Learning to Play Video Games with Intuitive Physics Priors

Abhishek Jaiswal (abhijais@cse.iitk.ac.in)
Department of CSE, IIT Kanpur
Kalyanpur, UP 208016 India

Nisheeth Srivastava (nsrivast@cse.iitk.ac.in)
Department of CSE, IIT Kanpur
Kalyanpur, UP 208016 India

Abstract

Video game playing is an extremely structured domain where algorithmic decision-making can be tested without adverse real-world consequences. While prevailing methods rely on image inputs to avoid the problem of hand-crafting state space representations, this approach systematically diverges from the way humans actually learn to play games. In this paper, we design object-based input representations that generalize well across a number of video games. Using these representations, we evaluate an agent's ability to learn games similar to an infant - with limited world experience, employing simple inductive biases derived from intuitive representations of physics from the real world. Using such biases, we construct an object category representation to be used by a Q-learning algorithm and assess how well it learns to play multiple games based on observed object affordances. Our results suggest that a human-like object interaction setup capably learns to play several video games, and demonstrates superior generalizability, particularly for unfamiliar objects. Further exploring such methods will allow machines to learn in a human-centric way, thus incorporating more human-like learning benefits.

Keywords: Category Learning; Object-based Reinforcement Learning; Generalization; Inductive priors; Intuitive physics

Introduction

Deep reinforcement learning (DRL) algorithms have shown professional to superhuman competency in gaming environments such as MuJoCo, and Atari (Shakya, Pillai, & Chakrabarty, 2023; Goodfellow, Shlens, & Szegedy, 2014). But, at the same time, like other black box deep learning models, they can break with even slight modifications of the environment (Justesen et al., 2018; Goodfellow et al., 2014).

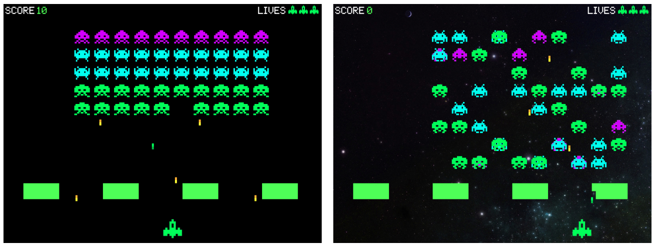

Figure 1: Simple Variations, Crippling Results - Deep Learning Models break even with a slight variation of the environment (Right image - randomized enemy positions).

For example, to contrast human and machine-level learning, Figure 1 shows two variants of the space invaders game we tested. DQN was trained on the basic version on the left for two million iterations and tested on the variant with partially randomized enemy positions on the right. The base variant's average score was 510, whereas the right variant could score only 280; both averaged over ten runs. A random agent also reached an average score of 270. Thus, even with this simple modification, a Deep Reinforcement Learning (DRL) model may fail. On the contrary, humans play through such variants with ease.

Even though DRL is setting new records on the Atari benchmarks, defeating human players in live settings, and achieving superhuman scores on games (Hessel et al., 2018; Mnih et al., 2015), they still fail at generalizing and transferring the learned knowledge to novel domains (Kansky et al., 2017) - a task which humans do exceptionally well. We find this a big issue in the recent Machine learning (ML) research trends where the focus is increasingly shifting towards feeding massive amounts of data to a black-box model, giving it enough facility to memorize all the variations it could and then breaking the previous benchmark results.

Thus, this paper focuses on techniques to make machines learn in a more human-like fashion. The first distinction between the recent ML direction and the human way of learning is its input. DRL takes images as input which essentially means that the whole world for them is a collection of pixels. In contrast, when humans look at the world, they do not see pixels; instead, they see objects. , which is the interest of the works on object-based reinforcement learning.

Here also, we take a slightly different approach which we find more interesting. Instead of working directly with the objects and assuming their properties as given, we try to look at the game world from a fresh perspective, somewhat similar to the view of a child who does not come with an oversized baggage of existing knowledge.

To this end, we try to learn game playing using common human inductive biases. The broad idea is to incorporate human-like learning trajectories in machines to test if they could be made more closer to human behavior than the current ML paradigms. With this line of work, we aim to leverage the same advantages humans show in generalization and zero-shot transfer. For the existing model-free algorithms, this is a difficult task because as soon as a new object becomes part of its input the algorithm sees pixel combinations never seen before.

Developing an agent with human-like abilities is a longstanding journey. This study evaluates and presents the following contributions as steps along this path. First, we try to incorporate the thinking of a first-time player in game playing using commonly agreeable inductive biases. Second and more importantly, we bring in the idea of using object categories instead of direct object-based inputs. We test our approach using a simple Q-learning agent (Watkins & Dayan, 1992) against DQN (Mnih et al., 2015) to contrast human and machine-like learning. We also show that by using such a paradigm, we can see generalization trends practically unattainable by resource-hungry pixel-based mechanistic ML agents.

Like many deep learning methods, these algorithms work as a black box (Kumar, Dasgupta, Daw, Cohen, & Griffiths, 2023), often struggling with inexplicability and poor sample efficiency(Mohan, Zhang, & Lindauer, 2023). More so, akin to their counterparts in deep learning (Goodfellow et al., 2014), they are susceptible to errors with even slight modifications of features (Lu, Shahn, Sow, Doshi-Velez, & Li-wei, 2020). On the contrary, humans, against AI, despite being defeated on many of the gaming benchmarks, do much better at learning task abstractions to reuse the acquired knowledge (Kansky et al., 2017). Humans demonstrate superior learning trajectories, learning games quickly and also performing well on modifications (Tsividis, Pouncy, Xu, Tenenbaum, & Gershman, 2017).

Theory-based RL is a form of Model-based method where the model is defined in terms of rich ontological symbolic representations pertaining to physical objects, their relations, and interactions. Using various intuitive theories, theory-based RL explicitly tries to incorporate human ways of learning (Tsividis et al., 2021). Such intuitive theories stem from a core knowledge representation of the world visible even in infants who can segregate the visual input into ontological structures such as objects, goals, and physics (Baillargeon, 2004; Spelke, 1990; Spelke & Kinzler, 2007; Csibra, 2008). Humans also have been shown to make internal models using theory representation(Tomov, Tsividis, Pouncy, Tenenbaum, & Gershman, 2023). Similarly, semantic and syntactic biases, such as those used in theory-based RL, show a strong resemblance to human-like learning (Pouncy & Gershman, 2022). Humans show a wide range of flexibility in adapting to variations within the same task domain. As such, (Pouncy, Tsividis, & Gershman, 2021) have shown evidence that such flexibility, a hallmark of human intelligence, can arise by representations composed of objects and interactions within a model-based framework. Thus, theory-based RL has shown a promising resemblance to human-like learning, But being highly dependent on a strong model of the environment, it has significant practical limitations.

To make sense of these observations, humans utilize various priors that help them explore efficiently.Dubey, Agrawal, Pathak, Griffiths, and Efros (2018) explore and quantify such priors for video gaming tasks. Our work builds upon such principles to learn a working structure of the world.

Tsividis et al. (2021) worked on the idea of making machines learn more like humans, starting from early childhood using strong theories about the working of the world. We take a slightly different approach and learn the affordances from object specifications rather than using pre-defined rules of interactions. Much like them, we also levy inductive biases for this task, which we understand to be a product of evolution, such as agent identification, threat perception, and goal attribution. In a related setting, using program induction, Ellis et al. (2023) developed DreamCoder – growing learning capabilities from a child-like state. Similarly, Ding et al. (2023) used language instructions and human demonstrations to learn concepts, acting like a baby learning from environmental interactions

We explore this learning task by leveraging inductive biases and agent-object interactions with a focus on categorization, complementing, and yet differentiating with previous works.

Spelke (1990) reason that infants perceive objects based on perceptual units moving together, moving separately, interacting on contact, and maintaining their shapes and sizes while in motion. We leverage these interactions to learn fundamental affordances such as avoid, touch, and block through our Reinforcement Learning (RL) agent's actions trained exclusively on object-specific properties that are interpretable and in alignment with concepts of infant learning.

Thus, in this domain, akin to the work of Ding et al. (2023) in the space of natural languages, we try to answer a simple question - can we enable an agent to learn like a small child, testing the hypothesis in game settings. Humans look at the world in terms of objects and their interactions; This is one of their core knowledge (Spelke & Kinzler, 2007). Drawing on this insight, we shape the task of object reasoning around basic principles of core knowledge and show that we can achieve game-playing in a more cognitive and less mechanistic manner. Specifically, instead of just looking at objects in isolation, we bring in the concept of meaningful object categories. This notion is inspired from compelling evidence on humans learning object categories in specific brain regions (Kriegeskorte et al., 2008; DiCarlo, Zoccolan, & Rust, 2012). Then, we build upon this representation to learn category-level affordances and test our hypothesis on multiple tasks considered easy for humans but proven challenging for machines.

Learning How to Play

We look at the game screen from the view of a fresh player who is coming with a very minimal baggage of experience. Such players would see certain entities stand out on the screen by virtue of their specific forms, colors, or movements but would not know the affordances for the different objects - a task necessary to accomplish their desire to win (Csibra, 2008). Thus, the first step in this learning process would be detecting what we control on the screen, i.e., the agent representing the player in the game. Agent detection is one of the key ideas in human-like learning and also a differentiator from large-scale pattern matching in machine-like learning (De Freitas et al., 2023). After knowing the where and how of the agent, the next step would be to devise a strategy to move about in the world, necessitating knowledge of at least a minimal set of affordances associated with other game entities, for which we devise a set of representative object categories and learn simple affordances associated with each category. Refer to Figure 2 for an overview of the complete pipeline.

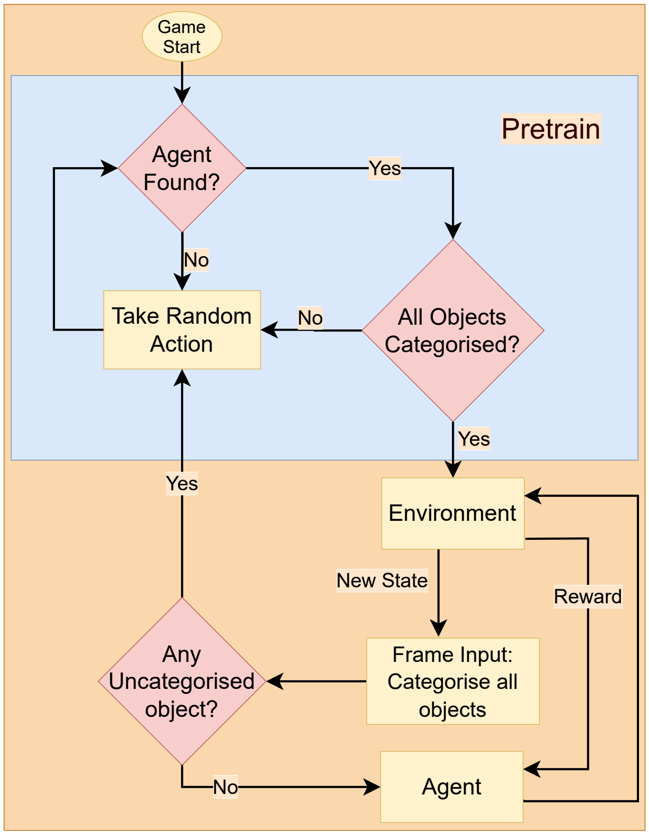

Figure 2: Flowchart of the whole pipeline

Categories

Theory-based Reinforcement Learning methods, even if showing human-like learning traits, use an object-interaction definition known a priori and focus on exploration and planning with a very strong world understanding (Tsividis et al., 2021). We take a different route here. Rather than working with the objects directly, we focus on relevant object categories based on a primitive evolutionary understanding of object affordances. Such affordances are understood either from the past or the current game experience. The game categories are motivated by a general sense of perception where we identify objects as static or moving and learn from experience if they are dangerous or useful. Humans also learn such object categories having similar affordances over isolated entities and tend to generalize strategies from previously learned knowledge to unseen situations (Perfors & Tenenbaum, 2009; Medin, Wattenmaker, & Hampson, 1987).

For all our games, we utilize only these five simple categories motivated by the affordances they could provide:

- Agent - Agent is detected using a minimal set of inductive biases depending upon the complexity of the environment.
- Static objects - This category involves objects whose position does not change in consecutive frames. In simple games, they could be harmless and provide secondary benefits like protection from bullets. In a more complicated setting, they could be collectable objects necessary to win the game.
- Moving-Good objects - If an object is displaced from its previously occupied position, we classify it as moving. They are the interesting and primary interacting entities apart from the agent. The advantageous category constitutes those objects that give positive rewards on touching. From a game perspective, such objects would be pickables such as keys or eatables such as food.
- Moving-Bad objects - Such objects are the prime obstacle in the game. They give negative rewards or kill the agent on touching. As we perceive a threat and move away, the primary affordance associated with this class is to avoid them.
- Agent objects - Objects spawned by the agent, like agent bullets, constitute this category. The idea behind detecting these objects is that they appear very close to the agent immediately after a key press.

  After a player learns these categories, downstream classification becomes instantaneous. Similar to humans, we store characteristic properties of these categories, such as color. Once the colors are identified, new objects can be immediately assigned their respective categories.

Identifying the Agent

As discussed in the previous section, of all object categories, the agent is of primal importance and requires special attention. By analyzing different games and life settings, we propose a set of inductive biases to mimic how a new player would detect the agent in the game.

Inductive biases

Even though identifying objects and the associated properties occurs concurrently and continuously, we try to solve the agent identification problem by utilizing as little information as possible. Thus, we initialize use only standalone properties and then integrate action responses as the environments get more complicated.

Inductive Bias 1 - Uniqueness. This property suggests that the agent is expected to have a unique form. On the game screen, if two objects appear visually similar, they are less likely to be the agent.

Inductive Bias 2 - Permanence. From a gaming perspective, "permanence" refers to the sustained existence of an entity on a game screen. As the game world is centered around the agent, other objects would enter and exit the world, but the agent is expected to persist at all times unless killed by an undesirable interaction.

Inductive Bias 3 - Action-Object Motion binding. The agent is meant for action. As a final conclusive test, we would assess all the objects for their mobility with different key presses, the intent being that the agent, as an active principle in the game, would be dynamic rather than passive unless killed by an undesirable interaction. Moreover, as a specific key is pressed repeatedly, only the agent is expected to consistently manifest a repeated action, as outlined by (De Freitas et al., 2023).

Agent Action Key Bindings

This involves learning the activities an agent does in response to different key presses. It is a form of reinforcement where the player presses keys to observe the agent's behavior. Through repeated iterations of this exercise, the player gradually discerns the mapping of each action to a specific outcome on the screen. We apply a similar principle by taking random actions in our games and observing the changes in the agent's position to assess action-key bindings. Thus, evaluating movement action key bindings is straightforward. For bullet firing, we check for the generation of a new object near the agent immediately after a key is pressed. If this occurrence repeats until a specified threshold, we assign the key's affordance as "Fire."

Implementations

Since object detection is a well-researched field, for our tests, we start with a preexisting list of objects. Then, we categorize these objects into the previously mentioned groups based solely on their bounding box and color.

Incorporating the above object definitions, we try to learn game playing using the Q learning algorithm (Watkins & Dayan, 1992). For Q-learning to work, we need to process our object definitions to construct a state representation that is concise and, at the same time, rich enough for the agent to have sufficient winning information. All the games are fed with the parsed object category information, and we build a state relevant to each game setting. Specifically, we take 2k+1 relative orientation bits, two bits to denote the left and right boundary, and 1 bit to mark the presence of agent bullet if applicable to the game. The 2k+1 orientation bits represent the time it takes for the agent to reach each bit while stationed at the k^{th} bit and store the time it would take for a moving object to cover the horizontal distance from the agent.

GVGAI Games

We modify the MyAliens game from GVGAI (Perez-Liebana et al., 2019) into two variants to test our hypotheses on human-like learning.

MyAliens - variant 1 (MyAliensV1) In this game, the objective is to avoid getting hit by any moving object falling from the top till timeout. All the moving objects kill the agent on touching, so the agent has to learn to avoid them as long as possible.

MyAliens -variant 2 (MyAliensV2) This game has two types of moving objects - one giving positive rewards and one killing the agent. The agent has to learn to collect five positively rewarding objects before timeout to win the game .

Custom Games

Additionally, we also test two custom games to check our hypothesis on more visually exciting games.

Roadrash - Car Driving In this game, the player car has to avoid crashing into the incoming traffic cars. There are only two categories present - agent and moving bad objects. The vehicles can drive only in 4 lanes, making the game very challenging under heavy traffic.

SpaceInvaders This game is based on the classic Atari Space Invaders and has the same features with better visuals. The enemies travel horizontally and then move a row down while shooting bullets at the agent spaceship. The agent can shoot only one bullet at a time.

Generalization Experiments

Our goal here is not to defeat a Deep Reinforcement learning algorithm but to show that using a methodology like ours has certain benefits that opens up new avenues for mimicking human learning characteristics. For all the tests, we train DQN for 10e6 with linear exploration decay from 1 to 0.01.

First, we test our games to see their learning capacity compared against a DQN agent for all four games. For this, we plot the normalized average scores over 20 runs for model vs. epoch, where epoch is defined as one game run loop. We normalize the scores as follows

$$Normalized\ score = \frac{actual\ score}{maximum\ achievable\ score}$$

MyAliensV1 has five levels with different placement of spawn points for moving enemies with a maximum score of 50, +10 for winning each level, and -10 for losing. We test MyAliensV2 for three levels, with a maximum achievable score of 30. Apart from winning or losing ± 10, the agent also gets a reward of +1 for collecting a food item. The level is won when ten such items are collected.

In Roadrash, the agent needs to survive for 300 steps. Humans, after learning how to play a game, would easily adapt to slight variants of the game. SpaceInvaders has two levels with a maximum achievable score of 1000, from +10 received for killing each of the 50 enemy spaceships over the two levels.

To test the generalized ability of our agent to mimic human-like learning, we run the agent in different variations of the games. For all such cases, we train the game only on the base variant of the game.

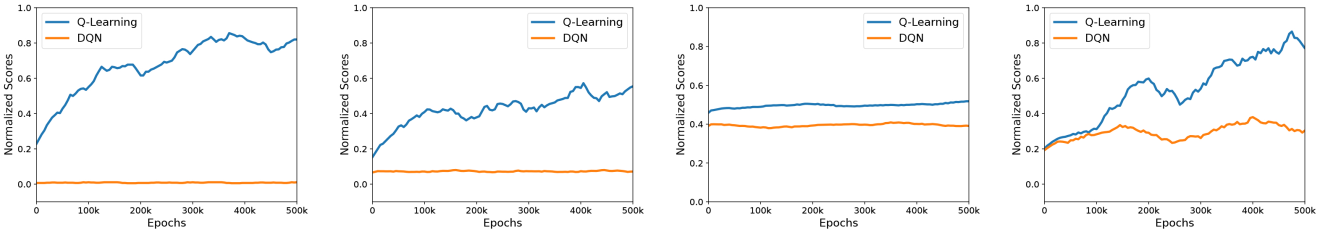

Figure 3: Affordance-based Q learning vs. Image-based DQN Normalized Score per epoch plots. a) MyAliensV1 - DQN is probably still exploring as it could not learn any valuable action. b)MyAliensV2 - Both algorithms found difficulty; Q-learning still shows signs of learning, but DQN could not clear even the first levels for both variants of MyAliens. c)Roadrash - Very stochastic game with many occasions where avoiding collision is impossible. Q-learning still does better than DQN. d)SpaceInvaders - our algorithm easily learns gameplay using its object-based representation.

Table 1: Normalized score for DQN vs. our method averaged over 20 runs of the games. All the models are trained for 1 Million epochs. SpaceInvaders 2 levels and a maximum achievable score of 1000. MyAliensV1 has five levels with a maximum score of 50 and MyAliensV2 has three levels with a maximum score of 30. Roadrash ends if the agent can avoid collision for 300 steps, and the score is measured in the number of steps moved.

Modifications	**MyAliensV1**		**MyAliensV2**		**Roadrash**		**Space Invaders**	
	DQN	Ours	DQN	Ours	DQN	Ours	DQN	Ours
Random Action	- 0.20		- 0.33		0.27		0.27	
Base-Variant	- 0.08	0.80	- 0.23	0.57	0.45	0.50	0.51	1.0
Mod-Position	- 0.20	0.74	- 0.27	0.47	NA	NA	0.28	0.42
Mod-ColorSize	- 0.20	0.80	- 0.27	0.57	0.40	0.48	0.31	1.0
Mod-Image	NA	NA	NA	NA	0.38	0.47	0.30	1.0

We explore three types of game variations:

- Mod-Position: Changes the default position of moving enemies.
- Mod-ColorSize: Alters the size and color of image objects.
- Mod-Image: Substitutes default game images.

GVGAI games do not allow modification of object size, and all the objects within the game are made from unit-sized colored rectangles. Thus, the Mod-Image variant is not applicable as no images are loaded. In Roadrash, enemy cars are spawned randomly. Therefore, the Mod-Position variant will not make any difference in the game.

We train the DQN algorithm using a batch size 32 on each game run loop with experience replay. The Q-Learning agent is trained only once on each run on the latest experience. Thus, even on the same level of epochs, DQN weights are updated 32 times more than Q-Learning.

Results and Discussion

To test the efficacy of our category-level representations, we run two kinds of tests. First, we compare DQN and our method under varying training durations. Scores from the trained models from different training epochs are plotted, and we analyze the agent's normalized score (Figure 3). One Epoch is defined as one run of the game loop. Even though Q-Learning updates 32 times less than DQN, it is able to learn correct decisions quickly. We plot the results up to 0.5 million epochs, equivalent to approximately 2 hours of gameplay at 60 frames per second, and in most cases, DQN did not show any improvement owing to its sample inefficiency.

Our algorithm demonstrates strong performance in MyAliensV1, successfully winning all five levels. In SpaceInvaders also, it wins both the levels. However, MyAliensV2 presents a more challenging scenario, requiring the agent to distinguish between moving-good and moving-bad categories and collect ten of the good ones before a timeout to win the level. Here also, our method does well, but the performance degrades as compared to MyAliensV1, primarily because our agent has only a 9-bit state space representation, i.e., it can see nearest objects only within a range of four units on both the left and right sides. Given that the moving-good category is dispersed over a broader x-range of thirty bits, the agent often struggles to locate the moving-good category within its narrow field of vision. Consequently, the time elapses before the agent can collect the required ten items, impacting its overall performance in this more complex scenario.

The escalating difficulty in subsequent levels, coupled with a reduction in the number of moving-good spawn points, adds to the complexity of the task. We tested a broader state representation with 25 bits, but the learning became computationally intractable, and the agent struggled to learn meaningful affordances. Nevertheless, even with a limited view, our agent could clear the first two levels and failed after collecting a few items from the moving-good category on the third level in most runs (Table 1).

For Roadrash, even though our method does better, we do not see substantial performance gain with training. The game has four lanes, with enemies spawning stochastically in any of them. Thus, in many cases, all four lanes get blocked, and a crash becomes unavoidable. In other situations, avoiding accidents requires precise control because of the crowded structure of game objects. So, even a reasonably learned agent could not perform very well in this game, and the performance was more-or-less stagnant. Nonetheless, our algorithm still fairs better against the DQN agent.

Our second set of comparisons focuses on the transferability of the acquired knowledge. For deep learning algorithms, object level alterations, such as changing object colors, can have devastating consequences (Lake, Ullman, Tenenbaum, & Gershman, 2017). On the other hand, humans can easily manage such variations. Our results indicate that, unlike a DQN agent, our category-based method exhibits no notable deterioration, aligning more with human-level gameplay (Table 1).

This is primarily because, at the category level, the state representations remain relatively stable despite the above generalization modifications of the games. Consequently, our algorithm's performance does not degrade with these variations. It's noteworthy that both models in these comparisons are trained for one million epochs. However, for MyAliensV1 and MyAliensV2, DQN is still in its exploration phase, exhibiting minimal performance improvement, and the introduced variations further degrade its performance. This is particularly evident in SpaceInvaders, where the DQN agent while displaying some learning traces in the base variant, regresses to the level of a random agent when faced with varying input pixel combinations. As the Roadrash game is challenging from the start, there is little difference after making a difficult game more difficult.

Among all the alterations, only SpaceInvaders Mod-Position resulted in a substantial decline in Q-learning performance. This is primarily due to position modifications creating new, and previously unseen, state representations. In this setting, as the enemies get randomly arranged, some enemies get placed too close to the agent. As such states are previously unseen, a table-based Q-learning agent struggles to navigate this variation (Table 1). Such instances could potentially be avoided by using techniques to extrapolate for unseen states based on prior experience. Apart from this, other game modifications consistently exhibit performance similar to the unmodified original versions of the games, as is also expected from a human player.

Thus, our comparisons show that an object-based representation, even if applied using a model-free algorithm, offers much better sample efficiency (Figure 3). This improvement is evident in results with environmental perturbations, such as varying enemy positions and differently shaped enemies, among other variations. The primary factor contributing to this enhanced performance is the category-based representation, in which minor perturbations do not alter the game representation significantly, which might be more prominent if all objects are treated as separate entities and is definitely visible for pixel-based model-free methods like DQN.

The results are visible with Q learning, a discrete state algorithm. We anticipate similar result translation with continuous state approximations, a clear recipe for future work.

Conclusion

Making machines learn and act like humans is an important goal in Artificial General Intelligence(AGI) (Lake et al., 2017; Tsividis et al., 2017; Pouncy, 2022). We look at this task from the eyes of a novice player discovering gameplay dynamics. Building such a state in machines is an interdisciplinary task. Drawing inspiration from previous works in cognitive psychology, we try to develop a category-inspired concept of object representation. Building upon this state representation, we show that machines can exhibit certain similarities to human-like learning in game playing. While numerous studies try to reach this overarching goal of AGI, using object representation and probabilistic generative modeling (Ellis et al., 2023; Tsividis et al., 2021), we do not find the affordance-based category representation in any of them, which is our novel contribution. While we try to emulate infant-like learning in game playing, we utilize just the visual cues for decision-making instead of working in a model-based setting, and exploring the effect of our representations with such planning would be an exciting way forward.

Limitations

Due to its model-free nature, our agent is still not as sample-efficient as a human, but it does well on the generalizability task. Works in theory-based RL paradigm, take the model as given and explore planning within such a framework (Tsividis et al., 2021; Pouncy, 2022). In contrast, our approach starts from a near-blank state, and we hope to extend this concept to learn a model of the environment based solely on core knowledge and experience.

In our demonstration, we adopt a Q learning-based approach to illustrate that learning to play games from first principles and category-level information can be incorporated in machines to instill certain human-like learning traits, such as fast generalization for new objects and interpretable affordance-based behavior. On a granular analysis, we find that most of the instances where Affordance-based Q learning fails are those that have previously unseen states. This mirrors situations where humans find themselves in new and uncertain environments. What is expected of humans in such scenarios is to generalize from previous experience to make an informed decision, a definite recipe for future work.

References

Baillargeon, R. (2004). Infants' physical world. *Current directions in psychological science*, *13*(3), 89–94.

Csibra, G. (2008). Goal attribution to inanimate agents by 6.5-month-old infants. *Cognition*, *107*(2), 705–717.

De Freitas, J., Uğuralp, A. K., Oğuz-Uğuralp, Z., Paul, L., Tenenbaum, J., & Ullman, T. D. (2023). Self-orienting in human and machine learning. *Nature Human Behaviour*, 1–14.

DiCarlo, J. J., Zoccolan, D., & Rust, N. C. (2012). How does the brain solve visual object recognition? *Neuron*, *73*(3), 415–434.

Ding, M., Xu, Y., Chen, Z., Cox, D. D., Luo, P., Tenenbaum, J. B., & Gan, C. (2023). Embodied concept learner: Self-supervised learning of concepts and mapping through instruction following. In *Conference on robot learning* (pp. 1743–1754).

Dubey, R., Agrawal, P., Pathak, D., Griffiths, T. L., & Efros, A. A. (2018). Investigating human priors for playing video games. *arXiv preprint arXiv:1802.10217*.

Ellis, K., Wong, L., Nye, M., Sable-Meyer, M., Cary, L., Anaya Pozo, L., … Tenenbaum, J. B. (2023). Dreamcoder: growing generalizable, interpretable knowledge with wake–sleep bayesian program learning. *Philosophical Transactions of the Royal Society A*, *381*(2251), 20220050.

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Hessel, M., Modayil, J., Van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., … Silver, D. (2018). Rainbow: Combining improvements in deep reinforcement learning. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 32).

Justesen, N., Rodriguez Torrado, R., Bontrager, P., Khalifa, A., Togelius, J., & Risi, S. (2018). Illuminating generalization in deep reinforcement learning through procedural level generation. In *Neurips workshop on deep reinforcement learning*.

Kansky, K., Silver, T., Mély, D. A., Eldawy, M., Lázaro-Gredilla, M., Lou, X., … George, D. (2017). Schema networks: Zero-shot transfer with a generative causal model of intuitive physics. In *International conference on machine learning* (pp. 1809–1818).

Kriegeskorte, N., Mur, M., Ruff, D. A., Kiani, R., Bodurka, J., Esteky, H., … Bandettini, P. A. (2008). Matching categorical object representations in inferior temporal cortex of man and monkey. *Neuron*, *60*(6), 1126–1141.

Kumar, S., Dasgupta, I., Daw, N. D., Cohen, J. D., & Griffiths, T. L. (2023). Disentangling abstraction from statistical pattern matching in human and machine learning. *PLoS computational biology*, *19*(8), e1011316.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and brain sciences*, *40*, e253.

Lu, M., Shahn, Z., Sow, D., Doshi-Velez, F., & Li-wei, H. L. (2020). Is deep reinforcement learning ready for practical applications in healthcare? a sensitivity analysis of duel-ddqn for hemodynamic management in sepsis patients. In *Amia annual symposium proceedings* (Vol. 2020, p. 773).

Medin, D. L., Wattenmaker, W. D., & Hampson, S. E. (1987). Family resemblance, conceptual cohesiveness, and category construction. *Cognitive psychology*, *19*(2), 242–279.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., … others (2015). Human-level control through deep reinforcement learning. *nature*, *518*(7540), 529–533.

Mohan, A., Zhang, A., & Lindauer, M. (2023). Structure in reinforcement learning: A survey and open problems. *arXiv preprint arXiv:2306.16021*.

Perez-Liebana, D., Lucas, S. M., Gaina, R. D., Togelius, J., Khalifa, A., & Liu, J. (2019). *General video game artificial intelligence* (Vol. 3) (No. 2). Morgan & Claypool Publishers. (`https://gaigresearch.github.io/gvgaibook/`)

Perfors, A., & Tenenbaum, J. (2009). Learning to learn categories..

Pouncy, T. (2022). *Theory-based reinforcement learning: A computational framework for modeling human inductive biases in complex decision making domains*. Unpublished doctoral dissertation, Harvard University.

Pouncy, T., & Gershman, S. J. (2022). Inductive biases in theory-based reinforcement learning. *Cognitive Psychology*, *138*, 101509.

Pouncy, T., Tsividis, P., & Gershman, S. J. (2021). What is the model in model-based planning? *Cognitive Science*, *45*(1), e12928.

Shakya, A. K., Pillai, G., & Chakrabarty, S. (2023). Reinforcement learning algorithms: A brief survey. *Expert Systems with Applications*, 120495.

Spelke, E. S. (1990). Principles of object perception. *Cognitive science*, *14*(1), 29–56.

Spelke, E. S., & Kinzler, K. D. (2007). Core knowledge. *Developmental science*, *10*(1), 89–96.

Tomov, M. S., Tsividis, P. A., Pouncy, T., Tenenbaum, J. B., & Gershman, S. J. (2023). The neural architecture of theory-based reinforcement learning. *Neuron*, *111*(8), 1331–1344.

Tsividis, P. A., Loula, J., Burga, J., Foss, N., Campero, A., Pouncy, T., … Tenenbaum, J. B. (2021). Human-level reinforcement learning through theory-based modeling, exploration, and planning. *arXiv preprint arXiv:2107.12544*.

Tsividis, P. A., Pouncy, T., Xu, J. L., Tenenbaum, J. B., & Gershman, S. J. (2017). Human learning in atari. In *2017 aaai spring symposium series*.

Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine learning*, *8*, 279–292.