Research Project Proposal:
Deep Learning AI for Racing Games

Emilio Capo
emilio.capo@mail.polimi.it
Computer Science and Engineering (CSE)

POLITECNICO
MILANO 1863

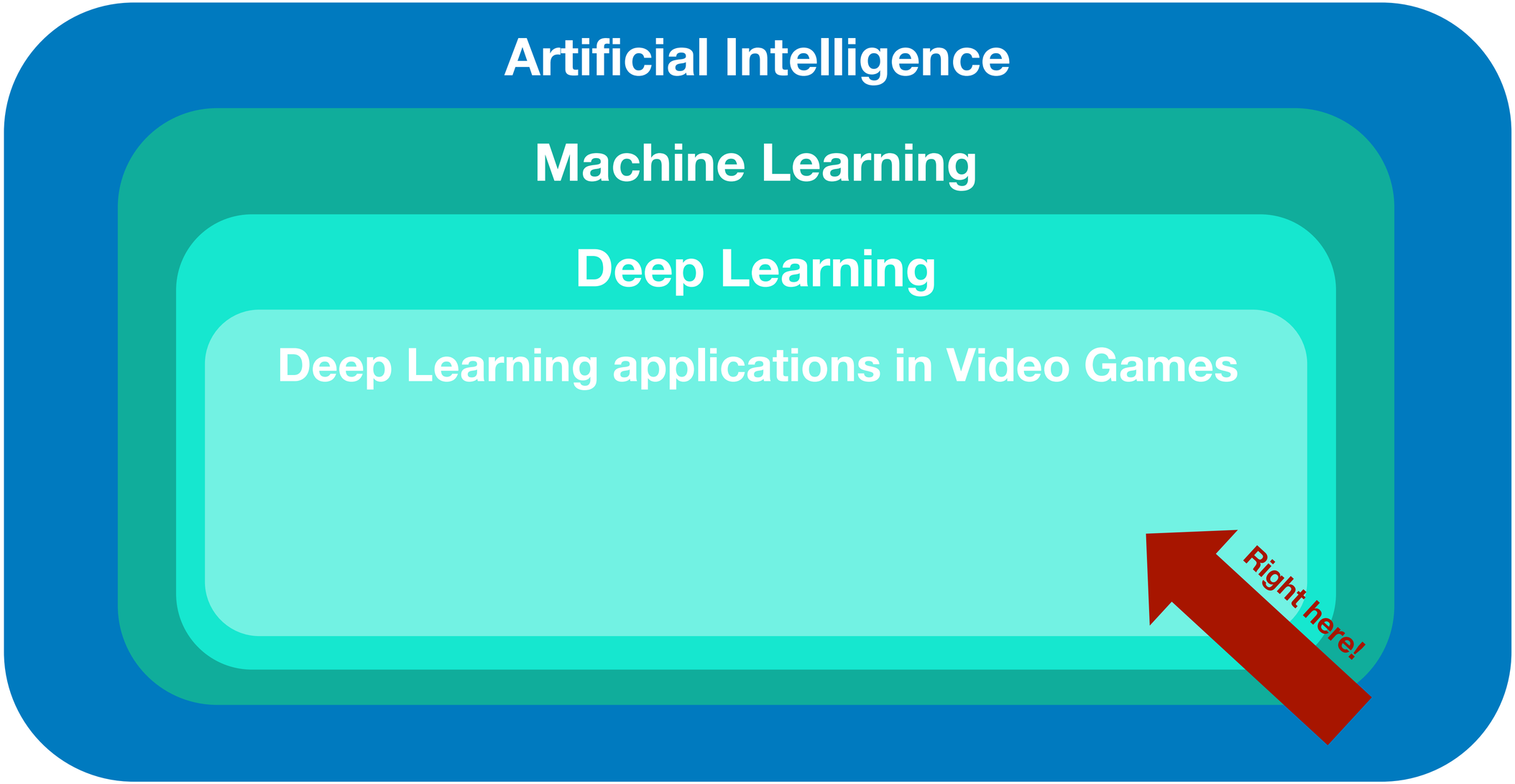
HP-SR
in Information Technology

Outline

- **Research Area**
  - Why Deep Learning for video games?
  - Classification and open issues
- **Research Topic**
  - Why Deep Learning for racing games?
  - Classification and research opportunities
- **Research Project Structure**
  - Content
  - Tasks and tentative schedule

Research Area

Where are we?


Artificial Intelligence
Machine Learning
Deep Learning
Deep Learning applications in Video Games
Right here!

Why Video Games?

(Industry's point of view)

- AI techniques currently used in video games are reaching their **limits.**
  - High dependence on hard-coded behaviours (low generalization)
- Behaviour realism fails to keep the pace of graphics realism.

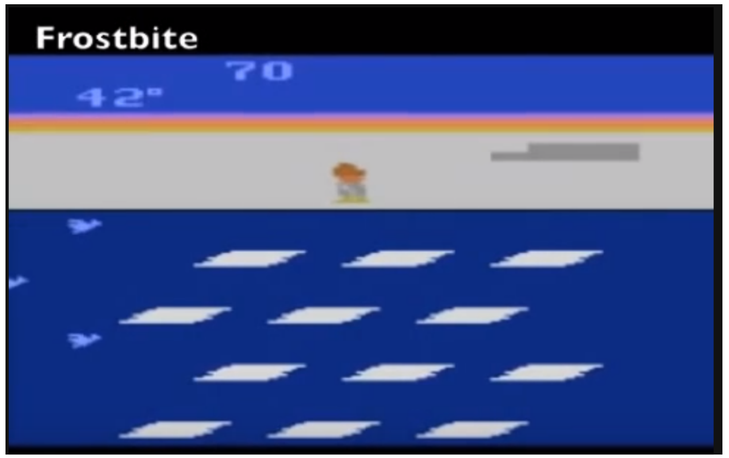
Example: Assassin's Creed Odissey (2018)

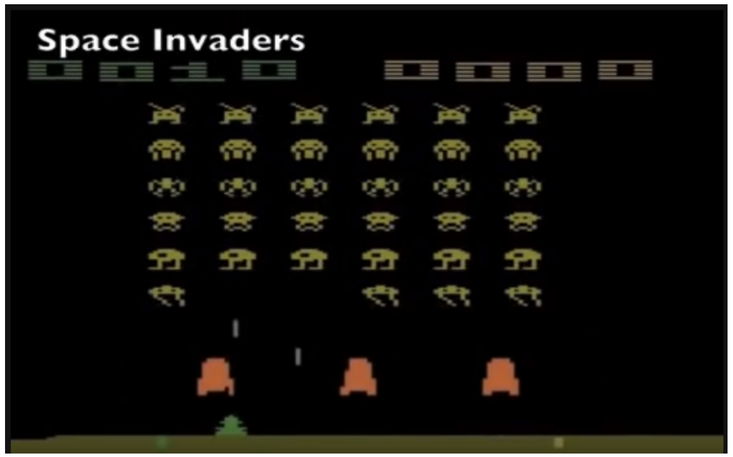
Why Video Games?

(AI researchers' point of view)

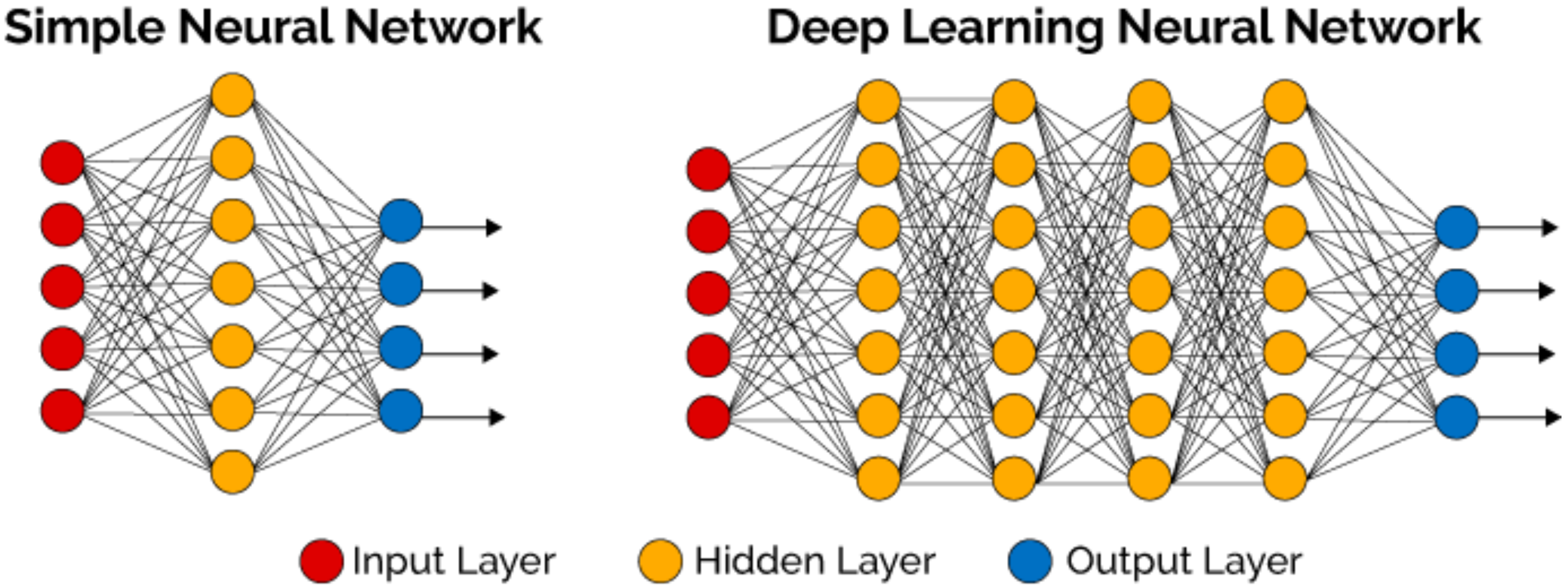
- Video games provide **richness and variety in learning environments.**

- Some available game frameworks are used as **benchmarks.**

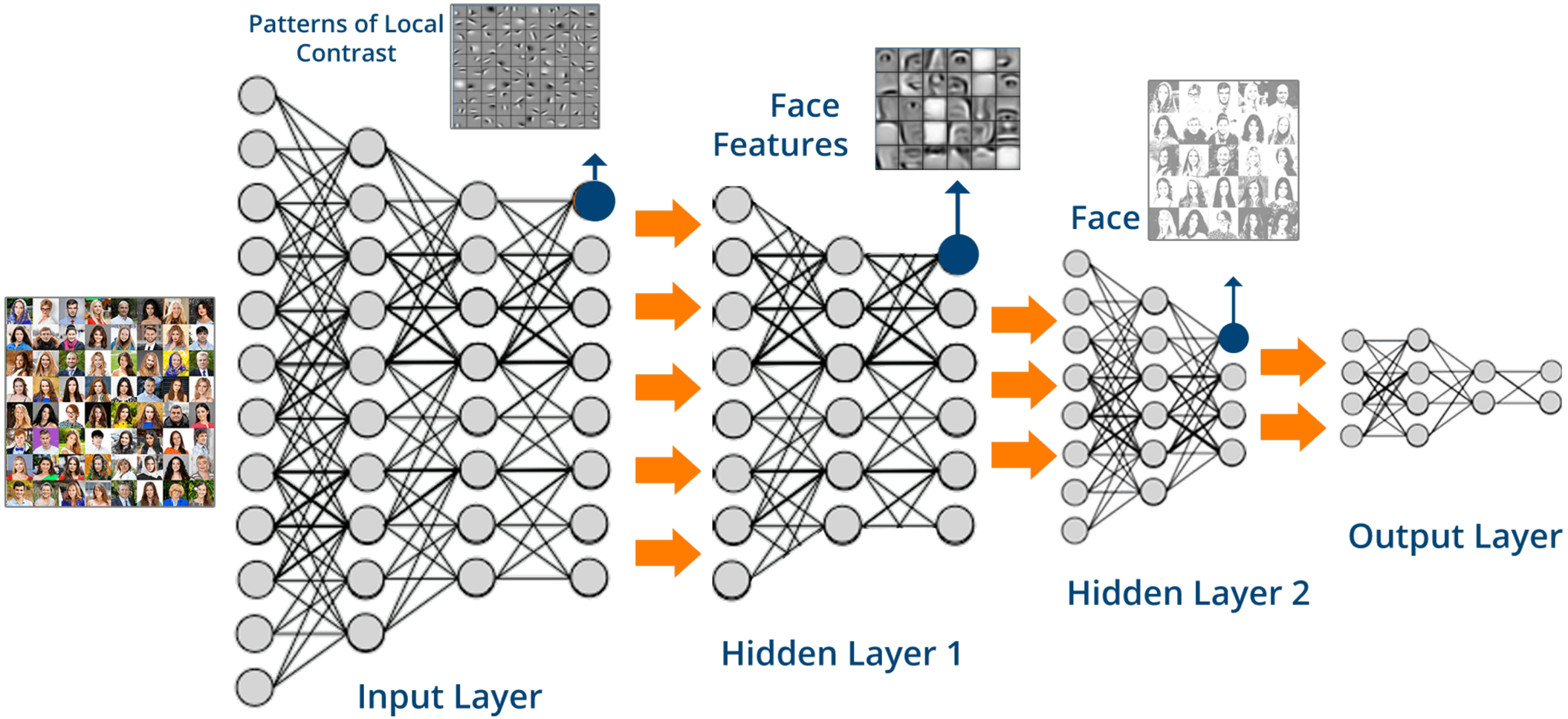




What is Deep Learning?



Advantages of a Deep architecture

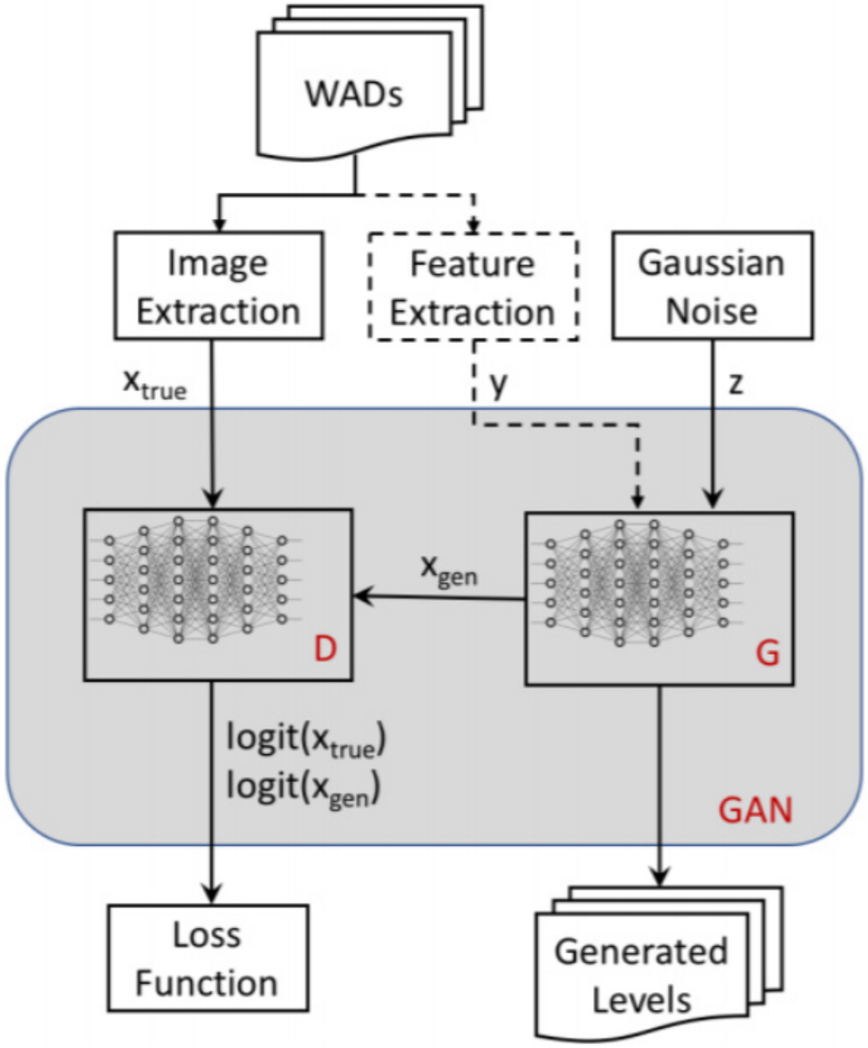


Why Deep Learning for Games?

- **Novel** machine learning technique.
  - Able to solve complex task.
  - Data recorded from games available in large datasets online.
- Promising **generalization** capabilities.
  - More spontaneous and believable behaviours *(Industry's point of view)*.
  - Potential applications in the field of General AI *(AI researchers' point of view)*.

DL Applications in Games (1)

- Deep Learning can used to support the game in its development stage.
  - Procedural Content Generation (PCG);
  - Tools for game designer support;
  - …



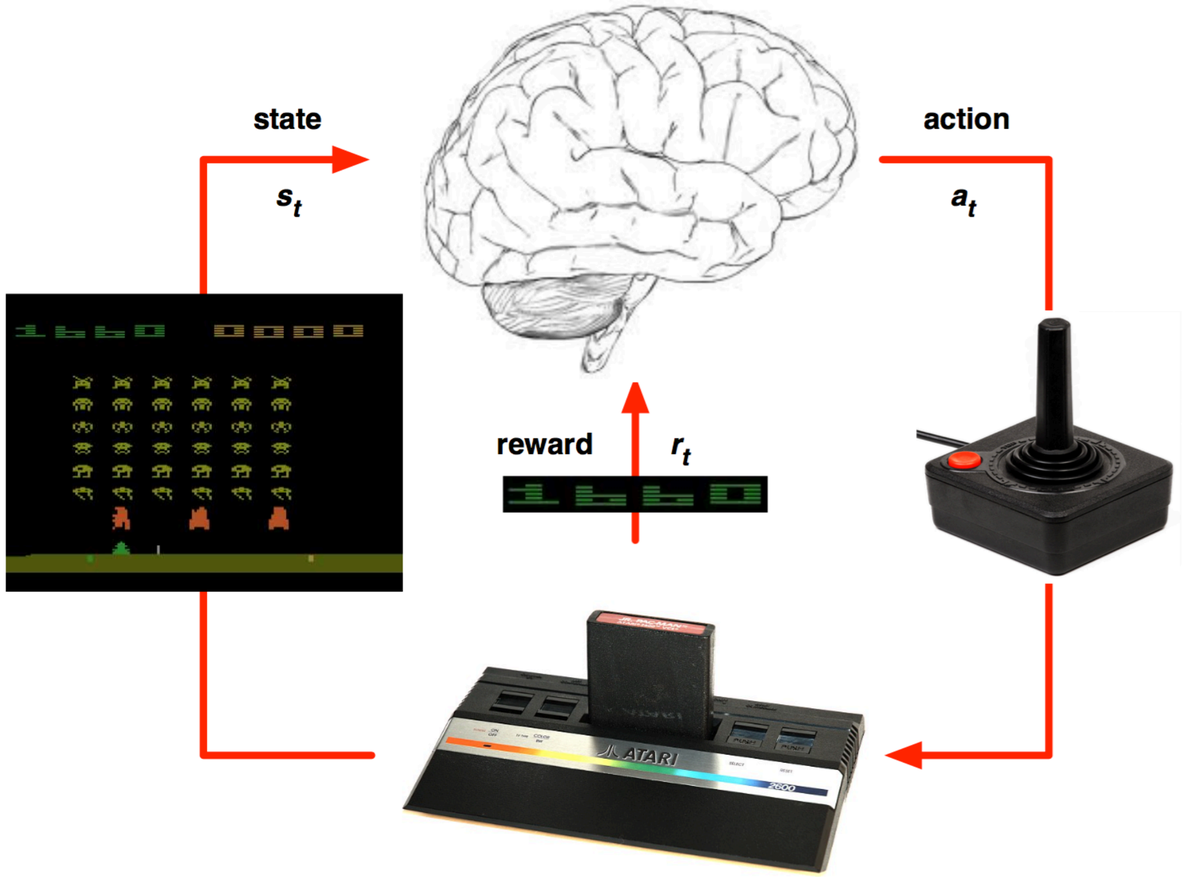
DL Applications in Games (2)

- For this research, we considered its application to model the behaviour of game agents.

- Efforts in this sense mainly align along two directions:

  - Non-Player Character (NPC) Behaviour Modeling;

  - General Video Games AI (GVGAI).



Classification

- Per **Learning Paradigm**:
  - Supervised Learning
  - Reinforcement Learning
  - Evolutionary Learning
- Per **Game Genre**:
  - Arcade games
  - Racing games
  - First-Person Shooters
  - Real-Time Strategy games

Open issues

- GVGAI for playing multiple diverse games;
- Learning to play games with very sparse rewards;
- Respecting computational and stability constraint for industry application;
- Human-like game playing (agents' believability);
- Agents with adjustable level of performance.

Research Topic

Why Racing Games?

- AI in Racing Games currently «cheats» the player.
  - Simplified physics and vehicle model.
  - AI basically performs actions that are impossible for the player.
- Adversarial contexts are handled according to simple heuristics.

Example: Forza Motorsport 7 (2017)

Three opportunities for innovation

- **Player Emulation**
  - The agent learns the player's driving style to be able to substitute him in different contexts.
- **Player Support**
  - The agent teaches the player how to drive or which components setting is better for a given track.
- **Player Evaluation**
  - The agent functions as a race judge, evaluates guilt in conflicts or evaluating the player's performance.

Main axes for classification

- Agent's **output emission**
  - Discrete
  - Continuous
- Agent's **input representation**
  - Fully visual (maps images to actions)
  - Fully sensorial (maps sensor data to actions)
  - Hybrid (maps images to data, then data to actions)

Research opportunities

- Efficiently handling the «real» physical model.
  - Main research focus on mastering the task.
  - Industry’s feasibility constraints are usually not considered.
- Management of group adversarial contexts.
  - Poorly explored research direction.

Research Project Structure

Project proposal

The research project has the objective to possibly tackle both issues:

- Main focus on **efficient individual racing**;
- Secondary focus on **adversarial contexts management**.

Ultimately, the goals are:

- Design an effective neural network architecture and learning paradigm;
- Apply the network in a real game (Racecraft by Vae Victis) and exploit previous knowledge.

Project structure

The project is structured in 4 main tasks:

- Architecture Design
- Implementation
- Experiments
- Writing

Architecture Design

This task mainly concerns the definition of the network topology and the learning paradigm.

Consequently, in can be divided in two sub-tasks:

- Representation and domain knowledge integration;
- Learning paradigm engineering.

Representation and domain knowledge integration

- Definition of input-output shape based on domain knowledge.
  - From TORCS research community;
  - From a collaboration with Vae Victis.
- Two main direction of exploration:
  - Input: Image / Image + Data;
  - Output: Direct / Layered (Hierarchical).

Racing line approach

- An optimal “racing line” is attached to the track.
- The AI tries to follow it as precisely as possible.
- Input: current status (blue vector) + target position and speed (green point).
- Output: control action.
- Might involve heuristics and approximations.

Learning paradigm engineering

This task mainly concerns the definition of the approach to use for training.

- A possibility is a hybrid supervised-reinforcement learning approach.
- The company’s track generation feature can be exploited to obtain diverse training data.

Implementation

This tasks includes both prototyping of the different network designs and the choice for implementation language and tools.

- Different designs will be compared to derive the most efficient topology for our goals;

- Tools choice will have an impact on integration with testing benchmarks.

Experiments

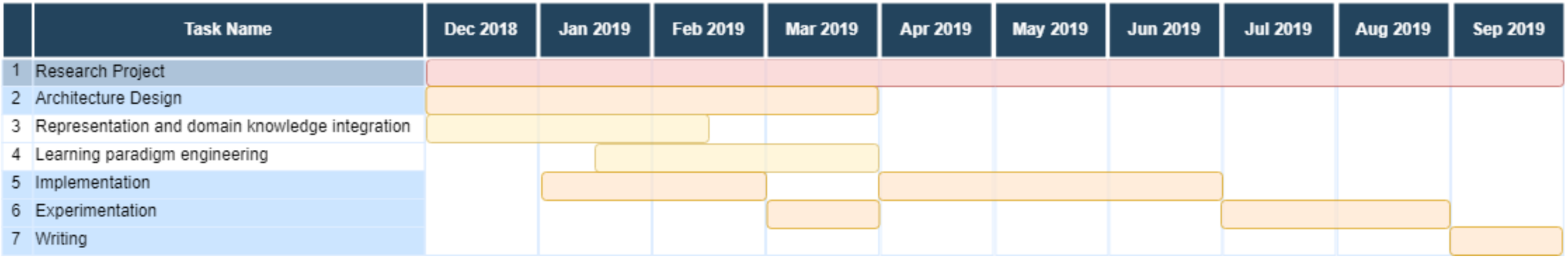
This task will actually involve the experimentation on the derived prototype networks.

- Evaluation of prototypes.
- Evaluation of the final work.

Performance metrics

- Performance
  - How well does the agent perform compared to other techniques?
- Preliminary user study
  - What is the players' opinion on the agent's performance?
- Agent versus player comparison
  - How coherently does the agent react to human behaviour?

Tentative schedule



	Task Name	Dec 2018	Jan 2019	Feb 2019	Mar 2019	Apr 2019	May 2019	Jun 2019	Jul 2019	Aug 2019	Sep 2019
1	Research Project										
2	Architecture Design										
3	Representation and domain knowledge integration										
4	Learning paradigm engineering										
5	Implementation										
6	Experimentation										
7	Writing										

- The tasks stretch along different periods.
- Some tasks require feedback from others, thus they require interruptions.

Emilio Capo
emilio.capo@mail.polimi.it
Computer Science and Engineering (CSE)

POLITECNICO
MILANO 1863

HONOURS PROGRAMME
HP-SR
in Information Technology