A Survey of Inverse Reinforcement Learning in Aviation and Future Outlooks *

Rupal Nigam †
University of Illinois Urbana-Champaign, Urbana, IL, 61801

Jimin Choi‡
University of Michigan Ann Arbor, Ann Arbor, MI, 48109

Niket Parikh §
University of Illinois Urbana-Champaign, Urbana, IL, 61801

Max Z. Li ¶
University of Michigan Ann Arbor, Ann Arbor, MI, 48109

Huy T. Tran ‖
University of Illinois Urbana-Champaign, Urbana, IL, 61801

Many problems in aviation can be characterized as sequential decision-making problems under uncertainty, such as air traffic management and flight delay prediction. One approach to solving such problems is to learn from expert demonstrations. Inverse reinforcement learning (IRL) is one such class of methods, where a reward function is learned given expert demonstrations. However, while IRL has shown promising results in recent years, for example in autonomous vehicle path planning, we identify a significant, quantifiable gap in applications of IRL in aviation relative to other domains. In this paper, we formally introduce IRL, provide a comprehensive summary of foundational methods, review how IRL has been applied in the field thus far, discuss challenges that may explain the IRL gap, and explore potential future applications of IRL for aviation.

I. Introduction

The aviation field is characterized by many problems that require decision-making in complex environments. Combined with increasingly available data in aviation, this has led to significant research into machine learning and other data-driven techniques capable of helping solve such problems, including efforts from academia, industry (e.g., [1, 2]), and government stakeholders (e.g., [3–5]). More specifically, reinforcement learning (RL) is emerging as a promising approach to address, for example, air traffic control [6–9], autonomous conflict resolution [10] and separation assurance [11, 12], autonomous taxiing [13], landing [14], and emergency response maneuvers [15]. RL is a subset of machine learning where agents can interact with and learn from a simulated or real-world environment through a trial-and-error process, to maximize the accumulated reward they earn by taking different actions [16]. A comprehensive survey of how RL has been leveraged to solve various aviation problems has been conducted in [17].

Within RL, defining an appropriate reward function is a crucial yet challenging task [18]. This problem is particularly difficult in aviation-specific applications, due to the complex nature of aviation tasks. For example, accurately capturing and representing operator intent (such as pilots and air traffic controllers) is difficult, yet crucial for safe and efficient operations. While explicit rules and operational norms are readily available, their application in dynamic and unforeseen situations remains challenging. Human decision-making often involves nuanced judgments, risk tolerance, and adaptive strategies that are difficult to fully codify. This lack of a clear objective function hinders the development of effective

*This paper was presented as "A Survey of Current Applications of Inverse Reinforcement Learning in Aviation and Future Outlooks" at AIAASCITECH 2025 Forum, Orlando, FL, January 6–10, 2025.

†Corresponding author: rupaln2@illinois.edu, Ph.D. Student, The Grainger College of Engineering, Department of Aerospace Engineering
‡Ph.D. Student, Aerospace Engineering
§Ph.D. Student, The Grainger College of Engineering, Department of Aerospace Engineering
¶Assistant Professor, Aerospace Engineering
‖Assistant Professor, The Grainger College of Engineering, Department of Aerospace Engineering

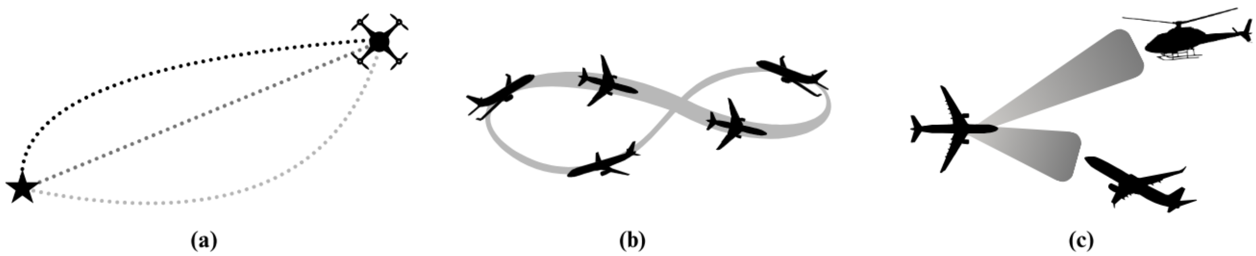

Fig. 1 IRL is a promising approach for various aviation problems, such as (a) UAV control and path planning, (b) complex maneuvers, and (c) air traffic management.

autonomous systems and predictive models. Inverse reinforcement learning (IRL) offers a promising approach to address these challenges. Instead of explicitly defining rewards, IRL algorithms aim to infer an underlying reward function from expert demonstrations or collected trajectory data [19]. By learning the implicit goals and preferences of operators, IRL can enhance situational awareness, improve human-machine collaboration, and contribute to more robust and adaptable autonomous aviation systems. Furthermore, learning a reward function through IRL can improve transfer capabilities between different tasks relative to RL, as reward functions are inherently more transferable than decision-making policies [20]. Figure 1 depicts example IRL applications in aviation. However, despite the widespread use and success of IRL methods in numerous fields, the aviation field has seen much fewer applications of IRL.

This work explores the potential of IRL for understanding and enhancing autonomous decision-making in aviation systems. We delve into the theoretical foundations of these techniques (Section II), survey foundational IRL methods that have been successfully applied in related domains (Section III), and discuss their application to specific aviation tasks thus far (Section IV). Additionally, we put forth and discuss relevant datasets, simulations, and potential future applications of IRL in aviation (Section V). Our contributions are thus:

1) formally introducing IRL, its benefits, and its challenges,
2) providing a summary, categorization, and comparison of various foundational IRL methods,
3) reviewing some significant milestones achieved by IRL methods in aviation, and
4) identifying relevant datasets, simulations, and potential aviation problems unexplored by IRL to encourage its use in aviation.

II. Background

A. Reinforcement Learning

RL models an agent and its interactions with an environment as a Markov decision process (MDP), defined as $M \coloneqq \langle \mathcal{S}, \mathcal{A}, p, r, \gamma \rangle$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $p : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the state transition function, $r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is a reward function, and $\gamma \in [0, 1)$ is the discount factor which assigns smaller weights to future rewards. Let $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ be a policy for the agent. At time step t, the agent is in state $s_t \in \mathcal{S}$ and executes an action $a_t \sim \pi(\cdot|s_t)$, after which the system transitions to state $s_{t+1} \sim p(\cdot|s_t, a_t)$ and the agent receives reward $r(s_t, a_t, s_{t+1})$. The performance of π can be described by its value function. The value function of policy π, $V^\pi : \mathcal{S} \to \mathbb{R}$, denotes the expected discounted cumulative reward obtained when starting from state s and following policy π,

$$V^\pi(s) \coloneqq \mathbb{E}_{p,\pi}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, s_{t+1}) \mid s_0 = s\right]. \tag{1}$$

Similarly, the action-value function of policy π, $Q^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, denotes the expected discounted cumulative reward obtained when starting from state s, taking action a, and then following policy π to choose subsequent actions,

$$Q^\pi(s, a) \coloneqq \mathbb{E}_{p,\pi}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, s_{t+1}) \mid s_0 = s, a_0 = a\right]. \tag{2}$$

The typical goal in RL is to find an optimal policy, π^*, which maximizes the value function $V^\pi(s)$ at every state $s \in \mathcal{S}$. The agent learns the optimal policy through repeated interactions with the environment, as visualized in Figure 2. The

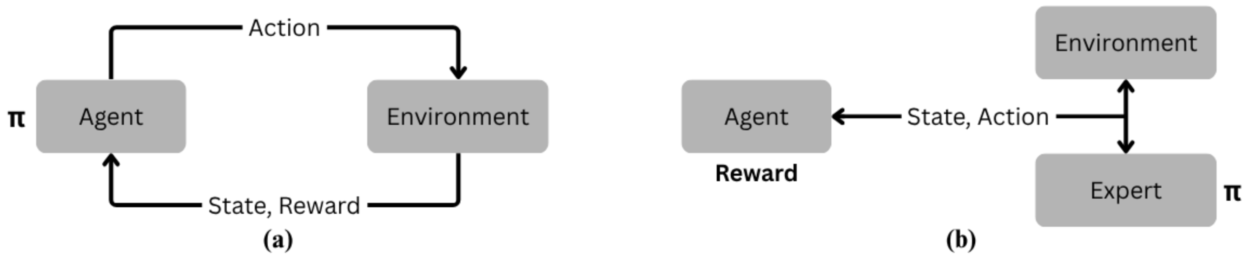

Fig. 2 (a) RL learns an optimal policy by choosing actions and receiving rewards. (b) IRL learns a reward function by observing the expert's states and actions.

MDP framework can be extended to model multi-agent decision-making problems through a multi-agent MDP (MMDP), defined as $\mathbf{M} \coloneqq \langle \mathcal{S}, \mathcal{N}, \{\mathcal{A}^i\}_{i\in\mathcal{N}}, p, r, \gamma \rangle$, where $\mathcal{N} = \{1, \ldots, N\}$ is the set of agents and $\mathcal{A}^i$ is the action space of agent i. Then $p : \mathcal{S} \times \mathcal{A}^1 \times \cdots \times \mathcal{A}^N \times \mathcal{S} \to [0, 1]$ is the state transition function and $r : \mathcal{S} \times \mathcal{A}^1 \times \cdots \times \mathcal{A}^N \times \mathcal{S} \to \mathbb{R}$ is the team reward function. Similar to the single-agent case, the typical goal for an MMDP is to find the joint policy, $\boldsymbol{\pi} : \mathcal{S} \times \mathcal{A}^1 \times \cdots \times \mathcal{A}^N \to [0, 1]$, which maximizes the value function $V^{\boldsymbol{\pi}}(s)$ at every state $s \in \mathcal{S}$.

B. Inverse Reinforcement Learning

In IRL, an agent observes an expert's behavior (or is given a dataset of collected expert trajectories) and uses that to infer the expert's *reward function* [19]. This problem is modeled as an MDP without the reward function, $M_{\backslash r} \coloneqq \langle \mathcal{S}, \mathcal{A}, p, \gamma \rangle$, with a dataset of expert trajectories, $\mathcal{D} = \{\tau_i\}_{i=1}^{N_e}$, where $\tau_i = \{(s_0, a_0), \ldots, (s_t, a_t), \ldots\}$ denotes the ith trajectory and N_e is the number of available expert trajectories. Expert trajectories are assumed to be finite with potentially varying lengths unless specified otherwise.

Whereas in RL the agent aims to learn the optimal policy π^* given r (and sometimes p), in IRL the agent seeks to learn r given the dataset of trajectories $\mathcal{D}$ (and sometimes p). This framework thus enables agents to learn a reward function from human behavior and expert demonstrations for complex tasks where an appropriate reward function may be difficult to define. For example, defining a reward function to train a robot arm to manipulate various objects is complicated as each actuation must be encoded for each type of object and desired location. Instead, providing relevant expert demonstrations can enable the robot to learn an appropriately parameterized reward function through IRL. As with RL, the IRL problem formulation can also be extended to multi-agent systems [21]. Figure 2 visualizes the RL and IRL problem setups.

C. Imitation Learning

Imitation learning (IL), also known as learning from demonstration, apprenticeship learning, or behavior cloning, is a related learning paradigm in which an agent aims to *mimic* the behavior of an expert by observing its actions and their corresponding outcomes. While this method addresses problems similar to IRL, for example, scenarios where designing a reward function is challenging, a key difference is that IL aims to recover a policy π from demonstrations instead of a reward function. We include this subsection because many aviation works surveyed in this paper include IL methods, and thus it is important to clarify the similarities and differences between the two learning paradigms.

IL can be more efficient than IRL, as it does not include the computationally expensive step of forward RL to evaluate the current reward function. Additionally, IL does not assume any structure over the reward function, and thus can be easier to implement. However, IL also has some limitations. The quality of the learned policy is highly dependent on the quality of the expert demonstrations, since poor or suboptimal demonstrations can lead to suboptimal policies. While this is also a challenge for IRL methods, it is exacerbated in IL methods. IRL can include methods to handle suboptimal demonstrations during optimization steps, whereas IL primarily aims to mimic the demonstrated behaviour. IL also struggles more when generalizing to new situations not represented in the expert demonstrations, since a policy is less transferrable than a reward function when the environment's underlying dynamics change [16, 22]. Finally, it has been empirically found that when the reward function is sparse, IRL methods perform better than IL methods [22].

Algorithm 1 Inverse Reinforcement Learning Algorithm Template

Require:
 $\mathcal{D}$: A set of expert demonstrations
 M: $\langle \mathcal{S}, \mathcal{A}, p, \gamma \rangle$
Ensure:
 $\hat{r}$: The estimated expert reward function
1: Initialize $\hat{r}$
2: **while** not done **do**
3: Compute the optimal policy π^* for $\hat{r}$ using an RL algorithm
4: Update $\hat{r}$ to maximize the desired objective, such as the likelihood of $\mathcal{D}$ under π^*
5: done $\leftarrow$ whether π^* generated by $\hat{r}$ matches the expert demonstrations $\mathcal{D}$ within the desired margin of error, or a maximum number of iterations is reached

D. Challenges in IRL

While IRL is an effective learning paradigm for addressing problems with unknown reward functions, there are many open challenges, which we summarize here. When considering multiple agents in IRL problems, additional challenges are induced (challenges 5-7). Addressing these challenges is crucial for applying IRL to real-world problems, such as those related to aviation systems, many of which require multi-agent considerations. While some, such as access to expert demonstrations, are challenges inherent to the application, others can be addressed through the reward optimization process.

1) Underspecification of Rewards: A fundamental challenge in IRL is the inherent ambiguity in inferring reward functions from observed expert behavior [23]. Multiple reward functions may explain the behavior for a given set of expert trajectories.
2) Computationally Expensive: To evaluate the current estimate of the learned reward function, we must perform forward RL many times. This process can be computationally intensive, especially for complex environments. As the state and action space of the MDP grows, the computational complexity for each algorithm iteration also increases.
3) Generalization: Although reward functions are more transferable than policies [19], they still have limitations in terms of generalization. The performance of the reward function may be limited to settings identical or similar to the expert demonstrations.
4) Quality of Experts: The quality of the expert demonstrations significantly impacts the accuracy and effectiveness of IRL. In many real-world scenarios, access to high-quality expert demonstrations may be limited or expensive to attain. Furthermore, expert demonstrations are not necessarily optimal, which can impact the learning process.
5) Scalability: As the number of agents present in the environment increases, the complexity of the problem grows exponentially. This is due to the increased dimensionality of the state and joint action spaces.
6) Additional Ambiguity: In multi-agent settings, there may be multiple Nash equilibria or global reward functions that could explain the observed behavior of the agents. The assumptions and conditions for team optimality may be unknown, making it difficult to narrow down and identify unique reward functions that accurately capture the underlying incentives of the agents.
7) Expert Interactions: A unique aspect of multi-agent scenarios is that the expert policies may be interdependent. Thus, the optimal behavior of one agent may depend on the actions of other agents.

III. Foundational Methods for IRL

The majority of IRL methods follow a general algorithm template [24], shown in Algorithm 1. As described in Section II and visualized in Figure 2, the agent is given access to either a dataset of state-action pairs or an expert policy (from which a dataset is generated). The agent's objective then is to optimize a reward function that best describes the input. The reward function is parameterized by weights which are updated during each iteration in the algorithm. During each iteration, the current estimate of the reward function is evaluated using forward RL to generate a corresponding policy (line 3). We then compute how closely this policy matches the expert policy. Based on the resulting similarity, the parameters of the reward function are updated (line 4) until a desired minimum difference is achieved. The subsequent sections describe common classes of IRL methods; more detailed expositions are given in [24, 25]. We include multi-agent extensions where applicable.

A. Feature Matching

Feature expectation matching is a widely used approach to formulate IRL problems. First, it is assumed that the reward function can be expressed as a sum of weighted features, $\phi : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}^n$, such that $r_\phi(s, a, s') = w^\intercal \phi(s, a, s')$, where $w \in \mathbb{R}^n$ is a vector of weights. Given a feature function ϕ, the idea is to choose weights such that the optimal policy for a chosen linear reward, π^{r_ϕ}, achieves the same feature expectation as the expert policy, π^E, $\mathbb{E}_{\pi^{r_\phi}}[\phi(\tau)] = \mathbb{E}_{\pi^E}[\phi(\tau)]$, where $\phi(\tau) = \sum_{(s_i, a_i, s_{i+1}) \in \tau} \phi(s_i, a_i, s_{i+1})$, $\mathbb{E}_\pi[\phi(\tau)] = \Pr(\tau|\pi)\phi(\tau)$, and $\Pr(\tau|\pi)$ is the probability of sampling trajectory τ from the distribution over trajectories induced by π. In practice, it can be difficult to compute this expectation. Another challenge, as discussed in Section II.D, is that this optimization problem can have many possible solutions. Maximum margin methods propose different margins to choose one among the many solutions.

B. Maximum Margin Optimization Methods

The objective of maximum margin optimization methods is to find a reward function that is better at explaining the given expert demonstrations than alternative reward functions by a set margin. The definition of this margin varies by method. The margin maximized in [19] is

$$\sum_{s \in \mathcal{S}} \left(Q^\pi(s, a^*) - \max_{a \in \mathcal{A} \setminus a^*} Q^\pi(s, a) \right). \tag{3}$$

Here, the goal is to maximize the sum of differences between the value of the optimal action a^* and the value of the second-best action. This ensures that single-step deviations from the optimal policy are as costly as possible.

An alternative formulation of the maximum margin optimization problem involves feature expectation matching. This formulation is sometimes referred to as apprenticeship learning in the literature [18]. First, we assume the reward function can be expressed as a linear sum of weighted features. Next, we define the expected state visitation frequency of a policy as

$$\psi^\pi(s, a) = \mathbb{E}_{p,\pi} \left[\sum_{t=0}^{\infty} \gamma^t \phi(s_t, a_t, s_{t+1}) \mid s_0 = s, a_0 = a \right]. \tag{4}$$

ψ is sometimes known as the successor feature function [26]. Now, we can re-define the action-value function as $Q^\pi(s, a) = \psi^\pi(s, a)^\intercal w$. Substituting this into the margin from Equation (3) yields

$$\sum_{s \in S} \left(\psi^\pi(s, a^*)^\intercal w - \max_{a \in A \setminus a^*} \psi^\pi(s, a)^\intercal w \right). \tag{5}$$

Early works solved the optimization problem in Equation (5) by formulating it as a linear or quadratic program [27]. Subsequent works, such as [28], optimize the reward function using gradient descent. A benefit of this class of IRL methods is that the optimization problems are relatively straightforward to formulate. However, observe that the learned reward's performance in these formulations depends on the quality of the expert data. To help alleviate this, [29] proposes an extension, MWAL, to handle the case where the expert data might be sub-optimal, while [30] handles the case where there may be limited expert data. Additionally, [31] incorporate safety constraints in their extension, CEGAL, and [32] extends this to the multi-agent setting.

C. Maximum Entropy Methods

Maximum entropy methods aim to find the behavior strategy that is constrained to match feature expectations and maximize entropy, thereby addressesing the challenge of ambiguity described in Section II.D. The first paper to propose a technique in this class of IRL methods is [33]. The work finds the distribution over all possible trajectories in an MDP that maximizes entropy while matching feature expectations of the given expert behavior. The motivation behind using entropy to choose among degenerate solutions comes from the principle of maximum entropy [34], which implies that the solution with maximum entropy minimizes bias. In practice, this translates to making minimum commitments beyond the necessary feature expectation matching constraint.

Mathematically, this maximization can be formalized as

$$\max_{\Delta} - \sum_{\tau \in (\mathcal{S} \times \mathcal{A})^l} \Pr(\tau) \log \Pr(\tau), \tag{6}$$

where Δ is the space of all distributions [24] and τ is a trajectory of length l. That is, the agent's goal is to find the reward function that maximizes the likelihood that the given expert trajectories will occur. Specifically, the probability of the expert policy generating a trajectory for given reward function parameters is exponentially greater as the reward increases. This was first formalized as a convex, non-linear optimization problem in [33].

Later work solves this problem using neural networks, enabling maximum entropy methods to be used in a wider class of problems [35]. For example, entropy-based IRL methods have been extended to continuous and high-dimensional domains [36] and causal entropy [37]. A potential drawback of maximum entropy methods is the assumption that a transition model of the underlying MDP is available. RE-IRL introduces the concept of relative entropy to extend maximum entropy learning to a model-free setting [38]. The relative and maximum entropy concepts are later unified in [39]. Furthermore, recent works are looking into extending maximum entropy methods for multi-agent IRL [40, 41].

D. Adversarial-based Methods

In [42], an analogy between model-free maximum entropy and general adversarial networks (GANs) is drawn, which jump-started a new line of research into adversarial-based IRL methods. While not strictly an IRL method, GAIL [43] has been widely adopted in the field of IL and has been used solely or in conjunction with IRL techniques in aviation applications. GAIL employs a generative adversarial framework, where a generator network produces agent policies and a discriminator network distinguishes between the generator's policies and expert demonstrations. While GAIL is effective in learning complex policies, it is sensitive to the quality of expert demonstrations and lacks convergence guarantees. MA-GAIL [44] extends GAIL to the multi-agent setting, where it can imitate the policy of each agent present in cooperative, competitive, and mixed-motive scenarios. OptionGAN [45] extends GAIL by considering multiple reward functions using the options framework in RL. At a high level, the options framework decomposes a policy into multi-step actions or options. OptionGAN excels in one-shot transfer settings, addressing a primary challenge mentioned in Section II.D.

AIRL [46] is an adversarial IRL method that can learn robust reward functions, even in the presence of changes to environment dynamics. This method is similar to GAIL, but also recovers the reward function instead of just the expert policy. However, AIRL can be challenging to train due to the complex interplay between the generator and discriminator networks. H-AIRL [47] extends AIRL to a hierarchical framework, enabling the learning of hierarchical policies and reward functions. MA-AIRL [48] extends AIRL to multi-agent settings by assuming a correlated equilibrium among agents, which is a more general assumption than that of Nash equilibrium.

Overall, adversarial-based IRL methods help address the challenges of noisy expert demonstrations, scalability to complex environments, and generalization to similar settings not covered by the given demonstrations. While GANs can be difficult to train and computationally expensive, the increased availability of computing resources in recent years has led to many IRL methods in this category having great success in applications such as robotics [47, 49], autonomous driving [50], and mobility [51].

E. Bayesian Methods

In Bayesian IRL methods, the state-action pairs in a trajectory are modeled as observations that are used to perform Bayesian updates of prior distributions to update the distribution over possible reward functions, as follows,

$$\Pr(\tau \mid r) = \Pr\left((s_0, a_0) \mid r\right) \Pr\left((s_1, a_1) \mid r\right) \ldots \Pr\left((s_T, a_T) \mid r\right). \tag{7}$$

Typically, the likelihood of each state-action pair is modeled as an exponential distribution in terms of the action-value function. Then, using Bayes' theorem and given an expert trajectory, one can compute the likelihood of a reward function.

BIRL models the likelihood function as a Boltzmann distribution [52] and considers various candidate functions to represent the priors over possible reward functions. Some examples include Beta density, Laplacian, and uniform density. GPIRL [53] extends this method and models the reward function as a Gaussian process, enabling it to be a nonlinear function of weighted features. MLIRL [54] directly maximizes the likelihood estimation in Equation (7) by modifying the representation of the policy so that it is differentiable. Thus, standard gradient ascent methods can be used to maximize the likelihood and converge to an optimum. These probabilistic methods help mitigate the challenge of noisy expert demonstrations mentioned in Section II.D.

F. Classification and Regression Methods

Classification and regression methods are typically more straightforward to formulate and implement but may result in less accurate performance because IRL is not a simple supervised learning problem. We can translate IRL to a classification problem by treating state-action pairs in $\mathcal{D}$ as the data and labels, respectively. Since the size of $\mathcal{A}$ is often greater than two, we typically have a multi-class classification problem. The goal in such problems is to minimize the data-label classification error. A natural way to provide this quantification in the IRL setting is using the action-value function. Recall that this function can also be expressed using a linear parameterization as in Equation (4). Thus, our goal now is to find the linear weights, w, that minimize the classification error between the pairs given in $\mathcal{D}$ and pairs generated through our current estimate of the reward.

SCIRL implements the multi-class classification formulation of IRL [55]. This method requires knowledge of the transition dynamics, which may be a limitation in some applications. CSI is an extension of SCIRL and uses regression techniques while also estimating a transition dynamics model [56]. For problems with simpler reward functions, transition dynamics, and sufficient expert demonstrations, methods in this category can be more straightforward to use than other categories without greatly sacrificing performance.

G. Miscellaneous

This section presents IRL methods that do not fall into the above categories. Many of these methods are also more recent and therefore have not been included in prior IRL surveys [24, 25].

1) IQ-Learn [57]: This method learns a single function capable of representing both the policy and reward function by introducing a new update rule. This is in contrast to the dual adversarial network optimization necessary for methods in Section III.D. This method can be implemented on top of an existing RL algorithm with less than 15 additional lines of code.
2) BASIS[58]: This method leverages RL pre-training to quickly recover a reward function. During pre-training, the RL agent builds a ‘basis’ that spans the space of all possible goals for the given MDP. When given expert demonstrations for a new unseen goal, the agent can accurately recover the expert’s reward function in less than 100 trajectories.
3) XIRL [59]: Cross-Embodiment IRL addresses the challenge of learning from demonstrations performed by agents with different embodiments, such as end-effector dynamics. By leveraging self-supervised learning and temporal cycle-consistency constraints, XIRL can learn robust visual reward functions from videos of diverse expert demonstrations.
4) GraphIRL [60]: This method leverages graph-based representations to learn reward functions from diverse videos. By abstracting videos into graphs, GraphIRL can focus on the underlying structure of the task, reducing the impact of irrelevant visual information. Tasks are described by entity interactions that form a graph abstraction free of irrelevant information.
5) IRL without RL [61]: As mentioned in Section II.D, the forward RL optimization step often required in IRL is computationally expensive. To address this, this approach uses the state distribution of the expert to accelerate learning, where the learner agent only considers state distributions relevant to that of the expert’s, which drastically reduces the amount of environment interactions necessary for algorithm convergence.

H. Example IRL Problem Formulation: Autonomous Driving

As we discuss in Section IV.B, the current work on IRL for aviation applications is limited. However, IRL has been applied in many other fields, such as robotics, controls, autonomous driving, video games, and healthcare. Autonomous driving has many similarities to aviation problems, such as advanced air mobility. Some analogous tasks include trajectory prediction, traffic control, and learning from human operators. Thus, we summarize the typical problem setup for applying IRL to autonomous driving to help the reader understand how the same might be done for an aviation problem.

Prior to applying an IRL algorithm, it is crucial to understand the dataset of trajectories given as well as the context and problem to be solved. Analyzing and pre-processing the data can provide insights such as what information is available, the timescale to consider, and the length of trajectories. These design decisions inform the following steps of defining an MDP and selecting an appropriate IRL method.

The state space of an autonomous vehicle typically includes information about the vehicle’s position, heading, velocity, and the positions and velocities of other vehicles and obstacles in the environment. The action space can include continuous actions, such as steering angle and throttle, as well as discrete actions, such as gear shifts and brake

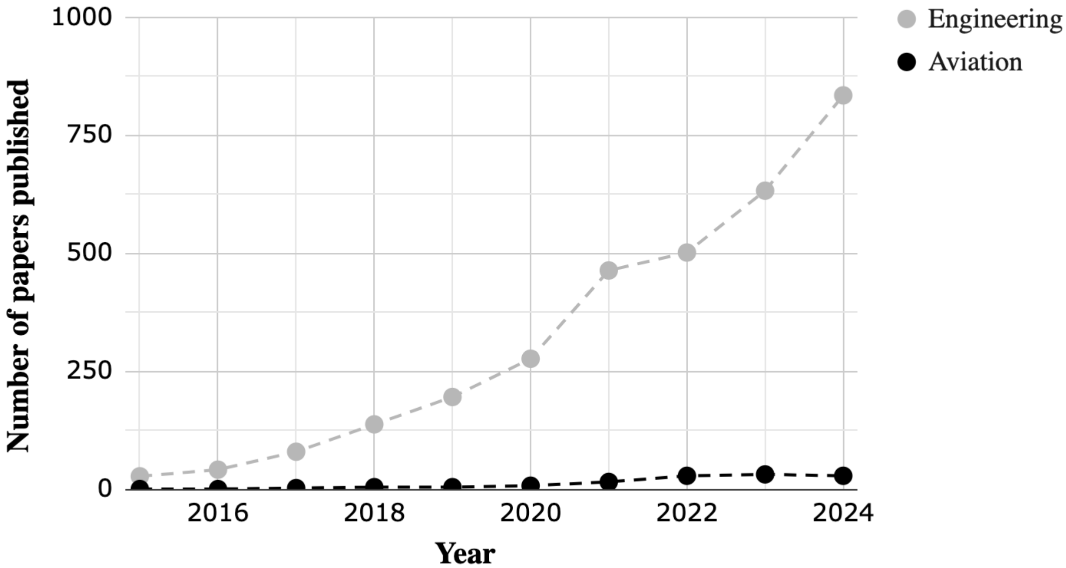

Fig. 3 Number of IRL papers published across engineering and in aerospace engineering.

application [62]. If a transition dynamics model is available, model-based IRL methods can be used. Otherwise, the dynamics must be learned or model-free IRL methods must be used.

Reward functions in autonomous driving often aim to incentivize safe, efficient, and comfortable driving behavior. Common features defined for estimating the reward function therefore include distance traveled, speed, collision avoidance, traffic rule adherence, and passenger comfort [62]. The definition of reward features to consider can be informed by the problem's context, defined by a subject-matter expert, or learned from the data itself using machine learning methods. Once the structure of the reward function has been determined, the appropriate IRL method can be used to optimize and uncover the underlying reward function associated with the dataset.

IV. Applications of IRL in Aviation

IRL has been widely applied to various systems, such as robotics and autonomous driving, demonstrating its significant potential. However, research on IRL in aviation remains relatively limited. Figure 3 illustrates the number of IRL papers published across various domains and within the aerospace engineering sector. The data was retrieved from Dimensions, a database of research articles indexed by Crossref, PubMed, PubMed Central, arXiv.org, and more than 160 other publishers directly. Papers counted here include those containing the key phrase "inverse reinforcement learning." Engineering papers were filtered using Dimension's research area category "Engineering", with aerospace engineering papers further filtered by the "Aerospace Engineering" subcategory. Therefore, some discrepancy may exist in the number of papers presented, as this figure encompasses both air and space applications. Additionally, this search does not include IL papers, although aerospace IL papers are included in the subsequent review. A linear increase in published IRL papers is observed in all engineering domains since 2015, indicating growing research interest and application in these fields. Notably, the Engineering category does not include Computer and Information Sciences, which are the dominant fields for fundamental IRL research. In contrast, within aerospace, a notable rise in the number of IRL papers emerged only after 2020, although a few papers were sporadically published earlier. As of the preparation of this survey manuscript, the total number of IRL papers in all engineering disciplines is over 3000, whereas those specific to aerospace number just over 100. Despite its promising potential, this contrast highlights the relatively premature use of IRL in aviation compared to other engineering industries.

We now discuss several aviation-specific challenges that may explain this gap and review previous studies using IRL for aviation.

A. Aviation-specific Challenges for IRL

Applying IRL to aviation problems faces unique challenges that may contribute to the significant gap in aviation applications compared to other domains. We discuss several of these challenges below.

1. Ensuring Safety and Reliability

Since many aspects of aviation are safety-critical, algorithms used in this domain require validation and verification, ideally supported by theoretical guarantees. For example, aviation systems are highly sensitive to errors or failures, which can lead to serious loss of life or significant economic losses. Therefore, rigorous validation procedures that assess whether an algorithm meets the intended operational requirements are essential when using IRL algorithms to minimize these risks and comply with aviation safety standards. These procedures include simulation testing, real-world flight testing, and stress testing for multiple scenarios. At each stage of validation, the limit of the algorithm needs to be thoroughly examined, and only those that pass can be used in production. Verifying that inferred reward structures behave safely and reliably in real-world situations is also crucial. Verification, in this context, ensures that these inferred structures operate as intended under real-world conditions. For example, in the case of a drone flight controller, the algorithm needs testing to ensure it can manage different environmental factors and unexpected situations. This involves evaluating its performance in decision-making with different variables, such as changes in wind, obstacles, or signal disruptions. Such evaluations are costly and difficult to implement. Furthermore, theoretical guarantees regarding the safety and reliability of policies optimized for inferred rewards are currently limited.

2. Lack of Expert Data Accessibility

Obtaining high-quality aviation data for IRL can be particularly challenging. One reason is that aviation data, particularly as it relates to aircraft control data and flight logs, can be relatively access-restricted due to safety and security concerns. For example, some real-time operational data provided by the US Federal Aviation Administration (FAA) Air Traffic Control System Command Center (ATCSCC) is only available to users with restricted access. IRL datasets should also include examples of contingency situations to understand behaviors in these rare, but important, real-world situations—unfortunately, such data is rarely shared in detail. For example, engine or control system failures in flight are rare, and when they do occur, detailed data from Flight Data Recorders (e.g., black boxes) can be sometimes difficult to obtain [63].

Moreover, aviation data is often incomplete or noisy due to a variety of factors. Inferring an accurate reward structure from such incomplete and/or noisy data is challenging, despite the increasing availability of data in recent years. The extent of this availability greatly depends on the specific application and geographical location [64]. Several simulators exist to address these limitations; however, using simulation data also has drawbacks because a simulated environment cannot be expected to faithfully reproduce all real-world complexities.

3. Lack of Standard Testbeds and Benchmarks

The aviation domain suffers from a lack of open source, standardized test environments and benchmarks for evaluating IRL algorithms. While this is a broader issue in IRL, it is particularly problematic in aviation due to the complex and highly regulated nature of the field. Diverse operational conditions, stringent safety requirements, and the need to integrate with existing systems make it difficult to create uniform testbeds. Without standardized benchmarks, assessing and comparing the efficiency, reliability, and safety of these algorithms is challenging. Reliable benchmarks are crucial not only to verify algorithm accuracy but also to ensure consistent performance across different, often unpredictable environments.

For example, evaluating an algorithm that optimizes aircraft takeoff and landing routes requires a testbed that provides a variety of scenarios and environments. These testbeds are required to reflect different weather conditions, terrain, airport infrastructure, and traffic density. Although there is a significant need, testbeds that effectively address complex scenarios are still limited. Furthermore, the absence of internationally recognized benchmarks makes it difficult to compare algorithms across various studies consistently.

4. Scalability

Applying IRL algorithms to large-scale aviation problems faces significant scalability challenges, particularly in multi-agent contexts. Aviation often involves numerous agents, including aircraft, air traffic controllers, ground support teams, and weather monitoring systems. Multi-agent IRL methods, still relatively new, are crucial but complex as each agent must consider the behaviors of others. For example, air traffic controllers must monitor and manage multiple aircraft in real time to ensure safety and efficiency, requiring an IRL algorithm capable of modeling and learning these intricate interactions.

The expansion of the state space, often called the curse of dimensionality, presents another scalability issue. In aviation, position, speed, altitude, and weather conditions contribute to an exponentially growing state space, making data management complex and demanding high-performance computing resources. For instance, optimizing routes for hundreds of aircraft in a busy airspace generates a vast state space, challenging algorithms to process this efficiently. Multi-agent missions also require contextual decision-making, such as coordinating takeoffs, landings, and collision avoidance. Algorithms must learn and reason about optimal decisions by adapting to different behavioral situations.

5. Legal and Regulatory Requirements

The aviation sector has very strict regulations and legal requirements, which can slow the introduction of new technologies. Introducing new algorithms into aviation requires approval from multiple regulatory bodies, which can be a lengthy process. Organizations such as the FAA and the European Aviation Safety Agency (EASA) review new technologies and systems to ensure they comply with current stringent safety standards. This approval process necessitates experimental results to prove the reliability and safety of the technology. For example, introducing an automated flight control system requires multiple test flights to demonstrate the system's performance and safety, along with a range of reports on the results. It also requires assessing how the new technology is compatible with existing laws and regulations. These challenges are further compounded when implementing IRL in the highly regulated airline industry, as the stochastic and complex nature of IRL algorithms must align with stringent safety and regulatory standards.

6. Multidisciplinary Systems Engineering Aspects

Aircraft are a prime example of a highly complex systems engineering domain, where factors such as aeronautics, propulsion, control, and structures must be closely linked for safe and efficient operation. These variables are interdependent, meaning a change in one can significantly impact the others. Additionally, aircraft operation relies on real-time data exchange with systems like air traffic control, weather observation, and ground support, requiring accurate data integration to ensure safety. For instance, systems providing real-time location information and optimal flight paths must prioritize data accuracy and processing speed to avoid safety concerns. These complexities make it challenging for IRL algorithms to analyze behaviors and account for the many interactions between systems.

B. Existing Work on IRL in Aviation

Despite these challenges, there is a growing body of work applying IRL to aviation. We categorize these existing applications under two primary objectives: mimicking expert policies and predicting aviation systems. Table 1 lists existing work focused on learning from and mimicking decision-making strategies experienced pilots employ in diverse scenarios. Table 2 lists existing work focused on predicting system behaviors, such as air traffic flow management, aircraft trajectory optimization, and fault diagnosis in aviation systems. Studies utilizing IL approaches, such as GAIL [43] and Dagger [65], are also included in these tables. We include papers with these approaches because IL achieves similar objectives to IRL from an application perspective (e.g., replicating pilot decision-making or optimizing air traffic control strategies).

We see that Table 1 contains more papers than Table 2 (17 versus seven), likely due to mimicking expert data representing a more typical IRL problem than predicting system behavior. We also see that Table 1 encompasses a more diverse array of problem types, reflecting a comparatively mature research domain, while Table 2 focuses on trajectory prediction, illustrating the growing potential for IRL in understanding system behavior. A common trend in Table 1 (mimicking expert policies) is using both simulated and real-world data to enhance model robustness. For example, [66] leverages real pilot data and simulated aircraft models to enhance the demonstration dataset. Some works also integrate IRL with other techniques, such as planning [9] and particle swarm optimization [67], to improve the effectiveness of the implemented IRL method. In contrast, Table 2 (predicting system behaviors), shows a stronger emphasis on real-world data. Both tables reveal a reliance on advanced IRL techniques. These distinctions highlight the broad applicability of IRL in aviation, driving innovations in decision-making and predictive analytics. Figure 4 visualizes the percentage of these papers that fall into the categories we identified within the problem type addressed, the IRL algorithm leveraged, and the source of expert data used. The majority of papers fall under the pilot control imitation category and aim to learn aircraft maneuvers. This reflects the difficulty of designing objective reward functions for such complex maneuvers. We also see that most papers leverage either IL algorithms or adversarial IRL algorithms, the latter of which have gained popularity in recent years due to their performance in complex problem domains. Real world data is a popular expert data source, likely because high-fidelity and accurate simulations are difficult to make and expensive to run.

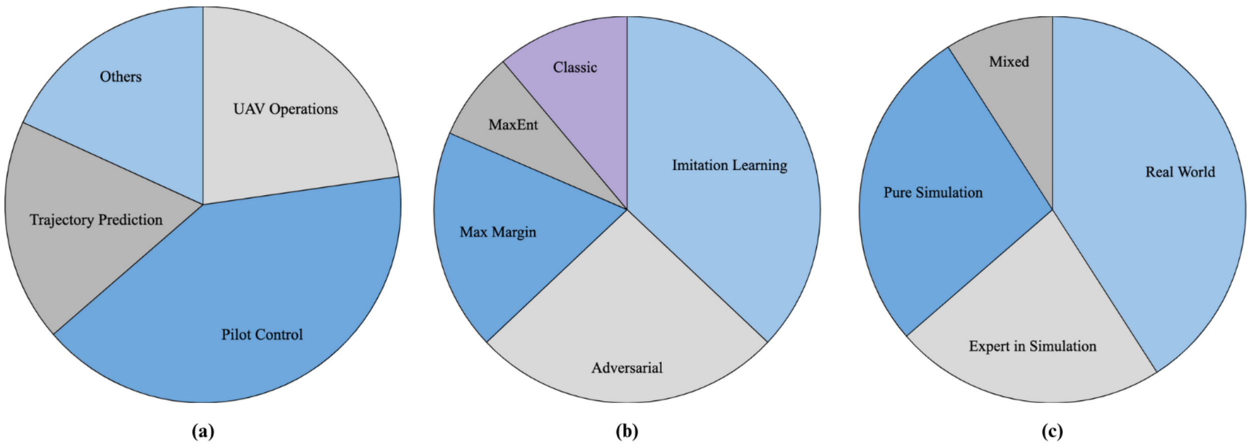

Fig. 4 Distribution of existing IRL works in aviation across categories identified in (a) problem type, (b) IRL method class, and (c) expert data source.

The following sections discuss the studies in Tables 1 and 2 in more detail, categorized by problem type to better highlight the common methodologies and advancements within each category. The "Others" category encompasses studies that do not neatly fit into the other specified categories, or contain a sparse number of studies.

1. UAV Operations

This subsection includes several studies that apply IRL methods to enhance autonomous behavior, coordination, and control of UAVs. A study investigates autonomous learning for multiple drones with limited vision and no central control, developing a distributed motion model that uses onboard cameras for navigation and obstacle avoidance [68], demonstrating robustness and scalability in decentralized control. Another study introduces a Federated Imitation Learning Control (FILC) algorithm for UAV swarm coordination in urban traffic monitoring [69]. By combining inter-agent GAIL and intra-agent self-IL (SIL) models, the algorithm adapts to dynamic traffic conditions.

In trajectory tracking, an IRL algorithm has been proposed to learn control skills for a multirotor UAV by extracting representative trajectories from demonstrations and learning a reward function [67]. This method has been validated through simulations and experiments, successfully imitating expert behavior. Joint path planning and power allocation for cellular-connected UAVs are addressed in another study, which uses apprenticeship learning via IRL with Q-learning and deep Q-networks (DQN) to minimize interference and maximize throughput [70]. It highlights IRL's effectiveness over behavioral cloning in scenarios with system errors or incomplete expert data. Finally, an algorithm for autonomously generating UAV combat maneuver strategies is presented, utilizing SA-DDPG and IRL to account for sensor measurement errors [71]. The research improves the learning speed and robustness of combat strategies, with future work to include control strategies for weapons and sensors.

2. Aircraft Trajectory Prediction

This subsection encompasses studies using IRL and IL methods to enhance the accuracy and robustness of predicting flight paths. One study treats the trajectory prediction problem as an IL task by employing GAIL along with trajectory clustering and classification, effectively predicting whole flight trajectories from origin to destination [72]. Similarly, there is another paper introducing an apprenticeship learning approach, which uses expert trajectories to develop a reward function through IRL [73]. It employs a DQN-based neural structure for Q-value function approximation, demonstrating promising initial results.

A recent study further contributes to this field by modeling flight trajectories as a multi-modal IL problem using Triple-GAIL and Info-GAIL, with different expert modalities to generate accurate trajectories [74]. Meanwhile, another study applies GAIL with a critic model for state advantage estimation, which reduces variance in policy gradients by evaluating the benefit of an action relative to the state value, enabling accurate trajectory prediction [75]. Finally, further research addresses the transferability of trajectory prediction models using GAIL [76]. This study highlights the model's effectiveness in handling different trajectory patterns and proposes a pipeline for model transfer across OD pairs, incorporating zero-shot generalization techniques to improve accuracy with minimal training time.

3. Pilot Control Imitation

This subsection covers studies that utilize IRL and IL methods to replicate expert pilot behaviors and enhance autonomous aircraft control systems. The research investigates the application of GAIL and AIRL for learning flight control skills by imitating expert demonstrations, focusing on essential elements such as pitch angle and pitch rate, achieving performance comparable to human experts [77]. Another study proposes combining behavior cloning, transfer learning, and reinforcement learning to develop robust, agile maneuver models capable of performing complex aerobatic maneuvers with limited pilot demonstration data [66].

One study presents a method for pilot performance modeling using observer-based IRL, demonstrating the ability to replicate a surrogate LQR pilot's cost functional and feedback matrix [78]. Another work explores IL for aerobatic maneuvering in fixed-wing aircraft [79]. Research involving a feed-forward neural network trained via behavior cloning for simple flight tasks highlights the need for diverse training data to develop robust policies [80]. Furthermore, there is research that applies AIRL with multimodal information, including instrument data (e.g., pitch, true airspeed) and visual data (e.g., runway angles and distances) to acquire expert piloting skills for landing [81]. Furthermore, a neural network autopilot has been investigated to mimic guidance, navigation, and control algorithms. By employing Dagger to mitigate data distribution mismatches, the autopilot achieves stable flight conditions effectively [82].

4. Others

This section highlights studies addressing various challenges in aviation, including control system design, air traffic management, and event prediction. One research addresses traditional brake control limitations by designing a model-free RL brake control algorithm, which decomposes the braking task and uses IRL to normalize the reward function, ensuring control safety and efficiency [83]. In adverse event prediction, the ADOPT algorithm is introduced to discover precursors in multivariate time series data, demonstrating promising results with real aviation data [84].

There also exist learning algorithms that leverage expert demonstrations to fly autonomous helicopters at an expert human pilot level [85, 86]. The system infers desired trajectories and builds control systems, outperforming state-of-the-art methods and even expert human pilots on various difficult maneuvers. Differential dynamic programming is employed for the control design of new aerobatic maneuvers, utilizing apprenticeship learning to determine the reward function and model [86].

Several papers contribute to managing aircraft traffic in the airspace and on the airport surface. One study develops a method for search-based planning using cost functions learned through inverse optimal control, encoding the implicit safety criteria of human ATC trajectories [9]. In another study, IRL and behavioral cloning (BC) models are compared for Ground Delay Program (GDP) implementation, revealing that BC models outperform IRL models [87]. Important insights are gained through feature importance scores, providing a deeper understanding of the decision-making process behind GDP implementations. Finally, GAIL models aircraft taxi speeds while considering operational factors like traffic and take-off time [13]. The resulting policy outperforms baseline models in spatial and temporal performance metrics.

Table 1 IRL works in aviation that primarily aim to mimic the decision-making strategies of observed expert behavior or collected data.

Problem Type	IRL Method	Expert Data	Reference
UAV Operations			
Control of drone swarms with limited vision and communication	Imitation Learning: Dagger [65]	Simulated data from NMPC controller with privileged information that would otherwise need communication or vision	Wan et al. [68]
Consistent UAV swarm control	Adversarial and IL: GAIL [43]	Real taxi data from Rome and a simulated platform	Yang et al. [69]

Problem Type	IRL Method	Expert Data	Reference
Controller for multirotor UAV	IRL using particle swarm optimization	Simulation dataset using LQR control	Choi et al. [67]
Joint path planning and power allocation of a cellular-connected UAV	Max Margin: Apprenticeship learning via IRL [18]	Expert in an open-source simulator environment	Shamsoshoara et al. [70]
Aerial combat maneuver strategy generation	MaxEnt IRL [33]	Generated aerial combat counter trajectories using simulations	Kong et al. [71]
Pilot Control Imitation			
Autonomous aerobatic helicopter flight and inference of trajectory preferences	Max Margin: Apprenticeship learning via IRL [18]	Human pilot flying desired maneuvers in a real helicopter	Abbeel and Coates [85, 86]
Mimic air traffic controllers	Max Entropy: Planning, inverse optimal control, and MaxEnt IRL [33]	Real airplane landings at Seattle-Tacoma airport from FlightAware	Tolstaya et al. [9]
Flight control of pitch angle in the presence of turbulence	Adversarial and IL: GAIL [43] and AIRL [46]	Expert data obtained from human-in-the-loop simulations	Takase et al. [77]
A pilot-assist system that recommends paths personalized to suit the pilots' preferences and skill levels	Regression: Regularized History Stack Observer (RHSO)	Simulated data from a surrogate LQR controller that mimics velocity commands sent by a remote controller to a quadcopter	Town et al. [78]
Autopilot for fixed-wing unmanned aerial systems	Imitation Learning: Dagger [65]	Simulated data from an intelligent path planning algorithm deciding waypoints	Shukla et al. [82]
Aircraft control and maneuver generation	Imitation Learning: Confidence-Dagger	Real pilot data from Turkish aerospace industry and open-source aircraft model	Sever et al. [66]
Fighter maneuvers for fixed-wing aircraft	Imitation Learning: Behavioral cloning using transfer learning	Human pilot data from a flight simulator	Sandstorm et al. [80]
Landing control of fixed-wing aircraft	Adversarial IRL	Human dataset using a flight simulator	Suzuki et al. [81]
Aerobatic maneuver of fixed-wing aircraft	Imitation Learning: Behavioral Cloning using LSTM	Human pilots provided sample maneuvers in simulation	Freitas et al. [79]
Others			
Aircraft anti-skid braking control to achieve both safety and efficiency under varying operational conditions	Max Margin: Apprenticeship learning via IRL [18]	Simulated data for braking performance under various working conditions	Jiao et al. [83]
Aircraft taxi speed modeling between two pre-defined locations	Adversarial and IL: GAIL [43]	Real-world aircraft ground movement taxiing data from the Singapore Changi Airport	Pham et al. [13]

Table 2 IRL works in aviation that primarily aim to predict a system's behavior by learning about the underlying objectives of observed expert behavior.

Problem Type	IRL method	Expert Data	Reference
Aircraft Trajectory Prediction			
Prediction of trajectories from origin to destination	Adversarial and IL: GAIL [43] along with clustering and classifying trajectories	Real flight trajectories from Barcelona to Madrid and corresponding weather data from NOAA	Bastas et al. [72]
Disentangle and identify different experts	Adversarial and IL: Multi-modal GAIL	Real flight trajectories from Paris to Istanbul	Spatharis et al. [74]

Problem Type	IRL method	Expert Data	Reference
Considers positional and meteorological features to predict entire trajectories	Max Margin: Quadratic programming	Real flight trajectories from Barcelona to Madrid and corresponding weather data from NOAA	Spatharis et al. [73]
Imitate and predict expert trajectories with different objectives	Adversarial and IL: GAIL [43]	Real flight trajectories (BCN-MAD, LHR-FCO, HEL-LIS / LIS-FRA)*	Kravaris et al. [75, 76]
Others			
Predict hourly ground delay program actions	Classification: Cascaded Supervised Inverse reinforcement learning (CSI) [56]	Real ground delay program data from EWR and SFO airports	Bloem et al. [87]
Precursor analysis in aviation for go-around	Max Margin: IRL solved as a linear programming problem	Real go-around and nominal flights landed at Dallas Ft. Worth International Airport	Janakiramen et al. [84]

V. Discussion

While Section IV shows there is a growing body of work applying IRL to aviation, as highlighted by Figure 3, there is still a notable gap in applications of IRL to aviation relative to other domains. We now present relevant datasets, simulations, and potential new applications to encourage the use of IRL in aviation.

A. Relevant Datasets

One reason IRL is difficult to apply in aviation is the limited number of datasets [88]. Here, we summarize key aviation datasets and how they can be utilized for IRL research. We focus on publicly available data applicable to learning models and data analysis, including the datasets used in the studies discussed in Section IV.B. Table 3 summarizes these datasets and the following sections discuss them in more detail.

1. Flight Operations

Flight operation data provide insights into aircraft movements, performance metrics, and air traffic flow management [89–94]. This data category is vital for understanding and modeling the dynamic behavior of flights using IRL. Flight operation data, such as aircraft trajectories, speed, and altitude, can be used to train IRL models to understand optimal flight patterns and air traffic control strategies. By analyzing the behavior of efficient flights, IRL can infer the reward functions that govern ideal flight paths, efficient fuel usage, and minimal delays. These models can support the development of advanced air traffic flow management systems that optimize flight routes, reduce congestion, and enhance overall flight performance.

2. Weather

Weather conditions heavily influence flight operations, making weather data essential for trajectory prediction and operational planning. Accurate weather data help IRL models predict and adapt under varying atmospheric conditions [95–99]. Weather data can be integrated into IRL models to understand how pilots and automated systems optimally respond to weather conditions. By observing actions taken during various weather scenarios, IRL can learn the implicit reward structures that guide decisions such as route changes, altitude adjustments, and speed variations. For example, an IRL model could combine historical flight data and weather conditions to mimic flight paths, minimizing fuel consumption and avoiding turbulence.

3. Airport

Data on airport capacity and usage are essential for improving airport efficiency. These data can, for example, help optimize the scheduling and sequencing of aircraft takeoffs and landings, which can be modeled using IRL techniques [100–102]. By understanding the reward structures associated with minimizing delays and optimizing resource use,

*Barcelona-El Prat Airport (BCN), Adolfo Suárez Madrid-Barajas Airport (MAD), London Heathrow International Airport (LHR), Rome Fiumicino Airport (FCO), Helsinki-Vantaa Airport (HEL), Lisbon-Humberto Delgado Airport (LIS), Frankfurt Airport (FRA)

IRL models can inform the development of algorithms that enhance air traffic flow management and overall airport efficiency. For example, An IRL model could be used to simulate different airport utilization scenarios to optimize aircraft spacing and sequencing, thereby reducing delays and increasing the overall efficiency of airport operations. Additionally, IRL can help design dynamic scheduling systems that adapt in real-time to varying airport conditions, such as sudden changes in weather or unexpected aircraft arrivals, further improving operational resilience.

4. Aviation Accidents

Accident datasets play a crucial role in safety analysis and learning from past incidents to enhance future operations. By examining the sequences of actions and events leading to accidents, these models can identify patterns and underlying cost functions that may have led to unsafe behaviors [103–108]. Understanding these cost functions can lead to developing new safety protocols to mitigate similar risks. Additionally, these models can simulate "what-if" scenarios, providing insights into how different actions might have prevented an incident, thus contributing to the design of more robust safety measures.

5. Audio Recordings for Air Traffic Management (ATM)

Audio data from air traffic control (ATC) communications help to shed light on pilot-controller interactions and improve speech recognition systems. These interactions can be transcribed and analyzed to understand communication patterns, which can then be modeled to improve decision-making systems in aviation [109–112]. Using IRL to study these interactions, researchers can infer the implicit rewards and decision-making criteria controllers use to manage air traffic safely and efficiently. IRL can be applied to learn optimal communication strategies for managing high-traffic situations, helping to develop systems that assist controllers in making quick and effective decisions.

6. Passenger

These datasets analyze passenger satisfaction, focusing on aspects such as in-flight services. Overall flight performance is also examined, including punctuality and baggage handling efficiency. Furthermore, the economic impact of operational strategies, such as dynamic pricing models or changes to flight frequency and routes, is assessed. Together, these insights contribute to a better understanding of the human factors and economic considerations essential for developing more robust aviation operation models [113, 114]. Through IRL, it is possible to model how airline decisions impact passenger satisfaction, schedule adherence, and economic outcomes. For instance, an IRL model could be utilized to analyze passenger feedback and ticket sales data to identify optimal pricing strategies that balance passenger satisfaction and airline revenue. By discerning the reward structures that prioritize these factors, IRL models can assist in optimizing schedules, enhancing in-flight services, improving operational efficiency, and maximizing economic benefits.

7. Aircraft and Drone Surveillance Data

These datasets support the surveillance and tracking of air vehicles through various imaging and detection technologies and are used for developing models that enhance detection and identification capabilities [115–119]. By learning from human experts and previous detections, IRL models can improve the accuracy and reliability of detection systems. For example, IRL can optimize aircraft detection under various lighting and weather conditions, enhance automatic surveillance systems, and improve air traffic control by providing real-time updates on aircraft movements. Additional applications include developing drone navigation systems for obstacle avoidance, detecting landing platforms, and optimizing search and rescue missions.

Table 3 Summary of various aviation datasets and their key features.

Dataset	**Description**	**Size**	**Last Updated**	**Availability**
Flight Operations				
OpenSky Network [89]	ADS-B data, state vector data, climbing segments, and more	23+ trillion messages (since 2013)	Continuously updated	Open (registration required)

Dataset	Description	Size	Last Updated	Availability
FlightAware [90]	Real-time aircraft tracking data and airport status	Over 30,000 terrestrial ADS-B receivers worldwide	Continuously updated	Open (live), paid API for historical
FAA Operations and Performance Data [91]	Traffic flow management and operational performance	Monthly IFR traffic data (~2,000 airports, city-pair level)	Continuously updated	Open (save/download with login)
Eurocontrol Demand Data Repository [92]	European air traffic demand and flow data	Monthly European traffic data	Continuously updated	Restricted (institutional access)
Plane Finder ADS-B Data [93]	Real-time flight tracking using ADS-B signals	86M+ positions per day	Continuously updated	Open (live), paid API for historical
FlightRadar24 [94]	Real-time flight tracking and runway usage data	Over 50,000 ADS-B ground receivers globally	Continuously updated	Open (live), paid API for historical
Weather				
Aviation Weather Center [95]	Real-time and forecasted weather conditions	Decades of global meteorological data	Continuously updated	Free API/web; account for advanced features
Meteoblue [96]	Aviation weather observations API	Decades of global meteorological data	Continuously updated	Commercial API
National Oceanic and Atmospheric Administration (NOAA) Metar [97]	Comprehensive weather data	Decades of global meteorological data	Continuously updated	Open (manual download)
AVWX Engine [98]	Aviation Weather parsing engine	-	Mar 2025	Open (basic API), paid plans available
Open-Meteo [99]	Open source weather API, providing accurate weather forecasts around the world	80 years historical archive (50 TB)	Continuously updated	Open (non-commercial)
Airports				
OpenAIP Airports [100]	Global airport data with airspace classification	~46,400 airports worldwide	Continuously updated	Open (API/download via account)
OurAirports [101]	Worldwide airport data	~29,500 airports worldwide	Continuously updated	Open
FAA Aviation Data [102]	Detailed U.S. airport and operation data	-	Continuously updated	Open (web downloads), API key required for some datasets
Aviation Accidents				
Aviation Safety Network [103]	Comprehensive coverage of aviation accidents since 1919	Over 23,000 accidents	Continuously updated	Open
FAA Accident and Incident Data [104]	Data on preliminary accidents, incidents, and runway incursions	Recent 10 business days	Continuously updated	Open
GitHub Aviation Accidents Data Repository [105]	Scripts for data extraction from sources like Aviation Safety Network and Plane Crash Info databases	Over 100k accident records	Feb 2019	Open
Plane Crash Map [106]	Visualizes aviation accidents in the USA	-	Oct 2018	Open

Dataset	Description	Size	Last Updated	Availability
Aviation Herald [107]	Daily reports on global aviation accidents and incidents with analysis and media	~31,700 articles	Continuously updated	Open
Bureau of Aircraft Accidents Archives [108]	Global accident database with summaries and causes of accidents	Over 37,000 accident records	Continuously updated	Open
Audio Recordings				
TartanAviation [109]	Multimodal dataset including ATC speech, images, and ADS-B trajectory data	3.1M images; 3,374 hr ATC speech; 661 days ADS-B trajectory	Feb 2023	Open
ATCOSIM Corpus [110]	Contains of ATC speech data from real-time simulations	10 hr ATC speech	May 2007	Open
ATCO2 Corpus [111]	Large-scale dataset for Automatic Speech Recognition and Natural Language Understanding research in ATC communications	5,281 hr ATC speech	Oct 2022	Full dataset via ELRA
LiveATC Recordings [112]	Archived recordings of actual ATC communications	-	Continuously updated	Open (archives kept 7 days)
Passengers				
UK Civil Aviation Authority Passenger Survey Report [113]	Data on UK air passengers including satisfaction scores	~3,500 repondents per wave	Feb 2025	Open, paid for full history
GitHub Airline Passenger Satisfaction Repository [114]	Data analysis project focusing on passenger satisfaction metrics	Over 120k passenger records	May 2023	Open
Surveillance Data				
AVOIDDS [115]	Labeled images of aircraft under various conditions	72,000 labeled images	Jun 2023	Open
FGVC-Aircraft [116]	Images of different aircraft model variants	10,200 images	Jun 2013	Open (non-commercial)
VisDrone [117]	Large-scale benchmark dataset covering frames and static images	Over 2.6M bounding boxes	Sep 2023	Open
Semantic Drone [118]	Semantic segmentation dataset for drone landing zone detection	40,169 labeled objects	Jan 2019	Open
University-1652 [119]	Dataset for geolocation estimation using drone images	50,218 images for 1,652 buildings	May 2025	Open (request required)

B. Relevant Simulations

Flight simulators provide controlled environments where various flight scenarios can be simulated, data can be collected, and models validated. They are crucial for developing and testing real-life IRL models in a safe and controlled setting. Flight simulators allow researchers to create custom scenarios that mimic real-world conditions in a cost-efficient manner [120–122]. By observing the actions taken by expert pilots within these scenarios, IRL can be employed to derive reward functions that explain expert behavior. These reward functions can then be used to develop automated flight control systems or improve pilot training programs. For instance, simulators can help model how pilots handle emergencies or adverse weather conditions, which are critical for enhancing safety and efficiency in aviation. Relevant simulators and their short descriptions are provided in Table 4.

Additionally, simulators enable the exploration of rare or dangerous flight conditions that are impossible to study through live flight tests [123–126]. They can be integrated with IRL algorithms to refine pilot assistance systems and automate routine tasks. Recent advancements in simulator fidelity, such as high-resolution terrain data and advanced weather simulation, have further enhanced their utility in research. These high-fidelity environments help bridge the gap between simulated and real-world conditions, making simulator-derived insights increasingly relevant. Moreover, simulators can include many variables such as aircraft type, flight path, and environmental conditions, offering a comprehensive toolkit for in-depth aviation studies [127–129].

Table 4 Summary of various flight simulator datasets and their primary applications.

Dataset	Description
HARFANG 3D Framework [120]	Versatile 3D engine for real-time visualization and simulation in aircraft modeling and air combat scenarios
X-Plane 11 Desktop [121]	Realistic flight simulator featuring detailed aircraft and global scenery
Microsoft AirSim [122]	High-fidelity simulator designed for training and testing autonomous systems built on Unreal Engine
Virtual BattleSpace [123]	Military-focused simulator, including a flight module
Prepar3D [124]	For military and commercial flight training with detailed aircraft and environments
DCS World [125]	Realistic combat scenarios and equipment
Aerofly FS [126]	Realistic simulator with good graphics
FlightGear [127]	Open-source flight simulator with high customization options
BlueSky Simulator [128]	Open-source simulation tool designed for air traffic management research
DYNAMO3 [129]	Aircraft 4D trajectory generation and optimization tool for ATM research

C. Potential Applications for IRL in Aviation

Expanding on the research areas discussed in Section IV.B, this section examines potential advancements using IRL in aviation. Additionally, it includes potential research applications using the datasets and simulators discussed in Sections V.A and V.B.

1. Advanced Air Traffic Management and Communication

Utilizing flight operation data and ATC communication data, the extraction and analysis of air traffic flow patterns can be accomplished through IRL models. This involves identifying patterns in data related to flight trajectories, speeds, altitudes, and communication exchanges between pilots and air traffic controllers. By imitating these patterns, IRL-based models can identify optimal air traffic management strategies. These models can enhance overall air traffic efficiency by dynamically adjusting flight paths and altitudes to avoid congested airspaces and minimize delays.

Combining flight operation data with weather data, IRL models can be developed to predict optimal flight trajectories under various meteorological conditions [72, 73, 87]. These models analyze historical flight data to understand how weather conditions, such as wind speed, turbulence, and visibility, affect flight paths and performance. By leveraging this analysis, IRL-based trajectory prediction models can minimize fuel consumption and delays.

IRL can also be employed to derive reward functions from efficient conflict resolution strategies within air traffic control. By systematically analyzing instances of past conflicts and their resolutions, IRL models can learn the implicit reward structures that guide optimal decision-making in conflict scenarios [9]. These reward functions can then be used to develop automated systems that are capable of resolving air traffic conflicts with minimal human intervention.

2. Pilot Training and Simulation

Expanding beyond merely mimicking pilot behavior [66, 78, 79], we suggest using high-quality flight simulators and expert pilot behavior data to enhance pilot training programs. By integrating detailed flight simulation tools with data on pilot actions and decisions, IRL models can recreate real-world flight conditions and challenges. These models can include various scenarios, such as emergencies, adverse weather conditions, and critical system failures. This approach allows trainee pilots to practice handling real-world situations in a controlled setting, building the skills and confidence to manage unexpected challenges.

Additionally, by continuously analyzing the specific strengths and areas for improvement of each pilot, IRL models can tailor training scenarios to address specific skills that require upkeep. For instance, if a pilot consistently struggles with handling turbulence, an IRL model could generate scenarios focusing on these skills. Furthermore, the adaptive nature of these programs means that as pilots improve, the difficulty and complexity of the scenarios can be adjusted accordingly, providing a continuously challenging and developmental training environment.

3. UAV Operations

Implementing IRL models to optimize UAV missions using visual data is a promising approach to enhance the coordination and efficiency of drone swarms [68]. Swarm intelligence involves the collective behavior of decentralized, self-organized UAVs working together to achieve complex tasks. By leveraging IRL, models can be trained to understand and replicate the strategies employed by highly efficient drone swarms in various scenarios [69, 70]. This includes optimizing search patterns, maximizing coverage, and ensuring effective communication among the UAVs. In complex operational environments, such as disaster zones or rugged terrains, IRL-based strategies can significantly improve a UAV swarm's ability to locate and assist in rescue missions.

Using flight simulation data, IRL can also be employed to develop and enhance autonomous UAV flight control systems [67, 71]. These systems ensure UAVs can navigate effectively without human intervention, particularly in challenging environments. IRL models can learn from simulated flight data to optimize algorithms for obstacle avoidance and route planning. By analyzing previous flight scenarios and learning from the actions that led to successful control, IRL models can improve decision-making, resulting in a more reliable and efficient flight control system that can adapt to various obstacles.

Finally, IRL-based drone navigation systems can be implemented to optimize obstacle avoidance and landing zone detection. Utilizing visual data analysis, these systems can learn to navigate complex environments precisely. Trained to identify and respond to potential obstacles, these models would ensure safe and efficient flight operations. By continuously learning from many flight scenarios, IRL-based navigation systems can enhance the autonomy and reliability of drones, allowing them to perform complex tasks, such as delivery, surveillance, and search and rescue operations, with minimal human input.

4. Aircraft Maintenance and Monitoring

Predictive maintenance involves analyzing real-time and historical data from various aircraft systems to identify patterns and predict potential failures before they occur. By incorporating IRL, these models can learn from the operational behavior of aircraft and the maintenance actions taken to prevent breakdowns. For example, IRL models could identify the optimal times for maintenance activities, balancing the need to perform maintenance. The use of IRL in predictive maintenance would enable airlines to develop schedules that maximize aircraft availability and reliability, ultimately reducing costs associated with unexpected failures and extensive repairs.

Similarly, IRL could be an effective tool for developing health monitoring systems for aircraft components, which are essential for maintaining aircraft safety. These systems assess the condition of key components by analyzing sensor data. IRL models could therefore be used to predict failures and identify issues before malfunctions occur. This proactive approach allows maintenance teams to perform necessary interventions during scheduled maintenance rather than reacting to unexpected failures. Consequently, it enhances safety and ensures the optimal functioning of aircraft parts, leading to more efficient and cost-effective operations.

5. Enhancing Airline Operations

Accurately predicting delays and taxi times is critical for efficient airport and airline operations [13], and this can be effectively achieved by combining airport operational data through IRL models. These models would analyze historical data to learn patterns and factors contributing to delays and extended taxi times, such as adverse weather conditions, air traffic congestion, and airport logistics. IRL can thus support airport management in making informed decisions to mitigate delays, improve gate assignments, predict hold times, and streamline taxiing procedures.

For airlines, these predictive capabilities can enhance turnaround efficiency, ensuring aircraft readiness for subsequent departures. Beyond operational improvements, IRL can also be applied to passenger data to optimize pricing strategies and schedule management. By learning from patterns in passenger booking behavior, demand fluctuations, and pricing sensitivity, IRL-based approaches can dynamically adjust ticket prices and manage seat inventory, maximizing revenue while meeting passenger demand.

6. Improving Flight Safety

Utilizing IRL models to analyze historical accident data could offer valuable insights into aviation safety by helping to identify risk patterns and contributing factors [84]. These models could process extensive datasets, including weather conditions, mechanical failures, pilot errors, and air traffic control communications, to learn from the sequences of events preceding accidents. By understanding these patterns, airlines and regulatory bodies could develop new safety protocols and preventive measures that address the root causes of accidents.

Recreating accident scenarios using flight simulators and analyzing potential alternative actions through IRL is another possible step towards preventing similar incidents in the future. Using IRL, these simulations could test various alternative scenarios and responses, and be used to identify which actions could have prevented past accidents. Such insights are especially useful in autonomous flight systems that make real-time decisions in case of emergencies. By integrating IRL, autonomous systems can learn to make informed, rapid decisions, improving their ability to handle real-world emergencies effectively.

VI. Conclusion

In conclusion, the aviation industry holds significant potential for the application of IRL methods. IRL offers a promising approach to learning from expert demonstrations and inferring underlying reward functions. These methods have already achieved impressive successes in related fields, such as robotics and autonomous driving. By leveraging IRL methods, the community can develop more robust, adaptable, and human-like autonomous systems for aviation while also understanding existing systems.

However, significant challenges remain, including handling complex and dynamic environments while assuring safety and reliability. These challenges are particularly relevant to aviation problems. While some work has introduced methods to mitigate these challenges, further research, as well as careful consideration of ethical and regulatory implications, is necessary for IRL methods to be deployed in aviation systems. To help encourage IRL research in aviation, we summarize relevant open-source datasets and simulations and put forth potential novel research applications. By continuing to overcome these hurdles and exploring new problems with IRL, we can contribute to the advancement of safe, efficient, and sustainable aviation systems.

Acknowledgments

The authors would like to thank Dr. Husni Idris for insightful and helpful discussions. This work was funded in part by NASA TTT Award 80NSSC23M0221 and ONR N00014-20-1-2249.

References

[1] Jiang, Y., Tran, T. H., and Williams, L., "Machine learning and mixed reality for smart aviation: Applications and challenges," *Journal of Air Transport Management*, Vol. 111, 2023, p. 102437. https://doi.org/https://doi.org/10.1016/j.jairtraman.2023.102437, URL https://www.sciencedirect.com/science/article/pii/S0969699723000807.

[2] Airbus A^3, "ADAM: Autonomous Drone Air Mobility," , 2024. URL https://acubed.airbus.com/projects/adam/, retrieved 2 December 2024.

[3] Federal Aviation Administration, "Info-Centric National Airspace System," , 2024. URL https://www.faa.gov/about/office_org/headquarters_offices/ang/icn, retrieved 2 December 2024.

[4] Johnson, M. G., Jung, Y. C., Szatkowski, G. N., and Scott, T. J., "Digital Information Platform (DIP) Overview," *NASA DIP RFI with Airlines for America (A4A)*, 2021. URL https://ntrs.nasa.gov/citations/20210019267.

[5] Eurocontrol, "Artificial Intelligence," , 2024. URL https://www.eurocontrol.int/artificial-intelligence, retrieved 2 December 2024.

[6] Deniz, S., and Wang, Z., "A Multi-Agent Reinforcement Learning Approach to Traffic Control at Future Urban Air Mobility Intersections," *AIAA SCITECH 2022 Forum*, American Institute of Aeronautics and Astronautics, San Diego, CA & Virtual, 2022. https://doi.org/10.2514/6.2022-1509, URL https://arc.aiaa.org/doi/10.2514/6.2022-1509.

[7] Ghosh, S., Laguna, S., Lim, S. H., Wynter, L., and Poonawala, H., "A Deep Ensemble Method for Multi-Agent Reinforcement Learning: A Case Study on Air Traffic Control," *Proceedings of the International Conference on Automated Planning and Scheduling*, Vol. 31, 2021, pp. 468–476. https://doi.org/10.1609/icaps.v31i1.15993, URL https://ojs.aaai.org/index.php/ICAPS/article/view/15993.

[8] Brittain, M., and Wei, P., "Autonomous Air Traffic Controller: A Deep Multi-Agent Reinforcement Learning Approach," *Reinforcement Learning for Real Life (RL4RealLife) Workshop in the 36 th International Conference on Machine Learning*, 2019, p. 10. https://doi.org/10.48550/arXiv.1905.01303.

[9] Tolstaya, E., Ribeiro, A., Kumar, V., and Kapoor, A., "Inverse Optimal Planning for Air Traffic Control," *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 7535–7542. https://doi.org/10.1109/IROS40897.2019.8968460, URL https://ieeexplore.ieee.org/abstract/document/8968460, iSSN: 2153-0866.

[10] Zhao, P., and Liu, Y., "Physics Informed Deep Reinforcement Learning for Aircraft Conflict Resolution," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 7, 2022, pp. 8288–8301. https://doi.org/10.1109/TITS.2021.3077572, URL https://ieeexplore.ieee.org/document/9430767/.

[11] Brittain, M., and Wei, P., "Scalable Autonomous Separation Assurance With Heterogeneous Multi-Agent Reinforcement Learning," *IEEE Transactions on Automation Science and Engineering*, Vol. 19, No. 4, 2022, pp. 2837–2848. https://doi.org/10.1109/TASE.2022.3151607, URL https://ieeexplore.ieee.org/document/9717997/.

[12] Brittain, M. W., Yang, X., and Wei, P., "Autonomous Separation Assurance with Deep Multi-Agent Reinforcement Learning," *Journal of Aerospace Information Systems*, Vol. 18, No. 12, 2021, pp. 890–905. https://doi.org/10.2514/1.I010973, URL https://arc.aiaa.org/doi/10.2514/1.I010973.

[13] Pham, D.-T., Tran, T.-N., Alam, S., and Duong, V. N., "A Generative Adversarial Imitation Learning Approach for Realistic Aircraft Taxi-Speed Modeling," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 3, 2022, pp. 2509–2522. https://doi.org/10.1109/TITS.2021.3119073, URL https://ieeexplore.ieee.org/abstract/document/9576546.

[14] Rodriguez-ramos, A., Sampedro, C., Bavle, H., Moreno, I. G., and Campoy, P., "A Deep Reinforcement Learning Technique for Vision-Based Autonomous Multirotor Landing on a Moving Platform," *Journal of Intelligent & Robotic Systems*, 2018, pp. 1010–1017. https://doi.org/10.1109/IROS.2018.8594472, publisher: Journal of Intelligent & Robotic Systems ISBN: 9781538680933.

[15] Bae, J. H., Jung, H., Kim, S., Kim, S., and Kim, Y.-D., "Deep Reinforcement Learning-Based Air-to-Air Combat Maneuver Generation in a Realistic Environment," *IEEE Access*, Vol. 11, 2023, pp. 26427–26440. https://doi.org/10.1109/ACCESS.2023.3257849.

[16] Sutton, R. S., and Barto, A. G., *Reinforcement Learning: An Introduction*, 2^{nd} ed., The MIT Press, 2018.

[17] Razzaghi, P., Tabrizian, A., Guo, W., Chen, S., Taye, A., Thompson, E., Bregeon, A., Baheri, A., and Wei, P., "A Survey on Reinforcement Learning in Aviation Applications," , Nov. 2022. URL http://arxiv.org/abs/2211.02147.

[18] Abbeel, P., and Ng, A. Y., "Apprenticeship learning via inverse reinforcement learning," *Proceedings of the twenty-first international conference on Machine learning*, Association for Computing Machinery, New York, NY, USA, 2004, p. 1. https://doi.org/10.1145/1015330.1015430, URL https://dl.acm.org/doi/10.1145/1015330.1015430.

[19] Ng, A. Y., and Russell, S. J., "Algorithms for Inverse Reinforcement Learning," *Proceedings of the Seventeenth International Conference on Machine Learning*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000, p. 663–670. URL https://dl.acm.org/doi/10.5555/645529.657801.

[20] Russell, S., "Learning agents for uncertain environments (extended abstract)," *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, Association for Computing Machinery, New York, NY, USA, 1998, p. 101–103. https://doi.org/10.1145/279943.279964.

[21] Natarajan, S., Kunapuli, G., Judah, K., Tadepalli, P., Kersting, K., and Shavlik, J., "Multi-Agent Inverse Reinforcement Learning," *2010 Ninth International Conference on Machine Learning and Applications*, 2010, pp. 395–400. https://doi.org/10.1109/ICMLA.2010.65.

[22] Piot, B., Geist, M., and Pietquin, O., "Learning from Demonstrations: Is It Worth Estimating a Reward Function?" *Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML/PKDD 2013)*, Prague, Czech Republic, 2013. https://doi.org/10.1007/978-3-642-40988-2_2.

[23] Ng, A. Y., Harada, D., and Russell, S. J., "Policy Invariance Under Reward Transformations: Theory and Application to Reward Shaping," *Proceedings of the Sixteenth International Conference on Machine Learning*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999, p. 278–287. URL https://dl.acm.org/doi/10.5555/645528.657613.

[24] Arora, S., and Doshi, P., “A survey of inverse reinforcement learning: Challenges, methods and progress,” *Artificial Intelligence*, Vol. 297, 2021, p. 103500. https://doi.org/10.1016/j.artint.2021.103500, URL https://www.sciencedirect.com/science/article/pii/S0004370221000515.

[25] Adams, S., Cody, T., and Beling, P. A., “A survey of inverse reinforcement learning,” *Artificial Intelligence Review*, Vol. 55, No. 6, 2022, pp. 4307–4346. https://doi.org/10.1007/s10462-021-10108-x.

[26] Barreto, A., Dabney, W., Munos, R., Hunt, J. J., Schaul, T., van Hasselt, H., and Silver, D., “Successor features for transfer in reinforcement learning,” *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Curran Associates Inc., Red Hook, NY, USA, 2017, p. 4058–4068. URL https://dl.acm.org/doi/10.5555/3294996.3295161.

[27] Ratliff, N. D., Bagnell, J. A., and Zinkevich, M. A., “Maximum margin planning,” *Proceedings of the 23rd International Conference on Machine Learning*, Association for Computing Machinery, New York, NY, USA, 2006, p. 729–736. https://doi.org/10.1145/1143844.1143936.

[28] Ratliff, N. D., Silver, D., and Bagnell, J. A., “Learning to search: Functional gradient techniques for imitation learning,” *Autonomous Robots*, Vol. 27, 2009, pp. 25–53. https://doi.org/10.1007/s10514-009-9121-3, URL https://api.semanticscholar.org/CorpusID:7137185.

[29] Syed, U., and Schapire, R. E., “A game-theoretic approach to apprenticeship learning,” *Proceedings of the 21st International Conference on Neural Information Processing Systems*, Curran Associates Inc., Red Hook, NY, USA, 2007, p. 1449–1456. URL https://dl.acm.org/doi/10.5555/2981562.2981744.

[30] Boularias, A., and Chaib-Draa, B., “Apprenticeship learning with few examples,” *Neurocomput.*, Vol. 104, 2013, p. 83–96. https://doi.org/10.1016/j.neucom.2012.11.002.

[31] Zhou, W., and Li, W., “Safety-Aware Apprenticeship Learning,” 2018. https://doi.org/10.48550/arXiv.1710.07983.

[32] Zhao, J., “Safety-Aware Multi-Agent Apprenticeship Learning,” 2022. https://doi.org/10.48550/arXiv.2201.08111.

[33] Ziebart, B. D., Maas, A., Bagnell, J. A., and Dey, A. K., “Maximum entropy inverse reinforcement learning,” *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 3*, AAAI Press, 2008, p. 1433–1438. URL https://dl.acm.org/doi/10.5555/1620270.1620297.

[34] Jaynes, E. T., “Information Theory and Statistical Mechanics,” *Phys. Rev.*, Vol. 106, 1957, pp. 620–630. https://doi.org/10.1103/PhysRev.106.620, URL https://link.aps.org/doi/10.1103/PhysRev.106.620.

[35] Wulfmeier, M., Ondruska, P., and Posner, I., “Maximum Entropy Deep Inverse Reinforcement Learning,” , Mar. 2016. https://doi.org/10.48550/arXiv.1507.04888, URL http://arxiv.org/abs/1507.04888.

[36] Wu, Z., Sun, L., Zhan, W., Yang, C., and Tomizuka, M., “Efficient Sampling-Based Maximum Entropy Inverse Reinforcement Learning with Application to Autonomous Driving,” , Jun. 2020. URL http://arxiv.org/abs/2006.13704.

[37] Gleave, A., and Toyer, S., “A Primer on Maximum Causal Entropy Inverse Reinforcement Learning,” , Mar. 2022. https://doi.org/10.48550/arXiv.2203.11409, URL http://arxiv.org/abs/2203.11409.

[38] Boularias, A., Kober, J., and Peters, J., “Relative Entropy Inverse Reinforcement Learning,” *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research, Vol. 15, edited by G. Gordon, D. Dunson, and M. Dudík, PMLR, Fort Lauderdale, FL, USA, 2011, pp. 182–189. URL https://proceedings.mlr.press/v15/boularias11a.html.

[39] Snoswell, A. J., Singh, S. P. N., and Ye, N., “Revisiting Maximum Entropy Inverse Reinforcement Learning: New Perspectives and Algorithms,” *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2020, pp. 241–249. https://doi.org/10.1109/SSCI47803.2020.9308391.

[40] Inga, J., Bischoff, E., Köpf, F., and Hohmann, S., “Inverse Dynamic Games Based on Maximum Entropy Inverse Reinforcement Learning,” , 2020. URL https://arxiv.org/abs/1911.07503.

[41] Neumeyer, C., Oliehoek, F. A., and Gavrila, D. M., “General-Sum Multi-Agent Continuous Inverse Optimal Control,” *IEEE Robotics and Automation Letters*, Vol. 6, No. 2, 2021, pp. 3429–3436. https://doi.org/10.1109/LRA.2021.3060411.

[42] Finn, C., Levine, S., and Abbeel, P., “Guided Cost Learning: Deep Inverse Optimal Control via Policy Optimization,” *Proceedings of the 33rd International Conference on Machine Learning*, JMLR.org, 2016, p. 49–58.

[43] Ho, J., and Ermon, S., "Generative adversarial imitation learning," *Proceedings of the 30th International Conference on Neural Information Processing Systems*, Curran Associates Inc., Red Hook, NY, USA, 2016, p. 4572–4580.

[44] Song, J., Ren, H., Sadigh, D., and Ermon, S., "Multi-Agent Generative Adversarial Imitation Learning," *Advances in Neural Information Processing Systems*, Vol. 31, Curran Associates, Inc., 2018.

[45] Henderson, P., Chang, W.-D., Bacon, P.-L., Meger, D., Pineau, J., and Precup, D., "OptionGAN: Learning Joint Reward-Policy Options Using Generative Adversarial Inverse Reinforcement Learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32, No. 1, 2018. https://doi.org/10.1609/aaai.v32i1.11775.

[46] Fu, J., Luo, K., and Levine, S., "Learning Robust Rewards with Adversarial Inverse Reinforcement Learning," *International Conference on Learning Representations*, 2018.

[47] Chen, J., Lan, T., and Aggarwal, V., "Hierarchical Adversarial Inverse Reinforcement Learning," *IEEE Transactions on Neural Networks and Learning Systems*, 2023, pp. 1–10. https://doi.org/10.1109/TNNLS.2023.3305983.

[48] Yu, L., Song, J., and Ermon, S., "Multi-Agent Adversarial Inverse Reinforcement Learning," *Proceedings of the 36th International Conference on Machine Learning*, PMLR, 2019, pp. 7194–7201.

[49] Sun, J., Yu, L., Dong, P., Lu, B., and Zhou, B., "Adversarial Inverse Reinforcement Learning With Self-Attention Dynamics Model," *IEEE Robotics and Automation Letters*, Vol. 6, No. 2, 2021, pp. 1880–1886. https://doi.org/10.1109/LRA.2021.3061397.

[50] Wang, P., Liu, D., Chen, J., Li, H., and Chan, C.-Y., "Decision Making for Autonomous Driving via Augmented Adversarial Inverse Reinforcement Learning," *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 1036–1042. https://doi.org/10.1109/ICRA48506.2021.9560907.

[51] Alsaleh, R., and Sayed, T., "Markov-game modeling of cyclist-pedestrian interactions in shared spaces: A multi-agent adversarial inverse reinforcement learning approach," *Transportation research part C: emerging technologies*, Vol. 128, 2021, p. 103191.

[52] Ramachandran, D., and Amir, E., "Bayesian Inverse Reinforcement Learning," *International Joint Conference on Artificial Intelligence*, 2007.

[53] Levine, S., Popovic, Z., and Koltun, V., "Nonlinear Inverse Reinforcement Learning with Gaussian Processes," *Advances in Neural Information Processing Systems*, 2011.

[54] Babes-Vroman, M., Marivate, V., Subramanian, K., and Littman, M., "Apprenticeship Learning About Multiple Intentions," *International Conference on Machine Learning*, 2011.

[55] Klein, E., Geist, M., Piot, B., and Pietquin, O., "Inverse Reinforcement Learning through Structured Classification," *Advances in Neural Information Processing Systems*, Vol. 25, edited by F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/559cb990c9dffd8675f6bc2186971dc2-Paper.pdf.

[56] Klein, E., Piot, B., Geist, M., and Pietquin, O., "A Cascaded Supervised Learning Approach to Inverse Reinforcement Learning," *ECML PKDD*, 2013.

[57] Garg, D., Chakraborty, S., Cundy, C., Song, J., Geist, M., and Ermon, S., "IQ-Learn: Inverse soft-Q Learning for Imitation," , Nov. 2022. https://doi.org/10.48550/arXiv.2106.12142, URL http://arxiv.org/abs/2106.12142.

[58] Abdulhai, M., Jaques, N., and Levine, S., "Basis for Intentions: Efficient Inverse Reinforcement Learning using Past Experience," , Aug. 2022. https://doi.org/10.48550/arXiv.2208.04919, URL http://arxiv.org/abs/2208.04919.

[59] Zakka, K., Zeng, A., Florence, P., Tompson, J., Bohg, J., and Dwibedi, D., "Xirl: Cross-embodiment inverse reinforcement learning," *Conference on Robot Learning*, PMLR, 2022, pp. 537–546.

[60] Kumar, S., Zamora, J., Hansen, N., Jangir, R., and Wang, X., "Graph Inverse Reinforcement Learning from Diverse Videos," *Proceedings of The 6th Conference on Robot Learning*, Proceedings of Machine Learning Research, Vol. 205, edited by K. Liu, D. Kulic, and J. Ichnowski, PMLR, 2023, pp. 55–66. URL https://proceedings.mlr.press/v205/kumar23a.html.

[61] Swamy, G., Wu, D., Choudhury, S., Bagnell, J. A., and Wu, Z. S., "Inverse reinforcement learning without reinforcement learning," *Proceedings of the 40th International Conference on Machine Learning*, JMLR.org, 2023.

[62] Kiran, B. R., Sobh, I., Talpaert, V., Mannion, P., Sallab, A. A. A., Yogamani, S., and Pérez, P., "Deep Reinforcement Learning for Autonomous Driving: A Survey," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 6, 2022, pp. 4909–4926. https://doi.org/10.1109/TITS.2021.3054625.

[63] Jin, H., and Kim, J., "Jeju Air black box data missing from crucial minutes before crash," , 2025. URL https://www.reuters.com/world/asia-pacific/south-korea-jeju-air-jet-blackboxes-stopped-recording-4-minutes-before-crash-2025-01-11/, retrieved 23 July 2025.

[64] Li, M. Z., and Ryerson, M. S., "Reviewing the DATAS of aviation research data: Diversity, availability, tractability, applicability, and sources," *Journal of Air Transport Management*, Vol. 75, 2019, pp. 111–130. https://doi.org/https://doi.org/10.1016/j.jairtraman.2018.12.004.

[65] Ross, S., Gordon, G., and Bagnell, D., "A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning," *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research, Vol. 15, PMLR, Fort Lauderdale, FL, USA, 2011, pp. 627–635.

[66] Sever, G. G., Demir, U., Satir, A. S., Sahin, M. C., and Ure, N. K., "An integrated imitation and reinforcement learning methodology for robust agile aircraft control with limited pilot demonstration data," *Aerospace Science and Technology*, Vol. 158, 2025, p. 109682. https://doi.org/https://doi.org/10.1016/j.ast.2024.109682.

[67] Choi, S., Kim, S., and Jin Kim, H., "Inverse reinforcement learning control for trajectory tracking of a multirotor UAV," *International Journal of Control, Automation and Systems*, Vol. 15, No. 4, 2017, pp. 1826–1834. https://doi.org/10.1007/s12555-015-0483-3.

[68] Wan, Y., Tang, J., and Zhao, Z., "Imitation Learning of Complex Behaviors for Multiple Drones with Limited Vision," *Drones*, Vol. 7, No. 12, 2023. https://doi.org/10.3390/drones7120704.

[69] Yang, B., Shi, H., and Xia, X., "Federated Imitation Learning for UAV Swarm Coordination in Urban Traffic Monitoring," *IEEE Transactions on Industrial Informatics*, Vol. 19, No. 4, 2023, pp. 6037–6046. https://doi.org/10.1109/TII.2022.3192675.

[70] Shamsoshoara, A., Lotfi, F., Mousavi, S., Afghah, F., and İsmail Güvenç, "Joint path planning and power allocation of a cellular-connected UAV using apprenticeship learning via deep inverse reinforcement learning," *Computer Networks*, Vol. 254, 2024, p. 110789. https://doi.org/https://doi.org/10.1016/j.comnet.2024.110789.

[71] Kong, W., Zhou, D., Yang, Z., Zhao, Y., and Zhang, K., "UAV Autonomous Aerial Combat Maneuver Strategy Generation with Observation Error Based on State-Adversarial Deep Deterministic Policy Gradient and Inverse Reinforcement Learning," *Electronics*, Vol. 9, No. 7, 2020. https://doi.org/10.3390/electronics9071121.

[72] Bastas, A., Kravaris, T., and Vouros, G. A., "Data Driven Aircraft Trajectory Prediction with Deep Imitation Learning," , 2020. URL https://arxiv.org/abs/2005.07960.

[73] Spatharis, C., Blekas, K., and Vouros, G. A., "Apprenticeship learning of flight trajectories prediction with inverse reinforcement learning," *11th Hellenic Conference on Artificial Intelligence*, Association for Computing Machinery, New York, NY, USA, 2020, p. 241–249. https://doi.org/10.1145/3411408.3411427.

[74] Spatharis, C., Blekas, K., and Vouros, G. A., "Modelling flight trajectories with multi-modal generative adversarial imitation learning," *Applied Intelligence*, Vol. 54, 2024, pp. 7118–7134. https://doi.org/10.1007/s10489-024-05519-6.

[75] Kravaris, T., Bastas, A., and Vouros, G. A., "Predicting aircraft trajectories via imitation learning," *Adaptive and Learning Agents Workshop at AAMAS*, 2021.

[76] Kravaris, T., and Vouros, G. A., "Transferable aircraft trajectory prediction with generative deep imitation learning," *International Journal of Data Science and Analytics*, 2024. https://doi.org/10.1007/s41060-024-00574-1.

[77] Takase, R., "Learning Flight Control Skills By Imitating Expert Demonstrations," *32nd Congress of the International Council of the Aeronautical Sciences*, Shanghai, China, 2021.

[78] Town, J., Morrison, Z., and Kamalapurkar, R., "Pilot Performance Modeling via Observer-Based Inverse Reinforcement Learning," *IEEE Transactions on Control Systems Technology*, Vol. 32, No. 6, 2024, pp. 2444–2451. https://doi.org/10.1109/TCST.2024.3410128.

[79] Freitas, H., Camacho, R., and Castro Silva, D., "Imitation learning for aerobatic maneuvering in fixed-wing aircraft," *Journal of Computational Science*, Vol. 81, 2024, p. 102343. https://doi.org/https://doi.org/10.1016/j.jocs.2024.102343.

[80] Sandström, V., Luotsinen, L., and Oskarsson, D., "Fighter Pilot Behavior Cloning," *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2022, pp. 686–695. https://doi.org/10.1109/ICUAS54217.2022.9836131.

[81] Suzuki, K., Uemura, T., Tsuchiya, T., Beppu, H., Hazui, Y., and Ono, H., "Learning Aircraft Pilot Skills by Adversarial Inverse Reinforcement Learning," *2023 Asia-Pacific International Symposium on Aerospace Technology (APISAT 2023) Proceedings*, Springer Nature Singapore, Singapore, 2024, pp. 1431–1441.

[82] Shukla, D., Keshmiri, S., and Beckage, N., "Imitation Learning for Neural Network Autopilot in Fixed-Wing Unmanned Aerial Systems," *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2020, pp. 1508–1517. https://doi.org/10.1109/ICUAS48674.2020.9213850.

[83] Jiao, Z., Bai, N., Sun, D., Liu, X., Li, J., Shi, Y., Hou, Y., and Chen, C., "Aircraft Anti-skid Braking Control Based on Reinforcement Learning," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 59, No. 6, 2023, pp. 9219–9232. https://doi.org/10.1109/TAES.2023.3319600.

[84] Janakiraman, V. M., Matthews, B., and Oza, N., "Discovery of Precursors to Adverse Events using Time Series Data," *Proceedings of the 2016 SIAM International Conference on Data Mining (SDM)*, 2016, pp. 639–647. https://doi.org/10.1137/1.9781611974348.72.

[85] Abbeel, P., Coates, A., Quigley, M., and Ng, A. Y., "An application of reinforcement learning to aerobatic helicopter flight," *Proceedings of the 20th International Conference on Neural Information Processing Systems*, MIT Press, Cambridge, MA, USA, 2006, p. 1–8.

[86] Coates, A., Abbeel, P., and Ng, A. Y., "Apprenticeship learning for helicopter control," *Commun. ACM*, Vol. 52, No. 7, 2009, p. 97–105. https://doi.org/10.1145/1538788.1538812.

[87] Bloem, M., and Bambos, N., "Ground Delay Program Analytics with Behavioral Cloning and Inverse Reinforcement Learning," *Journal of Aerospace Information Systems*, Vol. 12, No. 3, 2015, pp. 299–313. https://doi.org/10.2514/1.I010304.

[88] Li, M. Z., and Ryerson, M. S., "Reviewing the DATAS of aviation research data: Diversity, availability, tractability, applicability, and sources," *Journal of Air Transport Management*, Vol. 75, 2019, pp. 111–130. https://doi.org/https://doi.org/10.1016/j.jairtraman.2018.12.004.

[89] Schäfer, M., Strohmeier, M., Lenders, V., Martinovic, I., and Wilhelm, M., "Bringing Up OpenSky: A Large-scale ADS-B Sensor Network for Research," *Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, IEEE Press, 2014, pp. 83–94. https://doi.org/10.1109/IPSN.2014.6846743.

[90] FlightAware, "FlightAware: Flight Tracking / Flight Status," https://www.flightaware.com/, retrieved 22 October 2024.

[91] Federal Aviation Administration, "Traffic Flow Management System Counts," https://aspm.faa.gov/tfms/sys/main.asp, retrieved 22 October 2024.

[92] EUROCONTROL, "Demand Data Repository (DDR2)," https://www.eurocontrol.int/ddr, retrieved 22 October 2024.

[93] Plane Finder Ltd, "Plane Finder: Live Flight Tracker," https://planefinder.net/, retrieved 22 October 2024.

[94] Flightradar24 AB, "FlightRadar24: Live Flight Tracker," https://www.flightradar24.com/, retrieved 22 October 2024.

[95] National Weather Service, "Aviation Weather Center," https://www.aviationweather.gov/, retrieved 22 October 2024.

[96] Meteoblue AG, "Meteoblue Weather Data," https://www.meteoblue.com/, retrieved 22 October 2024.

[97] National Oceanic and Atmospheric Administration, "NOAA METAR Data," https://www.ncdc.noaa.gov/data-access/quick-links#metar, retrieved 22 October 2024.

[98] duPont, M., "AVWX-Engine," https://github.com/avwx-rest/avwx-engine, retrieved 22 October 2024.

[99] Open-Meteo.com, "Open-Meteo," https://open-meteo.com/, retrieved 22 October 2024.

[100] OpenAIP, "OpenAIP Airports Database," https://www.openaip.net/, retrieved 22 October 2024.

[101] Megginson, D., "OurAirports Database," https://ourairports.com/data/, retrieved 22 October 2024.

[102] Federal Aviation Administration, "Aviation Data & Statistics," https://www.faa.gov/data_research/aviation_data_statistics/, retrieved 22 Octobor 2024.

[103] Aviation Safety Network, “Aviation Safety Network Database,” https://aviation-safety.net/database/, retrieved 23 October 2024.

[104] Federal Aviation Administration, “Accident and Incident Data,” https://www.faa.gov/data_research/accident_incident, retrieved 23 October 2024.

[105] Hehnke, L., “aviation-accidents-data,” https://github.com/lhehnke/aviation-accidents-data, retrieved 23 October 2024.

[106] Baker, L. C., “Plane Crash Map,” https://www.planecrashmap.com/, retrieved 23 October 2024.

[107] Hradecky, S., “The Aviation Herald,” https://avherald.com/, retrieved 23 October 2024.

[108] Bureau of Aircraft Accidents Archives, “Bureau of Aircraft Accidents Archives,” https://www.baaa-acro.com/, retrieved 23 October 2024.

[109] Wang, W., Zhu, D., Wang, X., Hu, Y., Qiu, Y., Wang, C., Hu, Y., Kapoor, A., and Scherer, S., “TartanAir: A Dataset to Push the Limits of Visual SLAM,” *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 4909–4916. https://doi.org/10.1109/IROS45743.2020.9341801.

[110] Hofbauer, K., Petrik, S., and Hering, H., “The ATCOSIM Corpus of Non-Prompted Clean Air Traffic Control Speech,” *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC‘08)*, European Language Resources Association (ELRA), Marrakech, Morocco, 2008.

[111] Gomez, J. P. Z., Veselý, K., Szöke, I., Blatt, A., Motlicek, P., Kocour, M., Choukri, K., Nigmatulina, I., Cevenini, C., Tart, A., Cernocký, J., and Klakow, D., “ATCO2 corpus: A Large-Scale Dataset for Research on Automatic Speech Recognition and Natural Language Understanding of Air Traffic Control Communications,” *Journal of Data-centric Machine Learning Research*, 2024.

[112] Pascoe, D., “LiveATC Recordings,” https://www.liveatc.net/recordings.php, retrieved 23 October 2024.

[113] UK Civil Aviation Authority, “UK Aviation Consumer Survey,” Consumer research, Civil Aviation Authority, 2023.

[114] Ghosal, A., “Airline Passenger Satisfaction Analysis,” https://github.com/aryakghosal/airline-passenger-satisfaction-analysis, retrieved 23 October 2024.

[115] Smyers, E., Katz, S., Corso, A., and Kochenderfer, M., “AVOIDDS: A dataset for vision-based aircraft detection,” Stanford Digital Repository, 2023. https://doi.org/10.25740/hj293cv5980.

[116] Maji, S., Rahtu, E., Kannala, J., Blaschko, M., and Vedaldi, A., “Fine-Grained Visual Classification of Aircraft,” , 2013. URL https://arxiv.org/abs/1306.5151.

[117] Du, D., Zhu, P., Wen, L., Bian, X., Lin, H., Hu, Q., Peng, T., Zheng, J., Wang, X., Zhang, Y., Bo, L., Shi, H., Zhu, R., Kumar, A., Li, A., Zinollayev, A., Askergaliyev, A., Schumann, A., Mao, B., Lee, B., Liu, C., Chen, C., Pan, C., Huo, C., Yu, D., Cong, D., Zeng, D., Pailla, D. R., Li, D., Wang, D., Cho, D., Zhang, D., Bai, F., Jose, G., Gao, G., Liu, G., Xiong, H., Qi, H., Wang, H., Qiu, H., Li, H., Lu, H., Kim, I., Kim, J., Shen, J., Lee, J., Ge, J., Xu, J., Zhou, J., Meier, J., Choi, J. W., Hu, J., Zhang, J., Huang, J., Huang, K., Wang, K., Sommer, L., Jin, L., Zhang, L., Huang, L., Sun, L., Steinmann, L., Jia, M., Xu, N., Zhang, P., Chen, Q., Lv, Q., Liu, Q., Cheng, Q., Chennamsetty, S. S., Chen, S., Wei, S., Kruthiventi, S. S. S., Hong, S., Kang, S., Wu, T., Feng, T., Kollerathu, V. A., Li, W., Dai, W., Qin, W., Wang, W., Wang, X., Chen, X., Chen, X., Sun, X., Zhang, X., Zhao, X., Zhang, X., Zhang, X., Chen, X., Wei, X., Zhang, X., Li, Y., Chen, Y., Toh, Y. H., Zhang, Y., Zhu, Y., Zhong, Y., Wang, Z., Wang, Z., Song, Z., and Liu, Z., “VisDrone-DET2019: The Vision Meets Drone Object Detection in Image Challenge Results,” *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 213–226. https://doi.org/10.1109/ICCVW.2019.00030.

[118] Mostegel, C., Maurer, M., Heran, N., Puerta, J., and Fraundorfer, F., “Semantic drone dataset,” *Graz University of Technology Institute of Computer Graphics and Vision*, 2019.

[119] Zheng, Z., Wei, Y., and Yang, Y., “University-1652: A Multi-view Multi-source Benchmark for Drone-based Geo-localization,” Association for Computing Machinery, New York, NY, USA, 2020, p. 1395–1403. https://doi.org/10.1145/3394171.3413896.

[120] NWNC HARFANG, “HARFANG 3D Framework,” https://www.harfang3d.com/, retrieved 22 October 2024.

[121] Laminar Research, “X-Plane 11,” https://www.x-plane.com/, retrieved 22 October 2024.

[122] Shah, S., Dey, D., Lovett, C., and Kapoor, A., "AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles," *Field and Service Robotics*, Springer International Publishing, Cham, 2018, pp. 621–635. https://doi.org/https://doi.org/10.1007/978-3-319-67361-5_40.

[123] Bohemia Interactive Simulations, "Virtual BattleSpace 3," https://bisimulations.com/products, retrieved 22 October 2024.

[124] Lockheed Martin Corporation, "Prepar3D," https://www.prepar3d.com/, retrieved 22 October 2024.

[125] Eagle Dynamics, "Digital Combat Simulator World," https://www.digitalcombatsimulator.com/, retrieved 22 October 2024.

[126] IPACS, "Aerofly FS," https://www.aerofly.com/, retrieved 22 October 2024.

[127] FlightGear Project, "FlightGear Flight Simulator," https://www.flightgear.org/, retrieved 22 October 2024.

[128] Hoekstra, J., and Ellerbroek, J., "BlueSky ATC Simulator Project: An Open Data and Open Source Approach," *7th International Conference on Research in Air Transportation*, 2016.

[129] Dalmau, R., Melgosa, M., Vilardaga, S., and Prats, X., "A fast and flexible aircraft trajectory predictor and optimiser for atm research applications," *8th International Conference for Research in Air Transportation (ICRAT)*, 2018.