CEA Comité Español de Automática
XLVII JORNADAS DE AUTOMÁTICA CÓRDOBA 2026
UNIVERSIDADE DA CORUÑA

Jornadas de Automática

Temporal Expert-Based Reward Learning for Inverse Reinforcement Learning

Francisco J. Naranjo-Campos[a,*]; Juan G. Victores[a];
Almudena Alcaide[b]; Jesica Rivero[b]; Carlos Balaguer[a]

[a]*Robotics Lab, Departamento de Ingeniería de Sistemas y Automática, Universidad Carlos III de Madrid, C/ Butarque, nº15, Leganés, 28911, Madrid, Spain.*
[b]*Departamento de Tecnología Accesible e I+D, Dirección de Accesibilidad Universal e Innovación, Inserta Innovación, Fundación Once, C/ de Sebastián Herrera, nº15, 28012, Madrid, Spain.*

To cite this article: Naranjo-Campos, F. J., Victores, J. G., Alcaide, A., Rivero, J., Balaguer, C. 2026. Temporal Expert-Based Reward Learning for Inverse Reinforcement Learning. Jornadas de Automática, 47. https://doi.org/10.17979/ja-cea.2026.47.13710

Abstract

Learning reward functions from expert demonstrations removes the need for manual reward engineering in reinforcement learning applied to robotic manipulation. Existing methods, however, require trajectory quality annotations, episode success labels, or produce implicit rewards that are difficult to inspect. This paper presents TEXB-IRL (*Temporal EXpert-Based reward learning for Inverse Reinforcement Learning*), which learns a dense neural reward function directly from unlabeled expert demonstrations. TEXB-IRL exploits intra-trajectory temporal ordering through two complementary objectives: a temporal consistency loss that enforces monotonically increasing reward along expert trajectories, and an expert–agent separation loss that anchors the reward scale. The policy is optimized with PPO over the learned reward. Preliminary experiments in simulation with the PAL TIAGo++ robot show competitive results against state of the art algorithms.

Keywords: Inverse reinforcement learning, Imitation learning, Reward learning, Robotic manipulation, Deep reinforcement learning

Aprendizaje Temporal de Recompensas a partir de Expertos para Aprendizaje por Refuerzo Inverso

Resumen

El aprendizaje de funciones de recompensa a partir de demostraciones expertas elimina la necesidad de diseñar manualmente la recompensa en el aprendizaje por refuerzo aplicado a la manipulación robótica. Sin embargo, los métodos existentes requieren anotaciones de calidad sobre las trayectorias, etiquetas de éxito de episodio, o producen recompensas implícitas difíciles de inspeccionar. Este artículo presenta TEXB-IRL (*Temporal EXpert-Based reward learning for Inverse Reinforcement Learning*), que aprende una función de recompensa neuronal densa directamente a partir de demostraciones expertas sin etiquetar. TEXB-IRL explota el orden temporal intratrayectoria mediante dos objetivos complementarios: una pérdida de consistencia temporal que impone una recompensa monótonamente creciente a lo largo de las trayectorias expertas, y una pérdida de separación experto–agente que ancla la escala de la recompensa. La política se optimiza con PPO sobre la recompensa aprendida. Experimentos preliminares en simulación con el robot PAL TIAGo++ muestran resultados competitivos frente algortimos del estado del arte.

Palabras clave: Aprendizaje por refuerzo inverso, Aprendizaje por imitación, Aprendizaje de recompensa, Manipulación robótica, Aprendizaje por refuerzo profundo

*Corresponding author: frnaranj@ing.uc3m.es

1. Introduction

Deep reinforcement learning (RL) has shown strong results in continuous robotic control, but its performance depends on well-designed reward functions. Manually designing such rewards requires task-specific engineering that often fails to scale to new environments or robots with high-dimensional observations. This challenge is particularly severe in robotic manipulation, where diverse tasks, object configurations, and kinematic structures complicate reward specification. Expert demonstrations, by contrast, are often readily available from human operators or motion planners and directly encode the desired behavior. Inverse reinforcement learning (IRL) (Ng and Russell, 2000; Abbeel and Ng, 2004) addresses this problem by inferring reward functions directly from demonstrations.

Adversarial imitation methods such as GAIL (Ho and Ermon, 2016) and AIRL (Fu et al., 2018) encode the reward in a discriminator trained through expert–agent comparison, achieving strong results on continuous control benchmarks. However, the resulting reward remains implicit, being encoded in the discriminator rather than in an interpretable state-based reward function, which complicates failure diagnosis and deployment on physical robotic systems (Ravichandar et al., 2020). Temporally structured approaches such as T-REX (Brown et al., 2019) and TW-CIRL (Li et al., 2026) dispense with feature engineering by leveraging trajectory ordering, but require trajectory quality annotations or binary success labels. Our prior work, EXBAL-IRL (Naranjo-Campos et al., 2024), combines interpretability and temporal exploitation via *reverse discounting* over expert-centered features, achieving competitive results on TIAGo++ tasks. However, its dependence on linear feature representations limits scalability to higher-dimensional observations.

In this paper we present TEXB-IRL (*Temporal Expert-Based Reward Learning for Inverse Reinforcement Learning*), a method conceived as the successor to EXBAL-IRL that replaces hand-crafted feature representations with a learned neural reward function while preserving its expert-centered temporal intuition (Figure 1). TEXB-IRL is based on the observation that temporal ordering within expert trajectories provides a dense, annotation-free supervision signal: states appearing later in a successful demonstration are, on average, closer to task completion than earlier ones. The method formalizes this intuition through a differentiable intra-trajectory temporal ranking objective, complemented by an expert–agent separation loss that anchors the reward scale without requiring trajectory quality annotations, binary success labels, or manually engineered features. The main contributions are:

- (i) a deep IRL method based on *intra-trajectory temporal ranking* combined with an expert–agent separation mechanism;
- (ii) a formulation conceived as the natural successor to EXBAL-IRL, preserving its expert-centered temporal intuition while eliminating its scalability limitations; and
- (iii) a preliminary validation on TIAGo++ manipulation tasks in simulation, comparing against EXBAL-IRL, GAIL, and TW-CIRL.

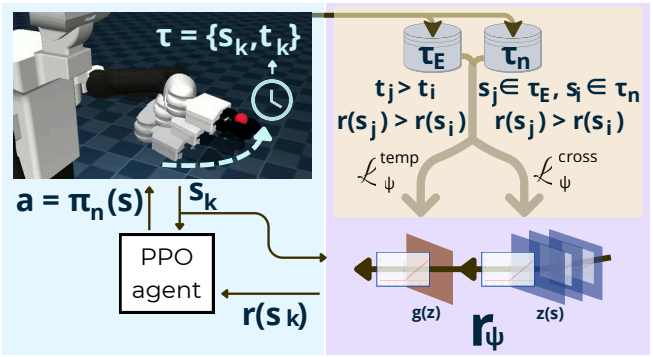

Figure 1: Overview of TEXB-IRL. *Left*: the PPO agent interacts with the environment receiving states s_k and learned rewards $r_\psi(s_k)$; expert demonstrations are stored with their normalized intra-trajectory timestep t_k. *Right (top)*: the reward model is updated via two ranking losses: $\mathcal{L}\psi^{\text{temp}}$, which enforces $r(s_j) > r(s_i)$ for $t_j > t_i$ within each expert trajectory, and $\mathcal{L}\psi^{\text{cross}}$, which separates expert states (τ_E) from agent states (τ_n). *Right (bottom)*: architecture of the reward model r_ψ, composed of an encoder $z(s)$ and a scalar head $g(z)$.

2. Related Work

2.1. IRL and imitation learning.

The problem of recovering a reward function from expert behavior was formalized by Ng and Russell (2000) and extended through AL-IRL (Abbeel and Ng, 2004), which infers the reward as a linear combination of features by matching expert–agent feature expectations. Maximum Entropy IRL (Ziebart et al., 2008) resolves reward ambiguity via the maximum entropy principle; Guided Cost Learning (Finn et al., 2016) extends this to deep networks through iterative policy samples that maximize reward for the expert over the agent. Adversarial methods GAIL (Ho and Ermon, 2016) and AIRL (Fu et al., 2018) reformulate reward learning as distribution matching and achieve competitive performance on continuous control benchmarks, although the implicit reward they produce complicates inspection and failure diagnosis on physical systems (Ravichandar et al., 2020).

2.2. Temporal structure in reward learning.

The temporal ordering within a demonstration provides a natural supervision signal. T-REX (Brown et al., 2019) and D-REX (Brown et al., 2020) learn the reward from comparisons between quality-ranked trajectories, enabling extrapolation beyond the demonstrated performance level but requiring trajectory-level ranking annotations. TW-CIRL (Li et al., 2026) uses the episode timestep as a direct supervision signal by assigning rewards proportional to the step in successful episodes, at the cost of requiring binary success/failure episode labels. TEXB-IRL learns solely from unlabeled demonstrations, with no quality or outcome annotations.

2.3. EXBAL-IRL as starting point.

EXBAL-IRL (Naranjo-Campos et al., 2024) introduced an approach based on features constructed from expert trajectories, together with a reverse discounting mechanism designed to preserve the relevance of goal states. In AL-IRL, the accumulated feature computation used to optimize the reward function typically applies a conventional discount factor that

prioritizes early trajectory states. However, in scenarios with few expert demonstrations and under the assumption that final states represent the goal pursued by the expert, this discount progressively reduces the influence of those terminal states. To mitigate this, EXBAL-IRL proposed a reverse discount that increases the importance of later trajectory samples.

The method was validated on manipulation tasks with the TIAGo++ robot, both in simulation and on the physical robot, demonstrating that expert-centered reward inference can be effectively applied in real robotic systems. Nevertheless, its main limitation lies in its dependence on a manually defined feature set, which hinders scalability to higher-dimensional observation spaces. TEXB-IRL addresses this limitation by replacing linear feature expectation matching with a learned *ranking* objective, retaining the temporal intuition of EXBAL-IRL and eliminating the need for manual feature engineering.

3. Methodology

TEXB-IRL learns a dense neural reward function from unlabeled demonstrations and uses it to optimize the policy. The framework jointly enforces two properties: (i) within each expert trajectory, states closer to task completion receive higher reward; (ii) expert states consistently receive higher reward than states visited by the current policy.

3.1. Expert demonstrations

Expert data consists of N demonstration trajectories:

$$\mathcal{D}_E = \{\tau_n\}_{n=1}^{N}, \quad \tau_n = (s_1^n, \ldots, s_k^n, \ldots, s_{T_n}^n) \tag{1}$$

Each state s_k^n is annotated with a normalized intra-trajectory timestep:

$$t_k^n = \frac{k}{T_n}, \quad t_k^n \in [0, 1] \tag{2}$$

which encodes task progress without requiring quality labels. The expert buffer stores tuples (s^E, t^E, n), preserving trajectory identity for intra-trajectory pair sampling.

3.2. Reward function

The reward function $r_\psi : \mathcal{S} \to \mathbb{R}$ is parameterized as a two-stage multilayer perceptron (MLP):

$$z = \phi_\psi(s), \qquad r_\psi(s) = g_\psi(z) \tag{3}$$

The encoder ϕ_ψ maps states through a fully connected neural architecture with ReLU activations, producing a compact latent representation suitable for continuous control observations. A separate reward head g_ψ maps this latent representation to a scalar reward value. Gradient clipping with norm 0.5 is applied during reward model updates to stabilize optimization and prevent large updates caused by strongly violated ranking constraints.

3.3. Learning objective

The loss function consists of two terms: a temporal term and an expert–agent separation term. Both losses use the *softplus* function $\sigma_+(x) = \ln(1 + e^x)$, a smooth relaxation of the ReLU hinge loss that preserves non-zero gradients near the margin boundary and improves optimization stability.

The temporal loss enforces monotonic reward progression within expert trajectories. Pairs (s_i, s_j) are sampled from the *same* trajectory with $t_j > t_i$ and state-space separation $\|s_j - s_i\| > \delta$, to exclude spatially identical pairs despite different timestamps:

$$\mathcal{L}_{\text{temp}} = \mathbb{E}_{(s_i,s_j)}\Big[\, \sigma_+\Big(\alpha(t_j - t_i) - \big(r_\psi(s_j) - r_\psi(s_i)\big)\Big)\Big] \tag{4}$$

where $\alpha > 0$ controls the required reward margin per unit of temporal distance. The loss vanishes when $r_\psi(s_j) - r_\psi(s_i) \geq \alpha(t_j - t_i)$. Restricting pairs to the same trajectory is essential: across different trajectories, a later timestep does not necessarily imply greater task progress.

Expert states must receive strictly higher reward than states visited by the current policy:

$$\mathcal{L}_{\text{cross}} = \mathbb{E}_{(s^E,s^\pi)}\Big[\, \sigma_+\Big(m_c - \big(r_\psi(s^E) - r_\psi(s^\pi)\big)\Big)\Big] \tag{5}$$

where $m_c > 0$ is a fixed margin. This term prevents degenerate solutions (e.g., a constant reward that trivially satisfies the temporal constraint) and anchors the scale of the learned reward to the expert–agent distinction.

Finally, the joint loss is:

$$\mathcal{L} = \mathcal{L}_{\text{temp}} + \lambda\, \mathcal{L}_{\text{cross}} \tag{6}$$

where λ weights the temporal structure against expert–agent separation. This formulation combines dense temporal supervision with policy discrimination, enabling reward learning without trajectory-level annotations or manually engineered features.

3.4. Policy optimization

At each timestep the environment reward is replaced by $r_t = r_\psi(s_t)$. The policy $\pi_\phi(a \mid s)$ is optimized with PPO (Schulman et al., 2017). Agent states s^π for $\mathcal{L}_{\text{cross}}$ are drawn directly from the PPO rollout buffer, requiring no additional replay mechanism. Training alternates between:

- Policy step: collect rollouts with π_ϕ; assign $r_t = r_\psi(s_t)$.
- Reward update: sample from $\mathcal{D}_E$ and the rollout buffer; minimize $\mathcal{L}$ for K steps (Adam, gradient clipping).

3.5. Algorithm

Algorithm 1: TEXB-IRL Training

Input: Buffer $\mathcal{D}_E$; parameters $\alpha, m_c, \lambda, \delta$; update freq F; steps K
Init: r_ψ, π_ϕ randomly
repeat
1. Collect rollouts with π_ϕ; assign $r_t \leftarrow r_\psi(s_t)$
2. **if** F steps since last reward update **then**
   2a. Sample $\{(s^E, t^E, n)\}$ from $\mathcal{D}_E$
   2b. Sample $\{s^\pi\}$ from PPO rollout buffer
   2c. Minimize $\mathcal{L} = \mathcal{L}_{\text{temp}} + \lambda\, \mathcal{L}_{\text{cross}}$ (K steps, Adam)
3. Update π_ϕ via PPO with reward r_ψ

until convergence

4. Experiments

We evaluate TEXB-IRL on both low-dimensional and robotic manipulation tasks to assess reward interpretability, learning stability, and policy performance under learned rewards alone. Comparisons are conducted against EXBAL-IRL, GAIL, and TW-CIRL using identical PPO training settings.

4.1. Experimental Setup

TEXB-IRL is evaluated in two Gymnasium-based [1] environments with complementary roles:

- Pendulum-v1 [2] serves as a fast-convergence benchmark. The observation is $(\cos\theta, \sin\theta, \dot{\theta})$, corresponding to the effective state space $(\theta, \dot{\theta})$. The objective is to stabilize the pendulum in the upright equilibrium ($\theta = 0, \dot{\theta} = 0$) while minimizing angular velocity and control effort. Its low-dimensional structure enables direct visualization of the learned reward landscape and qualitative comparison against the ground-truth environment reward.
- TIAGo Reach: the PAL Robotics TIAGo++ [3] robot must reach a random Cartesian target using only the learned reward r_ψ, replicating the reference scenario of EXBAL-IRL (Naranjo-Campos et al., 2024). The robot is simulated in MuJoCo [4] and the 7-DoF right arm is controlled via Cartesian delta commands resolved through KDL inverse kinematics. The action space consists of end-effector pose increments $(\Delta x, \Delta y, \Delta z, \Delta\phi, \Delta\theta, \Delta\psi) \in [-1, 1]^6$. Targets are sampled uniformly in $x \in [0.60, 0.75]$ m, $y \in [-0.30, 0.00]$ m, and $z \in [0.65, 0.85]$ m, with orientation in $[-\pi/4, \pi/4]$ per axis. Observations are 18-dimensional, composed of end-effector pose (6), velocity (6), and target pose (6). The ground-truth environment reward is

$$r_{env} = -(d_{pos} + 0.1\, d_{rot}), \tag{7}$$

where d_{pos} is the Euclidean distance to the target and d_{rot} the L2 distance between rotation vectors. This reward is used exclusively for evaluation; during training it is replaced by r_ψ. A visualization of the TIAGo Reach environment is shown in Figure 2.

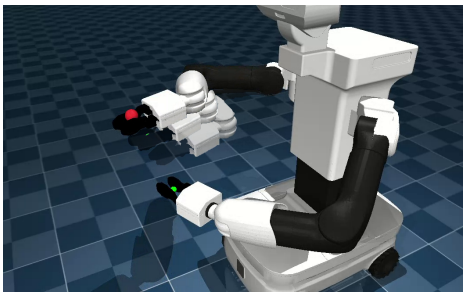

Figure 2: TIAGo Reach environment in MuJoCo. The TIAGo++ robot must reach randomly sampled Cartesian targets using Cartesian end-effector control. Colored markers indicate target positions (red) and end-effector (green) references during expert trajectory generation.

Expert demonstrations are collected using a proportional controller driving the end-effector toward the task goal. Each demonstration is stored as a sequence of states with normalized timesteps $t_k = k/T$.

TEXB-IRL is compared against three baselines:

- EXBAL-IRL (Naranjo-Campos et al., 2024): direct predecessor based on reverse discounting over expert-centered handcrafted features.
- GAIL (Ho and Ermon, 2016): adversarial imitation learning baseline using a discriminator-derived reward.
- TW-CIRL (Li et al., 2026): temporally supervised contrastive IRL approach requiring episode success/failure labels.

All methods use the same PPO policy optimizer, network architecture, and expert demonstrations for fair comparison. The TEXB-IRL reward model employs a 128-dimensional MLP encoder trained with Adam. Unless otherwise stated, experiments use $\alpha = 0.05$, $m_c = 0.05$, and $\lambda = 1.0$.

4.2. Pendulum-v1 Evaluation

Figure 3 reports the original environment reward used for evaluation. All four methods converge to the $[-1, -2]$ range by the end of training, differing mainly in sample efficiency and stability. GAIL converges fastest, though with wide variance across seeds. TEXB-IRL follows closely, stabilizing with progressively reduced variance. EXBAL-IRL converges more slowly from an initial range of -6 to -8, reaching comparable performance with low variance. TW-CIRL reaches similar final performance but exhibits pronounced oscillations and consistently high variance, reflecting unstable optimization under binary episode-success supervision.

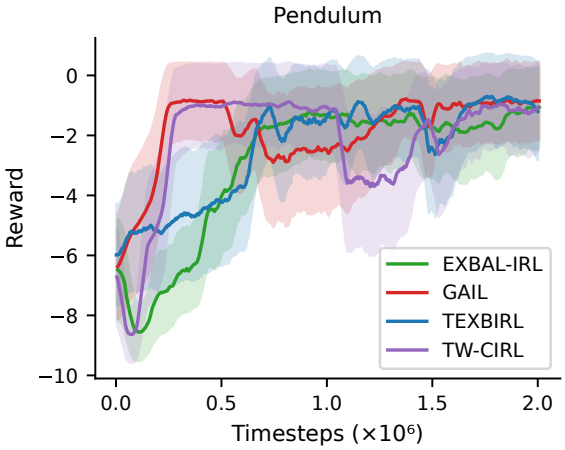

Figure 3: Learning curves on Pendulum-v1. Original environment reward over training timesteps (mean ± standard deviation over 5 random seeds).

[1] See `https://gymnasium.farama.org/`, accessed on May 11, 2026.
[2] See `https://gymnasium.farama.org/environments/classic_control/pendulum/`, accessed on May 10, 2026.
[3] See `https://github.com/pal-robotics-forks/mujoco_menagerie/tree/140ae8d30b430d9d8d8f0c42e031b93b59cb2968/pal_tiago_dual`, accessed on May 10, 2026.
[4] See `https://mujoco.readthedocs.io/en/stable/overview.html`, accessed on May 10, 2026.

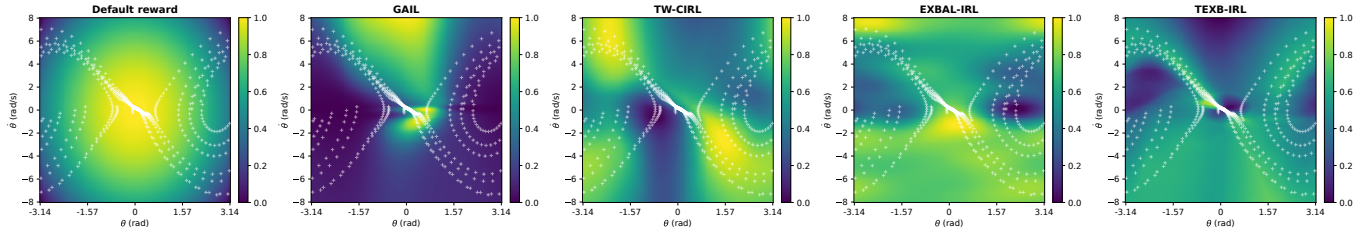

Figure 4: Learned reward landscapes on Pendulum-v1 in the $(\theta, \dot{\theta})$ state space. White crosses mark expert state visits. From left to right: default environment reward, GAIL, TW-CIRL, EXBAL-IRL, and TEXB-IRL. All values normalized to [0, 1] per panel.

Figure 4 shows the learned reward landscapes in the $(\theta, \dot{\theta})$ space. The environment reward forms a plateau at the upright configuration ($\theta \approx 0$). TEXB-IRL recovers the closest qualitative structure: high reward concentrates near $\theta \approx 0$ and propagates along the expert trajectory envelope, consistent with the temporal ranking objective assigning higher reward to states closer to task completion. EXBAL-IRL produces a V-shaped pattern equally aligned with the expert paths. GAIL concentrates high reward in a narrower, partially shifted region. TW-CIRL shows the least coherent landscape, with the high-reward region displaced away from the upright configuration.

4.3. TIAGo Reach Evaluation

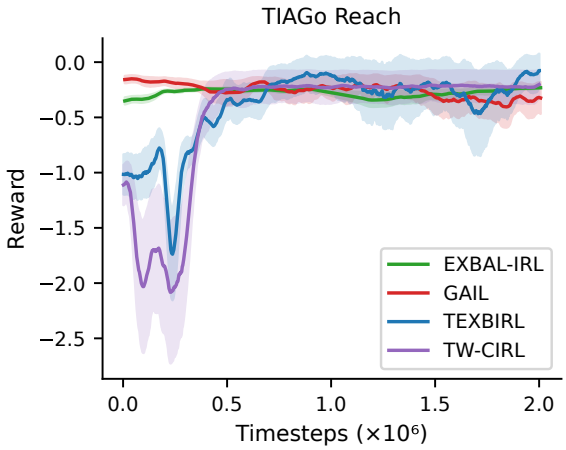

Figure 5: Learning curves on TIAGo Reach. Original environment reward over training timesteps (mean ± standard deviation over 5 random seeds).

Figure 5 shows the original environment reward used for evaluation. EXBAL-IRL achieves the best initial performance, owing to its handcrafted features that provide an immediately informative reward signal and enable stable optimization from the beginning of training. GAIL exhibits similarly stable behavior, converging with low variance across seeds. TEXB-IRL undergoes an initial performance drop as the reward model bootstraps from interaction data, but progressively recovers and ultimately matches the final performance of EXBAL-IRL and GAIL. In contrast, TW-CIRL shows a steeper degradation during early training and only partial recovery, stabilizing at a lower performance level for the remainder of training.

5. Discussion

The results show that the temporal structure implicit in expert demonstrations can serve as a sufficient signal for learning useful reward functions without manual feature engineering. TEXB-IRL retains the temporal intuition of EXBAL-IRL while translating it to a deep, scalable framework.

On Pendulum-v1, all methods reach comparable final performance; however, TEXB-IRL recovers the reward landscape whose qualitative structure most closely follows the expert trajectories around the upright equilibrium region. This suggests that the temporal ranking objective captures meaningful notions of task progress directly from demonstration ordering, which is further reflected in the training dynamics. GAIL achieves the fastest initial convergence but with high variance across seeds. TW-CIRL reaches a similar final reward level yet exhibits the most unstable training profile, with persistent oscillations throughout training that suggest sensitivity to the binary episode-success labels it requires.

On TIAGo Reach, EXBAL-IRL and GAIL exhibit faster initial convergence thanks to structured signals available from the start via handcrafted features and adversarial training, respectively. TEXB-IRL instead requires an initial adaptation phase to jointly learn the latent representation and the reward function. Once past this phase, it reaches comparable performance, demonstrating that intra-trajectory temporal ordering contains sufficient information to guide learning in continuous robotic tasks.

Furthermore, the weaker behavior of TW-CIRL suggests that modeling progress through ordinal relations between states is more robust than binary success/failure episode supervision, especially in scenarios with variable goals.

Overall, the experiments indicate that TEXB-IRL eliminates the dependence on manual features while maintaining competitive performance, at the cost of a higher bootstrapping cost during the early training phase.

6. Conclusions

This work presented TEXB-IRL, a deep inverse reinforcement learning method that learns dense neural reward functions from unlabeled demonstrations without manual feature engineering or additional supervision. The method extends the temporal intuition of EXBAL-IRL to a deep ranking-based

framework, removing the dependence on handcrafted representations while preserving expert-centered temporal reasoning.

Results on Pendulum-v1 and TIAGo++ reaching tasks show that TEXB-IRL can learn effective reward signals directly from the temporal ordering of expert demonstrations. In addition to achieving competitive policy performance, TEXB-IRL learns reward landscapes whose qualitative structure closely follows expert behavior and task progression, indicating that temporal ranking alone provides meaningful supervision for reward learning.

Although TEXB-IRL exhibits a slower initial adaptation phase than methods with explicit structural priors, it ultimately achieves performance comparable to EXBAL-IRL and GAIL, while providing greater scalability to higher-dimensional observation spaces and more stable convergence than TW-CIRL on Pendulum-v1.

Future work will extend validation to robotic tasks involving contact and more complex dynamics, formally analyze the theoretical properties of the temporal ranking objective, and evaluate robustness to the sim-to-real gap (e.g., via observation and actuation noise injection), prior to transferring the method to the physical TIAGo++ robot toward real-world deployment.

Acknowledgements

This research was funded through the R&D activities programme with reference TEC-2024/TEC-62 and acronym iRoboCity2030-CM, granted by the Community of Madrid through the Directorate General for Research and Technological Innovation via Order 5696/2024; by Asociación Inserta Innovación (part of Grupo Social ONCE) within the project "Robosoasis. New assistive and social robotics technology to improve autonomy in real environments", 2025/00715/001; and by Universidad Carlos III de Madrid (Escuela de Doctorado).

References

Abbeel, P., Ng, A. Y., 2004. Apprenticeship learning via inverse reinforcement learning. In: Proceedings of the 21st International Conference on Machine Learning (ICML). ACM.

Brown, D. S., Coleman, R., Srinivasan, R., Niekum, S., 2020. Safe imitation learning via fast Bayesian reward inference from preferences. In: Proceedings of the 37th International Conference on Machine Learning (ICML). pp. 1165–1177.

Brown, D. S., Goo, W., Narayanan, P., Niekum, S., 2019. Extrapolating beyond suboptimal demonstrations in inverse reinforcement learning from observations. In: Proceedings of the 36th International Conference on Machine Learning (ICML). pp. 783–792.

Finn, C., Levine, S., Abbeel, P., 2016. Guided cost learning: Deep inverse optimal control via policy optimization. In: Proceedings of the 33rd International Conference on Machine Learning (ICML). pp. 49–58.

Fu, J., Luo, K., Levine, S., 2018. Learning robust rewards with adversarial inverse reinforcement learning. In: International Conference on Learning Representations (ICLR).

Ho, J., Ermon, S., 2016. Generative adversarial imitation learning. In: Advances in Neural Information Processing Systems (NeurIPS). Vol. 29.

Li, Y., Gao, Y., Yang, N., Xia, S., 2026. TW-CRL: Time-weighted contrastive reward learning for efficient inverse reinforcement learning. Proceedings of the AAAI Conference on Artificial Intelligence 40 (28), 23301–23309. DOI: 10.1609/aaai.v40i28.39499

Naranjo-Campos, F. J., et al., 2024. Expert-trajectory-based features for apprenticeship learning via inverse reinforcement learning. Applied Sciences 14 (24), 11131. DOI: 10.3390/app142411131

Ng, A. Y., Russell, S. J., 2000. Algorithms for inverse reinforcement learning. In: Proceedings of the 17th International Conference on Machine Learning (ICML). Morgan Kaufmann, pp. 663–670.

Ravichandar, H., Polydoros, A. S., Chernova, S., Billard, A., 2020. Recent advances in robot learning from demonstration. Annual Review of Control, Robotics, and Autonomous Systems 3, 297–330. DOI: 10.1146/annurev-control-100819-063206

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.

Ziebart, B. D., Maas, A., Bagnell, J. A., Dey, A. K., 2008. Maximum entropy inverse reinforcement learning. In: Proceedings of the 23rd AAAI Conference on Artificial Intelligence. pp. 1433–1438.