Research Project Proposal: Deep Learning AI for Racing Games

Emilio Capo, emilio.capo@mail.polimi.it

1. Introduction to the problem

In the video game industry, the currently applied AI techniques are reaching their limits. Players are noticing that, while worlds keep being filled with more details and realism, agents' intelligence is not able to keep the pace.

This is especially true in racing games, where AI is typically provided with a simplified physics and vehicle model with respect to the ones the players is subject to. This leads to noticeable incoherencies, such as opponents overcoming physical limitations under the same conditions as the player. Moreover, these agents also struggle with handling adversarial contexts with different computer-controlled cars, which are usually solved through heuristics.

A recent AI technique that is proving to be quite efficient in solving many problems is deep learning, which elaborates great amounts of data to learn to perform a certain task. This technique allows for more spontaneous and less hardcoded behaviours, promising to be the next step in video games AI development.

From a research viewpoint, the genre of racing games is among the ones that were subject to less applications of this technology. However, chances are that the genre could benefit from a successful implementation of deep learning. Firstly, this would solve one of the industry's main concerns at the moment, that is providing a valid alternative to more traditional methods for AI development. The promise that this technology holds, however, is also that of imbuing games with innovative features. In the specific case of racing games, there are three main directions for innovation. One possibility is that of *player emulation*, perfecting a so called drivatar, that is an agent that learns the player's driving style and emulates it. This could then substitute the player either in online matches, in case of connection issues, or during less important stages of a single-player campaign. The AI could also play a major role in *player support*, teaching the player how to drive or giving suggestions for choosing the right parts or setup of a car for a specific track. Finally, it could be possible to enhance *player evaluation* through the use of virtual judges that could correctly assign punishments during race conflicts or simply evaluate the player performance.

The main obstacle to this goal is the computational feasibility. The representational structure of such an agent should be carefully designed to best exploit the domain-specific knowledge and efficiently manage the added cost of eventually using a visual technique.

2. Main related works

Applications of deep learning to racing games mainly align along two axes. One is that of agent's output emission, which can be either discrete or continuous. Work focusing on lane keeping [1] shows how DQNs performance suffer from discrete ouput, especially when steering, compared to DDAC, which features smoother steering thanks to its continuous input.

The other axis is the agent's input representation. In C. Chen et al. [2], input is mainly classified in two categories: direct learning of the image-output mapping, which thus features a fully-visual input, and mediated learning through sensor data, which makes use of sensor information, such as distance from the edges of the road, velocity and acceleration of the car. The work also highlights a valid third possible approach, which is hybrid: learning an image-data mapping and, then, exploiting the obtained data for making decision. In general, approach based on low-dimensional feature vectors perform better than high-dimensional pixel inputs [3].

Currently, different techniques have been applied to the problem. Evolved RNNs have been shown to perform comparably to TORCS built-in agents [4], while an agent combining this technique with a Max-Pooling CNN couldn't reach the same performances [5] [6], though being still able to control the car efficiently. Combining CNN with LSTM produced an agent that is able to stay on track [7]. Actor-critic (A3C) and DDPG are the techniques that have shown the most promising results [2] [8].

3. Research plan

The final goal of the research is being able to develop a deep learning agent that can efficiently manage individual racing and adversarial contexts.

The nature of the research is hybrid. Starting from a survey of the most commonly applied techniques to the problem and after considering possible edits to the main paradigms, the objective is to implement a fully functional agent and test it in a commercial game, thanks to the collaboration with Vae Victis, an Italian video game development company.

It is possible to identify five main tasks for this research project (Fig. 1).

- *Architecture design*, which concerns the definition of both the topology of the network and the paradigm to train it. This task can be logically divided into two steps.
    - *Representation and domain knowledge integration*, which consists of defining the network input and output, based on the domain knowledge gathered from previous works on the TORCS framework made by the community, as well as from the company we are collaborating with. This means defining on the input structure (image, image + data) and output management (layered approach, which consists in a high-level output to process, or hybrid approach, using the deep learning agent as a support for other agents).
    - *Learning paradigm engineering*, in which we define the approach to train the network. A possibility is working with a hybrid approach, using supervised learning for bootstrapping and reinforcement learning to enhance performances. One advantage is the chance to exploit the track generation feature of the cooperating company to produce large and diverse data for training.

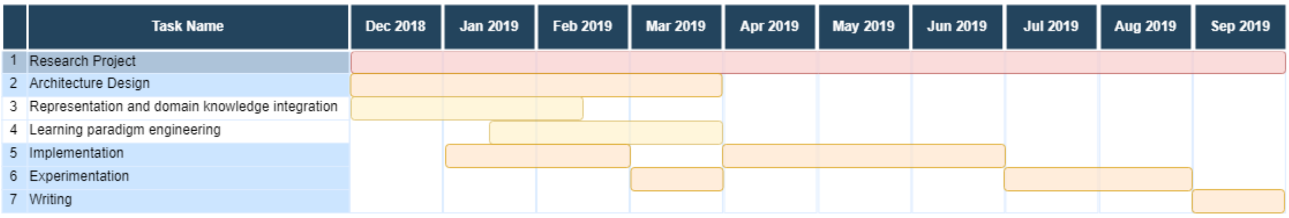

Figure 1 - A simple GANTT diagram of the project.

- *Implementation*, which involves prototyping and implementing the design defined in the previous phase. This also includes the choice of the implementation language and tools. The most convenient choice is to structure the work in C++, so to facilitate integration with the TORCS environment and with the video game developed by Vae Victis for testing.
- *Experimentation*, which can both be based on a comparison with different techniques and the exploitation of the video game's fanbase to test the reaction of the audience to the produced result.
- *Writing*, which is the final step in which the research and implementation results are written in the form of a scientific paper.

The work will be evaluated according to three main metrics. The first one is the performance, in terms of efficiency and effectiveness compared with other techniques and algorithms. The second one is through preliminary user study, collecting opinions from real players. Finally, the work will also be evaluated through an AI versus players comparison, to see how coherently the AI reacts to human behaviour.

References

[1] A. El Sllab, M. Abdou, E. Perot and S. Yogamani. End-to-End Deep Reinforcement Learning for Lane Keeping Assist.

[2] C. Chen, A. Seff, A. Kornhauser, and J. Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In Proceedings of the IEEE International Conference on Computer Vision, pages 2722– 2730, 2015.

[3] T. Lillicrap, J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. Continuous control with deep reinforcement learning. arXiv preprint arXiv:1509.02971, 2015

[4] J. Koutn´ık, G. Cuccu, J. Schmidhuber and Gomez, Faustino. Evolving Large-Scale Neural Networks for Vision-Based Reinforcement Learning.

[5] J. Koutn´ık, J. Schmidhuber and F. Gomez. Evolving deep unsupervised convolutional networks for vision-based reinforcement learning. In Proceedings of the 2014 conference on Genetic and evolutionary computation, pp. 541–548. ACM, 2014a.

[6] J. Koutn´ık, J. Schmidhuber and F. Gomez. Online evolution of deep convolutional network for vision-based reinforcement learning. In Animals to Animats 13, pp. 260–269. Springer, 2014b

[7] E. Perot, M. Jaritz, M. Toromanoff and R. D. Charette, "End-to-End Driving in a Realistic Racing Game with Deep Reinforcement Learning," in IEEE Computer Vision and Pattern Recognition Workshops, 2017.

[8] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In International Conference on Machine Learning, 2016.