

# A Survey of Imitation Learning: Algorithms, Recent Developments, and Challenges

Maryam Zare\*, Parham M. Kebria, *Member, IEEE*, Abbas Khosravi, *Senior Member, IEEE*, and Saeid Nahavandi, *Fellow, IEEE*

**Abstract**—In recent years, the development of robotics and artificial intelligence (AI) systems has been nothing short of remarkable. As these systems continue to evolve, they are being utilized in increasingly complex and unstructured environments, such as autonomous driving, aerial robotics, and natural language processing. As a consequence, programming their behaviors manually or defining their behavior through reward functions (as done in reinforcement learning (RL)) has become exceedingly difficult. This is because such environments require a high degree of flexibility and adaptability, making it challenging to specify an optimal set of rules or reward signals that can account for all possible situations. In such environments, learning from an expert’s behavior through imitation is often more appealing. This is where imitation learning (IL) comes into play - a process where desired behavior is learned by imitating an expert’s behavior, which is provided through demonstrations.

This paper aims to provide an introduction to IL and an overview of its underlying assumptions and approaches. It also offers a detailed description of recent advances and emerging areas of research in the field. Additionally, the paper discusses how researchers have addressed common challenges associated with IL and provides potential directions for future research. Overall, the goal of the paper is to provide a comprehensive guide to the growing field of IL in robotics and AI.

**Index Terms**—Imitation learning, learning from demonstrations, reinforcement learning, survey, robotics

## I. INTRODUCTION

Traditionally, machines and robots have been manually programmed to learn autonomous behavior [1]. Traditional methods require experts to provide specific, hard-coded rules regarding the actions that a machine must perform, as well as the characteristics of the environment in which the machine operates. However, developing such rules requires considerable time and coding expertise [2]. In order to automate the tedious manual hard-coding of every behavior, a learning approach is required [3]. Imitation learning provides an avenue for teaching the desired behavior by demonstrating it. IL techniques have the potential to reduce the problem of teaching a task to that of providing demonstrations, thus eliminating the

Maryam Zare, Parham M. Kebria, and Abbas Khosravi are with the Institute for Intelligent Systems Research and Innovation (IISRI), Deakin University, Waurn Ponds, 3216 VIC, Australia (\*Corresponding author: mzare@deakin.edu.au).

Saeid Nahavandi is with the Institute for Intelligent Systems Research and Innovation (IISRI), Deakin University, Waurn Ponds, 3216 VIC, Australia, and also with the Harvard Paulson School of Engineering and Applied Sciences, Harvard University, Allston, MA 02134 USA.

This work has been submitted to the IEEE for possible publication. Copyright may be transferred without notice, after which this version may no longer be accessible.

need for explicit programming or the development of task-specific reward functions [3]. The concept of IL is based on the premise that human experts are capable of demonstrating the desired behavior even when they are unable to program it into a machine or robot. As such, IL can be leveraged in any system that requires autonomous behavior similar to that of a human expert [1].

The main purpose of IL is to enable agents to learn to perform a specific task or behavior by imitating an expert through the provision of demonstrations [4]. Demonstrations are used to train learning agents to perform a task by learning a mapping between observations and actions. By utilizing IL, agents are able to transition from repeating simple predetermined behaviors in constrained environments to taking optimal autonomous actions in unstructured environments, without imposing too much burden on the expert [2]. As a result, IL approaches have the potential to offer significant benefits to a wide range of industries, including manufacturing [5], health care [6], autonomous vehicles [7], [8], and the gaming industry [9]. In these applications, IL allows subject-matter experts, who may not possess coding skills or knowledge of the system, to program autonomous behavior in machines or robots efficiently. Although the idea of learning by imitation has been around for some time, recent achievements in computing and sensing, along with a growing demand for artificial intelligence applications, have increased the significance of IL [10], [11]. Consequently, the number of publications in the field has increased significantly in recent years.

Multiple surveys of IL have been published over the past two decades, each focusing on different aspects of the field’s development (Fig. 1). Schaal [3] presented the first survey of IL, focusing on IL as a route to create humanoid robots. More recently, Osa et al. [1] provided an algorithmic perspective on IL, while Hussein et al. [12] provided a comprehensive review of the design options for each stage of the IL process. Most recently, Le Mero et al. [7] provided a comprehensive overview of IL-based techniques for end-to-end autonomous driving systems.

Despite the existence of a large number of surveys on IL, a new survey is necessary to capture the latest advances in this rapidly evolving field and provide an up-to-date overview of the state of the art. With the field gaining increasing interest and having diverse applications, a comprehensive survey could serve as an essential reference for newcomers, as well as provide an overview of different use cases. We acknowledge that IL is a constantly evolving field, with new algorithms, techniques, and applications being developed. Therefore, our

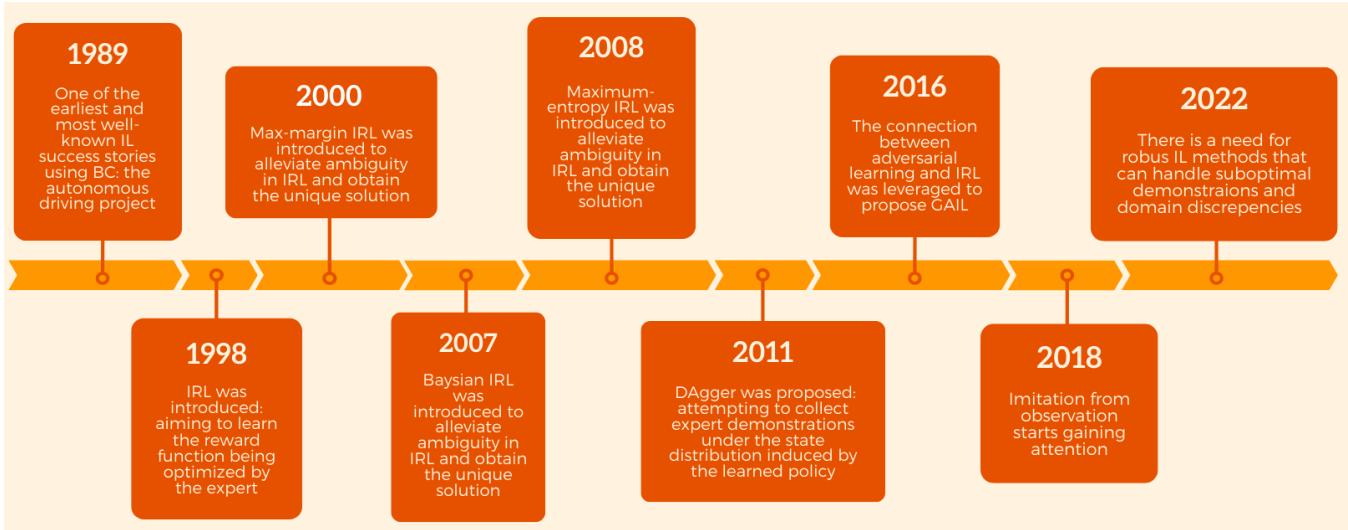


Fig. 1. A historical timeline of IL research illustrating key achievements in the field.

survey aims to consolidate the vast amount of research on IL, making it easier for researchers and practitioners to navigate. Moreover, we aim to identify gaps and challenges in the current research, providing a clear direction for future work. Lastly, we aim to make the concepts and techniques of IL more accessible to a wider audience, including researchers from related fields, to enhance the understanding of this area. Overall, we strongly believe that our survey will make significant contributions to advancing the field of IL and guide future research in this exciting area.

The objective of this survey paper is to present a comprehensive overview of the field of IL. To achieve this, we will organize our discussion of IL approaches based on historical and logical reasons. Initially, we will introduce the two broad categories of approaches to IL: behavioral cloning (BC) and inverse reinforcement learning (IRL). We will discuss their formulations, developments, strengths, and limitations. Additionally, we will explore how adversarial imitation learning (AIL) extends IRL by introducing an adversarial context to the learning process. We will underscore the benefits of integrating adversarial training into IL and assess the current progress in the AIL field. Furthermore, we will introduce imitation from observation (IfO) as a novel technique that aims to learn from state-only (no actions) demonstrations. We will discuss the significance of IfO and how it incorporates and extends the previous categories of BC, IRL, and AIL in different methods to tackle the challenge of learning from state-only observations. Finally, we will discuss the challenges that IL techniques encounter in real-world scenarios, such as sub-optimal demonstrations and domain discrepancies between the expert and the learner. We will conclude with a discussion of the different approaches to IL, their limitations, and future research directions that can be taken to address them.

## II. BEHAVIORAL CLONING

BC is an IL technique that treats the problem of learning a behavior as a supervised learning task [13], [14]. BC involves

training a model to mimic an expert's behavior by learning to map the state of the environment to the corresponding expert action. The expert's behavior is recorded as a set of state-action pairs, also known as demonstrations. During the training process, the model is provided with these demonstrations as inputs and is trained to learn a function that maps the current state to the corresponding expert action. Once the model is trained, it can use the learned function to generate actions for new states that it has not encountered before.

One advantage of BC is that it requires no knowledge of the underlying dynamics of the environment [13]. Instead, it relies solely on the provided demonstrations to learn the behavior. Additionally, BC is computationally efficient since it involves training a supervised learning model, which is a well-studied problem in machine learning.

Despite its simplicity, the BC approach has a significant drawback - the covariate shift problem [15]. This problem arises because during training, the learner is trained on states generated by the expert policy, but during testing, the learner is tested on states induced by its action [16]. As a result, the state distribution observed during testing can differ from that observed during training. The problem with BC supervised approach is that the agent does not know how to return to the demonstrated states when it drifts and encounters out-of-distribution states [17]. Covariate shift is particularly dangerous in safety-critical situations such as driving [18], as the agent may encounter novel situations that it has not seen during training, and its ability to recover from mistakes can be critical to avoid accidents. To address the covariate shift problem and improve the robustness of the BC approach, three broad research areas have been identified (Fig. 2).

The first and most popular area is interactive IL. Algorithms of this type are based on the assumption that the agent has access to an online expert who can be consulted during training. Dataset aggregation (DAgger) [14] is the earliest interactive IL method and proposes to train the agent on its own state distribution to resolve the train and test time

mismatch problem. DAgger queries the expert to relabel the data collected by the agent with the appropriate action that should have been taken. However, due to frequent queries, the human expert is subjected to a significant cognitive burden, resulting in inaccurate or delayed feedback that adversely affects the training process [19]. Consequently, determining when and how to engage human subjects is one of the key challenges of interactive IL algorithms [20].

Rather than providing continuous feedback, “human-gated” interactive IL algorithms [21], [22] extend DAgger to allow the expert to decide when to provide the corrective interventions. For example, human-gated DAgger (HG-DAgger) [21] rolls out the agent trajectory until the expert determines that the agent has reached an unsafe region of the state space. In this case, the human expert intervenes by taking control of the system and guiding the agent back to a safe state. Using this method, no constraints limit the amount of human intervention. Li et al. [19] propose a method that learns to minimize human intervention and adaptively maximize automation during training. To accomplish this, when the human expert issues intervention, it incurs a cost to the agent, which the agent learns to minimize during its training process.

However, the use of these algorithms depends on human experts constantly monitoring the agent to decide when to intervene, which imposes a significant burden on them. To tackle this challenge, there has been an increasing interest in “robot-gated” algorithms [20], [23]–[25] that allow robots to actively ask humans for intervention. For example, SafeDAgger [23] uses an auxiliary safety policy, which determines the likelihood of the agent deviating from the expert’s trajectory, as a signal for the agent to transfer control over to the expert. LazyDAgger [24] extends SafeDAgger to reduce the number of context switches between the expert and autonomous control. A recent robot-gated approach called ThriftyDAgger [20] aims to intervene only when states are sufficiently novel (out-of-distribution) or risky (prone to result in task failure). In addition, the intervention burden is reduced by limiting the total number of interventions to a budget specified by the user.

The second area of research addressing the covariate shift problem consists of algorithms that estimate the support of the expert occupancy measure and then specify a reward that encourages the agent to remain on the support of the expert [17], [26], [27]. The reward function is then optimized using RL. Unlike interactive IL, these algorithms do not assume access to an online expert and only rely on demonstrations and further interactions with the environment. The most popular algorithms are based on IRL, which addresses the covariate shift by training an RL agent to consistently match the demonstrations over time. A detailed discussion of these methods will be provided in section III and IV. However, these methods often use complex and unstable approximation techniques involving adversarial training to learn a reward function from demonstrations [17], [28]. In this section, a recent alternative line of research is reviewed that also employs RL, but instead of learning a reward function, it uses a simple fixed reward function. The key idea is to incentivize the agent to consistently remain on the support of the expert policy over time by encouraging it to return to demonstrated states when

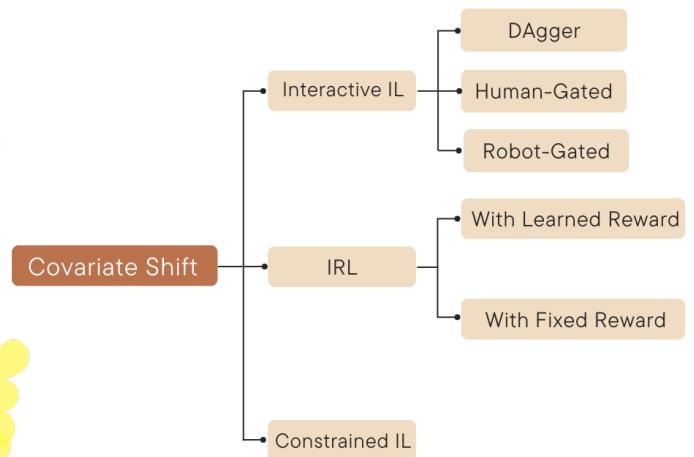


Fig. 2. A categorization of methods addressing the covariate shift problem. Interactive IL assumes access to an online expert. DAgger like algorithms require the expert to provide corrective labels for each action taken by the agent. On the other hand, human-gated and robot-gated methods provide corrective labels only when they are requested by the expert or agent, respectively. Unlike interactive IL, IRL methods do not require access to an online expert. These methods require an underlying RL algorithm to optimize a reward function (either learned from demonstrations or fixed). Lastly, by constraining the agent to known regions of the space covered by demonstrations, constrained IL attempts to address covariate shift problems that cannot be expressed or solved using the other two categories.

facing new states outside the expert’s support [29].

Wang et al. [26] estimate the support of the expert policy using a kernelized version of principal component analysis. The support estimation process produces a score that increases as a state-action pair moves closer to the support of the expert policy. The score is then used to construct an intrinsic reward function.

Reddy et al. [17] propose soft Q IL (SQIL). SQIL encourages the agent to imitate the expert in demonstrated states by using an extremely sparse reward function – assigns a constant reward of +1 to transitions inside expert demonstrations and a constant reward of 0 to all other transitions. The reward function encourages the agent to return to demonstrated states after encountering out-of-distribution states. The proposed model outperforms simple BC and shows good performance even with a limited number of demonstrations.

Brantley et al. [27] use expert demonstrations to train an ensemble of policies, with the variance of their predictions used as the cost. Inherently, the variance (cost) outside of expert support would be higher since ensemble policies will be more likely to disagree on states not seen in the demonstrations. An RL algorithm minimizes this cost in combination with a supervised BC cost. As a result, the RL cost assists the agent in returning to the expert distribution, whereas the supervised cost ensures that the agent mimics the expert within the expert’s distribution.

Lastly, the third area of algorithms aims to constrain the agent to known regions of the space covered by the demonstrator without relying on an interactive expert or leveraging RL. These methods are particularly beneficial and practical for real-world applications where safety constraints must be met, such as in healthcare, autonomous driving, and industrial

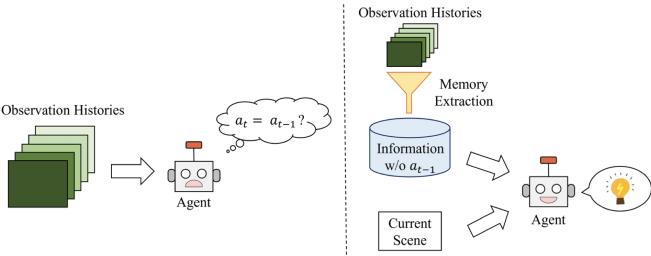


Fig. 3. Left: BC might learn a shortcut from prior observations that outputs the previous action as the current action. Right: a copycat-free memory extraction module. The shortcut is no longer available using historical information [34].

processes [30]. In [31], the authors attempt to overcome the covariate shift problem in autonomous driving by augmenting the imitation loss with additional losses to discourage bad driving. Moreover, additional data is provided to the agent in the form of synthetic perturbations to the expert's trajectory. The perturbations expose the model to non-expert behavior such as collisions and provide an important signal for the added losses to avoid these behaviors.

Wong et al. [32] propose a learned error detection system to indicate when an agent is in a potential failure state. In this way, the error detector can constrain the policy to execute only on states seen previously in the demonstrations and prevent potentially unstable behavior by resetting the agent to a well-known configuration or terminating the execution. [33] considers an offline IL setting where the agent is provided with two datasets: one small dataset of expert policy state-action pairs and one large dataset of state-action-next state transitions from a potentially suboptimal behavior policy. Their solution to the covariate shift problem consists of training a dynamics model on the sub-optimal demonstrations and applying a high penalty in regions of the state-action space that are not well covered by the data.

It is difficult for the supervised learning approach used in behavioral cloned policies to identify the underlying causes of expert actions, leading to a phenomenon known as “causal misidentification” [35]. In the training procedure, the causal structure of the interaction between the expert and the environment is not taken into account. Therefore, cloned policies might fail to distinguish nuisance correlates from the true causes of expert actions [35]. When training and testing distributions are the same, ignoring nuisance correlates might not pose a problem as they continue to hold in the test dataset. In BC, however, ignoring causality is particularly problematic because of the distributional shift [35]. The causal structure of the underlying sequential decision process in IL further exacerbates this problem. This is because the causal structure—past actions influence future observations—often creates additional complex nuisance correlates. To address this problem, [35] learns a mapping from causal graphs to policies and then uses targeted interventions—either expert queries or environment interactions—to find the optimal policy. Subsequent work by Wen et al. [36] explores a prominent class of causal confusion problems known as “the copycat problem.” This problem happens when the expert actions are highly correlated over

time. In this scenario, the agent learns to cheat by copying the expert's previous actions. To address the copycat problem, [36] proposes an adversarial approach for learning a feature representation that ignores the information about the known nuisance correlates—the previous actions—while retaining the necessary information to predict the next action. Chuang et al. [34] extend [36] to high-dimensional image observations. Using a memory extraction module, they attempt to extract historical features from observation histories while removing as much information as possible regarding previous actions (Fig. 3).

Traditionally, BC involves training an explicit neural network [37]. Unfortunately, conventional explicit models struggle to model discontinuities, making policies incapable of switching decisively between different behaviors. This issue arises because explicit models are unable to represent discontinuities with neural networks built with continuous activation functions, which is almost always the case with neural networks [37]. In contrast, implicit models are capable of representing sharp discontinuities, despite only having continuous layers in the network [37].

The implicit BC model presented in [38] turns BC into an energy-based modeling problem [39] by training a neural network that takes both observations and actions as input and outputs a single value that is low for expert actions and high for non-expert actions [37]. A trained implicit BC policy selects the action input with the lowest score for a given observation. This method requires more computation than explicit BC models, both during training and during inference. However, the results demonstrate that it can often outperform traditional explicit baselines in robotic manipulation tasks, both in the real world and in simulation.

### III. INVERSE REINFORCEMENT LEARNING

In addition to behavioral cloning, another key approach to imitation learning is IRL [40]. IRL involves an apprentice agent that aims to infer the reward function underlying the observed demonstrations, which are assumed to come from an expert who acts optimally [41]. Once the reward function is inferred, it is optimized to train an apprentice policy through RL [42].

RL agents, unlike the agents in BC, learn by continually interacting with their environment, observing the consequences of their actions, and altering their behavior to maximize long-term cumulative reward [43], [44]. This process involves using reinforcement signals to learn the long-term consequences of each action, allowing the agent to recover from mistakes [27]. Because of this capability, IRL is less sensitive to covariate shift compared to BC [14].

IRL has been widely used in a variety of applications, such as robotics manipulation, autonomous navigation, game playing, and natural language processing [45]–[47]. Nonetheless, devising an effective IRL algorithm for learning from demonstrations is a challenging task, mainly due to two major reasons.

Firstly, IRL can be computationally expensive and resource-intensive. In part, this is due to the fact that the agent must

interact repeatedly with its environment to accurately estimate the reward function [19], [48]. Additionally, the nature of this process can be inherently unsafe, particularly when dealing with high-risk applications such as autonomous driving or aircraft control [49].

Furthermore, a typical IRL approach follows an iterative process that involves alternating between reward estimation and policy training, which results in poor sample efficiency [46], [47], [50]. Consequently, there has been significant research aimed at addressing these issues to enhance the sample efficiency of IRL algorithms while maintaining the safety and accuracy of the learned policy. Some of these approaches include methods that utilize human guidance to reduce the number of interactions required to estimate the reward function accurately [51].

The second major challenge of IRL arises due to the inherent ambiguity in the relationship between the policy and the reward function. Specifically, a policy can be optimal with respect to an infinite number of reward functions [50], [52]. To address this challenge, researchers have proposed various methods to introduce additional structure into the reward function. There are roughly three categories of IRL methods that aim to address this ambiguity [53].

The first category is maximum-margin methods. The key idea in maximum-margin methods is to infer a reward function that explains the optimal policy more thoroughly than all other policies by a margin. These methods address the discussed ambiguity problem by converging on a solution that maximizes some margin. A foundational method in this category is the work of Ng et al. [50]. They estimate the reward function for which the given policy is optimal using a linear program while maximizing a margin. Another major work is maximum margin planning (MMP) [54] which seeks to find a weighted linear mapping of features to rewards so that the estimated policy is “close” to the demonstrated behaviors. [55], [56] build on and extend MMP to nonlinear hypothesis spaces by utilizing a family of functional gradient techniques.

The adoption of feature-based reward functions gave rise to a variety of approaches making use of feature expectations for margin optimization. Abbeel and Ng [45] propose two foundational methods (max-margin and projection) for maximizing the feature expectation loss margin without assuming access to the expert’s policy. As with many other IRL methods, these methods have the drawback of limiting the agent’s performance to the quality of the expert. To address this limitation, Syed and Schapire [57] propose a game-theoretic approach capable of training a policy with superior performance to an expert.

The second category of IRL algorithms aims to solve the ambiguity problem by maximizing the entropy of the resulting policy. MaxEntIRL [47] was the first IRL method to utilize maximum entropy. Ziebart’s work [47] demonstrates that the maximum entropy paradigm is capable of handling expert suboptimality and stochasticity by using the distribution over possible trajectories. Subsequent works [58], [59] extend the MaxEntIRL algorithm to continuous state-action spaces using path integral methods. Optimizing a full forward Markov Decision Process (MDP) iteratively becomes intractable in

high-dimensional continuous state-action spaces. To overcome this complexity, these studies exploit the local optimality of demonstrated trajectories.

In many prior methods, detailed features are manually extracted using domain knowledge, which can be linearly combined into a reward, such as the distance between the ball and the cup for a robotic ball-in-cup game [60]. While linear representations are sufficient in many domains, they may be overly simplistic for complex real-world tasks, particularly when reward values are derived from raw sensory data. Wulfmeier et al. [61] propose maximum entropy deep IRL, a generalization of MaxEntIRL that utilizes neural networks to model complex, nonlinear reward functions. In addition, instead of using pre-extracted features, the deep architecture is further extended to learn features through Convolution layers. This is an important step towards automating the learning process [12]. A further study by Fin et al. [62] proposes guided cost learning (GCL), which improves the sampling efficiency of [61] since [61] relies on a large number of expert transitions to estimate the reward function. GCL learns a nonlinear reward function in the inner loop of a policy optimization (in contrast to early IRL methods). This allows it to scale effectively to complex control problems. A further advantage of this method is that it utilizes the raw state of the system instead of predefined features to construct the reward function, which reduces the engineering burden.

Bayesian algorithms constitute the third category of IRL algorithms. Methods under this category use the expert’s actions as the evidence for updating the reward function estimate. A posterior distribution over candidate reward functions is derived from a prior distribution over the rewards and a likelihood of the reward hypothesis. Various models of likelihood have been proposed over the years. BIRL [63] is the earliest Bayesian IRL technique, based on a Boltzmann distribution for modeling the likelihood. A variety of distributions can be used as the prior over the reward functions [63]. For example, a Beta distribution is appropriate for planning problems with large rewards dichotomy. Analytically obtaining the posterior in the continuous space of reward functions is extremely difficult. To address this issue, [63] uses Markov chain Monte Carlo (MCMC) to derive a sample-based estimate of the posterior mean. Instead of computing the posterior mean, [64] computes the maximum a posteriori (MAP) reward function. [64] argues that the posterior mean is not the most suitable approach for reward inference as it integrates over the entire reward space, even those not consistent with the observed behavior, in the loss function. Levine et al. [65] propose a Bayesian IRL algorithm that uses a nonlinear function of features to represent the reward. They use a Gaussian process prior on reward values and a kernel function to determine the structure of the reward. The kernel’s hyperparameters are learned through the Bayesian GP framework, leading to the learning of the reward structure.

Classical BIRL algorithms are computationally intractable in complex high-dimensional environments since generating each posterior sample requires solving an entire MDP. This limitation has prevented these methods from scaling beyond small tabular settings. To overcome this limitation, [66] gen-

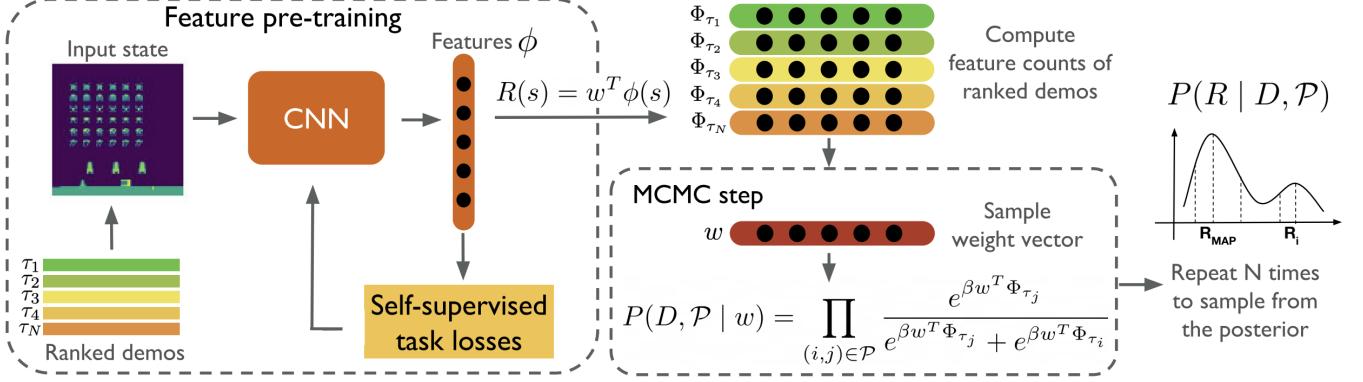


Fig. 4. A low-dimensional state feature embedding is pre-trained using ranked demonstrations [66]. A linear combination of learned features is used to derive the reward function. A pairwise ranking likelihood is used by MCMC proposal evaluations to estimate the likelihood of preferences over demonstrations given a proposal ( $w$ ). Utilizing the pre-computed embeddings of the ranked demonstrations makes MCMC sampling highly efficient; There is no need for data collection during inference or an MDP solver.

erates samples from the posterior distribution without using an MDP solver by proposing an alternative likelihood formulation that leverages preference labels over demonstrations (Fig. 4). Another approach to address the scalability issue is approximate variational reward IL (AVRIL) [67]. This approach simultaneously learns an imitating policy and an approximate posterior distribution over the reward in an offline setting. Unlike traditional sampling or MAP-based techniques, AVRIL relies on variational inference to estimate the posterior distribution accurately.

Most existing IRL algorithms rely on the unrealistic assumption that the transition model, and sometimes the expert's policy, are known beforehand [47], [50], [54], [65]. However, in real-world scenarios, the agent often has to estimate the expert policy and transition dynamics from samples, leading to errors in the recovered reward function [68], [69]. In their analysis, [69] breaks down this error into components from estimating the expert policy and transition model. Based on their analysis (in a tabular setting), they propose an efficient sampling strategy focused on transferring the learned reward function to a fully known target environment. It is assumed, however, that the agent can query the transition dynamics for arbitrary states and actions. To remove this assumption, active exploration for IRL (AceIRL) [68] focuses on developing an efficient exploration strategy. This strategy aims to explore both the environment dynamics and the expert policy such that an arbitrary IRL algorithm can infer the reward function as effectively as possible. By utilizing previous observations, AceIRL builds confidence intervals that capture feasible reward functions and find exploration policies that prioritize the most relevant regions of the environment.

#### IV. ADVERSARIAL IMITATION LEARNING

Scaling IRL algorithms to larger environments has been a major challenge despite their success in generating policies that replicate expert behavior [62], [70], [71]. This challenge arises due to the computational complexity of many IRL algorithms, which often require RL to be executed in an inner loop [46]. AIL offers a promising solution to the computational challenges of IRL by searching for the optimal

policy without fully solving an RL sub-problem at each iteration [46]. AIL involves a two-player game between an agent and an adversary (discriminator) where the adversary attempts to distinguish agent trajectories from expert trajectories [72]. The agent, on the other hand, endeavors to deceive the adversary by generating trajectories that closely resemble expert trajectories. Through this adversarial process, the agent gradually improves its imitation of the expert's behavior until it converges to a policy that closely resembles the expert's policy. AIL has demonstrated statistically significant improvements over existing methods in multiple benchmark environments, including robotics, autonomous driving, and game playing [46], [73], [74].

The effectiveness of AIL in addressing the limitations of IRL has spurred continued research in this area. The first AIL method that gained prominence is known as generative AIL (GAIL) [46]. In GAIL, the reward function measures the ability of the agent to imitate the expert's behavior. To do this, GAIL utilizes a discriminator network trained to distinguish between the expert's behavior and the agent's generated trajectories. The reward signal is then derived from the confusion of the discriminator, reflecting how difficult it is to tell whether a given trajectory is generated by the agent or the expert. By maximizing this reward signal, the agent is incentivized to generate trajectories that closely resemble the expert's behavior. Over the years, numerous improvements have been proposed to the original algorithm to improve its sample efficiency, scalability, and robustness [75], including changes to the discriminator's loss function [76] and switching from on-policy to off-policy agents [77].

In AIL, the objective is to enable the agent to generate trajectories that are similar to those of the expert. This involves the use of distance measures to quantify the similarity between the two. Different AIL methods employ various similarity measures to match the distribution over states and actions encountered by the agent with that of the expert [29]. For example, GAIL makes use of the Shannon-Jensen divergence, while some methods, such as AIRL [76], use the Kullback-Leibler divergence. However, recent research by Arjovsky et al. [78] has shown that replacing f-divergences with the

Wasserstein distance through its dual formulation can result in improved training stability, a technique that several AIL methods have implemented [77], [79]. Given these developments, exploring new similarity measures holds the potential to discover novel AIL methods.

Most AIL methods, just like GANs (generative adversarial networks) [80], use a min-max optimization approach to minimize the distance between the state-action distributions of the expert and agent, while maximizing a reward signal derived from the discriminator's confusion. However, this approach can be challenging to train due to issues such as vanishing gradients and convergence failure [28]. To overcome these challenges, methods such as primal wasserstein IL (PWIL) [29] have been developed, which approximates Wasserstein distances through a primal-dual approach.

## V. IMITATION FROM OBSERVATION

The prevailing paradigm in IL assumes that the learner has access to both states and actions demonstrated by an expert [81]. However, this often necessitates collecting data explicitly for IL purposes [81]. In robotics, for instance, the expert must teleoperate the robot or move its joints manually (kinesthetic learning) [82], and in gaming, the expert may require a special software stack. In both cases, considerable operator expertise is required, and useful demonstrations are limited to those recorded under artificial conditions. These limiting factors have motivated recent efforts in [83], where the expert's actions are unknown. In contrast to previous methods, imitation from observation is a more natural way to learn from experts and is more in tune with how humans and animals approach imitation in general. It is common for humans to learn new behaviors by observing other humans without being aware of their low-level actions (e.g., muscle commands). Humans learn a wide range of tasks, from weaving to swimming to playing games, by watching videos online. While there may be huge gaps in body shapes, sensing modalities, and timing, they show an incredible ability to apply the knowledge gained from online demonstrations [9].

Enabling agents to learn from demonstrations without the action information makes a large number of previously inapplicable resources, such as videos on the Internet, available for learning [84]. Additionally, it opens up the possibility of learning from agents with different embodiments whose actions are unknown or cannot be matched. The use of state-only demonstrations for IL is not new [85]. However, recent deep learning and visual recognition developments [86] have equipped researchers with more powerful tools to approach the problem, particularly when dealing with raw visual observations [48].

 Liu et al. [83] propose an imitation from observation method that learns an imitator policy from raw videos using context-aware translation. Their algorithm utilizes a context translation model that converts demonstrations from the expert's context (e.g., a third-person viewpoint) to the agent's context (e.g., a first-person viewpoint). The model is then used to predict the expert behavior in the context of the robot (Fig. 5). Using the predicted observations, a reward function is defined that is

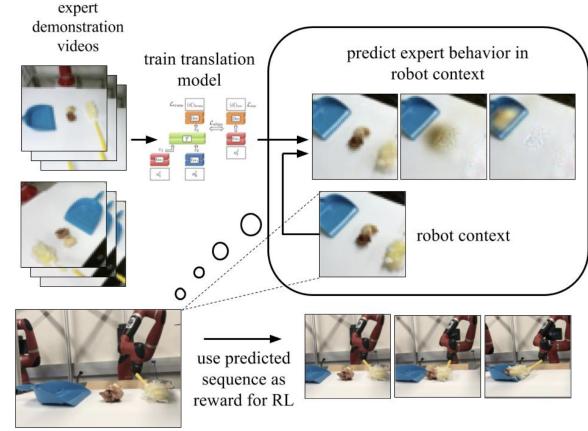


Fig. 5. A context translation model is trained on several videos of expert demonstrations [83]. The robot observes the context of the task it must perform during the learning process. The model then determines what an expert would do in the context of the robot.

made up of a penalty for deviating from the expert's translated features – encoded from input observations – and a penalty for encountering observations that are different from the translated observations. RL is then used to optimize the derived reward function. There are two drawbacks that limit the applicability of this method. First, it is assumed that demonstrations from different contexts are aligned in time which is rarely the case in the real world [87]. Second, learning the translation model requires a large number of demonstrations [83]. A further limitation is that it cannot address systematic domain shifts, such as differences in embodiment [83].

Sermanet et al. [88] introduce a self-supervised representation learning method using time-contrastive networks (TCN) that is invariant to different viewpoints and embodiments. TCN trains a neural network to learn an embedding of each video frame to extract features invariant to context differences, such as the camera angle. By using a triplet loss function, two frames occurring at the same time but with different modalities (i.e., viewpoints) are brought closer together in the embedding space while the frames from distant time-steps but with a visually similar frame are pushed apart (Fig. 6). In order to construct the reward function, Euclidean distance is calculated between the embedding of a demonstration and the embedding of an agent's camera images. RL techniques are used to optimize the reward function for learning imitation policies. A limitation of this technique is that it requires multi-viewpoint video for training, which is not readily available (e.g., over the Internet).

BC from observation (BCO) [89] aims to minimize the amount of post-demonstration environment interactions required for training RL algorithms of prior methods by taking a behavior cloning approach. BCO first learns an inverse dynamics model by letting an agent, who initially follows a random policy, interact with its environment and collect data [81]. After that, the model is used to infer the missing actions of the expert demonstrations. A BC algorithm is then used to map the states to the inferred actions and solve the problem as a regular IL problem. Using this approach, it is necessary to gather large amounts of data to learn the dynamics model

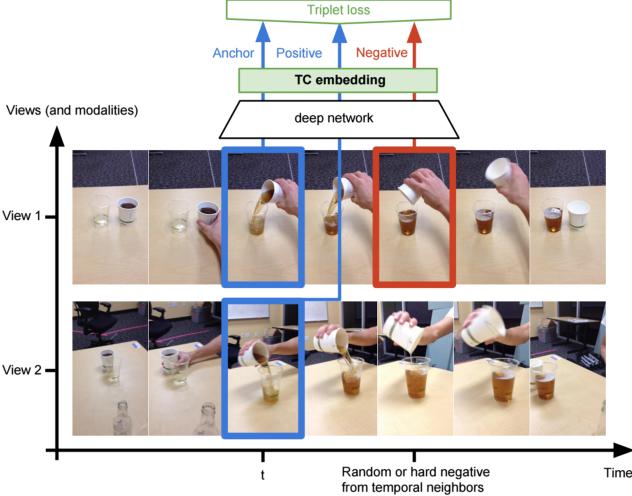


Fig. 6. The embedding space encourages co-occurring frames from different viewpoints to be in close proximity to each other, while images captured from the same viewpoint but at different times should be far apart [88].

online, especially in high-dimensional problems.

Subsequent work by [81] attempts to reduce the number of environment interactions required in BCO by learning a latent forward dynamics model in an offline manner [48]. The underlying assumption in this work is that predictable, though unknown, causes describe the classes of observed state transitions. The goal is to enable an agent to predict and imitate these latent causes [81]. To achieve this, a latent policy is learned, which estimates the probability that a latent action would be taken in an observed state. They then use a limited number of interactions with the environment to learn a mapping between the real-world actions the agent can take and latent actions identified by the model.

Generative adversarial imitation from observation (GAIfO) [84] adapts the GAIL objective to IfO by matching the expert and agent's state-transition distributions. By adopting an adversarial approach, this method can overcome the covariate shift problem encountered in the previous approaches [81], [89]. In addition, it is capable of handling demonstrations that are not time-aligned, unlike previous approaches. Using this approach is most successful when the expert and the agent operate in the same environment, under the same dynamics. However, it becomes more challenging to match state-transition distributions when dynamics differ since the expert's state transitions might not even be feasible in the agent's environment [90].

Jaegle et al. [91] introduce a non-adversarial IRL from observations approach using likelihood-based generative models. In this method, conditional state transition probabilities are matched between expert and learner. According to the authors' findings, their approach of matching conditional state transition probabilities tends to focus less on irrelevant differences between the expert and the learner settings than adversarial approaches such as GAIfO, which matches joint state-next-state probabilities. In particular, they argue that conditional state probabilities are less prone to erroneously penalize features that are not present in the demonstrations but lead to correct transitions.

Raychaudhuri et al. [87] propose a framework that learns state maps across the source and target domain from unpaired, unaligned demonstrations. This approach addresses embodiment, viewpoint, and dynamics mismatch. In order to preserve MDP dynamics during domain transformation, local and global alignments are performed. In local alignment, adversarial training is used to minimize divergence between the state-transition distributions of the true and transferred trajectories. Meanwhile, a learned temporal position function is used to enforce global alignment to ensure that states are placed in a consistent temporal position across the two domains. Finally, given the set of transferred demonstrations, BCO is used to learn the final policy. As with [83], this method relies on proxy tasks, i.e., expert demonstrations from both domains, which limits its application.

Aytar et al. [9] present a novel self-supervised framework for learning to play hard exploration Atari games without ever being explicitly exposed to the Atari environment by watching YouTube videos (Fig. 7). Learning from YouTube videos poses several challenges due to the existence of domain-specific variations (e.g., in color or resolution), and a lack of frame-by-frame alignment. To address these challenges, they first use self-supervised classification tasks, constructed over both vision and audio to map unaligned videos from multiple sources to a common representation. Then, a single YouTube video is embedded in this representation, and a sequence of checkpoints are placed along the embedding. This is used to create a reward function that encourages the agent to imitate human gameplay. During policy training, the agent is rewarded only when it reaches these checkpoints.

Brown et al. [97] introduce an IRL from observation technique for extrapolating the expert's intent from suboptimal ranked demonstrations. This work aims to improve the performance over a suboptimal expert in high-dimensional tasks by inferring the expert's intentions. They learn a state-based reward function such that a greater total return is assigned to higher-ranked trajectories. Utilizing ranking to construct a reward function in this way enables identifying features that are correlated with rankings, allowing for potentially better-than-demonstrator performance. Given the learned reward function, RL is used to optimize a policy.

Utilizing large amounts of navigation data from YouTube, [98] proposes a framework for learning scalable driving. First, a model is trained on a small labeled dataset to map monocular images to Bird's Eye View (BEV), facilitating learning from the unconstrained nature of YouTube videos (e.g. in viewpoints or camera parameters). Since many publicly available driving datasets include action labels, this assumption is reasonable. This trained model is used to generate pseudo-labels over a large unlabeled dataset. Lastly, a generalized policy is trained on the pseudo-labeled dataset and fine-tuned on the clean labels of the small labeled dataset.

## VI. CHALLENGES AND LIMITATIONS

### A. Imperfect Demonstrations

A common assumption in IL methods is that the demonstrations will be optimal, performed by an expert demonstrator

TABLE I  
SUMMARY OF EXISTING RESEARCH ON IMITATION LEARNING

Ref	Datasets	Inputs	Learning Type	Online/ Offline	Online Expert	Application
[93]	Sim	State, Image	Interactive IL	Online	Yes	Robotic Locomotion, Dependency Parsing
[17]	Sim	State, Image	Regularized BC	Online	No	Car Racing, Atari Games, Locomotion Control Tasks
[14]	Sim	State, Image	BC\DAgger	Online	Yes	Games, Handwriting Recognition
[21]	Sim, Real	State	BC\HG-DAgger	Online	Yes	Autonomous Driving
[22]	Sim	State	BC\ Human-gated Interactive IL	Online	Yes	Robotic Manipulation
[19]	Sim	State	Human-in-the-loop RL	Online	Yes	Autonomous Driving
[23]	Sim	Image	BC\SafeDagger	Online	Yes	Autonomous Driving
[24]	Sim, Real	State, Image	BC\LazyDagger	Online	Yes	Robotic Locomotion, Fabric Manipulation
[25]	Sim	State	BC\EnsembleDagger	Online	Yes	Inverted Pendulum, Locomotion
[20]	Sim, Real	State, Image	BC\ThriftyDagger	Online	Yes	Peg Insertion, Cable Routing
[26]	Sim	State	IL via Expert Support Estimation	Online	No	Robotic Locomotion, Autonomous Driving
[27]	Sim	State, Image	IL via Expert Support Estimation	Online	No	Atari Games, Continuous Control Tasks
[31]	Sim, Real	State	IL via Expert Support Estimation	Online	No	Autonomous Driving
[32]	Sim	State	BC	Online	No	Robotic Manipulation
[33]	Sim	State	Offline Imitation Learning	Offline	No	Robotic Locomotion
[35]	Sim	State	Causal Graph-Parameterized Policy Learning	Online	Yes	Atari Games, Robotic Locomotion
[36]	Sim	State	BC	Offline	No	Robotic Locomotion
[34]	Sim	State, Image	BC	Offline	No	Robotic Locomotion, Autonomous Driving
[38]	Sim, Real	State, Image	Implicit BC	Offline	No	Robotic Manipulation
[94]	Sim, Real	State	Maximum Entropy IRL	Online	No	Path Planning, Gridworld
[45]	Sim	State	Feature Based IRL	Online	No	Autonomous Driving, Gridworld
[47]	Real	State	Maximum Entropy IRL	Online	No	Predicting Driving Behavior, Route Recommendation
[54]	Real	State	Maximum-margin IRL	Online	No	Route Planning
[55]	Sim, Real	State, Image	Maximum-margin IRL\ MMPBOOST	Online	No	Path Planning, Legged Locomotion, Driving Obstacle Detection/Avoidance
[56]	Sim, Real	State, Image	Maximum-margin IRL	Online	No	Footstep Prediction, Grasp Prediction, Navigation Task
[57]	Sim	State	Maximum-margin IRL	Online	No	Car Driving Game
[58]	Sim	State	Maximum Entropy IRL	Online	No	2-D Point Mass Control System
[59]	Real	State	Maximum Entropy IRL	Online	No	Robotic Manipulation
[60]	Sim	State	Relative Entropy IRL	Online	No	Car Racing, Gridworld, Game
[61]	Sim	State	Maximum Entropy Deep IRL	Online	No	Objectworld, Binaryworld
[62]	Sim, Real	State	Maximum Entropy IRL	Online	No	Robotic Manipulation
[63]	Sim	State	Bayesian IRL	Online	No	Random Generated MDPs
[64]	Sim	State	Bayesian IRL\ MAP Inference	Online	No	Gridworld, Simplified Car Racing

TABLE I  
SUMMARY OF EXISTING RESEARCH ON IMITATION LEARNING (CONTINUED)

[65]	Sim	State	Nonlinear Bayesian IRL	Online	No	Objectworld, Highway Driving
[66]	Sim	Image	Bayesian IRL	Online	No	Atari Games
[67]	Sim, Real	State	Bayesian IRL	Offline	No	Medical Information Dataset, Physics-based Control
[69]	Sim	Image	IRL	Online	Yes	Gridworld, Random Generated MDPs, Chain MDP
[68]	Sim	State	IRL\Active Exploration	Online	No	Random MDP, Gridworld, Chain MDP, Double Chain
[46]	Sim	State	Generative Adversarial IL	Online	No	Physics-based Control, Robotic Locomotion
[76]	Sim	State	Adversarial IRL	Online	No	Random Generated MDPs, Continuous Control Tasks
[77]	Sim	State	Adversarial IRL	Online	No	Robotic Locomotion and Manipulation
[95]	Sim	State	Bayes-GAIL	Online	No	Robotic Locomotion
[96]	Sim	State	IRL	Online	No	Robotic Locomotion
[29]	Sim	State, Image	Adversarial IRL	Online	No	Robotic Locomotion and Hand Manipulation
[79]	Sim	State, Image	Generative Adversarial IL\InfoGAIL	Online	No	Synthetic 2D Example, Autonomous Highway Driving
[83]	Sim, Real	Image	Imitation from Observation	Online	No	Robotic Manipulation
[88]	Sim, Real	Image	IfO\Self-supervised Learning	Online	No	Robotic Manipulation, Human Pose Imitation
[89]	Sim	State	BC from Observation	Online	No	Physics-based Control, Robotic Locomotion
[81]	Sim	State, Image	Imitation from Observation	Online	No	Physics-based Control, CoinRun Game
[84]	Sim	State, Image	Generative Adversarial Imitation from Observation	Online	No	Physics-based Control, Robotic Locomotion
[90]	Sim	State	Imitation from Observation	Online	No	Robotic Locomotion
[91]	Sim	State	IRL from Observations	Online	No	Robotic Locomotion
[87]	Sim	State	Cross-domain IL from Observations	Offline	No	Physics-based Control, Robotic Locomotion
[9]	Sim	Image	Imitation from Observation	Online	No	Atari Games
[97]	Sim	State, Image	IRL from Observations	Online	No	Atari Games, Robotic Locomotion
[98]	Sim, Real	Image	Conditional IL from Observations	Offline	No	Autonomous Driving
[99]	Sim	State	Importance Weighting IL, GAIL	Online	No	Robotic Locomotion
[100]	Sim	State	BC from Noisy Demonstrations	Offline	No	Robotic Locomotion
[101]	Sim	State, Image	Weighted GAIL	Online	No	Atari Games, Robotic Locomotion
[102]	Sim	State	Weighted BC	Offline	No	Robotic Locomotion
[103]	Sim	State	BC	Offline	No	MiniGrid Environments, Robotic Manipulation, Chess Game-endings
[104]	Sim	State	GAIL with Modified Objective	Online	No	Robotic Locomotion
[105]	Sim	State	GAIL	Online	No	Physics-based Control
[106]	Sim, Real	Image	Cross-embodiment IRL	Online	No	Robotic Manipulation
[107]	Sim	State	IL via Optimal Transport	Online	No	Physics-based Control, Robotic Locomotion, 2D Maze Navigation
[108]	Sim	State	BC	Offline	No	Robotic Manipulation, Physics-based Control

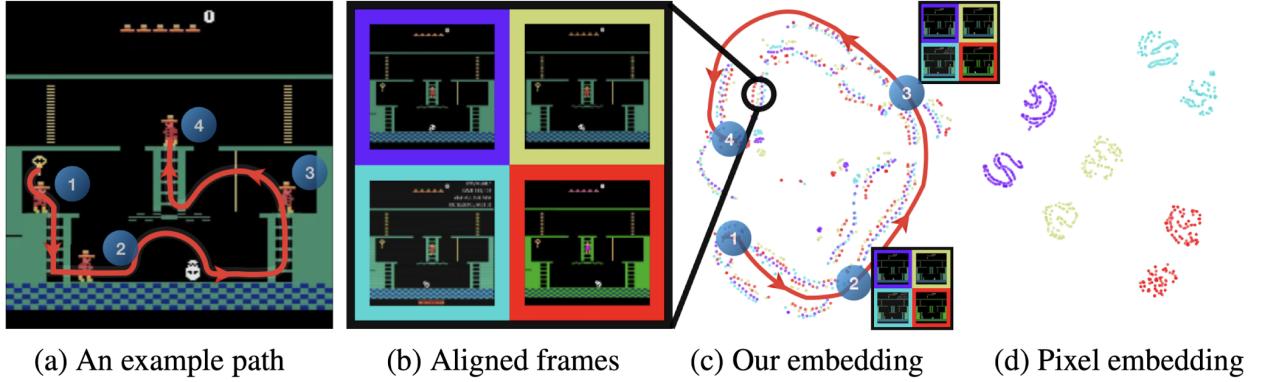


Fig. 7. For the path in (a), t-SNE projections [92] of trajectories using the proposed embedding (c) and raw pixels (d) are shown [9]. In b, an example frame of MONTEZUMA'S REVENGE is compared using four different domains: the Arcade Learning Environment, and three YouTube videos. Based on the embedding space, it can be seen that the four trajectories are well-aligned.

[2]. However, this assumption is too restrictive when it comes to learning from demonstrations in a variety of cases [2]. Firstly, it can be difficult to obtain large numbers of high-quality demonstrations from human experts [109], [110]. In many real-world tasks, this would be impossible for humans due to the amount of time and effort required. Additionally, humans are prone to making mistakes for various reasons, such as the presence of distractions, or limited observability of the environment [99], [100]. Secondly, it is necessary to leverage the scale and diversity of crowd-sourced datasets to learn robust effective IL policies [111]. However, a crowd-sourced dataset will inevitably have a wide range of behavior optimality since it is collected from users with varying levels of expertise.

The naive solution to imperfect demonstrations would be to discard the non-optimal ones. However, this screening process is often impractical since it requires significant human effort [100]. Therefore, researchers have been increasingly interested in developing methods that can learn from imperfect demonstrations.

Wu et al. [99] present two general approaches to address imperfect demonstrations by utilizing both confidence-scored and unlabeled data: two-step importance weighting IL (2IWIL) and generative adversarial IL with imperfect demonstration and confidence (IC-GAIL). Both approaches assume that a fraction of demonstrations are annotated with confidence scores (i.e. the probability that a given trajectory is optimal). 2IWIL is a two-step approach that first uses a semi-supervised classifier to generate confidence scores for the unlabeled demonstrations, and then performs standard GAIL with reweighted distribution [102]. To avoid error accumulation in two steps, IC-GAIL forgoes learning a classifier and performs occupancy measure matching with unlabeled demonstrations.

Sasaki et al. [100] propose an offline BC algorithm to learn from noisy demonstrations, obtained from a noisy expert, without any screening or annotations associated with the non-optimal demonstrations. The key idea is to leverage the learned policy to reweight the samples in the next iteration of weighted BC. The noisy expert action distribution is assumed to be a weighted mixture of two distributions: the action distribution of an optimal expert and a non-optimal one. The goal is to

change the weights so that the noisy expert action distribution mode gets closer to the optimal expert action distribution mode. This is achieved by reusing the old policy (i.e. the policy optimized in the previous iteration) as the weights for action samples in the weighted BC objective. However, this approach only converges to the optimal policy when optimal demonstrations constitute the majority of the data.

Wang et al. [101] investigate how to weight imperfect demonstrations in GAIL without requiring auxiliary information from an oracle. An automatic weight prediction method is proposed to assess the quality and significance of each demonstration for training. They demonstrate that the weight can be accurately estimated using both the discriminator and the agent policy in GAIL. In the training procedure, the weight estimation is conducted first to determine weight for each demonstration. Using weighted GAIL, the agent policy is then trained with weighted demonstrations. These two procedures interact alternately and are optimized as a whole.

Kim et al. [102] aim to overcome the distributional shift problem caused by the lack of sufficient expert demonstrations by using supplementary imperfect demonstrations with unknown optimality levels. They regularize a distribution-matching objective of IL by a KL divergence between the agent distribution and a mixture of expert and imperfect distributions. An optimal state-action distribution of this regularized objective is obtained using a dual-program technique [112]. Given the optimal state-action distribution, the expert policy is extracted by performing weighted BC.

Beliaev et al. [103] introduce IL by Estimating Expertise of Demonstrators (ILEED). It leverages information about demonstrators' identities to infer their expertise levels in an unsupervised manner. Each demonstrator is assigned a state-dependent expertise value, which indicates which demonstrators perform better in specific states, allowing them to combine their strengths in different states. ILEED develops and optimizes a joint model over a learned policy and expertise levels. As a result, the model is able to learn from the optimal behavior of each demonstrator and filter out the suboptimal behavior. Expertise levels are modeled by the inner product of two embeddings: an state embedding, and a demonstrator embedding. Each dimension of the embeddings vector corre-

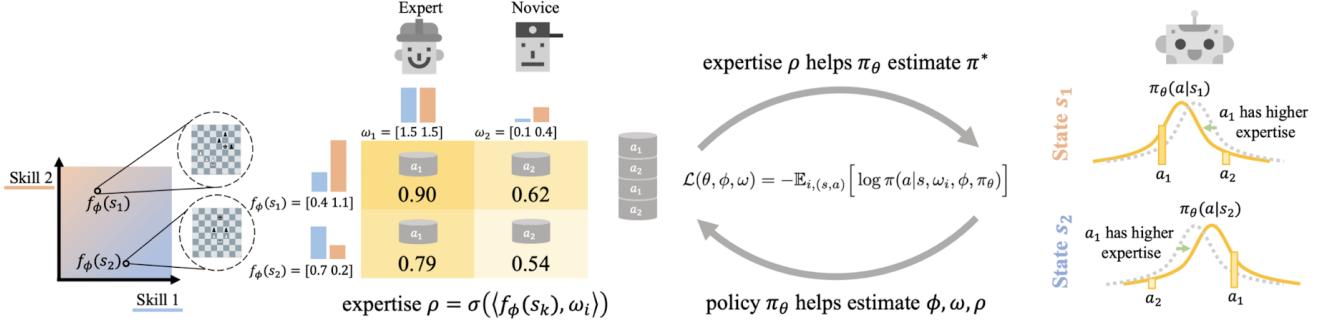


Fig. 8. IL by estimating expertise of demonstrators [103]. Left: Skills associated with states are encoded by state embeddings. Middle: Demonstrators' expertise,  $\rho$ , at a particular state is determined by the state embedding and the demonstrator embedding. Right: By utilizing the expertise levels, the model improves the learned policy, thereby improving the estimation of the state/demonstrator embeddings and the expertise level.

sponds to a latent skill, with the state embedding having a weighting of how relevant that skill is in acting correctly at that state and the demonstrator embedding representing how adept the demonstrator is at that skill (Fig. 8).

### B. Domain Discrepancies

The majority of prior research assumes that the expert and the agent operate under the same state and action space [107]. This assumption makes it easier to manually specify one-to-one correspondences between the actions of the expert and the agent. However, this will restrict the application of these algorithms to simple scenarios in which expert demonstrations come from the agent domain. In recent years, there has been increased interest in IL under a more relaxed and realistic assumption: given demonstrations of a task in the expert domain, train the agent to learn to perform the task optimally in its own domain [107]. This relaxed setting facilitates the collection of demonstrations by removing the requirement for in-domain expert demonstrations, improving the efficiency and scalability of IL. A variety of solutions have been proposed to address three main types of domain discrepancies: dynamics [90], [104], viewpoint [88], [105], and embodiment mismatch [106], [107], [113].

The transfer of knowledge between different domains in IL research often involves learning a mapping between the state-action spaces. Recent works [83], [88] utilize paired and time-aligned demonstrations from both domains (expert and agent) to learn a state mapping or encoding to a domain invariant feature space. Following this, they perform an RL step to learn the final policy on the given task. These studies are limited in their application due to the limited availability of paired demonstrations and the high cost of RL procedures [108]. To overcome these limitations, Kim et al. [108] propose a general framework to learn state and action maps from unpaired and unaligned demonstrations while having access to an online expert. In addition, they eliminate the need for an expensive RL step by leveraging the action map to perform zero-shot imitation. The work of [87] extends [108] to the imitation form observation setting and also eliminates the need for an online expert in [108]. All these methods rely on proxy tasks, which limits their applicability in real-world scenarios. Stadie

et al. [105] propose an adversarial framework for viewpoint-agnostic imitation that uses a discriminator to distinguish data coming from different viewpoints and maximizes domain confusion without proxy tasks. Zakka et al. [106] adopt a goal-driven approach that focuses on imitating task progress rather than matching fine-grained structural details. Several of these approaches have already been discussed in previous sections. The following is a detailed discussion of a few of the most recent methods.

Chae et al. [104] provide a framework for learning policies that can perform well under perturbed environment dynamics. The objective is to train a policy that is robust to continuous dynamics variation, using only a few samples from the continuum of environment dynamics. The sampled environments are used during both the demonstration collection phase and the policy interaction phase (Fig. 9). The problem is then formulated as the minimization of the weighted average of Jensen-Shannon divergences between the multiple expert policies and the agent policy.

Cross-embodiment IRL (XIRL) [106] attempts to extract an agent-invariant definition of a task from videos of different agents executing the same task differently due to embodiment differences. XIRL uses temporal cycle consistency (TCC) to learn visual embeddings, which identify key moments in videos of varying lengths and cluster them to encode task progression. In order to learn an embodiment-invariant reward function, XIRL uses the distance from a single goal state in the TCC embedding space. This method can be applied to any number of embodiments or experts (regardless of their skill level) since it does not require the manual pairing of video frames between the expert and the learner.

Fickinger et al. [107] examine how expert demonstrations can be used to train an imitator agent with a different embodiment without relying on explicit cross-domain latent space [106] or resorting to any form of proxy tasks [83], [87], [88]. Instead, they use the Gromov-Wasserstein distance between state-action occupancies of the expert and the agent to find isometric transformations that preserve distance measures between the two domains. Given trajectories from the expert and the agent domain, pseudo-rewards are computed based on the degree to which distances from a state to its neighbors in the agent domain are preserved in the expert domain. Using

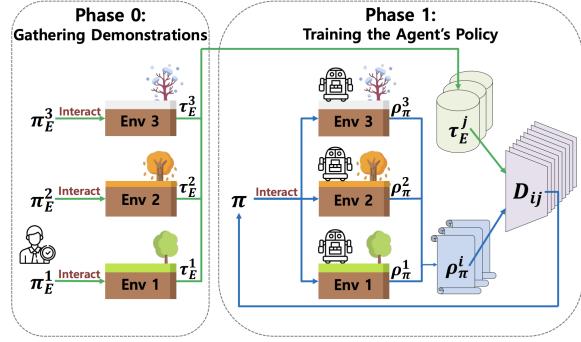


Fig. 9. IL against variations in environment dynamics [104]. Blue represents the flow of policy samples, while green represents the flow of expert demonstrations.

these pseudo-rewards, the policy is optimized using an RL algorithm.

## VII. OPPORTUNITIES AND FUTURE WORK

This survey paper provides a comprehensive overview of the field of IL, exploring its algorithms, categorizations, developments, and challenges. The paper starts by presenting a categorization of IL algorithms, identifying two general learning approaches, namely BC and IRL, and discussing their relative benefits and limitations. Additionally, the paper highlights the benefits of integrating adversarial training into IL and evaluates the current progress in the AIL field. The paper also introduces a novel technique called IfO that aims to learn from state-only demonstrations.

Through the examination of various IL algorithms, we have gained valuable insights into their strengths and limitations and identified some of the key challenges and opportunities for future research. One of the significant challenges across all categories of IL approaches is the need to collect diverse and large-scale demonstrations, which is crucial for training a generalizable policy that can be applied in the real world [111]. However, this poses a challenge, as readily available demonstration resources such as online videos present additional difficulties such as the varying levels of expertise among the demonstrators.

Another challenge in IL research is developing methods that enable agents to learn across domains with differences in dynamics, viewpoint, and embodiment. Overcoming these challenges is essential if we are to teach agents to learn from experts effectively and apply the insights from IL research to real-world scenarios. Therefore, future research should focus on developing algorithms that can learn from imperfect demonstrations, extract useful information, and enable cross-domain learning. Despite these challenges, the field of IL presents exciting opportunities for future research. As the field of AI continues to evolve and mature, we believe that IL will play a critical role in enabling agents to learn from demonstrations, adapt to new tasks and environments, and ultimately achieve more advanced levels of intelligence, paving the way for real-world applications of AI.

## ACKNOWLEDGMENTS

This research was partially supported by the Australian Research Council's Discovery Projects funding scheme (project DP190102181 and DP210101465).

## REFERENCES

- [1] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, J. Peters *et al.*, “An algorithmic perspective on imitation learning,” *Foundations and Trends® in Robotics*, vol. 7, no. 1-2, pp. 1–179, 2018.
- [2] 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, pp. 297–330, 2020.
- [3] S. Schaal, “Is imitation learning the route to humanoid robots?” *Trends in cognitive sciences*, vol. 3, no. 6, pp. 233–242, 1999.
- [4] M. Deng, Z. Li, Y. Kang, C. P. Chen, and X. Chu, “A learning-based hierarchical control scheme for an exoskeleton robot in human–robot cooperative manipulation,” *IEEE transactions on cybernetics*, vol. 50, no. 1, pp. 112–125, 2018.
- [5] Z. Zhu and H. Hu, “Robot learning from demonstration in robotic assembly: A survey,” *Robotics*, vol. 7, no. 2, p. 17, 2018.
- [6] J. Van Den Berg, S. Miller, D. Duckworth, H. Hu, A. Wan, X.-Y. Fu, K. Goldberg, and P. Abbeel, “Superhuman performance of surgical tasks by robots using iterative learning from human-guided demonstrations,” in *2010 IEEE International Conference on Robotics and Automation*. IEEE, 2010, pp. 2074–2081.
- [7] L. Le Mero, D. Yi, M. Dianati, and A. Mouzakitis, “A survey on imitation learning techniques for end-to-end autonomous vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, 2022.
- [8] P. M. Kebria, A. Khosravi, S. M. Salaken, and S. Nahavandi, “Deep imitation learning for autonomous vehicles based on convolutional neural networks,” *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 82–95, 2020.
- [9] Y. Aytar, T. Pfaff, D. Budden, T. Paine, Z. Wang, and N. De Freitas, “Playing hard exploration games by watching youtube,” *Advances in neural information processing systems*, vol. 31, 2018.
- [10] U. E. Ogenyi, J. Liu, C. Yang, Z. Ju, and H. Liu, “Physical human–robot collaboration: Robotic systems, learning methods, collaborative strategies, sensors, and actuators,” *IEEE transactions on cybernetics*, vol. 51, no. 4, pp. 1888–1901, 2019.
- [11] S. Sun, Z. Cao, H. Zhu, and J. Zhao, “A survey of optimization methods from a machine learning perspective,” *IEEE transactions on cybernetics*, vol. 50, no. 8, pp. 3668–3681, 2019.
- [12] 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.
- [13] D. A. Pomerleau, “Efficient training of artificial neural networks for autonomous navigation,” *Neural computation*, vol. 3, no. 1, pp. 88–97, 1991.
- [14] S. Ross, G. Gordon, and D. Bagnell, “A reduction of imitation learning and structured prediction to no-regret online learning,” in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 627–635.
- [15] D. A. Pomerleau, “Alvinn: An autonomous land vehicle in a neural network,” *Advances in neural information processing systems*, vol. 1, 1988.
- [16] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, “Domain generalization: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
- [17] S. Reddy, A. D. Dragan, and S. Levine, “Sqil: Imitation learning via reinforcement learning with sparse rewards,” in *International Conference on Learning Representations*, 2020.
- [18] J. Roche, V. De-Silva, and A. Kondoz, “A multimodal perception-driven self evolving autonomous ground vehicle,” *IEEE Transactions on Cybernetics*, 2021.
- [19] Q. Li, Z. Peng, and B. Zhou, “Efficient learning of safe driving policy via human–ai copilot optimization,” in *International Conference on Learning Representations*, 2022.
- [20] R. Hoque, A. Balakrishna, E. Novoseller, A. Wilcox, D. S. Brown, and K. Goldberg, “Thriftydagger: Budget-aware novelty and risk gating for interactive imitation learning,” in *Proceedings of the 5th Conference on Robot Learning*, vol. 164. PMLR, 2021, pp. 598–608.

- [21] M. Kelly, C. Sidrane, K. Driggs-Campbell, and M. J. Kochenderfer, “Hg-dagger: Interactive imitation learning with human experts,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8077–8083.
- [22] A. Mandlekar, D. Xu, R. Martín-Martín, Y. Zhu, L. Fei-Fei, and S. Savarese, “Human-in-the-loop imitation learning using remote tele-operation,” *arXiv preprint arXiv:2012.06733*, 2020.
- [23] J. Zhang and K. Cho, “Query-efficient imitation learning for end-to-end simulated driving,” in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, ser. AAAI’17. AAAI Press, 2017, p. 2891–2897.
- [24] R. Hoque, A. Balakrishna, C. Puterman, M. Luo, D. S. Brown, D. Seita, B. Thamanjeyan, E. Novoseller, and K. Goldberg, “Lazydagger: Reducing context switching in interactive imitation learning,” in *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2021, pp. 502–509.
- [25] K. Menda, K. Driggs-Campbell, and M. J. Kochenderfer, “Ensembledagger: A bayesian approach to safe imitation learning,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 5041–5048.
- [26] R. Wang, C. Ciliberto, P. V. Amadori, and Y. Demiris, “Random expert distillation: Imitation learning via expert policy support estimation,” in *International Conference on Machine Learning*. PMLR, 2019, pp. 6536–6544.
- [27] K. Brantley, W. Sun, and M. Henaff, “Disagreement-regularized imitation learning,” in *International Conference on Learning Representations*, 2020.
- [28] M. Arjovsky and L. Bottou, “Towards principled methods for training generative adversarial networks,” *arXiv preprint arXiv:1701.04862*, 2017.
- [29] R. Dadashi, L. Hussenot, M. Geist, and O. Pietquin, “Primal wasserstein imitation learning,” in *International conference on learning representations*, 2021.
- [30] K. Brantley, “Expert-in-the-loop for sequential decisions and predictions,” Ph.D. dissertation, 2021.
- [31] M. Bansal, A. Krizhevsky, and A. S. Ogale, “Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst,” in *Robotics: Science and Systems XV*, 2019.
- [32] J. Wong, A. Tung, A. Kurenkov, A. Mandlekar, L. Fei-Fei, S. Savarese, and R. Martín-Martín, “Error-aware imitation learning from teleoperation data for mobile manipulation,” in *Conference on Robot Learning*. PMLR, 2021, pp. 1367–1378.
- [33] J. Chang, M. Uehara, D. Sreenivas, R. Kidambi, and W. Sun, “Mitigating covariate shift in imitation learning via offline data with partial coverage,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 965–979, 2021.
- [34] C.-C. Chuang, D. Yang, C. Wen, and Y. Gao, “Resolving copycat problems in visual imitation learning via residual action prediction,” *arXiv preprint arXiv:2207.09705*, 2022.
- [35] P. De Haan, D. Jayaraman, and S. Levine, “Causal confusion in imitation learning,” *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [36] C. Wen, J. Lin, T. Darrell, D. Jayaraman, and Y. Gao, “Fighting copycat agents in behavioral cloning from observation histories,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 2564–2575, 2020.
- [37] P. Florence and C. Lynch, “Decisiveness in Imitation Learning for Robots,” <https://ai.googleblog.com/2021/11/decisiveness-in-imitation-learning-for.html>, 2022.
- [38] 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, 2021, pp. 158–168.
- [39] Y. Song and D. P. Kingma, “How to train your energy-based models,” *arXiv preprint arXiv:2101.03288*, 2021.
- [40] S. Russell, “Learning agents for uncertain environments,” in *Proceedings of the eleventh annual conference on Computational learning theory*, 1998, pp. 101–103.
- [41] B. Piot, M. Geist, and O. Pietquin, “Bridging the gap between imitation learning and inverse reinforcement learning,” *IEEE transactions on neural networks and learning systems*, vol. 28, no. 8, pp. 1814–1826, 2016.
- [42] B. Lian, W. Xue, F. L. Lewis, and T. Chai, “Robust inverse q-learning for continuous-time linear systems in adversarial environments,” *IEEE Transactions on Cybernetics*, 2021.
- [43] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
- [44] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, “Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications,” *IEEE transactions on cybernetics*, vol. 50, no. 9, pp. 3826–3839, 2020.
- [45] P. Abbeel and A. Y. Ng, “Apprenticeship learning via inverse reinforcement learning,” in *Proceedings of the twenty-first international conference on Machine learning*, 2004, p. 1.
- [46] J. Ho and S. Ermon, “Generative adversarial imitation learning,” *Advances in neural information processing systems*, vol. 29, 2016.
- [47] B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey *et al.*, “Maximum entropy inverse reinforcement learning,” in *Aaaai*, vol. 8. Chicago, IL, USA, 2008, pp. 1433–1438.
- [48] F. Torabi, G. Warnell, and P. Stone, “Recent advances in imitation learning from observation,” *arXiv preprint arXiv:1905.13566*, 2019.
- [49] M. Kuderer, S. Gulati, and W. Burgard, “Learning driving styles for autonomous vehicles from demonstration,” in *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 2641–2646.
- [50] A. Y. Ng, S. Russell *et al.*, “Algorithms for inverse reinforcement learning.” in *Icmi*, vol. 1, 2000, p. 2.
- [51] D. Hadfield-Menell, S. J. Russell, P. Abbeel, and A. Dragan, “Cooperative inverse reinforcement learning,” *Advances in neural information processing systems*, vol. 29, 2016.
- [52] K. Kim, S. Garg, K. Shiragur, and S. Ermon, “Reward identification in inverse reinforcement learning,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 5496–5505.
- [53] D. Jarrett, A. Hüyük, and M. Van Der Schaar, “Inverse decision modeling: Learning interpretable representations of behavior,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 4755–4771.
- [54] N. D. Ratliff, J. A. Bagnell, and M. A. Zinkevich, “Maximum margin planning,” in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 729–736.
- [55] J. Bagnell, J. Chestnutt, D. Bradley, and N. Ratliff, “Boosting structured prediction for imitation learning,” *Advances in Neural Information Processing Systems*, vol. 19, 2006.
- [56] N. D. Ratliff, D. Silver, and J. A. Bagnell, “Learning to search: Functional gradient techniques for imitation learning,” *Autonomous Robots*, vol. 27, no. 1, pp. 25–53, 2009.
- [57] U. Syed and R. E. Schapire, “A game-theoretic approach to apprenticeship learning,” *Advances in neural information processing systems*, vol. 20, 2007.
- [58] N. Aghasadeghi and T. Bretl, “Maximum entropy inverse reinforcement learning in continuous state spaces with path integrals,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 1561–1566.
- [59] M. Kalakrishnan, P. Pastor, L. Righetti, and S. Schaal, “Learning objective functions for manipulation,” in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 1331–1336.
- [60] A. Bouliarias, J. Kober, and J. Peters, “Relative entropy inverse reinforcement learning,” in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 182–189.
- [61] M. Wulfmeier, P. Ondruska, and I. Posner, “Maximum entropy deep inverse reinforcement learning,” *arXiv preprint arXiv:1507.04888*, 2015.
- [62] C. Finn, S. Levine, and P. Abbeel, “Guided cost learning: Deep inverse optimal control via policy optimization,” in *International conference on machine learning*. PMLR, 2016, pp. 49–58.
- [63] D. Ramachandran and E. Amir, “Bayesian inverse reinforcement learning,” in *IJCAI*, vol. 7, 2007, pp. 2586–2591.
- [64] J. Choi and K.-E. Kim, “Map inference for bayesian inverse reinforcement learning,” *Advances in Neural Information Processing Systems*, vol. 24, 2011.
- [65] S. Levine, Z. Popovic, and V. Koltun, “Nonlinear inverse reinforcement learning with gaussian processes,” *Advances in neural information processing systems*, vol. 24, 2011.
- [66] D. Brown, R. Coleman, R. Srinivasan, and S. Niekum, “Safe imitation learning via fast bayesian reward inference from preferences,” in *International Conference on Machine Learning*. PMLR, 2020, pp. 1165–1177.
- [67] A. J. Chan and M. van der Schaar, “Scalable bayesian inverse reinforcement learning,” in *International Conference on Learning Representations*, 2021.
- [68] D. Lindner, A. Krause, and G. Ramponi, “Active exploration for inverse reinforcement learning,” *arXiv preprint arXiv:2207.08645*, 2022.

- [69] A. M. Metelli, G. Ramponi, A. Concetti, and M. Restelli, “Provably efficient learning of transferable rewards,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 7665–7676.
- [70] J. Ho, J. Gupta, and S. Ermon, “Model-free imitation learning with policy optimization,” in *International Conference on Machine Learning*. PMLR, 2016, pp. 2760–2769.
- [71] S. Levine and V. Koltun, “Continuous inverse optimal control with locally optimal examples,” *arXiv preprint arXiv:1206.4617*, 2012.
- [72] A. Deka, C. Liu, and K. P. Sycara, “Arc-actor residual critic for adversarial imitation learning,” in *Conference on Robot Learning*. PMLR, 2023, pp. 1446–1456.
- [73] R. Bhattacharyya, B. Wulfe, D. J. Phillips, A. Kuebler, J. Morton, R. Senanayake, and M. J. Kochenderfer, “Modeling human driving behavior through generative adversarial imitation learning,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 3, pp. 2874–2887, 2023.
- [74] J. Song, H. Ren, D. Sadigh, and S. Ermon, “Multi-agent generative adversarial imitation learning,” *Advances in neural information processing systems*, vol. 31, 2018.
- [75] M. Orsini, A. Raichuk, L. Husenot, D. Vincent, R. Dadashi, S. Girgin, M. Geist, O. Bachem, O. Pietquin, and M. Andrychowicz, “What matters for adversarial imitation learning?” *Advances in Neural Information Processing Systems*, vol. 34, pp. 14 656–14 668, 2021.
- [76] J. Fu, K. Luo, and S. Levine, “Learning robust rewards with adversarial inverse reinforcement learning,” in *International Conference on Learning Representations*, 2018.
- [77] I. Kostrikov, K. K. Agrawal, D. Dwibedi, S. Levine, and J. Tompson, “Discriminator-actor-critic: Addressing sample inefficiency and reward bias in adversarial imitation learning,” in *International conference on learning representations*, 2019.
- [78] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein generative adversarial networks,” in *International conference on machine learning*. PMLR, 2017, pp. 214–223.
- [79] Y. Li, J. Song, and S. Ermon, “Infogail: Interpretable imitation learning from visual demonstrations,” *Advances in Neural Information Processing Systems*, vol. 30, 2017.
- [80] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” *Advances in neural information processing systems*, vol. 27, 2014.
- [81] A. Edwards, H. Sahni, Y. Schroecker, and C. Isbell, “Imitating latent policies from observation,” in *International conference on machine learning*. PMLR, 2019, pp. 1755–1763.
- [82] Y. Hu, G. Chen, Z. Li, and A. Knoll, “Robot policy improvement with natural evolution strategies for stable nonlinear dynamical system,” *IEEE Transactions on Cybernetics*, 2022.
- [83] Y. Liu, A. Gupta, P. Abbeel, and S. Levine, “Imitation from observation: Learning to imitate behaviors from raw video via context translation,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1118–1125.
- [84] F. Torabi, G. Warnell, and P. Stone, “Generative adversarial imitation from observation,” in *Imitation, Intent, and Interaction (I3) Workshop at ICML 2019*, June 2019.
- [85] A. J. Ijspeert, J. Nakanishi, and S. Schaal, “Movement imitation with nonlinear dynamical systems in humanoid robots,” in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, vol. 2. IEEE, 2002, pp. 1398–1403.
- [86] S. Choe, H. Seong, and E. Kim, “Indoor place category recognition for a cleaning robot by fusing a probabilistic approach and deep learning,” *IEEE Transactions on Cybernetics*, 2021.
- [87] D. S. Raychaudhuri, S. Paul, J. Vanbaar, and A. K. Roy-Chowdhury, “Cross-domain imitation from observations,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 8902–8912.
- [88] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain, “Time-contrastive networks: Self-supervised learning from video,” in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 1134–1141.
- [89] F. Torabi, G. Warnell, and P. Stone, “Behavioral cloning from observation,” in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, ser. IJCAI’18. AAAI Press, 2018, p. 4950–4957.
- [90] T. Gangwani, Y. Zhou, and J. Peng, “Imitation learning from observations under transition model disparity,” in *International Conference on Learning Representations*, 2022.
- [91] A. Jaegle, Y. Sulsky, A. Ahuja, J. Bruce, R. Fergus, and G. Wayne, “Imitation by predicting observations,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 4665–4676.
- [92] L. Van der Maaten and G. Hinton, “Visualizing data using t-sne.” *Journal of machine learning research*, vol. 9, no. 11, 2008.
- [93] W. Sun, A. Venkatraman, G. J. Gordon, B. Boots, and J. A. Bagnell, “Deeply aggregated: Differentiable imitation learning for sequential prediction,” in *International conference on machine learning*. PMLR, 2017, pp. 3309–3318.
- [94] M. Pflueger, A. Agha, and G. S. Sukhatme, “Rover-irl: Inverse reinforcement learning with soft value iteration networks for planetary rover path planning,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1387–1394, 2019.
- [95] W. Jeon, S. Seo, and K.-E. Kim, “A bayesian approach to generative adversarial imitation learning,” *Advances in Neural Information Processing Systems*, vol. 31, 2018.
- [96] F. Sasaki, T. Yohira, and A. Kawaguchi, “Sample efficient imitation learning for continuous control,” in *International conference on learning representations*, 2018.
- [97] D. Brown, W. Goo, P. Nagarajan, and S. Niekum, “Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations,” in *International conference on machine learning*. PMLR, 2019, pp. 783–792.
- [98] J. Zhang, R. Zhu, and E. Ohn-Bar, “Selfd: Self-learning large-scale driving policies from the web,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 316–17 326.
- [99] Y.-H. Wu, N. Charoenphakdee, H. Bao, V. Tangkaratt, and M. Sugiyama, “Imitation learning from imperfect demonstration,” in *International Conference on Machine Learning*. PMLR, 2019, pp. 6818–6827.
- [100] F. Sasaki and R. Yamashina, “Behavioral cloning from noisy demonstrations,” in *International Conference on Learning Representations*, 2020.
- [101] Y. Wang, C. Xu, B. Du, and H. Lee, “Learning to weight imperfect demonstrations,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 10 961–10 970.
- [102] G.-H. Kim, S. Seo, J. Lee, W. Jeon, H. Hwang, H. Yang, and K.-E. Kim, “Demodice: Offline imitation learning with supplementary imperfect demonstrations,” in *International Conference on Learning Representations*, 2022.
- [103] M. Beliaev, A. Shih, S. Ermon, D. Sadigh, and R. Pedarsani, “Imitation learning by estimating expertise of demonstrators,” in *Proceedings of the 39th International Conference on Machine Learning*. PMLR, 2022, pp. 1732–1748.
- [104] J. Chae, S. Han, W. Jung, M. Cho, S. Choi, and Y. Sung, “Robust imitation learning against variations in environment dynamics,” in *Proceedings of the 39th International Conference on Machine Learning*, vol. 162. PMLR, 2022, pp. 2828–2852.
- [105] B. C. Stadie, P. Abbeel, and I. Sutskever, “Third-person imitation learning,” in *International conference on learning representations*, 2017.
- [106] K. Zakka, A. Zeng, P. Florence, J. Tompson, J. Bohg, and D. Dwibedi, “Xirl: Cross-embodiment inverse reinforcement learning,” in *Conference on Robot Learning*. PMLR, 2022, pp. 537–546.
- [107] A. Fickinger, S. Cohen, S. Russell, and B. Amos, “Cross-domain imitation learning via optimal transport,” in *International Conference on Learning Representations*, 2022.
- [108] K. Kim, Y. Gu, J. Song, S. Zhao, and S. Ermon, “Domain adaptive imitation learning,” in *International Conference on Machine Learning*. PMLR, 2020, pp. 5286–5295.
- [109] M. Yang, S. Levine, and O. Nachum, “Trail: Near-optimal imitation learning with suboptimal data,” in *International Conference on Learning Representations*, 2022.
- [110] G. Xiang and J. Su, “Task-oriented deep reinforcement learning for robotic skill acquisition and control,” *IEEE transactions on cybernetics*, vol. 51, no. 2, pp. 1056–1069, 2019.
- [111] R. Ramrakhyta, E. Undersander, D. Batra, and A. Das, “Habitat-web: Learning embodied object-search strategies from human demonstrations at scale,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5173–5183.
- [112] J. Lee, W. Jeon, B. Lee, J. Pineau, and K.-E. Kim, “Optidice: Offline policy optimization via stationary distribution correction estimation,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 6120–6130.
- [113] W. Sheng, A. Thobbi, and Y. Gu, “An integrated framework for human-robot collaborative manipulation,” *IEEE Transactions on Cybernetics*, vol. 45, no. 10, pp. 2030–2041, 2015.