# Reinforcement Learning Project

Anonymous Authors (recall: the entire document should be anonymous)

***Abstract*—This is a rough guide to producing the project report. The structure outlined here is a suggestion, but you must use this template[1] (of course, replace these hints/instructions/examples with your own text); a limit of 5 pages *not including* references and an optional appendix.**

## I. Introduction ($\approx$ 1 page)

Reinforcement Learning (RL) has become a powerful paradigm for developing autonomous agents that learn optimal behaviors through interactions with their environments. In this study, we employ the CarRacing-v3 environment provided by Gymnasium, which presents a challenging control task in a racing scenario. The environment is characterized by a high-dimensional observation space and two distinct modes for the action space. Specifically, the observation space consists of a top-down $96 \times 96$ RGB image capturing both the car and the race track, thus requiring the use of deep convolutional neural networks (CNNs) for effective feature extraction.

Regarding the action space, CarRacing-v3 supports both continuous and discrete control modalities. In the continuous mode, the agent outputs three real-valued commands: steering, where values range from $-1$ (full left) to $+1$ (full right); gas; and braking. Conversely, in the discrete mode, the action space is reduced to five actions: do nothing, steer left, steer right, gas, and brake. This duality in action representation allows for a comprehensive evaluation of various RL algorithms under different control settings.

The reward structure of the environment underscores the challenge by combining two components: a penalty of $-0.1$ per frame and a reward of $+\frac{1000}{N}$ for each new track tile visited, where $N$ represents the total number of track tiles. For example, completing the race after visiting all $N$ tiles in 732 frames, results in a reward of $1000 - 0.1 \times 732 = 926.8$ points, as shown in [1]. This scheme incentivizes the agent to balance exploration (visiting tiles) with efficiency (minimizing frame usage), aligning its learning objectives with the task's overarching goal.

The primary objective of this project is to investigate and compare different RL policies across both discrete and continuous action modalities. For discrete action control, we implement methods such as Deep Q-Network (DQN) and SARSA. In contrast, for continuous action control, we explore approaches like the Cross-Entropy Method (CEM) and Self-Adaptive Evolution Strategy (SAES), and we also consider incorporating policy gradient techniques (e.g., Proximal Policy Optimization (PPO) and Soft Actor-Critic (SAC)). This comparative analysis is driven by our interest in understanding the strengths and limitations of each method in handling complex decision spaces.

The dual nature of the action space in CarRacing-v3 presents a significant challenge. When dealing with high-dimensional visual inputs, the necessity of effective feature extraction becomes paramount. To address this, our approach includes the development of a convolutional neural network architecture tailored to process the $96 \times 966$ RGB images, reducing their dimensionality while preserving essential spatial features required for decision making. Additionally, transitioning between discrete and continuous representations of actions requires careful algorithmic design and parameter tuning to ensure stable learning and convergence.

While previous studies have applied various RL techniques in simulated environments, many have tended to focus on either discrete or continuous action spaces separately. In our work, we adopt a comparative approach by evaluating different agents within the same CarRacing-v3 environment. This allows us to assess the performance of each method under similar conditions, examining aspects such as learning stability, computational complexity, and overall policy effectiveness.

At this initial stage, this work primarily outlines the methodology and anticipated challenges, rather than presenting final empirical results. Our approach involves designing the CNN-based feature extractor, implementing the chosen RL algorithms, and setting up a robust framework for comparing their performances. While our preliminary findings are yet to be finalized, we expect that this study will offer valuable insights into the practical implications of employing RL in high-dimensional, real-time control tasks.

Specific limitations of the current work include the preliminary nature of our experiments and the need for further tuning and validation. Future work will focus on extensive empirical evaluations, exploring additional policy gradient methods, and refining the network architecture to better handle the complexities of the CarRacing-v3 environment.

The code for this project is available at `[TEM QUE COLOCAR O NOSSO LINK AQUI, NÃO ESQUECER]`, providing a reproducible framework for future investigations and extensions of this work.

## II. Background ($\approx$ 1 page)

Here, tell the reader everything they need to know to understand your work, in your own words. The main emphasis in this section: be pedagogical, target your reader as someone who has followed the course, and needs to be reminded of relevant concepts to understand what you have done.

You must properly credit any source (textbook, article, course notes/slides) via a suitable reference, e.g., [2], Chapter 3, or [3], or even a blog post you found on the web [**?**].

[1] There are also Word templates for this format (https://www.ieee.org/conferences/publishing/templates.html) if you wish; but don't forget you will submit a pdf

Here is the place to introduce your notation, (e.g., state $s$, policy $\pi_\theta$ parametrized by $\theta$, trajectory $\boldsymbol{\tau} \sim p_\theta$ from policy $\pi_\theta$ under environment $p$), but make sure each part of your notation is stated somewhere.

## III. Methodology/Approach (≈ 1-2 pages)

As this report is about reinforcement learning, it should involve discussion of at least one environment, and at least one agent. Since you already introduced the general background and concepts you are using in Section II, here you can get straight into the specific details of *your* approach; basically, providing details on item **??** in Section I. What did you implement, what experiments did you do (and why); including details of your approach to hyper-parameterization.

For example, if you propose a new environment, you would define the state space, action space, reward function, transition function; here.

As elsewhere in the report, don't hesitate to use diagrams, figures, and screenshots wherever they can be useful. And precisely and ambiguously credit any work (code, theory, or otherwise) that you are using, building from, or comparing to; via an appropriate reference.

## IV. Results and Discussion (≈ 1-2 pages)

Show the results (tables, plots, ...), and – most importantly – discuss and *interpret* the results. Do not just narrate what you did and report results in the tables, but also *discuss the implications of the results*. Method A beats method B; but: Why? How? In which contexts?

Negative results are also results. Your agent performed poorly or not as expected? If you can explain why then this is indeed an important result.

Always discuss limitations, whether observed in your results or suspected in different scenarios.

Make use of plots, e.g., Fig. **??**, tables, etc.; anything that illustrates the performance of your agent in the environment under different configurations. Make sure to clearly indicate the parametrization behind each result (e.g., $\gamma$, or whatever is relevant to your experiments).

## V. Conclusions

Summarize the project briefly (one paragraph will do). Main outcome, lessons learned, suggestions of hypothetical future work. Reflect upon, but don't needlessly repeat, material from the conclusion.

## References

[1] Gymnasium. *Car Racing environment*, https://gymnasium.farama.org/environments/box2d/car_racing/, accessed 04 March 2025.
[2] Sutton and Barto. Reinforcement Learning, *MIT Press*, 2020.
[3] Read. Lecture IV - Reinforcement Learning I. In *INF581 Advanced Machine Learning and Autonomous Agents*, 2024.

## Appendix

This is the place to put work that you did but is not essential to understand the main outcome of your work. For example, additional results and tables, lengthy proofs or derivations. Material here does not count towards page limit (but also it will be optional for the reviewer/teacher to work through).