparable or lower variance than the baselines, indicating higher robustness to noise. Finally, we conducted real robot experiments and achieved a minimal sim2real gap ($\sim$6%),

REFERENCES

[1] V. E. Arriola-Rios, P. Guler, F. Ficuciello, D. Kragic, B. Siciliano, and J. L. Wyatt, "Modeling of deformable objects for robotic manipulation: A tutorial and review," *Frontiers in Robotics and AI*, vol. 7, p. 82, 2020.

[2] X. B. Peng, A. Kumar, G. Zhang, and S. Levine, "Advantage-weighted regression: Simple and scalable off-policy reinforcement learning," *arXiv preprint arXiv:1910.00177*, 2019.

[3] A. Nair, A. Gupta, M. Dalal, and S. Levine, "Awac: Accelerating online reinforcement learning with offline datasets," *arXiv preprint arXiv:2006.09359*, 2020.

[4] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne, "Deepmimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.

[5] J. O. Xingyu Lin, Yufei Wang and D. Held, "Softgym: Benchmarking deep reinforcement learning for deformable object manipulation," *Conference on Robot Learning (CoRL)*, 2020.

[6] R. P. Joshi, N. Koganti, and T. Shibata, "Robotic cloth manipulation for clothing assistance task using dynamic movement primitives," in *Proceedings of the Advances in Robotics*, 2017, pp. 1–6.

[7] Y. Wu, W. Yan, T. Kurutach, L. Pinto, and P. Abbeel, "Learning to manipulate deformable objects without demonstrations," *Robotics Science and Systems (RSS)*, 2019.

[8] Y. Tsurumine, Y. Cui, E. Uchibe, and T. Matsubara, "Deep reinforcement learning with smooth policy update: Application to robotic cloth manipulation," *Robotics and Autonomous Systems*, vol. 112, pp. 72–83, 2019.

[9] S. Kolathaya, W. Guffey, R. W. Sinnet, and A. D. Ames, "Direct collocation for dynamic behaviors with nonprehensile contacts: Application to flipping burgers," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3677–3684, 2018.

[10] P. Wriggers, *Nonlinear finite element methods*. Springer Science & Business Media, 2008.

[11] D. Sulsky, Z. Chen, and H. L. Schreyer, "A particle method for history-dependent materials," *Computer methods in applied mechanics and engineering*, vol. 118, no. 1-2, pp. 179–196, 1994.

[12] E. Heiden, M. Macklin, Y. S. Narang, D. Fox, A. Garg, and F. Ramos, "Disect: A differentiable simulation engine for autonomous robotic cutting," in *Robotics: Science and Systems*, 2021.

[13] S. Zimmermann, R. Poranne, and S. Coros, "Dynamic manipulation of deformable objects with implicit integration," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 4209–4216, 2021.

[14] J. M. Bern, P. Banzet, R. Poranne, and S. Coros, "Trajectory optimization for cable-driven soft robot locomotion." in *Robotics: Science and Systems*, vol. 1, no. 3, 2019.

[15] S. Duenser, J. M. Bern, R. Poranne, and S. Coros, "Interactive robotic manipulation of elastic objects," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 3476–3481.

[16] F. Allgöwer and A. Zheng, *Nonlinear model predictive control*. Birkhäuser, 2012, vol. 26.

[17] T. Y. James A. Preiss, David Millard and G. S. Sukhatme, "Tracking fast trajectories with a deformable object using a learned model," *IIEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[18] G. G. Ross, Stéphane and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," *Proceedings of the fourteenth international conference on artificial intelligence and statistics, JMLR Workshop and Conference Proceedings*, 2018.

[19] M. Laskey, J. Lee, R. Fox, A. Dragan, and K. Goldberg, "Dart: Noise injection for robust imitation learning," in *Conference on robot learning*. PMLR, 2017, pp. 143–156.

[20] G. W. Torabi, Faraz and P. Stone, "Behavioral cloning from observation," *arXiv preprint arXiv:1805.01954)*, 2018.

[21] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," *Advances in neural information processing systems*, 1988.

[22] P. Florence, C. Lynch, A. Zeng, O. A. Ramirez, A. Wahid, L. Downs, A. Wong, J. Lee, I. Mordatch, and J. Tompson, "Implicit behavioral cloning," in *Conference on Robot Learning*. PMLR, 2022, pp. 158–168.

[23] E. S. Ho, Jonathan, "Generative adversarial imitation learning," *Advances in neural information processing*, 2016.

[24] D. Michael, K. Andrey, B. Ashwin, M. Matl, R. M.-M. David Wang, A. Garg, S. Savarese, and G. Ken, "Mechanical search: Multi-step retrieval of a target object occluded by clutter," *International Conference on Robotics and Automation (ICRA).*, 2019.

[25] A. Kurenkov, A. Mandlekar, R. Martin-Martin, S. Savarese, and A. Garg, "Ac-teach: A bayesian actor-critic method for policy learning with an ensemble of suboptimal teachers," in *Conference on Robot Learning*. PMLR, 2020, pp. 717–734.

[26] A. Zeng, S. Song, K.-T. Yu, E. Donlon, F. R. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo *et al.*, "Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 3750–3757.

[27] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser, "Learning synergies between pushing and grasping with self-supervised deep reinforcement learning," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4238–4245.

[28] J. Yamada, Y. Lee, G. Salhotra, K. Pertsch, M. Pflueger, G. S. Sukhatme, J. J. Lim, and P. Englert, "Motion planner augmented reinforcement learning for robot manipulation in obstructed environments," in *Conference on Robot Learning*, 2020.

[29] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, I. Osband *et al.*, "Deep q-learning from demonstrations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[30] D. Horgan, J. Quan, D. Budden, G. Barth-Maron, M. Hessel, H. van Hasselt, and D. Silver, "Distributed prioritized experience replay," in *International Conference on Learning Representations*, 2018.

[31] T. Pohlen, B. Piot, T. Hester, M. G. Azar, D. Horgan, D. Budden, G. Barth-Maron, H. Van Hasselt, J. Quan, M. Večerík *et al.*, "Observe and look further: Achieving consistent performance on atari," *arXiv preprint arXiv:1805.11593*, 2018.

[32] I.-C. A. Liu, S. Uppal, G. S. Sukhatme, J. J. Lim, P. Englert, and Y. Lee, "Distilling motion planner augmented policies into visual control policies for robot manipulation," in *Conference on Robot Learning*, 2021.

[33] M. Laskey, C. Powers, R. Joshi, A. Poursohi, and K. Goldberg, "Learning robust bed making using deep imitation learning with dart," *arXiv e-prints*, pp. arXiv–1711, 2017.

[34] D. Seita, P. Florence, J. Tompson, E. Coumans, V. Sindhwani, K. Goldberg, and A. Zeng, "Learning to rearrange deformable cables, fabrics, and bags with goal-conditioned transporter networks," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 4568–4575.

[35] J. Matas, S. James, and A. J. Davison, "Sim-to-real reinforcement learning for deformable object manipulation," in *Conference on Robot Learning*. PMLR, 2018, pp. 734–743.

[36] A. S. Laskin, Michael and P. Abbeel, "Curl: Contrastive unsupervised representations for reinforcement learning," *International Conference on Machine Learning*, 2020.

[37] D. Yarats, I. Kostrikov, and R. Fergus, "Image augmentation is all you need: Regularizing deep reinforcement learning from pixels," in *International Conference on Learning Representations*, 2020.

[38] X. Lin, Y. Wang, Z. Huang, and D. Held, "Learning visible connectivity dynamics for cloth smoothing," in *Conference on Robot Learning*, 2021.

[39] R. Hoque, D. Seita, A. Balakrishna, A. Ganapathi, A. K. Tanwani, N. Jamali, K. Yamane, S. Iba, and K. Goldberg, "Visuospatial foresight for physical sequential fabric manipulation," *Autonomous Robots*, vol. 46, no. 1, pp. 175–199, 2022.

[40] X. Lin, Z. Huang, Y. Li, J. B. Tenenbaum, D. Held, and C. Gan, "Diffskill: Skill abstraction from differentiable physics for deformable object manipulations with tools," *International Conference on Learning Representation (ICLR)*, 2022.

[41] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International conference on machine learning*. PMLR, 2018, pp. 1861–1870.

[42] L. Pinto, M. Andrychowicz, P. Welinder, W. Zaremba, and P. Abbeel, "Asymmetric actor critic for image-based robot learning," *arXiv e-prints*, pp. arXiv–1710, 2017.