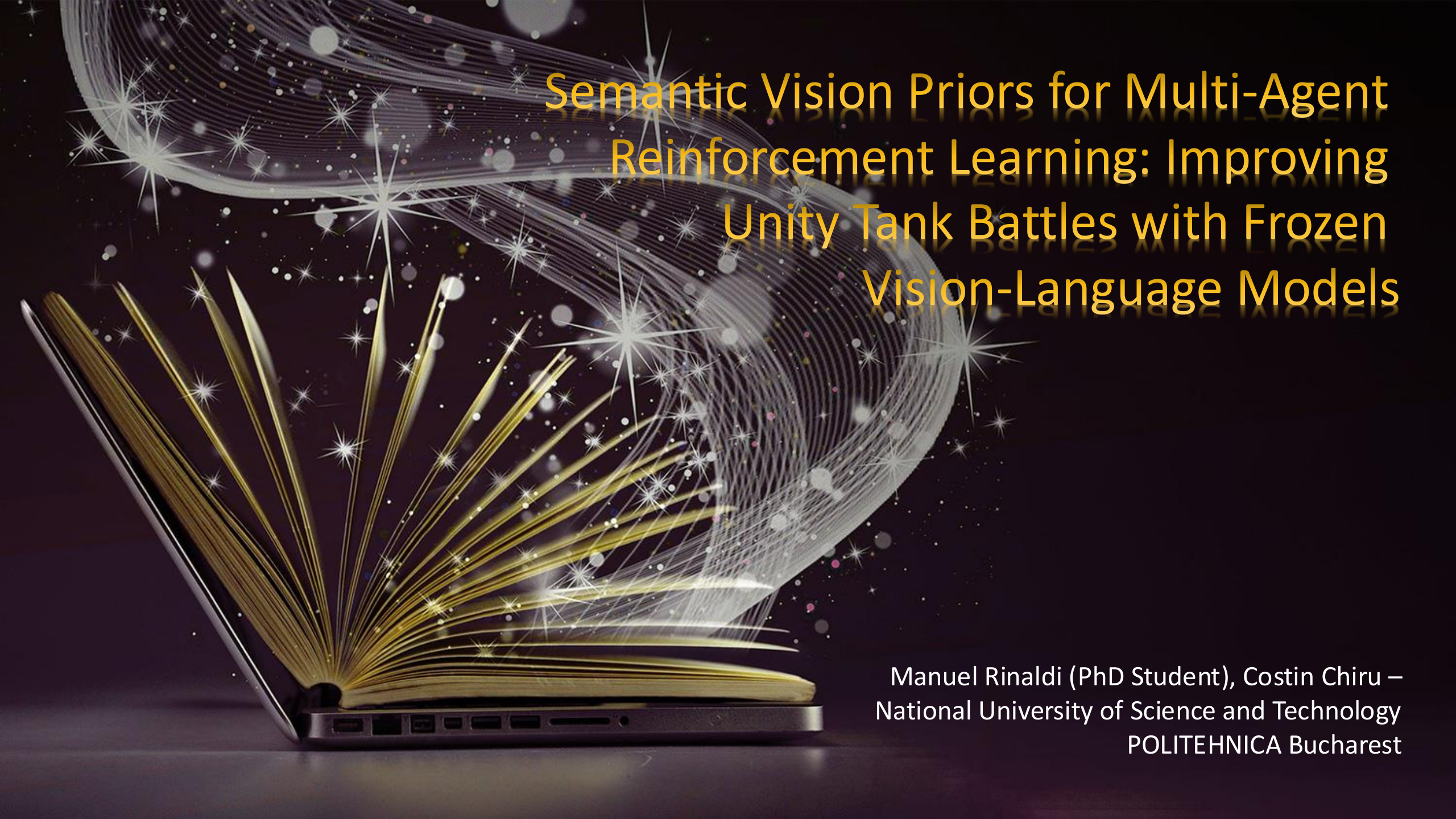


Semantic Vision Priors for Multi-Agent Reinforcement Learning: Improving Unity Tank Battles with Frozen Vision-Language Models

Manuel Rinaldi (PhD Student), Costin Chiru –
National University of Science and Technology
POLITEHNICA Bucharest

Agenda

01 Introduction

Environment, Methodology

02 Related Work

Self-Play Reinforcement Learning, Vision-Language Models in RL, Semantic Priors in Policy Learning

03 Results

Learning Performance, Self-Play ELOIntegration

04 Future Work

Reward Dynamics, Observation modalities, Generalization

Introduction

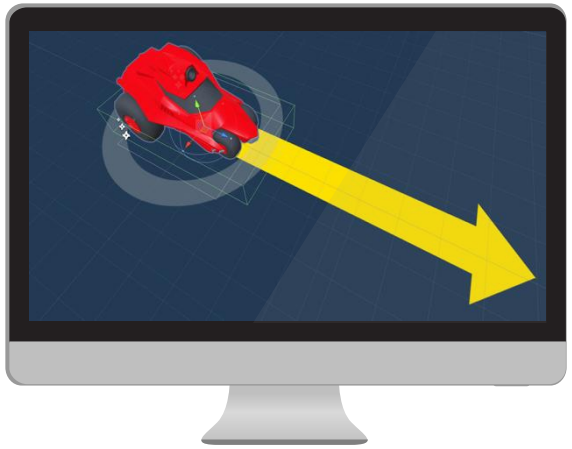
Environment

Environment

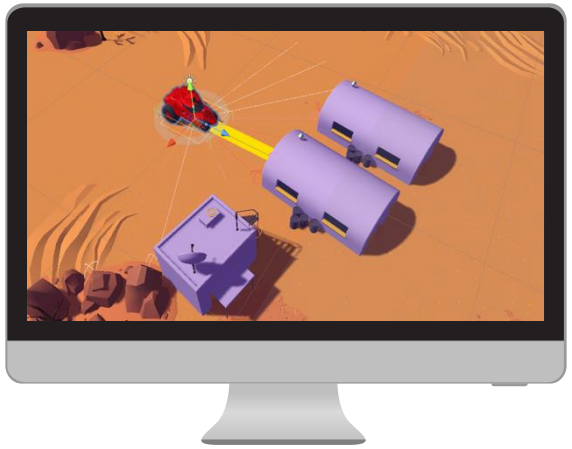
Full Environment

Obstacles

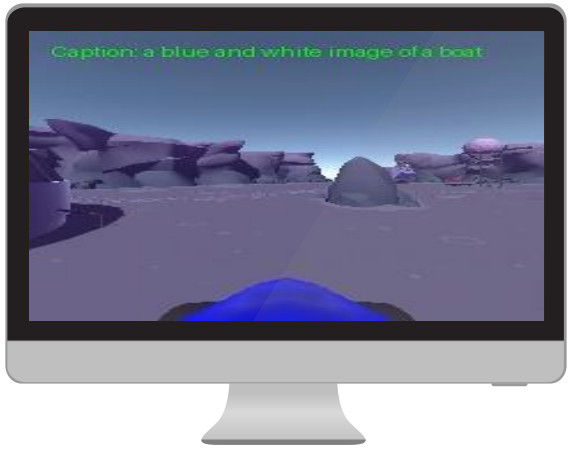
Spawn Point



Tank



RayCast

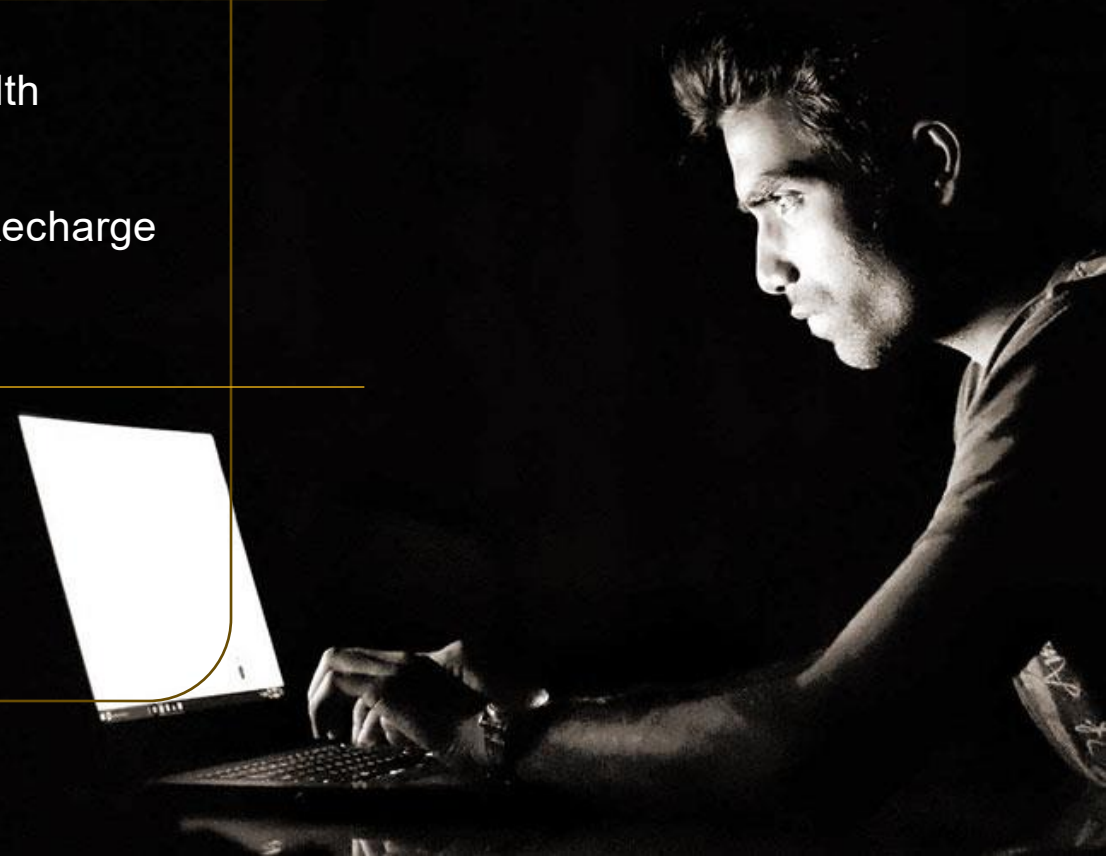


Tank Camera

Introduction

Methodology

Baseline	Raycast	Semantic Observation
Own Position	Own Position	Own Position
Enemy Position	Raycast Vectors	VLM Embeddings 512
Health	Health	Health
Shooting Recharge	Shooting Recharge	Shooting Recharge
	2M Steps	
	PPO	
	Self Play	



Related Work

Self-Play Reinforcement Learning

Self-Play RL (why it matters)

What it is & why it matters

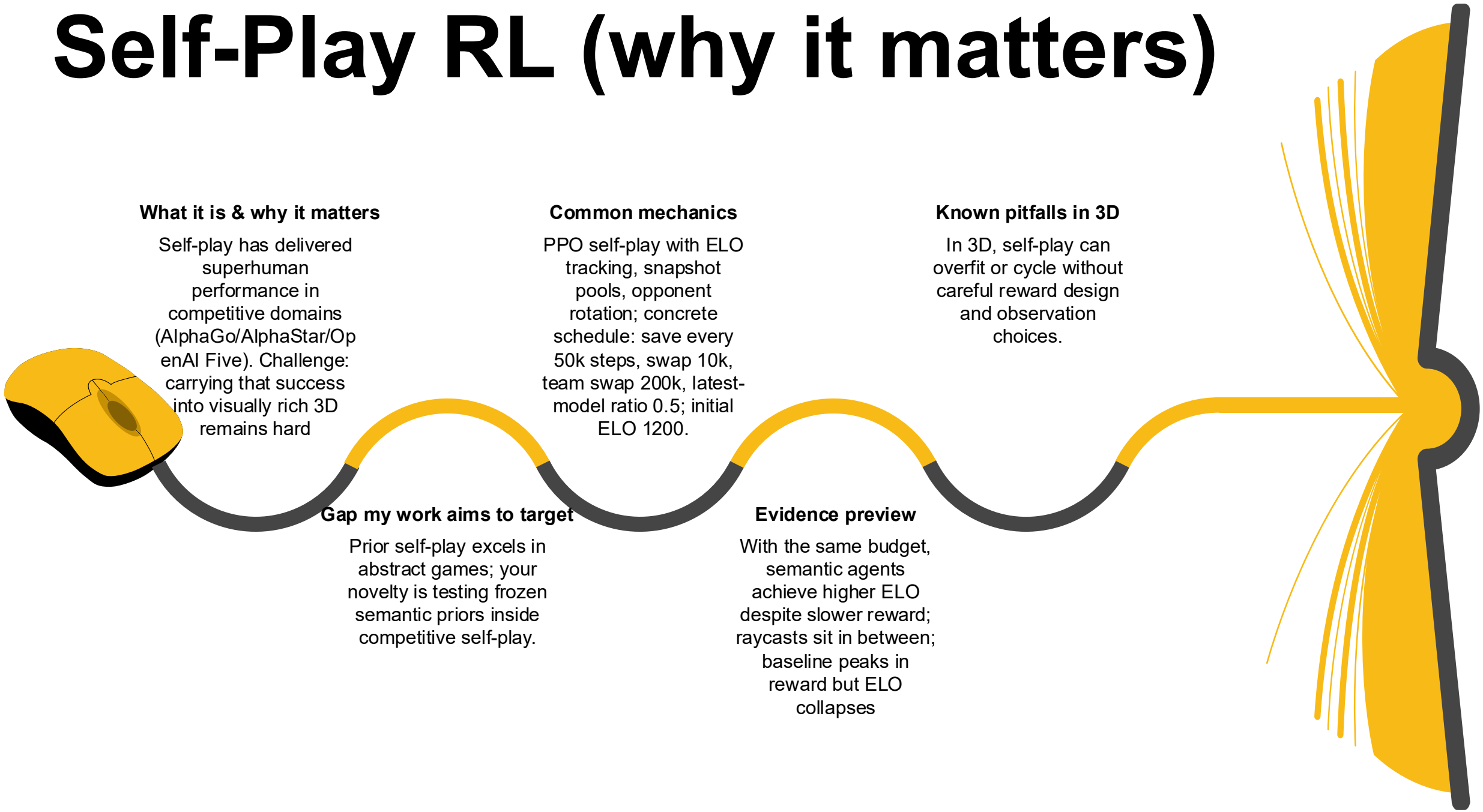
Self-play has delivered superhuman performance in competitive domains (AlphaGo/AlphaStar/OpenAI Five). Challenge: carrying that success into visually rich 3D remains hard

Common mechanics

PPO self-play with ELO tracking, snapshot pools, opponent rotation; concrete schedule: save every 50k steps, swap 10k, team swap 200k, latest-model ratio 0.5; initial ELO 1200.

Known pitfalls in 3D

In 3D, self-play can overfit or cycle without careful reward design and observation choices.



Gap my work aims to target

Prior self-play excels in abstract games; your novelty is testing frozen semantic priors inside competitive self-play.

Evidence preview

With the same budget, semantic agents achieve higher ELO despite slower reward; raycasts sit in between; baseline peaks in reward but ELO collapses

Related Work

Vision-Language Models in RL

Vision-Language Models in RL

What VLMs bring

Pretrained semantic representations (e.g., CLIP/BLIP-2) encode objects & relations; enable zero-shot transfer and better perception.

Where they've been used

My paper summarizes single-agent uses (navigation/manipulation) where VLMs speed learning; competitive MARL is underexplored

Why this matters for RL

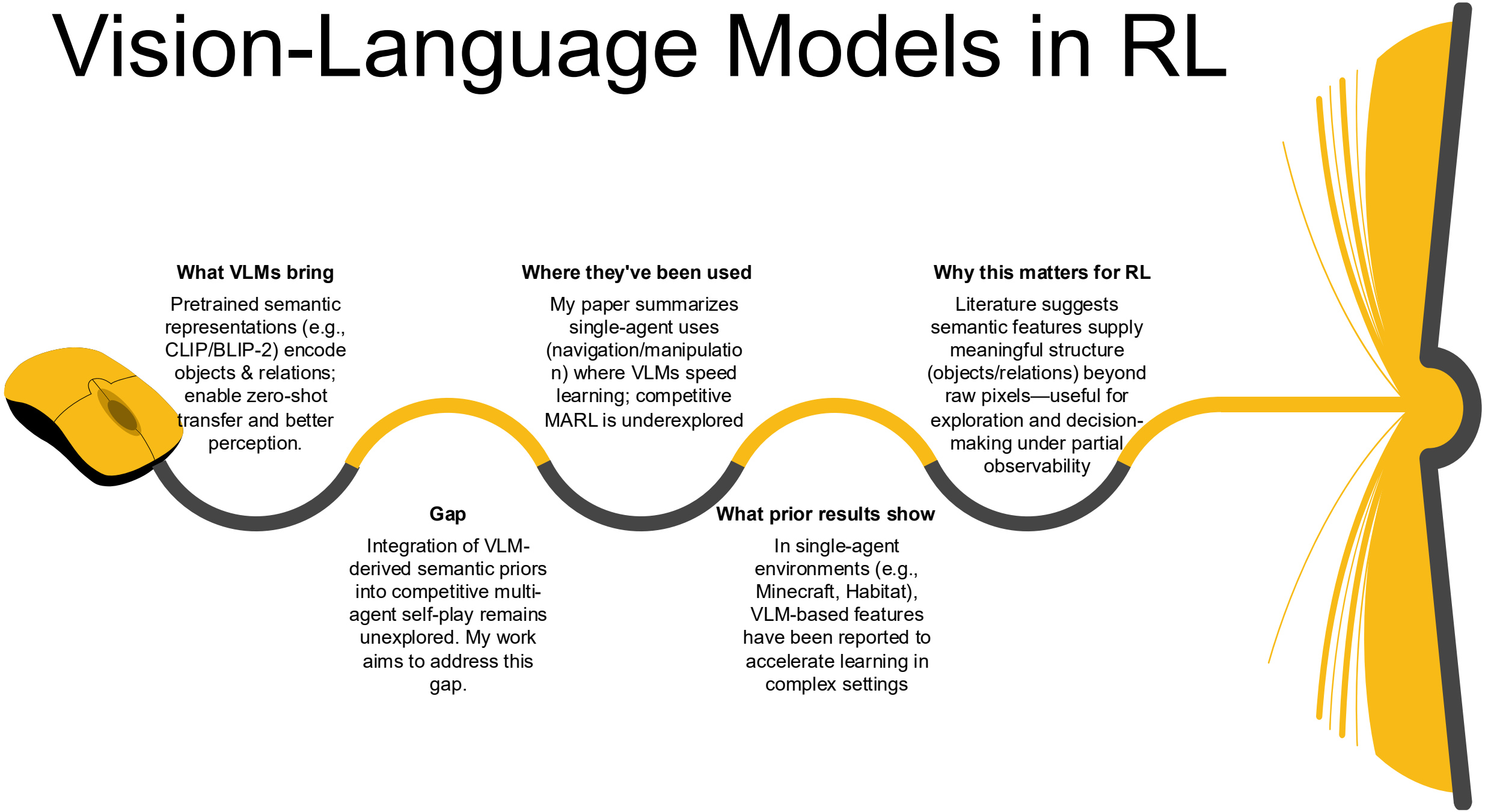
Literature suggests semantic features supply meaningful structure (objects/relations) beyond raw pixels—useful for exploration and decision-making under partial observability

Gap

Integration of VLM-derived semantic priors into competitive multi-agent self-play remains unexplored. My work aims to address this gap.

What prior results show

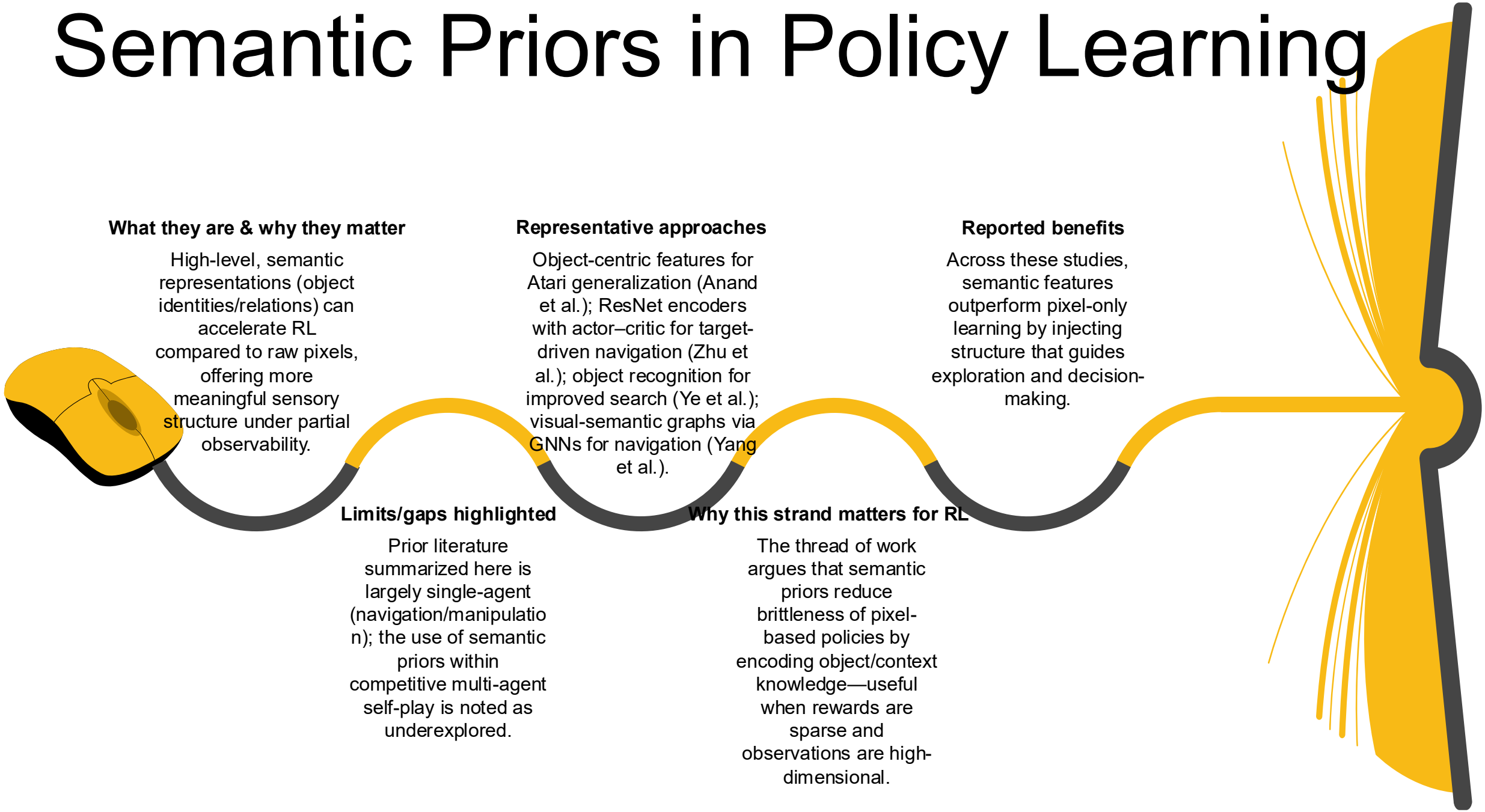
In single-agent environments (e.g., Minecraft, Habitat), VLM-based features have been reported to accelerate learning in complex settings



Related Work

Semantic Priors in Policy Learning

Semantic Priors in Policy Learning



What they are & why they matter

High-level, semantic representations (object identities/relations) can accelerate RL compared to raw pixels, offering more meaningful sensory structure under partial observability.

Representative approaches

Object-centric features for Atari generalization (Anand et al.); ResNet encoders with actor–critic for target-driven navigation (Zhu et al.); object recognition for improved search (Ye et al.); visual-semantic graphs via GNNs for navigation (Yang et al.).

Reported benefits

Across these studies, semantic features outperform pixel-only learning by injecting structure that guides exploration and decision-making.

Limits/gaps highlighted

Prior literature summarized here is largely single-agent (navigation/manipulation); the use of semantic priors within competitive multi-agent self-play is noted as underexplored.

Why this strand matters for RL

The thread of work argues that semantic priors reduce brittleness of pixel-based policies by encoding object/context knowledge—useful when rewards are sparse and observations are high-dimensional.

Results

Learning Performance

Learning Performance (2M steps)

Context: 1v1 Unity tanks, PPO self-play, fixed **2M steps**; three observation modalities (Baseline numeric, Raycasts, Semantic BLIP-2).

	Mean Reward	Final Reward	Notes
Baseline	13.53	19.63	Always attack do not explore
Raycast	6.79	8.33	Explore and attack once see the target
Semantic (VLM)	4.49	1.83	Only Explore didn't learn to attack

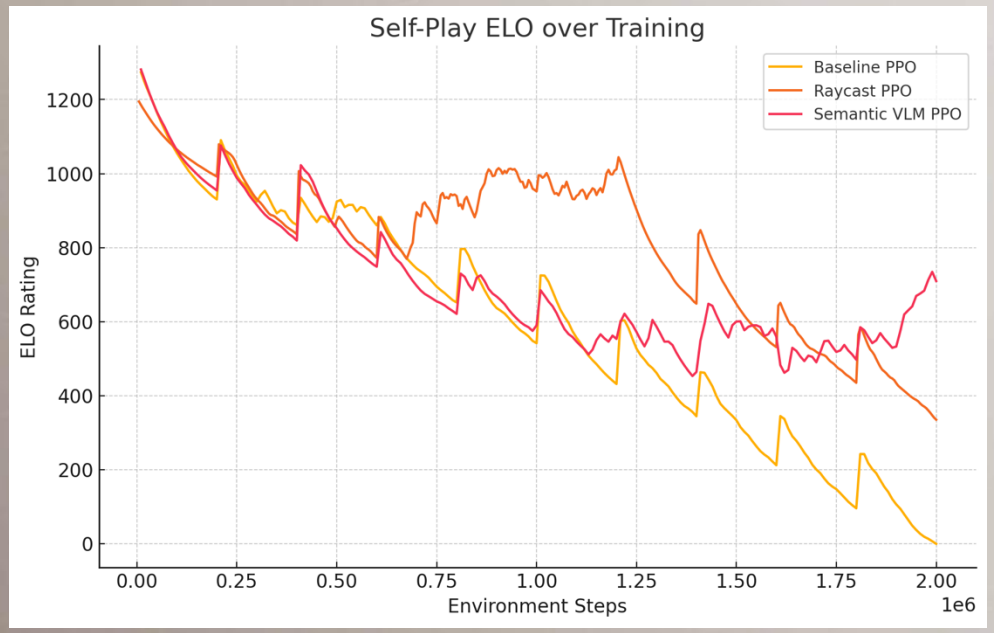
Overall: All three configurations **improve reward over time**; learning dynamics differ (fast vs moderate vs exploratory).

Results

Self-Play ELO

Self-Play ELO (2M steps)

	ELO	Notes
Baseline	−0.45	collapsed from 1273 → −0.45
Raycast	334.96	moderate/stabilized
Semantic (VLM)	709.66	highest ELO



Future Work

Reward Dynamics, Observation modalities, Generalization

Where to go from here?

Rewards

- Experiment using rewards that are either less frequent or more frequent.
- Test rewards that focus more on tactical elements to help the tank accomplish a specific goal.

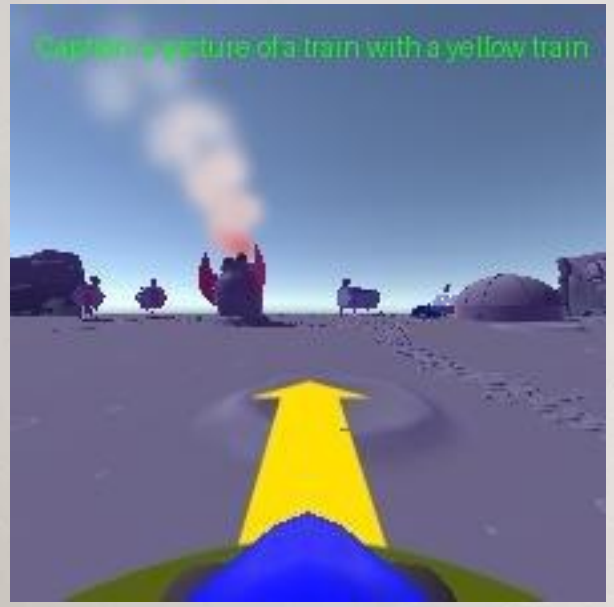
Semantics & Raycast

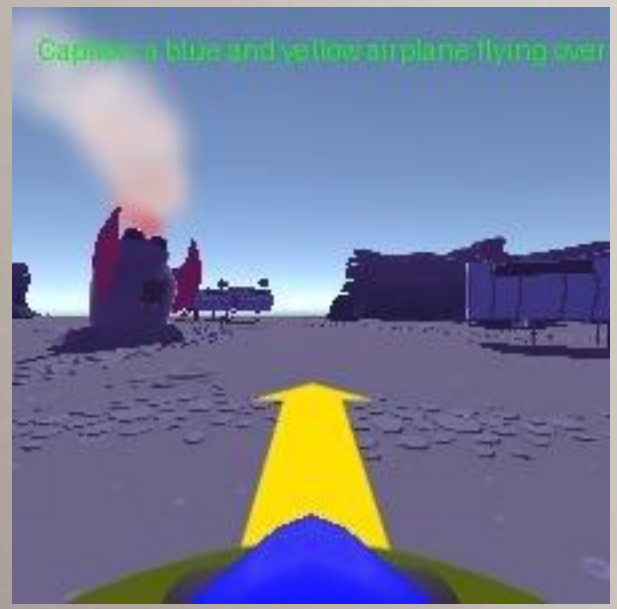
- Try using a larger VLM
- Test a fine-tuned VLM
- Explore additional prompt engineering techniques
- Attempt to generate strategies based on image analysis
- Combine VLM with Raycast for experimentation

Generalization

- Try out the three distinct levels offered
- Utilize Procedural Content Generation to vary the obstacles

Implications of Frozen VLM





THANK YOU