Scalable Lifelong Learning from Heterogeneous Demonstrations

Letian Chen*, Sravan Jayanthi*, Rohan Paleja, Daniel Martin, Viacheslav Zakharov, Matthew Gombolay
Georgia Institute of Technology
Atlanta, GA 30332
{letian.chen, sjayanthi, rpaleja3, dmartin1, vzakharov3, matthew.gombolay}@gatech.edu

***Abstract*— Learning from Demonstration (LfD) approaches empower end-users to teach robots novel tasks via demonstrations of the desired behaviors. However, current LfD frameworks are neither capable of fast adaptation to heterogeneous human demonstrations nor large-scale deployment in ubiquitous robotics applications. In this paper, we propose a novel LfD framework, Fast Lifelong Adaptive Inverse Reinforcement learning (FLAIR). Our approach (1) leverages learned strategies to construct policy mixtures for fast adaptation to new demonstrations; (2) distills common knowledge across demonstrations, achieving accurate task inference; and (3) expands its model when needed in lifelong deployments, maintaining a concise set of prototypical strategies that can approximate all behaviors via policy mixtures. We empirically validate that FLAIR achieves *adaptability* (i.e., the robot adapts to heterogeneous, user-specific task preferences), *efficiency* (i.e., the robot achieves sample-efficient adaptation), and *scalability* (i.e., the model grows sublinearly with the number of demonstrations while maintaining high performance). FLAIR surpasses benchmarks across three continuous control tasks with an average 57% improvement in policy returns and an average 78% fewer episodes required for demonstration modeling using policy mixtures. Finally, we demonstrate the success of FLAIR in a real-robot table tennis task.**

I. INTRODUCTION

Robots are becoming increasingly ubiquitous with recent advancements in Artificial Intelligence (AI), largely due to the success of Deep Reinforcement Learning (DRL) techniques in generating high-performance continuous control behaviors [1]–[3]. However, DRL's success heavily relies on sophisticated reward functions designed for each task. These hand-crafted reward functions typically require iterations of fine-tuning and consultation with domain experts to be effective [4]. Instead, Learning from Demonstration (LfD) approaches democratize access to robotics by having users demonstrate the desired behavior to the robot [5], removing the need for per-task reward engineering. Nevertheless, we must consider that end-users may adopt varying preferences and strategies in how they complete the same task [6]. An LfD framework that assumes homogeneity across the set of provided demonstrations could cause the robot to fail to infer the accurate intention, resulting in unwanted or even unsafe behavior [7], [8]. Embracing individual preferences can help robots achieve better performance and long-term acceptance from humans [9]. Personalization can also prove inefficient if each individual policy must be inferred separately. To avoid this, prior work, MSRD [10], decomposed shared and individual-specific reward information across heterogeneous demonstrations (i.e., demonstrations seeking to accomplish the same task with different styles). While MSRD makes significant improvements on the accuracy and efficiency in personalized policy modeling, the framework must be trained all-at-once and is unable to handle *incremental/lifelong learning*, a more realistic paradigm for real-world LfD applications.

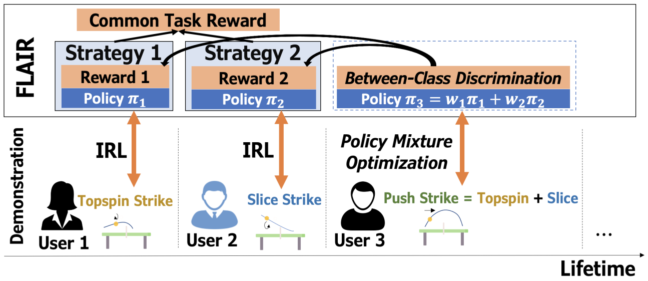

Fig. 1. This figure shows an illustration of the lifelong learning process with our proposed method, FLAIR. As each demonstrator performs their strike, FLAIR determines whether the demonstration is novel. If a demonstration can be explained by a *policy mixture* of previously learned strategies, FLAIR accepts the policy mixture without training a new strategy. If the policy mixture is not close to the demonstration, FLAIR creates a new strategy and a prototype policy for the demonstration.

Consider a real-world example of a series of humans teaching a robot how to play table tennis, a compelling robot learning platform utilized in prior work [11]–[13]. Users of the robot may have their own preferences for table tennis strike as shown in Figure I. To solve such lifelong robot LfD tasks, we introduce FLAIR: Fast Lifelong Adaptive Inverse Reinforcement learning. Instead of repeating a strategy that has been seen, when a third user demonstrates a behavior that could be explained by a mixture (i.e., a composition of known behaviors) of previously seen prototypical behaviors, FLAIR will design a policy mixture to adeptly model that behavior. We show FLAIR accomplishes *adaptivity*, *efficiency*, and *scalability* in LfD tasks in simulated and real robot experiments:

1) **Adaptive Learning**: We display FLAIR's *adaptivity* by showing it models demonstrations better than benchmarks and achieves an average of 57% higher task returns.
2) **Efficient Adaptation**: We demonstrate better sample *efficiency* of FLAIR by showing its mixture optimization needs an average of 78% fewer samples to model demonstrations compared with training a new policy.
3) **Lifelong Scalability**: We showcase the *scalability* of

[*] equal contribution

FLAIR in an experiment obtaining 100 demonstrations sequentially where FLAIR utilizes *policy mixtures* to achieve a precise representation of each demonstration.

4) **Real-Robot Deployment**: We demonstrate FLAIR's ability to construct successful policy mixtures achieving personalization in a table tennis robot experiment.

II. Related Work

Two common approaches in LfD are to either directly learn a policy, i.e., Imitation Learning (IL), or infer a reward to train a policy, i.e., Inverse Reinforcement Learning (IRL) [14]. IL learns a direct mapping from states to the actions demonstrated [15], [16]. Although a straightforward approach, IL suffers from correspondence matching issues and is not robust to changes in environment dynamics due to its mimicry of the demonstrated behaviors [17], [18]. IRL, on the other hand, infers the demonstrator's latent intent in a more robust and transferable form of a reward function [19].

Although traditional IRL approaches often overlook heterogeneity within demonstrations, there has been recent work that models heterogeneous demonstrations [20]–[25]. One intuitive way is to classify demonstrations into homogeneous clusters before applying IRL [6]. The Expectation Maximization (EM) algorithm also operates on a similar idea and iterates between E-step and M-step, where E-step clusters demonstrations and M-step solves the IRL problem on each cluster [26], [27]. When the number of strategies is unknown, a Dirichlet Process prior [28]–[30] or non-parametric methods [31] could be used. In these approaches, each reward function only learns from a portion of the demonstrations, making them prone to the issue of reward ambiguity [10]. Furthermore, these methods assume access to all demonstrations beforehand, which is not realistic for LfD algorithm deployment. We instead consider the more realistic setting of lifelong learning [32], where an agent adapts to new demostrations through its lifetime and continually builds its knowledge base.

III. Method

In this section, we introduce the problem setup and notations, then provide an overview of FLAIR, and its key components: *policy mixture* and *between-class discrimination*.

A. Problem Setup

In our problem setup, we consider a lifelong learning from heterogeneous demonstration process where demonstrations arrive in sequence, as illustrated in Figure I. We denote the i-th arrived demonstration as τ_i. Unlike prior work, FLAIR does not assume access to the ground-truth strategy label, c_{τ_i}. FLAIR learns a shared task reward $R_{\theta_{\text{Task}}}$, strategy rewards $R_{\theta_{\text{S-}j}}$, and policies corresponding to each strategy π_{ϕ_j}, similar to MSRD [33]. We define the number of prototype strategies created by FLAIR till demonstration τ_i as M_i, and $\eta_R(\tau) = \sum_{t=1}^{\infty} \gamma^{t-1} R_\theta(s_t)$ as trajectory τ's discounted cumulative reward with the inferred reward function R_θ.

Algorithm 1: FLAIR

```
Input : Demonstration modeling quality threshold $\epsilon$
1  $M_0 = 0$, MixtureWeights=[], $m$=[]
2  while lifelong learning from heterogeneous demonstration do
3      Obtain demonstration $\tau_i$
4      $\vec{w}_i, D_{\text{KL}}^{\text{mix}} \leftarrow$ MixtureOptimization($\tau_i, \{\pi_{\phi_j}\}_{j=1}^{M_i}$)
5      if $D_{KL}^{mix} < \epsilon$ then
6          MixtureWeights[i]$\leftarrow \vec{w}_i$, $M_{i+1} \leftarrow M_i$
7      else
8          $\pi_{\text{new}}, R_{\theta_{\text{S-}(M_i+1)}} \leftarrow$ AIRL($\tau_i$)
9          $D_{\text{KL}}^{\text{new}} \leftarrow \mathbb{E}_{\tau \sim \pi_{\text{new}}} D_{\text{KL}}(\tau_i, \tau)$
10         if $D_{KL}^{mix} < D_{KL}^{new}$ then
11             MixtureWeights[i]$\leftarrow \vec{w}_i$, $M_{i+1} \leftarrow M_i$
12         else
13             $M_{i+1} \leftarrow M_i + 1$
14             $m_{M_{i+1}} \leftarrow i$
15             MixtureWeights[i]$\leftarrow [\underbrace{0, 0, \cdots, 0}_{M_i \text{ zeros}}, 1]$
16     Update $R_{\theta_{\text{Task}}}, R_{\theta_{\text{S-}j}}, \pi_{\phi_j}$ by Between-Class Discrimination and MSRD
```

B. Fast Lifelong Adaptive Inverse Reinforcement Learning (FLAIR)

With FLAIR, we seek to accomplish two key goals: a) design policies that solve the task while personalizing to demonstrations (i.e., the standard objective in personalized LfD), and b) incorporate knowledge from demonstrated behaviors to facilitate precise, efficient, and scalable adaptation to future demonstrations (i.e., the characteristics required for a lifelong LfD framework). We present our method in pseudocode in Algorithm 1.

When a new demonstration τ_i becomes available, FLAIR decides whether to explain τ_i with previously learned policies (a highly efficient approach), or create a new strategy from scratch (a fallback technique). In the first case, FLAIR attempts to explain τ_i by constructing *policy mixtures* with previously learned strategies according to the demonstration recovery objective (line 4). If the trajectory generated by the mixture is close to the demonstration (evidenced by the KL-divergence between the *policy mixture* trajectory and the demonstration state distributions falling under a threshold, ϵ), FLAIR can bypass the computationally expensive new-strategy training (line 8).

Otherwise, if the mixture does not meet the quality threshold, ϵ, FLAIR trains a new strategy by AIRL [18] and compares the quality of the new policy to the *policy mixture* (Lines 8-10). If the mixture performs better, we accept the mixture weights to represent τ_i (line 11). If the new strategy performs better, we accept the new strategy as an additional prototype and update our reward and policy model (accordingly, in Line 13, we increment the number of strategies by one). Further, we call the demonstration, τ_i, the "pure" demonstration for strategy M_{i+1}, meaning strategy

TABLE I

THIS TABLE SHOWS LEARNED POLICY METRICS BETWEEN AIRL, MSRD, AND FLAIR. THE HIGHER ENVIRONMENT RETURNS / LOWER ESTIMATED KL DIVERGENCE, THE BETTER.

Domains	Inverted Pendulum			Lunar Lander			Bipedal Walker		
Methods	AIRL	MSRD	FLAIR	AIRL	MSRD	FLAIR	AIRL	MSRD	FLAIR
Environment Returns	-172.7	-166.4	$-\mathbf{38.5}^{**}$	-7418.1	-9895.3	$-\mathbf{6346.6}^{*}$	-30637.2	-74166.0	$-\mathbf{7064.0}^{**}$
Estimated KL Divergence	4.08	7.67	$\mathbf{4.01}^{**}$	72.0	70.9	$\mathbf{67.2}^{**}$	13.0	32.6	$\mathbf{12.1}^{**}$
Strategy Rewards	-5.73	-6.22	$-\mathbf{1.23}$	-12.67	-20.26	$-\mathbf{4.19}^{*}$	-5.31	-29.82	$-\mathbf{4.22}^{**}$

* Significance of $p < 0.05$

** Significance of $p < 0.01$

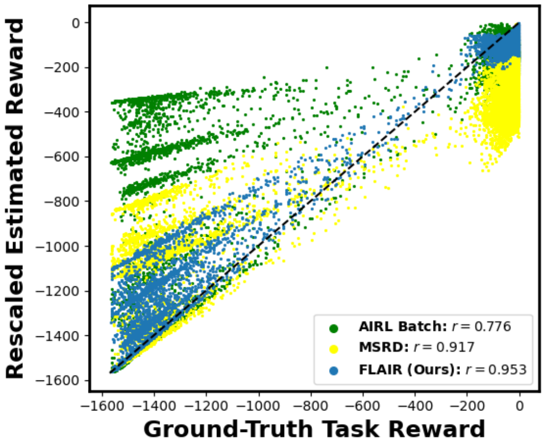

Fig. 2. This figure shows the correlation between the estimated task reward with the ground truth task reward for Inverted Pendulum. Each dot is a trajectory. FLAIR achieves a higher task reward correlation.

M_{i+1} purely represents demonstration τ_i (line 14). As such, the mixture weight for τ_i is a one-hot vector on strategy M_{i+1} (line 15).

C. Policy Mixture Optimization

To achieve efficient personalization for a new demonstration τ_i (Line 4 of Algorithm 1), we construct a *policy mixture* with a linear geometric combination of existing policies $\pi_1, \pi_2, \cdots, \pi_{M_i}$ (Equation 1), where $w_{i,j} \geq 0$ are learned weights such that: $\sum_{j=1}^{M_i} w_{i,j} = 1$.

$$\pi_{\vec{w}_i}(s) = \sum_{j=1}^{M_i} w_{i,j} a_j, \quad a_j \sim \pi_j(s) \tag{1}$$

As the ultimate goal of demonstration modeling is to recover the demonstrated behavior, we optimize the linear weights, $\vec{w}_i$, to minimize divergence between the trajectory induced by the mixture policy and the demonstration, illustrated in Equation 2.

$$\underset{\vec{w}_i}{\text{minimize}}\, \mathbb{E}_{\tau \sim \pi_{\vec{w}_i}} \left[\text{Div}(\tau_i, \tau) \right] \tag{2}$$

Specifically, we choose Kullback-Leibler divergence (KL-divergence) [34] on the state marginal distributions of trajectories in our implementation. We estimate the state distribution within a trajectory by the kernel density estimator [35]. Since the trajectory generation process is non-differentiable, we seek a non-gradient-based optimizer to solve the optimization problem. Specifically, FLAIR utilizes a naïve, random optimization method; it generates random weight vectors $\vec{w}_i$, evaluates Equation 2, and chooses the weight that achieves the minimization.

D. Between-Class Discrimination

In order to increase the strategy reward's discriminability between different strategies, we propose a novel learning objective named *Between-Class Discrimination* (BCD). BCD enforces the strategy reward to correctly discriminate mixture demonstrations from the pure demonstration: If demonstration τ_i has weight $w_{i,j}$ on strategy j (as identified in *Policy Mixture Optimization*), we could view the probability that τ_i happens under the strategy reward, $R_{\text{S-}i}$, should be $w_{i,j}$ proportion of the probability of the pure demonstration, τ_{m_j}. This property can be exploited to enforce a structure on the reward given to the pure-demonstration, τ_{m_j}, and mixture-demonstration τ_i, as per Lemma 1.

Lemma 1: Under the maximum entropy principal,

$$w_{i,j} = \frac{P(\tau_i; \text{S-}j)}{P(\tau_{m_j}; \text{S-}j)} = \frac{e^{\eta R_{\text{S-}j}(\tau_i)}}{e^{\eta R_{\text{S-}j}(\tau_{m_j})}}$$

Thus, we enforce the relationship of strategy rewards, S-j, evaluated on pure strategy demonstration, τ_{m_j}, and mixture strategy demonstration, τ_i with mixture weight $w_{i,j}$, as shown in Equation 3.

$$L_{\text{BCD}}(\theta^{\text{S-}j}) = \sum_{i=1}^{n} \left(e^{\eta \theta_{\text{S-}j}(\tau_i)} - w_{i,j} e^{\eta \theta_{\text{S-}j}(\tau_{m_j})} \right)^2 \tag{3}$$

An extreme case of BCD loss is when τ_i is the pure demonstration for another strategy, k (i.e., $m_k = i$). In this case, $w_{i,j} = 0$ (as τ_i is purely on strategy k), and Equation 3 degenerates to encourage the strategy j's reward to give as low as possible reward to τ_i. In turn, strategy rewards gain better discrimination between different strategies, facilitating more robust strategy reward learning, and contributing to the success in lifelong learning.

IV. RESULTS

In this section, we show that FLAIR achieves *adaptability*, *efficiency*, and *scalability* in modeling heterogeneous demonstrations. We test FLAIR on three simulated continuous control environments in OpenAI Gym [36]: Inverted Pendulum (IP) [37], Lunar Lander (LL), and Bipedal Walker (BW) [38]. We generate a collection of heterogeneous demonstrations by jointly optimizing an environment and diversity reward with DIAYN [39]. For all experiments excluding the scalability study, we use ten demonstrations. We compare FLAIR with AIRL and MSRD by running three trials of each method.

TABLE II

THIS TABLE DEPICTS POLICY METRICS BETWEEN FLAIR'S BEST MIXTURES, FLAIR'S WORST MIXTURES, AND LEARNING-FROM-SCRATCH WITH AIRL. THE SCORES ARE SHOWN AS AVERAGES ± STANDARD DEVIATIONS ACROSS PARTICIPANTS. BOLD DENOTES THE HIGHEST SCORES.

Metrics	FLAIR's Best Mixture	FLAIR's Worst Mixture	Learning-from-Scratch
Task Score	$\mathbf{69.9 \pm 10.3}$	63.1 ± 13.4	60.5 ± 10.9
Strategy Score	$\mathbf{101.3 \pm 14.8}$	73.1 ± 25.3	93.7 ± 17.1

A. Adaptability

Q1: ***Can FLAIR's policy mixtures perform well at the task?*** From ten demonstrations, FLAIR created 6.3 ± 0.5 strategies (average and standard deviation across three trials) in IP, 5.3 ± 1.2 in LL, and 3.3 ± 0.5 in BW. FLAIR's learned policies including the policy mixtures are signficantly more successful at the task (row "Environment Returns" in Table I), outperforming benchmarks in task performance with 77% higher returns in IP, 14% in LL, and 80% in BW than best baselines.

Q2: ***How closely does the policy recover the strategic preference?*** We show that FLAIR is statistically significantly better in estimated KL divergence than AIRL (average 4% better) and MSRD (average 18% better), shown in row "Estimated KL Divergence" in Table I, where KL divergence is evaluated between policy rollouts and demonstration state distributions. We further tested the learned policies' performance on ground-truth strategy reward functions given by DIAYN. The results on row "Strategy Rewards" illustrate FLAIR's better adherence to the demonstrated strategies.

Q3. ***How well does the task reward model the ground truth environment reward?*** We evaluate the learned task reward functions by calculating the correlation between estimated task rewards with ground-truth environment rewards. We construct a test dataset of 10,000 trajectories with multiple policies obtained during the "DIAYN+env reward" training. FLAIR's task reward achieves $r = 0.953$ in IP (shown in Figure 2), $r = 0.614$ in LP, and $r = 0.582$ in BW, with an average 18% (statistically significantly) higher correlation than best baselines.

B. Efficiency & Scalability

Q4. ***Can FLAIR's mixture optimization model demonstrations more efficiently than learning a new, separate policy?*** We study the number of episodes needed by FLAIR's mixture optimization and AIRL/MSRD policy training to achieve the same modeling performance of demonstrations. FLAIR requires 77% fewer episodes to achieve a high log likelihood of the demonstration relative to AIRL and 79% fewer episodes than MSRD.

Q5. ***Can FLAIR's success continue in a larger-scale LfD problem?*** We generate 95 mixtures with randomized weights from 5 prototypical policies for a total of 100 unique demonstrations to test how well FLAIR scales. We train FLAIR sequentially on the 100 demonstrations and observe FLAIR learns a concise set of 17 strategies in IP, 10 in LL, and 6 in BW that capture the scope of behaviors while also achieving a consistently strong return for each task (Figure 3). We find FLAIR on average is able to maintain or even exceed its 10-demonstration performance when scaling up to 100 demonstrations.

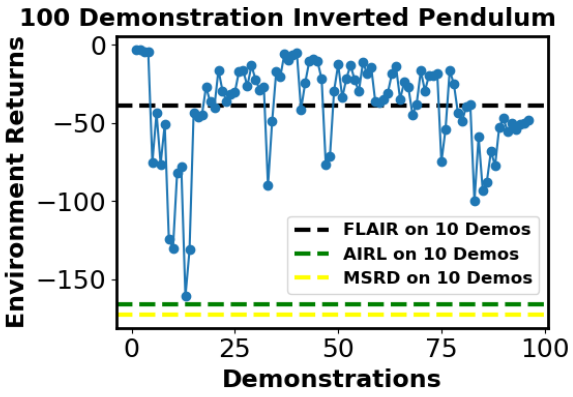

Fig. 3. This figure plots the returns of FLAIR policies in the scalability experiment for 100 demonstrations in IP, with AIRL, MSRD, and FLAIR 10-demonstration experiment performance as reference.

C. Real-World Robot Case Study: Table Tennis

We perform a real-world robot table tennis experiment where we utilize FLAIR's *policy mixtures* to model user demonstrations. We first collect demonstrations of four different strategies, push, slice, topspin, and lob, from human subjects via kinesthetic teaching. After training the four prototypical strategy policies, we assess how FLAIR's policy mixtures can succeed in new user demonstration modeling.

We quantitatively evaluate the fitness of the policy mixtures with respect to user preferences by a questionnaire, where we calculate task and strategy scores produced via a Likert scale. The strategy and task scores results are summarized in Table II. Statistical tests shows the FLAIR best mixture has significantly higher task reward than learning-from-scratch, and both FLAIR best mixture and learning-from-scratch have significantly higher strategy reward than the FLAIR worst mixture. Such result indicates the success of FLAIR's mixture optimization in identifying a policy mixture that accomplishes the task and fulfills the user's preference in the table tennis real-robot setup.

V. CONCLUSION

In this paper, we present FLAIR, a fast lifelong adaptive LfD framework. We demonstrate FLAIR's *adaptability* to novel personal preferences and *efficiency* by utilizing policy mixtures. We also illustrate FLAIR's *scalability* in how it learns a concise set of strategies to solve the problem of modeling a large number of demonstrations.

References

[1] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, and S. Levine, "Soft actor-critic algorithms and applications." *CoRR*, vol. abs/1812.05905, 2018.

[2] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. A. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, 2015.

[3] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[4] L. Matignon, G. J. Laurent, and N. Le Fort-Piat, "Reward function and initial values: Better choices for accelerated goal-directed reinforcement learning," in *Artificial Neural Networks – ICANN 2006*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 840–849.

[5] S. Schaal, "Learning from demonstration," in *Advances in Neural Information Processing Systems*, M. C. Mozer, M. Jordan, and T. Petsche, Eds., vol. 9. MIT Press, 1997.

[6] S. Nikolaidis, R. Ramakrishnan, K. Gu, and J. Shah, "Efficient model learning from joint-action demonstrations for human-robot collaborative tasks," in *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2015, pp. 189–196.

[7] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza, "Power to the people: The role of humans in interactive machine learning," *AI Magazine*, vol. 35, no. 4, pp. 105–120, Dec. 2014.

[8] E. F. Morales and C. Sammut, "Learning to fly by combining reinforcement learning with behavioural cloning," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2004, p. 76.

[9] I. Leite, C. Martinho, and A. Paiva, "Social robots for long-term interaction: A survey," *International Journal of Social Robotics*, vol. 5, no. 2, pp. 291–308, Apr 2013.

[10] L. Chen, R. R. Paleja, M. Ghuy, and M. C. Gombolay, "Joint goal and strategy inference across heterogeneous demonstrators via reward network distillation," in *Proceedings of the International Conference on Human-Robot Interaction (HRI)*, 2020.

[11] K. Mülling, J. Kober, O. Kroemer, and J. Peters, "Learning to select and generalize striking movements in robot table tennis," *Proceedings of the International Journal of Robotics Research (IJRR)*, vol. 32, no. 3, pp. 263–279, 2013.

[12] K. Muelling, A. Boularias, B. Mohler, B. Schölkopf, and J. Peters, "Learning strategies in table tennis using inverse reinforcement learning," *Biological cybernetics*, vol. 108, no. 5, pp. 603–619, 2014.

[13] L. Chen, R. Paleja, and M. Gombolay, "Learning from suboptimal demonstration via self-supervised reward regression," in *Proceedings of Conference on Robot Learning (CoRL)*, 2020.

[14] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, 2020.

[15] A. Chella, H. Dindo, and I. Infantino, "A cognitive framework for imitation learning," *Robotics and Autonomous Systems*, vol. 54, no. 5, pp. 403–408, 2006, the Social Mechanisms of Robot Programming from Demonstration.

[16] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne, "Imitation learning: A survey of learning methods," *ACM Computing Surveys (CSUR)*, vol. 50, no. 2, pp. 1–35, 2017.

[17] P. de Haan, D. Jayaraman, and S. Levine, "Causal confusion in imitation learning," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019.

[18] J. Fu, K. Luo, and S. Levine, "Learning robust rewards with adverserial inverse reinforcement learning," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[19] N. D. Daw and P. Dayan, "The algorithmic anatomy of model-based evaluation," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 369, no. 1655, p. 20130478, 2014.

[20] A. Y. Ng, S. Russell *et al.*, "Algorithms for inverse reinforcement learning." in *Icml*, vol. 1, 2000, p. 2.

[21] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the International Conference on Machine Learning (ICML)*. ACM, 2004.

[22] D. Ramachandran and E. Amir, "Bayesian inverse reinforcement learning," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. Morgan Kaufmann Publishers Inc., 2007, p. 2586–2591.

[23] B. D. Ziebart, A. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *Proceedings of the National Conference on Artificial intelligence (AAAI)*, 2008, pp. 1433–1438.

[24] B. D. Ziebart, "Modeling purposeful adaptive behavior with the principle of maximum causal entropy," Ph.D. dissertation, Carnegie Mellon University, 2010.

[25] R. Paleja, A. Silva, L. Chen, and M. Gombolay, "Interpretable and personalized apprenticeship scheduling: Learning interpretable scheduling policies from heterogeneous user demonstrations," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 6417–6428.

[26] M. Babes-Vroman, V. Marivate, K. Subramanian, and M. Littman, "Apprenticeship learning about multiple intentions." in *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, 01 2011, pp. 897–904.

[27] G. Ramponi, A. Likmeta, A. M. Metelli, A. Tirinzoni, and M. Restelli, "Truly batch model-free inverse reinforcement learning about multiple intentions," in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, S. Chiappa and R. Calandra, Eds., vol. 108. PMLR, 26–28 Aug 2020, pp. 2359–2369.

[28] J. Almingol, L. Montesano, and M. Lopes, "Learning multiple behaviors from unlabeled demonstrations in a latent controller space," in *Proceedings of the 30th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, S. Dasgupta and D. McAllester, Eds., vol. 28. Atlanta, Georgia, USA: PMLR, 17–19 Jun 2013, pp. 136–144.

[29] A. Bighashdel, P. Meletis, P. Jancura, and G. Dubbelman, "Deep adaptive multi-intention inverse reinforcement learning," in *Machine Learning and Knowledge Discovery in Databases. Research Track*, N. Oliver, F. Pérez-Cruz, S. Kramer, J. Read, and J. A. Lozano, Eds. Cham: Springer International Publishing, 2021, pp. 206–221.

[30] J. Choi and K.-e. Kim, "Nonparametric bayesian inverse reinforcement learning for multiple reward functions," in *Advances in Neural Information Processing Systems*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012.

[31] S. Rajasekaran, J. Zhang, and J. Fu, "Inverse reinforce learning with nonparametric behavior clustering," *arXiv preprint arXiv:1712.05514*, 2017.

[32] J. A. Mendez, S. Shivkumar, and E. Eaton, "Lifelong inverse reinforcement learning." in *NeurIPS*, 2018, pp. 4507–4518.

[33] L. Chen, R. Paleja, M. Ghuy, and M. Gombolay, "Joint goal and strategy inference across heterogeneous demonstrators via reward network distillation," in *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, 2020, pp. 659–668.

[34] S. Kullback and R. A. Leibler, "On Information and Sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79 – 86, 1951.

[35] L. F. Kozachenko and N. N. Leonenko, "Sample estimate of the entropy of a random vector," *Probl. Inf. Transm.*, vol. 23, no. 1-2, pp. 95–101, 1987.

[36] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," *CoRR*, vol. abs/1606.01540, 2016.

[37] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2012.

[38] C. Ericson, *Real-Time Collision Detection*. USA: CRC Press, Inc., 2004.

[39] B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine, "Diversity is all you need: Learning skills without a reward function," in *International Conference on Learning Representations*, 2019.