Intrinsic Goals for Autonomous Agents: Model-Based Exploration in Virtual Zebrafish Predicts Ethological Behavior and Whole-Brain Dynamics

Reece Keller[1,2*] **Alyn Kirsch** [2] **Felix Pei**[1] **Xaq Pitkow**[1,3]
Leo Kozachkov[4,†] **Aran Nayebi**[3,1,2,†]
[1]Neuroscience Institute, Carnegie Mellon University
[2]Robotics Institute, Carnegie Mellon University
[3]Machine Learning Department, Carnegie Mellon University
[4]IBM Thomas J. Watson Research Center, IBM Research
{rdkeller, fcpei, akirscht, xpitkow, anayebi}@andrew.cmu.edu
leokoz8@gmail.com

Abstract

Autonomy is a hallmark of animal intelligence, enabling adaptive and intelligent behavior in complex environments without relying on external reward or task structure. Existing reinforcement learning approaches to exploration in reward-free environments, including a class of methods known as *model-based intrinsic motivation*, exhibit inconsistent exploration patterns and do not converge to an exploratory policy, thus failing to capture robust autonomous behaviors observed in animals. Moreover, systems neuroscience has largely overlooked the neural basis of autonomy, focusing instead on experimental paradigms where animals are motivated by external reward rather than engaging in ethological, naturalistic and task-independent behavior. To bridge these gaps, we introduce a novel model-based intrinsic drive explicitly designed after the principles of autonomous exploration in animals. Our method (3M-Progress) achieves animal-like exploration by tracking divergence between an online world model and a fixed prior learned from an ecological niche. To the best of our knowledge, we introduce the first autonomous embodied agent that predicts brain data entirely from self-supervised optimization of an intrinsic goal—without any behavioral or neural training data—demonstrating that 3M-Progress agents capture the explainable variance in behavioral patterns and whole-brain neural-glial dynamics recorded from autonomously behaving larval zebrafish, thereby providing the first goal-driven, population-level model of neural-glial computation. Our findings establish a computational framework connecting model-based intrinsic motivation to naturalistic behavior, providing a foundation for building artificial agents with animal-like autonomy.

1 Introduction

Animals exhibit remarkable autonomy, navigating complex environments through self-directed, internally driven behaviors rather than solely responding to external rewards or immediate physiological needs. Unlike typical artificial agents designed to optimize explicit, predefined task objectives in well-defined problem settings, animals intrinsically explore and adapt in open-ended, naturalistic environments where goals are neither clear nor stable. This capacity for autonomous behavior allows

*Corresponding author.
†These authors jointly supervised this work.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

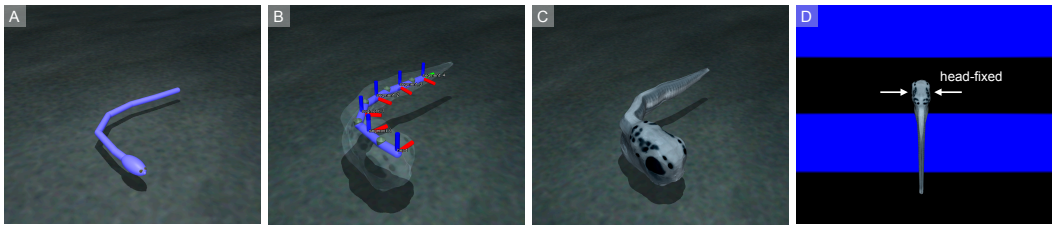

Figure 1: Simulation of the zebrafish agent in a physics-based virtual environment. A) The 6-link embodiment geometry [1] in a environment with dynamic fluid forces. B) The agent controls the torque exerted by motors at each joint (5 DoF) to swim and navigate its environment. C) A custom cosmetic skin to mimic the appearance of larval zebrafish. D) A virtual environment matching the experimental parameters of the open loop protocol Mu et al. [2]. The root joint located at the head is fixed during training.

organisms, including humans, to flexibly engage in abstract thinking, exploratory learning, and innovative problem-solving. Central to this autonomy is the ability to generate intrinsic goals, seek novel experiences, and continuously refine internal models of the world that inform future actions. Understanding the interplay between extrinsic and intrinsic motivations remains a fundamental challenge in both systems neuroscience and artificial intelligence, particularly for building robust agents capable of lifelong autonomy.

However, existing methods in reinforcement learning for intrinsic motivation using world-models —such as curiosity-driven exploration and variants [3–6]—exhibit inconsistent behavioral patterns due to difficulties distinguishing between controllable and uncontrollable stimuli, coping with non-stationary environments, and avoiding the "noisy TV" problem [7]. For example, agents driven by reinforcing prediction-error alone gets locked into pursuing inherently unpredictable or irrelevant aspects of the environment, hindering their ability to develop robust exploratory behaviors [3]. Moreover, existing methods typically fail to produce structured and stable behavioral transitions that characterize genuine autonomous behaviors observed in animals and children [8].

At the same time, systems neuroscience has historically overlooked the cellular basis of autonomy, favoring experimental paradigms centered on explicit external rewards to motivate task-dependent behaviors rather than free, unconstrained exploration. As a result, the cellular mechanisms underlying naturalistic autonomy—particularly those involving whole-brain interactions—remain poorly understood. Growing experimental, computational, and theoretical evidence suggests that non-neuronal cells, especially astrocytes, play a critical role in the generation of intelligent behavior [2, 9–14]. Owing to their close association with neurons, distinctive connectivity (with a single astrocyte capable of interfacing with up to a million nearby synapses) and ability to perform integrative computations using a hierarchy of timescales, astrocytes are well-positioned to support adaptive, naturalistic, goal-directed behavior [15–20].

A Dataset for Animal Autonomy To investigate these questions, we study autonomy in larval zebrafish—a uniquely valuable animal model due to their optical transparency, which affords whole-brain calcium imaging via light-sheet microscopy [21]. Given that the cellular basis of autonomous behavior remains largely unknown, whole-brain imaging allows us to search across the entire recorded population of over 250,000 cells (roughly 125K neurons and 125K astrocytes) to measure how our proposed model aligns with brain activity [2] [3]. We examine a particular cognitive behavior known as futility-induced passivity, an experimentally induced state of "giving up" observed across the animal spectrum [2, 22–24]. In zebrafish, futility-induced passivity occurs when their swimming effort in a virtual environment fails to simulate optic flow, a key visual feedback signal associated with motor-action under ethological circumstances. After some time in the passive state, zebrafish reattempt to swim–an exploratory behavior to assess whether the open-loop dynamics of the environment have changed. This cycle between active and passive states continues over the course of a trial.

Mu et al. [2] offers rigorous experimental evidence suggesting this transition to passivity is driven neuron-glial interactions in the lateral medulla oblongata (L-MO) that detect and accumulate sensori-motor feedback error. Noradrenergic (NE) neurons signal swim failures which slowly activate radial

[3]The dataset we use is open-source and freely available here.

astrocyte with extending processes to the NE cluster of L-MO. At a critical level of intracellular calcium, astrocyte processes downstream of L-MO activate GABAergic neurons that suppress premotor neurons relevant for swimming. With extensive manipulation experiments, Mu et al. [2] demonstrate that futility-induced passivity is not due to fatigue, struggle, or positional homeostasis. Specifically, when visual feedback from closed-loop trials was replayed during the open-loop condition, zebrafish still transitioned to a passive state. This indicates the behavior is mediated by an internal model that is used to detect active sensorimotor feedback error, providing strong evidence for model-based intrinsic motivation.

Contributions In this work, we study neural-glial computations and their relationship to autonomous animal behavior by training an embodied zebrafish agent with model-based intrinsic motivation and studying its emergent behavior and internal representations. We introduce a novel intrinsic reinforcement learning algorithm termed Model-Memory-Mismatch Progress (3M-Progress) alongside a virtual environment that captures the fundamental physics of the zebrafish embodiment. Our approach leverages an internal model that continually compares the agent's online memory formed by its current sensory experience against an ethologically relevant prior memory; intrinsic reward then reinforces transitions between behaviors that maximize the divergence between memories relative to its temporal history. 3M-Progress is uniquely capable of producing stable, ethologically-relevant behavioral transitions among several state-of-the-art exploration algorithms in reinforcement learning.

By training embodied agents with 3M-Progress, we successfully replicate both the robust behavioral patterns and whole-brain neural-glial dynamics in autonomously-behaving zebrafish. Capturing nearly all of the variance in neural and astrocytic activity, this marks the first predictive and normative model of neural-glial computation. Our agent was *not trained on any behavioral or neural data*, and thus represents the first autonomous embodied agent that predicts brain data completely from optimizing a self-supervised, intrinsic goal. To summarize, our technical contributions are:

- 3M-Progress, a novel intrinsic reward that leverages an ecological dynamics prior to guide exploration in new environments.
- Emergence of an autonomous behavior known as futility-induced passivity in 3M-Progress agents, closely matching larval zebrafish behavior.
- Alignment between whole-brain calcium response and 3M-Progress agents, providing the first goal-driven model of neural-glial computation.
- A general modeling perspective positioning intrinsic reinforcement learning as a computational framework for understanding autonomy in animals.

2 Related Work

Neural-Glial Models Although glial cells—especially astrocytes—are increasingly recognized as crucial to adaptive brain function [2, 9, 17, 14, 19, 20, 11, 25, 26], computational models of neural-glial interactions remain underdeveloped [27]. Existing models of neuron-astrocyte dynamics typically fall into two categories: phenomenological models that reproduce specific experimental findings like calcium oscillations or epileptic activity [28–31], and simplified, "bottom-up" mathematical models that explore theoretical principles based on astrocytes' unique morphology and anatomy [32–35, 12, 13]. While important, these models are not directly applicable to our setting because they are (a) not yet directly trained on ethological tasks, embodied, or quantitatively validated against real brain data, and (b) typically focused on a single astrocyte rather than a population of astrocytes. In contrast, we adopt a "top-down" approach: we train a general-purpose recurrent architecture to control an embodied agent to perform ethologically-relevant behavior and find that this imposes strong constraints on the learned representations, allowing us to identify units whose activity patterns closely match those observed in neurons and astrocytes of larval zebrafish.

Curiosity-driven Exploration Exploration using self-supervised world-models has demonstrated promising success in several standard reinforcement learning domains, and even more recently in language modeling [36]. Methods like learning progress [6] and Random Network Distillation (RND) [4] were primarily evaluated using either handcrafted object-centric or state observations with low-dimensionality embodiments, limiting the applicability to pixel-based environments or

continuous control. The Intrinsic Curiosity Module (ICM) [3] uses a pixel-encoder trained with an inverse-dynamics loss to predict features rather than raw states, but was evaluated on discrete pixel-based environments like Doom and Atari. While some recent works, such as LEXA [37] or Plan2Explore [38], extend intrinsic curiosity (specifically, Disagreement [5]) to continuous control from visual inputs, success of the exploration policy is defined relative to downstream task generalization. Since these methods do not independently evaluate the quality of the exploration policy, it remains unclear whether these algorithms are powerful enough to learn complex behaviors. In contrast, we investigate completely open-ended autonomous behavior in reward-free, continuous MDPs with high-dimensional observations, turning the focus on the ability of the exploration algorithm to develop ethological, interpretable behaviors independent of any downstream task.

Embodied AI in Neuroscience Several works have leveraged embodied AI to bridge computational models with neuroscience, including virtual animal models such as the virtual rodent [39, 40], which facilitates grounded studies of motor control by replicating rodent motor behaviors across various tasks using imitation learning; the virtual fruit fly, a biomechanically detailed model matching both the visual system and basic flight capacity used to study a diverse range of behaviors driven by imitation learning [41–43]; the OpenWorm project [44], a biophysically accurate simulation of the *C-elegan* nematode, but has yet to be combined with deep learning and task-optimization; and Zador et al.'s Embodied Turing Test position paper [45], which emphasizes developing AI models whose sensorimotor capabilities rival those of their biological counterparts. There are several existing works that apply task-optimization to control details musculoskeletal models [46–48], use robots to implement and validate neural circuits in zebrafish [49], and model animal-like social behavior or object perception using digital twins [50, 51].

Our work extends these directions by providing the first predictive and normative computational formalization of neural-glial interactions in embodied agents, thereby validating a circuit model recently proposed by Mu et al. [2]. Most importantly, our model is trained entirely via intrinsically-motivated exploration, unlike previous approaches that constrain behavior and their resulting neural representations by supervised learning [39–41]. To the best of our knowledge, this marks the first completely autonomous, embodied agent model of behavioral and brain data in neuroscience, pointing towards a promising computational framework for understanding naturalistic, task-independent behavior in biological systems using intrinsic reinforcement learning.

3 Methods

Virtual Zebrafish Environment Animals are physically coupled to the environment through their embodiment; this coupling is often referred to as the sensory-motor or perception-action feedback loop. Physical embodiment imposes strong constraints on both the sensory stream from which an agent learns meaningful representations of the world and the actuation system by which the agent manifests behavior. Following this top-down view of biological systems, we construct an embodied agent and custom virtual environment in the MuJoCo physics engine [52] specifically designed after the ethology of the zebrafish (Figure 1). Leveraging the procedurally generated n-link swimmer and built-in inertial fluid model from the Deepmind Control Suite (dm-control) [1], we construct an ethological environment (Figure 1A-C) in which the agent can freely behave in the presence of both passive and active fluid currents, similar to the dynamic water environments to which zebrafish are native. To evaluate our agent in the futility-induced passivity task, we construct a second environment closely matching the open-loop experimental protocol in Mu et al. [2] (Figure 1D). In this environment, agents passively experience a high-contrast grating moving away from the egocentric point-of-view, which simulates backward motion. In the closed-loop condition, the agent is head-free and can learn a positional-homeostasis policy to counteract the perceived backward flow. In the open-loop condition, the agent is head-fixed and its swim commands produce no movement. The passive speed of the moving grating, its colors, and sizing relative to the zebrafish body were determined from experimental parameters in Mu et al. [2]. These environments provide sensory and actuator configurations that closely match the basic structure of free swimming as well as the head-fixed protocol, facilitating a meaningful comparison between artificial and biological agents.

Agent Design The zebrafish agent is equipped with a recurrent sensory-cognitive architecture to support perception and action in continuous, high-dimensional environments. Because the autonomous zebrafish behavior recorded by Mu et al. [2] is primarily driven by visual input, we restrict

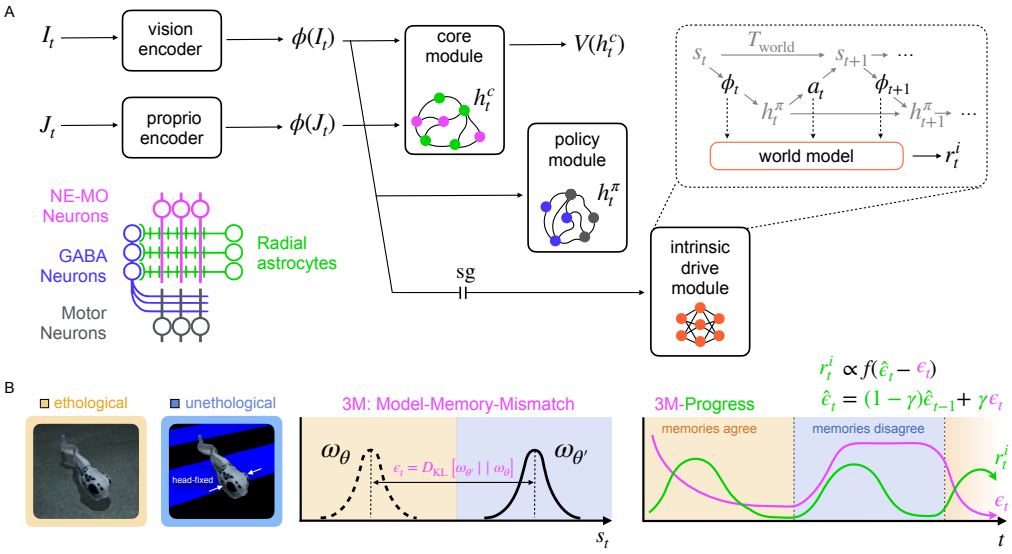

Figure 2: Agent architecture and 3M-Progress. A) Egocentric visual input (I_t) is encoded via a small residual network ϕ_I. Proprioceptive state observations (J_t) are encoded via a small multi-layer perceptron ϕ_J with shortcut paths to both the core and policy module. Sensory features are passed into recurrent LSTM core (h_t^c) and policy (h_t^π) modules that learn a state value function and stochastic policy, respectively. The intrinsic drive module consists of a small multi-layer perceptron that parameterizes a forward dynamics model on sensory features observed from an environment with dynamics T_{world}. B) 3M-Progress uses two memories created from environments with differing transition dynamics. Divergence between the ethological prior ω_θ and the current world-model $\omega_{\theta'}$ defines 3M, which is then is used as input to leaky integrator $\hat{\epsilon}$ to generate intrinsic reward r_t^i.

the sensory stream to a vision encoder operating on images with similar resolution to the visual acuity of zebrafish [53]. Sensory features are used by three distinct cognitive networks with modularized objectives. The core and policy modules are implemented as an actor-critic architecture using Long Short-term Memory (LSTM) networks [54] followed by feedforward decoders trained end-to-end with Proximal Policy Optimization (PPO) to output torques that are used as the control input to the agent's motors [55]. Specific agent implementation details can be found in Appendix E. Although PPO is a model-free reinforcement learning algorithm, the intrinsic drive module (IDM) learns a world model through online experience that approximates the world state-action transition dynamics in sensory feature-space (as described below). However, this internal model functions only to generate intrinsic reward and is not used for planning or model-based control.

Model-based Intrinsic Reinforcement Learning In classical reinforcement learning, policies are obtained by mapping a desired behavior to maximizing a reward function [56]. This function is one component of the Markov Decision Process (MDP) that formally specifies a task M as a tuple $(\mathcal{S}, \mathcal{A}, T, p_0, r)$, where $\mathcal{S}$ is the space of environment states, $\mathcal{A}$ is the space of actions, $T : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ is the transition dynamics (where $\mathcal{P}(\mathcal{S})$ is the set of probability densities over $\mathcal{S}$), p_0 is the distribution over initial states, and r is the reward function. Importantly, when reward is defined as part of the task MDP, it is *extrinsic* and provided from the environment. However, learning complex behaviors using extrinsic reward can fail for many real-world situations where r is extremely sparse (winning a long-horizon game like Go) or intractable to begin with (intellectual pursuits such as knowledge acquisition) without a powerful exploration mechanism to guide behavior. Model-based intrinsic motivation is a class of such mechanisms that leverage predictive world models to convey exploration-relevant information using prediction-error to the form an intrinsic reward r_t^i. In the absence of any extrinsic reward r_t^e, policy learning with intrinsic motivation is completely self-supervised. In this work, we consider intrinsic motivation driven by forward dynamics world models ω. Let θ parameterize a neural network and $\omega_\theta : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(S)$. For simplicity, we assume a fixed variance Gaussian density $\omega_\theta = \mathcal{N}(\phi(\mathbf{s}_{t+1}) \mid \phi(\mathbf{s}_t), \mathbf{a}_t; \mu_\theta, \sigma I)$. Here, $\phi(\cdot)$ denotes the

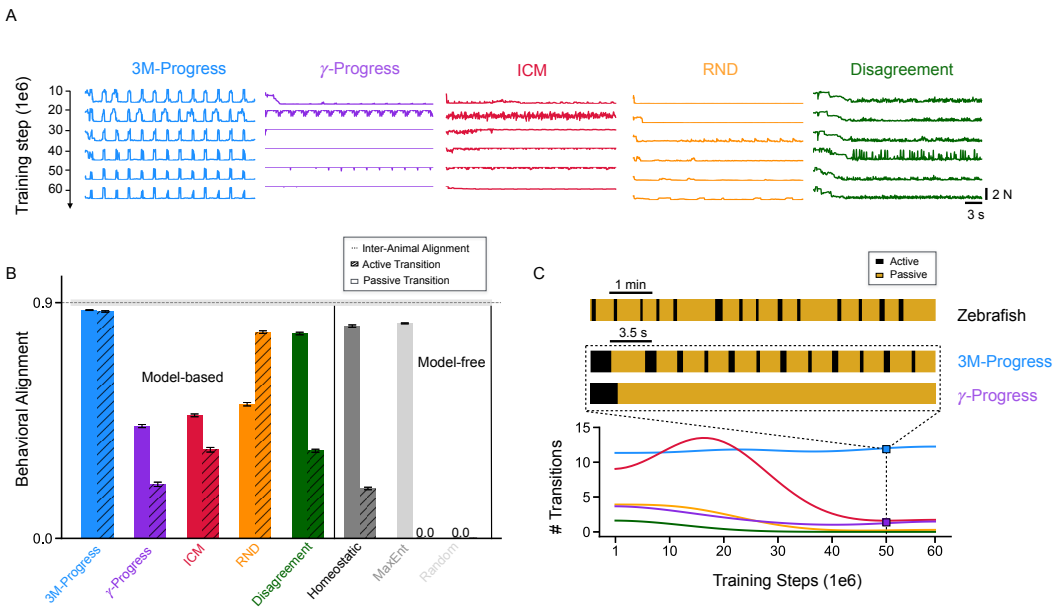

Figure 3: Model-behavioral alignment. A) Swim power traces of artificial agents with different intrinsic drives throughout training. B) Pearson's r correlation between agent swim power (joint torques) and zebrafish swim power (motor neuron activity) for active and passive behavioral transitions. C) (Top) Timecourse of active-passive transitions in zebrafish compared against stationary behavior from progress-driven agents for a single rollout. (Bottom) Average number of behavioral transitions per rollout across training for different intrinsic drives.

concatenated sensory embeddings from $\phi_I(\cdot)$ and $\phi_J(\cdot)$. This class of methods is appealing from two complimentary perspectives. On one hand, it is computationally straightforward; learning an internal model of an MDP's transition dynamics is a natural way to measure novelty of the states visited under the agent's policy—states with a high prediction error under the internal model indicate regions of the state-action space that are poorly understood or rarely visited. On the other hand, it is well motivated by experimental evidence in both neuroscience and psychology; numerous empirical studies suggest humans and animals depend on predictive models of the world for decision-making [57–61]. Thus, world models are well-positioned as a primary substrate for intrinsic motivation; however, we emphasize that a world model is best viewed as *only a substrate*, as exactly how it should be used for intrinsic motivation remains unclear.

Model-Brain and Inter-Animal Alignment To rigorously evaluate our embodied agents against whole-brain data collected by Mu et al. [2], neural-glial recordings from zebrafish were first aligned to periods specifically capturing behavioral state transitions between active swimming and futility-induced passivity. Correspondingly, latent states and predicted neural-glial responses from our agents were aligned to these same experimentally defined epochs using event-triggered averaging around transition events. To quantify model-brain alignment, we employed a stringent "One-to-One" mapping, in which each neural or astrocytic unit from the zebrafish brain was matched directly to the single most correlated artificial unit from the virtual embodied agent. Although this mapping is typically too restrictive for capturing inter-animal alignment in heterogeneous neural populations for brain-region-specific (e.g. *not* whole brain) data in other animals (as shown in prior sensory [62] and cognitive systems [63] where full linear regression is necessary), we found it was fully sufficient here due to the exceptional whole-brain reliability of zebrafish neural-glial responses during futility-induced passivity. Indeed, inter-animal alignment computed from pairwise correlations between individual zebrafish was nearly 100%, establishing a robust empirical ceiling for evaluating model performance.

Inter-animal alignment was computed as the correlation of neural-glial response patterns across pairs of zebrafish within aligned epochs, thus formally defining the upper bound of predictivity achievable by any candidate model (the mathematical details of the chosen metric can be found in Appendix F). This approach ensures that successful computational models achieve neural predictivity

indistinguishable from biological measurements, within the bounds set by natural biological variability. Together, these procedures allowed us to precisely quantify model-brain alignment; achieving close correspondence between artificial and biological responses under this strict One-to-One mapping criterion provides strong evidence that our intrinsic motivation-driven embodied agents effectively capture the detailed neural-glial dynamics underlying autonomous behavioral transitions in zebrafish.

Model-Behavioral Alignment To quantify similarity between biological zebrafish and artificial agents, we use Pearson's r as a metric between the respective swim power readouts for each system surrounding state transitions. In zebrafish, swim power was calculated as the standard deviation over a 10ms window of recorded tail motor nerve signals following Mu et al. [2]. We use the same passive and active transitions windows identified by Mu et al. [2] to compute the inter-animal behavioral alignment across 11 subjects by applying the metric to swim power after smoothing and normalization over a 20 second window surrounding the transition time. For the artificial agents, we take the norm of their joint torques as swim power and identify behavioral transitions as high-frequency changes in swim power above a threshold of 1 Newton. This threshold was determined empirically by recording joint activations during active and passive behaviors in the default swim task in the dm-control suite 6-link swimmer environment [1]. Model-behavioral consistency is then computed by applying the metric between segmented zebrafish data and the agent's swim power surrounding these transitions over a 20-step window.

4 Animal-like Exploration from First Principles: Lessons from Zebrafish

Autonomy in animals enables intelligent and robust decision-making in complex environments, even in the presence of high-entropy or unfamiliar dynamics such as noisy stimuli or laboratory habitats. Reflecting on characteristics of autonomous exploration in zebrafish, we propose two simple desiderata for intrinsic drives that capture animal-like autonomy:

1. *Animals do not perseverate on unpredictable stimuli or pursue stimuli they cannot causally interact with*. Mu et al. [2] demonstrate that zebrafish transition to passive behavioral states when motor commands elicit unpredictable sensory-feedback (unlearnable dynamics) or when sensory-feedback is withheld altogether (uncontrollable dynamics). An intrinsic objective for animal autonomy should avoid unpredictable or uncontrollable stimuli.
2. *Animals exhibit consistent decision-making strategies across repeated encounters of the same context*. Zebrafish exhibit stable behavioral state-switching in the open-loop experimental protocol across trials and subjects, ultimately converging on a single exploration policy [2]. An intrinsic objective for animal autonomy should converge to a stable behavioral policy.

Intrinsic objectives that rely on prediction-error alone [3, 4] reward stochastic environment dynamics and incentivize learning transitions in which the agent has no causal control[4]. Learning progress [6, 7, 64] and Disagreement [5] overcome this by leveraging temporal dynamics or statistics of a world model ensemble, but together with ICM and RND, are formulated as functions of the world model training loss and are thus non-stationary—learning on repeated behavioral strategies drive the training loss towards zero, and so any single behavioral strategy is transient since it is not consistently reinforced. Although these properties are suitable for exploration in some robotics domains, particularly when the policy is supplemented with an extrinsic task reward [36–38, 3, 5], they fail to capture the nature of autonomous exploration in animals.

3M-Progress In order to overcome these drawbacks, we introduce Model-Memory-Mismatch Progress (3M-Progress), a novel intrinsic drive that incorporates these simple normative properties of animal autonomy inspired by zebrafish. Curiosity, disagreement, and learning progress couple intrinsic reward to a moving world model, yielding a non-stationary policy objective. The RL problem becomes a two player minimax game, where the actor seeks states that increase prediction-error and the learner reduces it, leading to a reward landscape that flattens and precludes a stationary optimal policy[5]. Thus, each of the existing curiosity-driven exploration algorithms we consider fails to converge to an exploratory policy. However, learning progress [6] is unique among these algorithms

[4] Although [3] proposed an inverse dynamics feature space to avoid representing uncontrollable stimuli, it's efficacy has not been demonstrated outside of simple pixel-based environments with discrete actions.

[5] See Appendix C.1 for mathematical details.

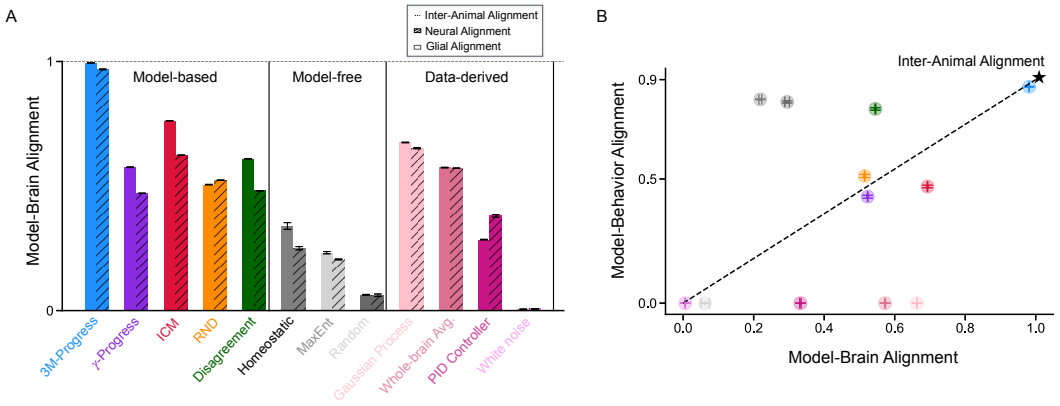

Figure 4: Model-brain alignment averaged across active and passive transitions. A) Noise-corrected Pearson's r correlation between whole-brain neural and glial units and artificial units from trained agents. B) Model scores on behavioral and whole-brain alignment.

in its ability to avoid perseveration on unpredictable and uncontrollable stimuli by leveraging the temporal dynamics of prediction-error. This is achieved by maintaining two world models, an online model and a long-term memory implemented as an exponential weight decay on the online model. The slow timescale memory provides a moving baseline that, when compared with the online model, creates an intrinsic reward that flattens as the temporal dynamics of the online model's predictions flatten—in other words, a computationally efficient and biologically plausible implementation of a time-derivative.

To combine the adaptive behavior afforded by derivative-like operations with an intrinsic goal that admits a stationary solution, we propose to decouple the two memories completely. We achieve this by learning the long-term memory ω_θ in a pretraining environment and deploying it as a fixed prior while the online model $\omega_{\theta'}$ is learned in a new environment with new transition dynamics. The frozen "ethological memory" functions as a static memory primitive: comparing its predictions with the online memory produces a residual that encodes an explicit bias toward transition dynamics that match the pretraining environment, which can then be leveraged to partition the state-action space in the new context into regions where memories systematically agree or disagree [6]. This reflects the idea that animals develop in an ecological niche with characteristic environment dynamics, distilled via experience as an internal world model ω_θ. In new environments with different transition dynamics, such as the experimental protocol in Mu et al. [2], animals can use this prior to seek or avoid regions that match their ecological niche as they learn new world model $\omega_{\theta'}$. To capture this, we define the model-memory-mismatch (ϵ_t), exponential filter ($\hat{\epsilon}_t$), and niche-aware intrinsic motivation r_t^i as

$$\epsilon_t := \mathcal{D}_{\mathrm{KL}}\left[\omega_\theta\left(\phi(\mathbf{s}_{t+1}) \mid \phi(\mathbf{s}_t), \mathbf{a}_t\right) \,||\, \omega_{\theta'}\left(\phi(\mathbf{s}_{t+1}) \mid \phi(\mathbf{s}_t), \mathbf{a}_t\right)\right], \tag{1}$$

$$\hat{\epsilon}_t = (1-\gamma)\hat{\epsilon}_{t-1} + \gamma\epsilon_t, \tag{2}$$

$$r_t^i = |\hat{\epsilon}_t - \epsilon_t|, \tag{3}$$

where $\mathcal{D}_{\mathrm{KL}}$ denotes the Kullback-Leibler divergence and γ is the filter timescale. The non-negativity of this divergence coarsely partitions the state-action space into niche-seeking (model-memory agreement: $\epsilon_t \approx 0$) and niche-avoidance (model-memory disagreement: $\epsilon_t \gg 0$). We design the intrinsic reward mechanistically after γ-Progress [6] to maintain a moving baseline, but filter model-memory-mismatch rather than model parameters.

Due to the moving baseline, the reward does not perseverate on any single partition, either agreement or disagreement. The symmetry enforced by the absolute value encourages periodic exploration between partitions by reinforcing deviations from the moving baseline in either direction. Unlike learning progress, this formulation does not saturate as learning unfolds since ϵ_t is computed using a fixed memory. In fact, as the prediction error from $\omega_{\theta'}$ stabilizes with more environment interactions, the signal-to-noise ratio in ϵ_t only increases, resulting in more robust behavioral patterns. Note that the absolute value is a specific choice of an activation function. The shape of this function and the progress horizon γ determine the relative time spent in each partition. All experiments and

[6] See Appendix C.2 for mathematical details.

baseline algorithms include an action penalty $r_t^a = -\lambda \|\mathbf{a}_t\|_2^2$ to encourage exploration of passive behavior. Figure 2B illustrates a specific example inspired by zebrafish, where ω_θ is learned in an ethological environment—the agent can freely behave and experiences passive fluid forces induced by self-motion as it swims and and active fluid forces as it learns positional homeostasis by resisting an opposing current. The agent is then put into an unethological experimental protocol where it is head-fixed and its swim commands do not elicit sensory feedback. As the agent behaves in this environment, it distills a new memory $\omega_{\theta'}$ from experience.

3M-Progress as a Normative Model of Neural-Glial Computation Owing to the formulation of 3M-Progress, the state-value function must implement units in the core module that are functionally equivalent to neurons in the Noradernergic cluster of the Medulla Oblongata (NE-MO) and radial astrocytes identified by Mu et al. [2] (2A). 3M-Progress detects sensory-motor mismatch using a prior memory as an *expectation* of how action is coupled to sensory-feedback under ethological environment dynamics (2B). This is functionally equivalent to signaling failed swim-attempts by NE-MO neurons in zebrafish. Similarly, the exponential filter is a discrete-time leaky integrator on model-memory-mismatch (NE-MO input), which is functionally equivalent to radial astrocytes that accumulate NE-MO signals during failed swim-attempts and decay during passivity (2C).

5 Experimental Results

3M-Progress Agents Replicate Behavioral Patterns Observed in Zebrafish We first assessed the ability of intrinsic motivation methods to replicate detailed behavioral patterns observed in biological zebrafish during autonomous exploration. Behavioral alignment was quantified by comparing locomotor trajectories and state transitions (active to passive and back) between artificial agents and zebrafish across multiple trials and subjects. 3M-Progress discovers ethologically-relevant behavioral state transitions by 10 million environment steps and sooner, whereas other intrinsic drives display transient strategies and stabilize on complete passivity or activity over training (Figure 3A). Agents trained with 3M-Progress exhibited the highest model-behavior alignment, successfully capturing the dynamics of state transitions recorded in the biological data (Figure 3B). Additionally, agents trained with traditional intrinsic motivation methods such as ICM, RND, Disagreement, and γ-progress showed significantly lower alignment and failed to capture the characteristic stable cycling between active and passive states (Figure 3B-C).

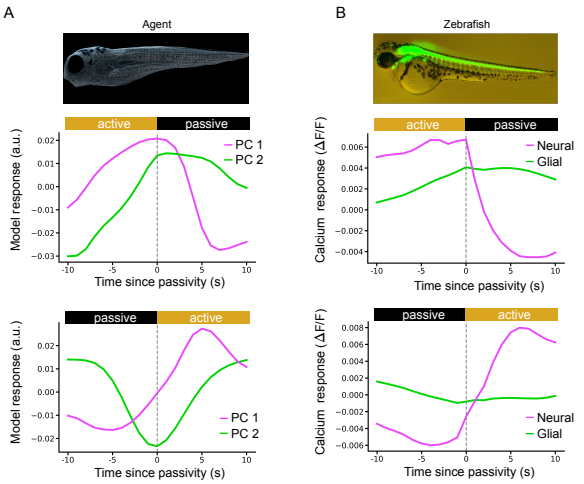

Figure 5: Latent dynamics of 3M-Progress agent's internal activations compared with normalized whole-brain neural-glial response in zebrafish. A) Principal components during passive and active transitions in the agent. B) Normalized average whole-brain neural-glial response during passive and active transitions in a zebrafish subject.

3M-Progress Agents Saturate Explainable Variance of Whole-Brain Neural-Glial Dynamics To evaluate how closely intrinsic motivation-driven agents matched zebrafish neural-glial dynamics, we compared model predictions to whole-brain calcium imaging data recorded during behavioral transitions of futility-induced passivity. Neural-glial alignment was quantified using a One-to-One mapping, where each recorded biological neuron and astrocyte was matched to the single most correlated artificial unit from the model. Strikingly, 3M-Progress agents captured nearly all of the explainable variance in neural and astrocytic activity, markedly outperforming existing intrinsic motivation algorithms, as well as baseline controls (Figure 4A). Model-free intrinsic drive controls include energy cost (homeostatic), entropy bonus (MaxEnt), and a randomly intialized agent. Data-derived controls include a Gaussian Process fit to neural-glial responses, proportional-integral-derivative (PID) control, average neural-glial population response,

and white-noise. More details on these controls can be found in Appendix D. In fact, the 3M-Progress agent is the only model that is almost completely aligned with *both* the behavioral alignment and neural-glial alignment, highlighting that accurate behavioral modeling can tightly constrain detailed neural dynamics (Figure 4B). Taken together, 3M-Progress agents pass the NeuroAI Turing Test on this dataset, a criterion emphasizing models that match both behavior and internal function [65].

Latent Dynamics of 3M-Progress Agents Reflect Underlying Neural-Glial Computations Given that 3M-Progress best matched both behavioral and neural-glial alignment among all candidate models, we then characterized the agent's internal dynamics by performing Principal Component Analysis (PCA) on its *in silico* neural-glial population during behavioral state transitions. PCA revealed that the dominant latent dimensions of the agent's core module closely mirrors whole-brain neural-glial dynamics measured from biological zebrafish, whereby glial responses accumulate evidence of motor futility via noradrenergic signaling to drive behavioral suppression, and neural responses reflect transient activation patterns associated with detecting mismatches between expected and actual sensory outcomes during unsuccessful swim attempts. This analysis demonstrates that the 3M-Progress mechanism for intrinsic motivation not only generates realistic behavior, but also robustly captures fundamental internal neural-glial computations underlying autonomous exploration and behavioral state transitions (Figure 5).

6 Discussion

Our work seeks to identify and computationalize intrinsic goals that enable autonomous, task-independent behavior in animals. Leveraging a unique whole-brain dataset recorded in larval zebrafish during an autonomous behavior known as futility-induced passivity, we identify two simple principles of intrinsic goals for autonomous agents: avoid perseveration on stimuli that are uncontrollable and unpredictable, and converge on a robust decision-making strategy. We introduced 3M-Progress, a novel intrinsic drive that operationalizes these principles by continually tracking divergence between an online world model and an ethologically relevant prior. Learned from experience in an ecological niche that captures the basic physics of a naturalistic zebrafish habitat, this prior guides exploration in new environments by partitioning the behavioral space into niche-seeking and niche-avoiding modes.

Unlike prior intrinsic motivation methods that suffer from behavioral inconsistency and non-stationarity, 3M-Progress reliably generated stable cycling behaviors closely matching those observed in biological zebrafish. Moreover, 3M-Progress agents were uniquely successful in capturing whole-brain neural-glial activity among all candidate models. We showed that artificial agent's internal latent dynamics mirror neural-glial computation, wherein astrocytic responses accumulate evidence of motor futility through noradrenergic signaling to trigger behavioral suppression, while neural populations transiently encode mismatches between expected and observed sensory outcomes. To the best of our knowledge, our work marks the first goal-driven model of neural-glial computation, as well as the first completely self-supervised embodied agent that predicts behavioral and brain data.

Our findings suggest two complementary evolutionary constraints for developing robust, animal-like autonomous agents: (1) maintaining an intrinsic drive informed by memory, and (2) continually monitoring divergence between this memory and new sensory experiences. Functionally, tracking this mismatch within reinforcement learning frameworks allows artificial agents to identify ineffective strategies, update internal models adaptively, and discover new behaviors. Such mechanisms offer significant potential for enhancing the autonomy of artificial systems, especially in open-ended environments lacking clear external rewards or goals.

Limitations and Future Work Our current analyses primarily focused on behavioral transitions within constrained virtual environments, limiting ecological realism. Future work could extend both the artificial environments and biological experiments to richer ecological settings and more complex behavioral repertoires, providing stronger and more diverse constraints on computational models of autonomy. The biomechanical realism of the body could also be expanded to include muscles and motor circuits, providing realistic constraints on the low-level controller. Additionally, while our model captures essential neuron-glial interactions at a population level, it abstracts away detailed biochemical signaling mechanisms and anatomy of astrocytes and neurons. Incorporating more biologically detailed models of these processes could aid in providing predictions about these mechanisms at a finer scale. Finally, 3M-Progress can be generalized in a variety of ways that extend beyond futility-induced passivity (see Appendix C.3)—we leave this to future work.

Acknowledgements

We thank Misha Ahrens, Chris Doyle, and Yu Mu for helpful discussions, as well as the anonymous reviewers for their helpful feedback on the initial manuscript draft. A.N. was supported in part by a grant from the Burroughs Wellcome Fund. X.P. and R.K. were supported in part by the National Science Foundation and DoD OUSD (R & E) under Cooperative Agreement PHY-2229929 (The NSF AI Institute for Artificial and Natural Intelligence, ARNI) and the Simons Collaboration in Ecological Neuroscience (SFI-AN-NC-SCN-00007276-10).

References

[1] Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

[2] Yu Mu, Davis V Bennett, Mikail Rubinov, Sujatha Narayan, Chao-Tsung Yang, Masashi Tanimoto, Brett D Mensh, Loren L Looger, and Misha B Ahrens. Glia accumulate evidence that actions are futile and suppress unsuccessful behavior. *Cell*, 178(1):27–43, 2019.

[3] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pages 2778–2787. PMLR, 2017.

[4] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

[5] Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement. In *International conference on machine learning*, pages 5062–5071. PMLR, 2019.

[6] Kuno Kim, Megumi Sano, Julian De Freitas, Nick Haber, and Daniel Yamins. Active world model learning with progress curiosity. In *International conference on machine learning*, pages 5306–5315. PMLR, 2020.

[7] Jürgen Schmidhuber. Formal theory of creativity, fun, and intrinsic motivation (1990–2010). *IEEE transactions on autonomous mental development*, 2(3):230–247, 2010.

[8] Nick Haber, Damian Mrowca, Stephanie Wang, Li F Fei-Fei, and Daniel L Yamins. Learning to play with intrinsically-motivated, self-aware agents. *Advances in neural information processing systems*, 31, 2018.

[9] Jun Nagai, Xinzhu Yu, Thomas Papouin, Eunji Cheong, Marc R Freeman, Kelly R Monk, Michael H Hastings, Philip G Haydon, David Rowitch, Shai Shaham, et al. Behaviorally consequential astrocytic regulation of neural circuits. *Neuron*, 109(4):576–596, 2021.

[10] Laurie M Robin, José F Oliveira da Cruz, Valentin C Langlais, Mario Martin-Fernandez, Mathilde Metna-Laurent, Arnau Busquets-Garcia, Luigi Bellocchio, Edgar Soria-Gomez, Thomas Papouin, Marjorie Varilh, et al. Astroglial cb1 receptors determine synaptic d-serine availability to enable recognition memory. *Neuron*, 98(5):935–944, 2018.

[11] Michael R Williamson, Wookbong Kwon, Junsung Woo, Yeunjung Ko, Ehson Maleki, Kwanha Yu, Sanjana Murali, Debosmita Sardar, and Benjamin Deneen. Learning-associated astrocyte ensembles regulate memory recall. *Nature*, pages 1–9, 2024.

[12] Leo Kozachkov, Ksenia V Kastanenka, and Dmitry Krotov. Building transformers from neurons and astrocytes. *Proceedings of the National Academy of Sciences*, 120(34):e2219150120, 2023.

[13] Leo Kozachkov, Jean-Jacques Slotine, and Dmitry Krotov. Neuron–astrocyte associative memory. *Proceedings of the National Academy of Sciences*, 122(21):e2417788122, 2025. doi: 10.1073/pnas.2417788122. URL `https://www.pnas.org/doi/abs/10.1073/pnas.2417788122`.

[14] Ciaran Murphy-Royal, ShiNung Ching, and Thomas Papouin. A conceptual framework for astrocyte function. *Nature Neuroscience*, pages 1–9, 2023.

[15] Zhiguo Ma, Tobias Stork, Dwight E Bergles, and Marc R Freeman. Neuromodulators signal through astrocytes to alter neural circuit activity and behaviour. *Nature*, 539(7629):428–432, 2016.

[16] Jillian L Stobart, Kim David Ferrari, Matthew JP Barrett, Chaim Glück, Michael J Stobart, Marc Zuend, and Bruno Weber. Cortical circuit activity evokes rapid astrocyte calcium signals on a similar timescale to neurons. *Neuron*, 98(4):726–735, 2018.

[17] Adi Doron, Alon Rubin, Aviya Benmelech-Chovav, Netai Benaim, Tom Carmi, Ron Refaeli, Nechama Novick, Tirzah Kreisel, Yaniv Ziv, and Inbal Goshen. Hippocampal astrocytes encode reward location. *Nature*, 609(7928):772–778, 2022.

[18] Roberta De Ceglia, Ada Ledonne, David Gregory Litvin, Barbara Lykke Lind, Giovanni Carriero, Emanuele Claudio Latagliata, Erika Bindocci, Maria Amalia Di Castro, Iaroslav Savtchouk, Ilaria Vitali, et al. Specialized astrocytes mediate glutamatergic gliotransmission in the cns. *Nature*, 622(7981):120–129, 2023.

[19] Kyungchul Noh, Woo-Hyun Cho, Byung Hun Lee, Dong Wook Kim, Yoo Sung Kim, Keebum Park, Minkyu Hwang, Ellane Barcelon, Yoon Kyung Cho, C Justin Lee, et al. Cortical astrocytes modulate dominance behavior in male mice by regulating synaptic excitatory and inhibitory balance. *Nature Neuroscience*, 26(9):1541–1554, 2023.

[20] Peter Rupprecht, Sian N Duss, Denise Becker, Christopher M Lewis, Johannes Bohacek, and Fritjof Helmchen. Centripetal integration of past events in hippocampal astrocytes regulated by locus coeruleus. *Nature Neuroscience*, pages 1–13, 2024.

[21] Misha B Ahrens, Michael B Orger, Drew N Robson, Jennifer M Li, and Philipp J Keller. Whole-brain functional imaging at cellular resolution using light-sheet microscopy. *Nature methods*, 10(5):413–420, 2013.

[22] Joseph V Brady, Robert W Porter, Donald G Conrad, and John W Mason. Avoidance behavior and the development of duodenal ulcers. *Journal of the Experimental Analysis of Behavior*, 1 (1):69, 1958.

[23] Robert M Camp and John D Johnson. Repeated stressor exposure enhances contextual fear memory in a beta-adrenergic receptor-dependent process and increases impulsivity in a non-beta receptor-dependent fashion. *Physiology & behavior*, 150:64–68, 2015.

[24] Natalija Popović, Nicanor Morales-Delgado, Ernesto De la Cruz-Sánchez, and Miroljub Popović. Rats conserve passive avoidance retention level throughout the light phase of diurnal cycle. *Physiology & Behavior*, 268:114234, 2023.

[25] Cagla Eroglu. Astrocytes, hidden puppet masters of the brain. *Science*, 388(6748):705–706, 2025. doi: 10.1126/science.adx7102. URL `https://www.science.org/doi/abs/10.1126/science.adx7102`.

[26] Katheryn B. Lefton, Yifan Wu, Yanchao Dai, Takao Okuda, Yufen Zhang, Allen Yen, Gareth M. Rurak, Sarah Walsh, Rachel Manno, Bat-Erdene Myagmar, Joseph D. Dougherty, Vijay K. Samineni, Paul C. Simpson, and Thomas Papouin. Norepinephrine signals through astrocytes to modulate synapses. *Science*, 388(6748):776–783, 2025. doi: 10.1126/science.adq5480. URL `https://www.science.org/doi/abs/10.1126/science.adq5480`.

[27] Ksenia V Kastanenka, Rubén Moreno-Bote, Maurizio De Pittà, Gertrudis Perea, Abel Eraso-Pichot, Roser Masgrau, Kira E Poskanzer, and Elena Galea. A roadmap to integrate astrocytes into systems neuroscience. *Glia*, 68(1):5–26, 2020.

[28] Leo Kozachkov and Konstantinos P Michmizos. The causal role of astrocytes in slow-wave rhythmogenesis: A computational modelling study. *arXiv preprint arXiv:1702.03993*, 2017.

[29] Maurizio De Pittà, Mati Goldberg, Vladislav Volman, Hugues Berry, and Eshel Ben-Jacob. Glutamate regulation of calcium and ip3 oscillating and pulsating dynamics in astrocytes. *Journal of biological physics*, 35(4):383–411, 2009.

[30] Mati Goldberg, Maurizio De Pittà, Vladislav Volman, Hugues Berry, and Eshel Ben-Jacob. Nonlinear gap junctions enable long-distance propagation of pulsating calcium waves in astrocyte networks. *PLoS computational biology*, 6(8):e1000909, 2010.

[31] Vladislav Volman, Maxim Bazhenov, and Terrence J Sejnowski. Computational models of neuron-astrocyte interaction in epilepsy. *Frontiers in computational neuroscience*, 6:58, 2012.

[32] Leo Kozachkov and Konstantinos P Michmizos. Sequence learning in associative neuronal-astrocytic networks. In *Brain Informatics: 13th International Conference, BI 2020, Padua, Italy, September 19, 2020, Proceedings 13*, pages 349–360. Springer, 2020.

[33] Vladimir Ivanov and Konstantinos Michmizos. Increasing liquid state machine performance with edge-of-chaos dynamics organized by astrocyte-modulated plasticity. *Advances in neural information processing systems*, 34:25703–25719, 2021.

[34] Maurizio De Pittà and Nicolas Brunel. Multiple forms of working memory emerge from synapse–astrocyte interactions in a neuron–glia network model. *Proceedings of the National Academy of Sciences*, 119(43):e2207912119, 2022.

[35] Lulu Gong, Fabio Pasqualetti, Thomas Papouin, and ShiNung Ching. Astrocytes as a mechanism for meta-plasticity and contextually-guided network function. *arXiv preprint arXiv:2311.03508*, 2023.

[36] Haoran Sun, Yekun Chai, Shuohuan Wang, Yu Sun, Hua Wu, and Haifeng Wang. Curiosity-driven reinforcement learning from human feedback. *arXiv preprint arXiv:2501.11463*, 2025.

[37] Russell Mendonca, Oleh Rybkin, Kostas Daniilidis, Danijar Hafner, and Deepak Pathak. Discovering and achieving goals via world models. *Advances in Neural Information Processing Systems*, 34:24379–24391, 2021.

[38] Ramanan Sekar, Oleh Rybkin, Kostas Daniilidis, Pieter Abbeel, Danijar Hafner, and Deepak Pathak. Planning to explore via self-supervised world models. In *International conference on machine learning*, pages 8583–8592. PMLR, 2020.

[39] Josh Merel*, Diego Aldarondo*, Jesse Marshall*, Yuval Tassa, Greg Wayne, and Bence Ölveczky. Deep neuroethology of a virtual rodent. *International Conference on Learning Representations*, 2020.

[40] Diego Aldarondo, Josh Merel, Jesse D Marshall, Leonard Hasenclever, Ugne Klibaite, Amanda Gellis, Yuval Tassa, Greg Wayne, Matthew Botvinick, and Bence P Ölveczky. A virtual rodent predicts the structure of neural activity across behaviors. *Nature*, pages 1–3, 2024.

[41] Roman Vaxenburg, Igor Siwanowicz, Josh Merel, Alice A Robie, Carmen Morrow, Guido Novati, Zinovia Stefanidi, Gert-Jan Both, Gwyneth M Card, Michael B Reiser, et al. Whole-body simulation of realistic fruit fly locomotion with deep reinforcement learning. *bioRxiv*, pages 2024–03, 2024.

[42] Victor Lobato-Rios, Shravan Tata Ramalingasetty, Pembe Gizem Özdil, Jonathan Arreguit, Auke Jan Ijspeert, and Pavan Ramdya. Neuromechfly, a neuromechanical model of adult drosophila melanogaster. *Nature Methods*, 19(5):620–627, 2022.

[43] Sibo Wang-Chen, Victor Alfred Stimpfling, Thomas Ka Chung Lam, Pembe Gizem Özdil, Louise Genoud, Femke Hurtak, and Pavan Ramdya. Neuromechfly v2: simulating embodied sensorimotor control in adult drosophila. *Nature Methods*, 21(12):2353–2362, 2024.

[44] Gopal P Sarma, Chee Wai Lee, Tom Portegys, Vahid Ghayoomie, Travis Jacobs, Bradly Alicea, Matteo Cantarelli, Michael Currie, Richard C Gerkin, Shane Gingell, et al. Openworm: overview and recent advances in integrative biological simulation of caenorhabditis elegans. *Philosophical Transactions of the Royal Society B*, 373(1758):20170382, 2018.

[45] Anthony Zador, Sean Escola, Blake Richards, Bence Ölveczky, Yoshua Bengio, Kwabena Boahen, Matthew Botvinick, Dmitri Chklovskii, Anne Churchland, Claudia Clopath, et al. Catalyzing next-generation artificial intelligence through neuroai. *Nature communications*, 14 (1):1597, 2023.

[46] Adriana Perez Rotondo, Alessandro Marin Vargas, Michael Dimitriou, and Alexander Mathis. Modeling sensorimotor processing with physics-informed neural networks. *bioRxiv*, pages 2024–09, 2024.

[47] Olivier Codol, Jonathan A Michaels, Mehrdad Kashefi, J Andrew Pruszynski, and Paul L Gribble. Motornet, a python toolbox for controlling differentiable biomechanical effectors with artificial neural networks. *Elife*, 12:RP88591, 2024.

[48] Alessandro Marin Vargas, Axel Bisi, Alberto S Chiappa, Chris Versteeg, Lee E Miller, and Alexander Mathis. Task-driven neural network models predict neural dynamics of proprioception. *Cell*, 187(7):1745–1761, 2024.

[49] Xiangxiao Liu, Matthew D Loring, Luca Zunino, Kaitlyn E Fouke, François A Longchamp, Alexandre Bernardino, Auke J Ijspeert, and Eva A Naumann. Artificial embodied circuits uncover neural architectures of vertebrate visuomotor behaviors. *Science Robotics*, 10(107): eadv4408, 2025.

[50] Justin N Wood, Lalit Pandey, and Samantha MW Wood. Digital twin studies for reverse engineering the origins of visual intelligence. *Annual Review of Vision Science*, 10(1):145–170, 2024.

[51] Joshua D McGraw, Donsuk Lee, and Justin N Wood. Parallel development of social behavior in biological and artificial fish. *Nature Communications*, 15(1):10613, 2024.

[52] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.

[53] Marion F Haug, Oliver Biehlmaier, Kaspar P Mueller, and Stephan CF Neuhauss. Visual acuity in larval zebrafish: behavior and histology. *Frontiers in zoology*, 7:1–7, 2010.

[54] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

[55] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[56] Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

[57] Kevin Kermani Nejad, Paul Anastasiades, Loreen Hertäg, and Rui Ponte Costa. Self-supervised predictive learning accounts for cortical layer-specificity. *Nature Communications*, 16(1):6178, 2025.

[58] Nathaniel D Daw, Samuel J Gershman, Ben Seymour, Peter Dayan, and Raymond J Dolan. Model-based influences on humans' choices and striatal prediction errors. *Neuron*, 69(6): 1204–1215, 2011.

[59] Samuel J Gershman. The successor representation: its computational logic and neural substrates. *Journal of Neuroscience*, 38(33):7193–7200, 2018.

[60] Andreja Bubic, D Yves Von Cramon, and Ricarda I Schubotz. Prediction, cognition and the brain. *Frontiers in human neuroscience*, 4:1094, 2010.

[61] Lee De-Wit, Bart Machilsen, and Tom Putzeys. Predictive coding and the neural response to predictable stimuli. *Journal of Neuroscience*, 30(26):8702–8703, 2010.

[62] Aran Nayebi*, Nathan CL Kong*, Chengxu Zhuang, Justin L Gardner, Anthony M Norcia, and Daniel LK Yamins. Mouse visual cortex as a limited resource system that self-learns an ecologically-general representation. *PLOS Computational Biology*, 19, 2023.

[63] Aran Nayebi, Alexander Attinger, Malcolm Campbell, Kiah Hardcastle, Isabel Low, Caitlin Mallory, Gabriel Mel, Ben Sorscher, Alex Williams, Surya Ganguli, Lisa M Giocomo, and Daniel LK Yamins. Explaining heterogeneity in medial entorhinal cortex with task-driven neural networks. *Advances in Neural Information Processing Systems*, 34, 2021.

[64] Joshua Achiam and Shankar Sastry. Surprise-based intrinsic motivation for deep reinforcement learning. *arXiv preprint arXiv:1703.01732*, 2017.

[65] Jenelle Feather*, Meenakshi Khosla*, N Murty*, and Aran Nayebi*. Brain-model evaluations need the neuroai turing test. *arXiv preprint arXiv:2502.16238*, 2025.

[66] Saran Tunyasuvunakool, Alistair Muldal, Yotam Doron, Siqi Liu, Steven Bohez, Josh Merel, Tom Erez, Timothy Lillicrap, Nicolas Heess, and Yuval Tassa. dm_control: Software and tasks for continuous control. *Software Impacts*, 6:100022, 2020.

[67] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

[68] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[69] Aran Nayebi, Rishi Rajalingham, Mehrdad Jazayeri, and Guangyu Robert Yang. Neural foundations of mental simulation: Future prediction of latent representations on dynamic scenes. *Advances in Neural Information Processing Systems*, 36:70548–70561, 2023.

[70] Alex H Williams, Erin Kunz, Simon Kornblith, and Scott Linderman. Generalized shape metrics on neural representations. *Advances in Neural Information Processing Systems*, 34:4738–4750, 2021.

Appendix

A Model-Brain Alignment per Transition and per Module

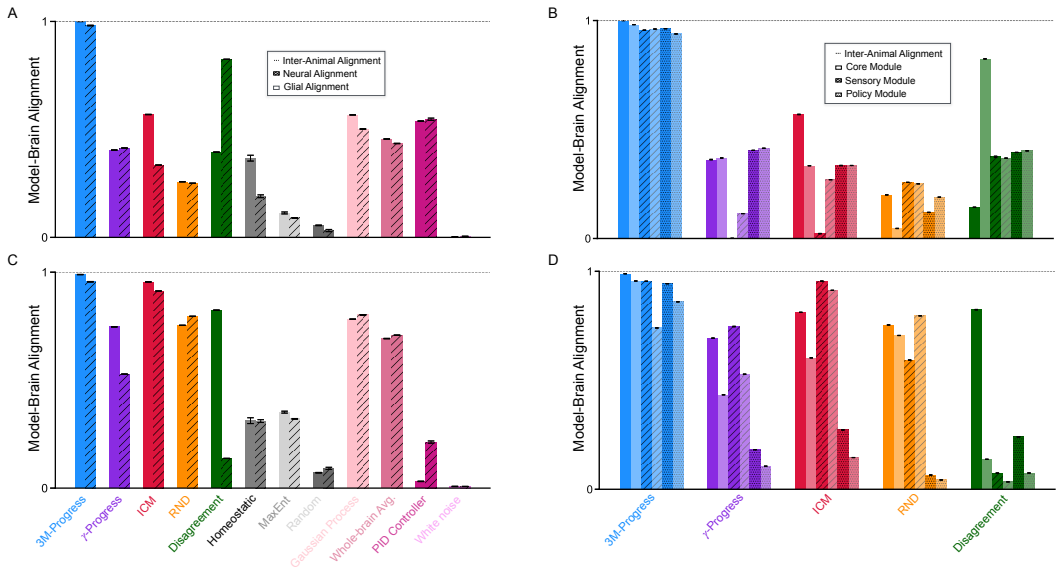

Figure 6: Model-Brain alignment for active and passive transitions and per module, excluding per module read-in and readout layers where applicable. A) Alignment for Active Transitions. B) Alignment per Agent Module for Active Transitions. C) Alignment for Passive Transitions. D) Alignment per Agent Network Module for Passive Transitions.

B Latent Dynamics of Baseline Agents

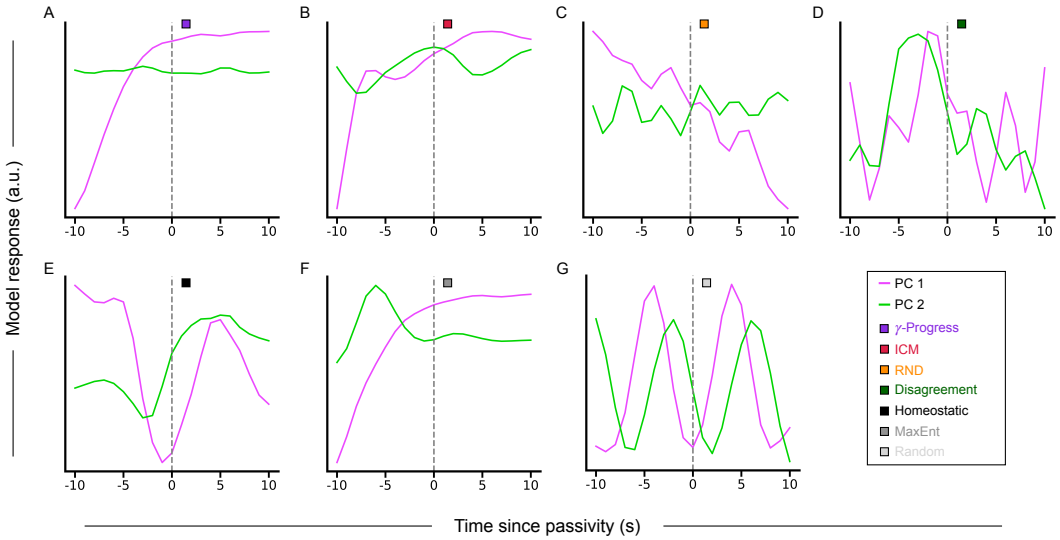

Figure 7: Latent dynamics of each agent-based control model. Dashed line indicates time since passivity. Coloring of PCs neural-glial cell-types (Fig. 2, Fig. 5) was chosen according to cell-types of the 3M-Progress agent PCs.

C Formal Intuitions

C.1 Curiosity-driven Exploration is Not Enough

Consider an MDP with discount $\gamma \in (0,1)$, occupancy $d_\pi(s,a) = (1-\gamma)\sum_{t\geq 0}\gamma^t \Pr_\pi(s_t{=}s, a_t{=}a)$, and transition kernel $\mathcal{T}(s' \mid s,a)$. Let a predictive world model with parameters θ be trained online to minimize

$$\mathcal{L}(\pi,\theta) \;=\; \mathbb{E}_{(s,a,s')\sim d_\pi \mathcal{T}}\big[\ell(s,a,s';\theta)\big],$$

and suppose the policy receives intrinsic reward determined by this predictor: $r^i(s,a,s';\theta) = g\big(\ell(s,a,s';\theta)\big)$ where $\ell \geq 0$ and $g:\mathbb{R}_{\geq 0}\to\mathbb{R}_{\geq 0}$ is monotone increasing with $g(0)=0$. The policy is updated to maximize

$$J(\pi,\theta) \;=\; \mathbb{E}_{(s,a,s')\sim d_\pi \mathcal{T}}\big[r^i(s,a,s';\theta)\big].$$

Under either alternating or simultaneous updates that decrease $\mathcal{L}$ in θ and increase J in π, the process cannot converge to a nontrivial, uniquely defined exploratory policy. The only stationary outcome is a degenerate collapse in which r^i is constant on the visited support; otherwise the policy-model coupling remains non-stationary and induces drift. That is, any stationary point $(\pi^\star,\theta^\star)$ satisfies one of the following cases:

1. *Reward collapse.* If $\theta^\star$ minimizes $\mathcal{L}(\pi^\star,\theta)$ on the support of $d_{\pi^\star}\mathcal{T}$ and the model class is realizable on that support, then $\ell(\cdot;\theta^\star)=0$ almost surely, hence $r^i(\cdot;\theta^\star)=0$ almost surely and $J(\pi,\theta^\star)=0$ for all π. The objective is flat and does not select a unique exploratory policy.
2. *No stationary policy.* If we freeze the model at $\theta^\star$, then $r^i(\cdot;\theta^\star)$ is a fixed reward. If residual error remains, then $\ell(\cdot;\theta^\star)$ (hence r^i) is not almost surely constant, so there exists a measurable set U on which the intrinsic advantage $A^{r^{\text{int}}}_{\pi^\star}(s,a) > 0$ for some actions. By the policy-gradient identity,

   $$\nabla J(\pi^\star,\theta^\star) \;=\; (1-\gamma)\,\mathbb{E}_{d_{\pi^\star}}\big[A_{\pi^\star}(s,a)\,\nabla \log \pi^\star(a \mid s)\big].$$

   For some state $s\in U$, suppose we increase $\pi^\star(a\mid s)$ slightly on an action with $A_{\pi^\star}(s,a) > 0$ and decrease it on other actions to preserve $\sum_a \pi^\star(a\mid s)=1$. This choice makes the inner product $\langle A_{\pi^\star}(s,\cdot), \nabla\log\pi^\star(\cdot\mid s)\rangle$ strictly positive on U, hence the expectation above is positive and the policy gradient is nonzero. Therefore $\pi^\star$ is not a local maximizer of the stationary objective. Any such occupancy shift then triggers predictor updates that reduce r^{int} on U, moving the high-reward region and preventing a nontrivial fixed point.

In both cases, these intrinsic signals do not converge to a stable, uniquely-defined exploratory policy. Either training drives the intrinsic signal to a constant (degenerate) value on the visited distribution, or residual heterogeneity in the reward keeps creating ascent directions that are then neutralized by predictor updates, preventing stabilization.

C.2 3M-Progress Partitions the Behavioral Space

Consider two reward-free MDPs, $\mathcal{M}_1 = (\mathcal{S},\mathcal{A},\mathcal{T}_1,p_0)$ and $\mathcal{M}_2 = (\mathcal{S},\mathcal{A},\mathcal{T}_2,p_0)$, that differ only in their transition densities. Further, suppose there exists a set $U\subset\mathcal{S}\times\mathcal{A}$ such that $\mathcal{T}_1(\cdot\mid s,a) = \mathcal{T}_2(\cdot\mid s,a)$ almost everywhere in s' for all $(s,a)\in U$. Let $p(\cdot\mid s,a)$ and $q(\cdot\mid s,a)$ be world models trained on data from $\mathcal{T}_1$ and $\mathcal{T}_2$, respectively, with pointwise consistency:

$$\mathcal{D}_{\mathrm{KL}}\left(\mathcal{T}_1(\cdot\mid s,a)\|p(\cdot\mid s,a)\right) \xrightarrow{p} 0, \quad \mathcal{D}_{\mathrm{KL}}\left(\mathcal{T}_2(\cdot\mid s,a)\|q(\cdot\mid s,a)\right) \xrightarrow{p} 0.$$

Assume for each (s,a), both kernels share the same support with no vanishing probabilities (i.e. $\exists c>0$ s.t. $\mathcal{T}_2(s'\mid s,a)\geq c$ whenever $\mathcal{T}_1(s'\mid s,a)>0$). Then it follows that for all $(s,a)\in U$,

$$\mathcal{D}_{\mathrm{KL}}\left(p(\cdot\mid s,a)\|q(\cdot\mid s,a)\right) \xrightarrow{p} 0,$$

and that for all $(s,a)\in U^c$,

$$\mathcal{D}_{\mathrm{KL}}\left(p(\cdot\mid s,a)\|q(\cdot\mid s,a)\right) \xrightarrow{p} \mathcal{D}_{\mathrm{KL}}(\mathcal{T}_1\|\mathcal{T}_2) > 0.$$

C.3 Beyond Futility-induced Passivity

Although we demonstrate 3M-Progress on a specific autonomous behavior known as futility-induced passivity, our algorithm applies to any exploration behavior in which a dynamics niche is reasonably specified. In our experiments, futility-induced passivity arises completely from the choice of the pretraining environment and the online learning environment; we choose these environments such that the transition dynamics between environments agree when the agent is passive, thus defining the ecological niche that guides exploration (see Appendix C.2: simply put, we choose environments such that there exists a subset U in which the transition dynamics locally agree). In general, the pretraining environment should encode a meaningful dynamics prior the agent can use for continual learning with environments whose physics vary systematically from the pretraining environment.

For example, suppose we pretrain an agent and world model on a foraging task (e.g. the virtual rodent environment in [66]). In a new environment that includes the opportunity for foraging and any of its constituent locomotor primitives (running, turning, jumping, etc.), the basic distribution-matching objective of 3M-Progress rewards the agent for trajectories whose dynamics are predictable under the pretrained world model (namely, foraging and constituent motor primitives):

$$r_t^i = f(\hat{\epsilon}_t - \epsilon_t); \quad \epsilon_t = \mathcal{D}_{\mathrm{KL}}(\omega_{\theta'} \| \omega_\theta); \quad \hat{\epsilon}_t = (1-\gamma)\hat{\epsilon}_t + \gamma\epsilon_t$$

where θ' is learned online and θ is pretrained. Niche-aware exploration is primarily mediated by the activation function f. When f is monotonic non-decreasing, such as a rectified linear unit, 3M-Progress is niche-seeking with the niche defined by the dynamics prior ω_θ. Conversely, if f is monotonic non-increasing, 3M-Progress is niche-avoiding and explores dynamics outside the prior. Non-monotonic functions allow some amount of symmetry between niche-seeking and niche-avoidance depending on the specific function shape.

The utility of this approach can also be appreciated when the pretraining stage involves a large diversity of dynamics, either from multiple environments and tasks or a single environment with multiple tasks. A world model with sufficient computational capacity that captures these diverse dynamics can be flexibly used in new environments to motivate exploration in a variety of ways. In the simplest case, for example, suppose we pretrain an ensemble of dynamics models $\{\omega_{\theta_j}\}_{j=1}^N$ on N separate environments and tasks. Maintaining independent temporal filters $\hat{\epsilon}_t^j$ for each prior, a deterministic intrinsic motivation can be defined as $r_t^i = \max\{f(\hat{\epsilon}_t^j - \epsilon_t^j)\}_{j=1}^N$. Alternatively, one can imagine various sampling schemes over the ensemble in order to drive specific exploration styles, such as ϵ-greedy or max-entropy. This extends 3M-Progress to niche-aware exploration over multiple niches, thereby allowing the flexibility of multiple modes of behavior in a single objective function. Each addition of a dynamics prior further partitions the behavioral space in the online environment, and allows the dynamics characteristic of each pretraining environment to be composed to form an exploration landscape of attractors (niche-seeking) or repellors (niche-avoidance) for online learning.

D Description of Control Models

D.1 Model-based Controls

The Intrinsic Curiosity Module (ICM) [3] defines intrinsic reward as the Shannon surprise of the forward model, $r_t^i := -\log \omega_\theta(\phi_t^I \mid \phi_t, \mathbf{a}_t)$. I denotes an augmented inverse feature space that is learned on-top of sensory embeddings from ϕ—ICM trains an additional embedding layer ϕ^I using an inverse dynamics model parameterized by θ_I by optimizing an MSE loss $L(\theta_I) = \mathbb{E}_{\pi_\theta} \left\| f(\theta_I; \phi_t^I, \phi_{t+1}^I) - \mathbf{a}_t \right\|_2^2$.

Random Network Distillation (RND) [4] defines a fixed random nonlinear projection of sensory features $g(\phi)$ and trains a predictor network $\hat{g}$ using an MSE loss $r_t^i := L(\theta_{\mathrm{RND}}) = \mathbb{E}_{\pi_\theta} \left\| g(\phi_t) - \hat{g}(\theta_{\mathrm{RND}}; \phi_t) \right\|_2^2$. With the distillation objective as the intrinsic reward, RND does not reinforce behaviors by scoring their predictability by a forward dynamics model as in ICM; instead, the random memory provides a simple exploration bonus for visiting novel states under the policy distribution.

Disagreement [5] learn an ensemble of world models $\{\omega_{\theta_j}\}_{j=1}^{N}$ and defines intrinsic reward as $r_i^t := \mathrm{Var}\left(\{\mu_{\theta_j} : j \in [N]\}\right)$ for N randomly initialized world models. Ensemble variances scores the stochasticity of the environment and reinforces state-action pairs for which models disagree.

γ-Progress [7] leverages the temporal history of Shannon surprise to define intrinsic reward using prediction gain, $r_i^t := \log \frac{\omega_{\theta_{new}}}{\omega_{\theta_{old}}}$, where θ_{new} parameterized a world model after learning on new transitions withheld from an lagging model θ_{old}.

D.2 Model-Free Controls

Homeostatic Agent One straightforward way to achieve a passive behavioral transition is to simply add an action-cost that outweighs any other positive reward signal. In the presence of a fixed or nonexistent extrinsic reward signal which the agent has no control over (such as in the open-loop protocol), an action-cost encourages the agent to become passive, corresponding to a metabolic constraint or homeostatic regulation of energy. We implement this cost as the magnitude of the force exerted by the agent's motors, $c(a_t) = \lambda \|\mathbf{a}_t\|_2$. In all our experiments, we set $\lambda = 1$.

Maximum Entropy Agent Maximum entropy RL is a general exploration strategy that provides a bonus reward proportional to the entropy of the current policy. That is, $r_t^i = \lambda \mathcal{H}\left[\pi(\mathbf{a}_t \mid \mathbf{s}_t)\right]$. In all our experiments, we set $\lambda = 1$.

Random Agent To test a random baseline model for embodied control, we use a randomly initialized agent with the same model architecture (described in section 3, Figure 2A, and in PPO implementation details below).

Whole-brain Average We use the average recorded neural-glial activity during a specified behavioral transition in one larval zebrafish to predict whole-brain neural-glial activity recorded from another larval zebrafish undergoing the same transition. The alignment under the metric described in section 3 is computed using the average response of each cell-type (neural and glial) from the source animal to predict the corresponding cell-type in the target animal. We use two subjects and report the total alignment as the sample-weighted average between scores from both source-target pairs.

Gaussian-Process We fit a separate Gaussian Process (GP) to each cell-type (neural and glial) for each subject, using a radial basis function (RBF) kernel and centering the prior mean at the average whole-brain response. Alignment under the metric described in section 3 is computed between the whole-brain data from the target cell-type from an individual subject and its corresponding GP as the source. We use two subjects and report the total alignment as the sample-weighted average between scores from both source-target pairs.

White-noise At each timestep t, the model's predicted next state is give by $x_{t+1} = x_t + \eta_t$, where $\eta_t \sim \mathcal{N}(0, 1)$ is white-noise. Because the metric that saturates inter-animal alignment is correlation-based, the mean and variance of this random walk are arbitrary.

PID Controller To implement the circuit-based word model proposed by Mu et al. [2] for the zebrafish brain's transition from active to passive states, we employ a threshold-based ("bang–bang") controller that switches the fish's swim power $P(t)$ between an active waveform $P_{\text{base}}(t)$ and complete cessation ($P(t) = 0$) once a cumulative "futility" signal exceeds a fixed GABAergic threshold. This controller can be interpreted as the high-gain limit of a saturated PID controller.

Perceived hydrostatic drift is modeled as a constant v_d. The motor plant converts swim power into a counter-drift locomotor velocity with gain g_{MS},

$$v_s(t) = v_d - g_{\text{MS}} P(t).$$

Visual mismatch is the product of forward stimulus velocity and ongoing motor drive but is rectified to ignore overshoot,

$$e(t) = \begin{cases} v_s(t)P(t), & \text{if } v_s(t) > 0, \\ 0, & \text{otherwise.} \end{cases}$$

A leaky integrator with time constant τ_F accumulates the mismatch (i.e., the futility):

$$\dot{F}(t) = -\lambda_F F(t) + k_F e(t),$$

where $\lambda_F = 1/\tau_F$ and k_F is a gain. A second leaky integrator converts sustained futility into an inhibitory drive,

$$\dot{G}(t) = -\lambda_G G(t) + k_G \max(0, F(t) - \theta_F),$$

$$P(t) = \begin{cases} P_{\text{base}}(t), & G(t) \le \theta_G, \\ 0, & G(t) > \theta_G. \end{cases}$$

To mimic experimental perturbations we set $g_{\text{MS}} = 0$ between $t_{\text{OL on}}$ and $t_{\text{OL off}}$, effectively clamping optic-flow feedback.

E Implementation Details

All code can be found in `https://github.com/neuroagents-lab/autonomous_zebrafish`.

Proximal Policy Optimization (PPO) In all our experiments, we train our agents with PPO using a clipped surrogate objective [55] with $\epsilon^{CLIP} = 0.2$ and normalized advantage function computed using Generalized Advantage Estimation (GAE) [67] with $\lambda^{GAE} = 0.95$. The policy network is an MLP with two hidden layers [128, 64] optimized by Adam [68] with learning rate $\alpha = 0.0003$ and gradients computed over 5 epochs of 1000-step trajectories with a 250-step batch size vectorized across 64 environments. This MLP parameterizes the policy as a 5-dimensional diagonal Gaussian distribution which is sampled to produce actions during training. For evaluation (the experiments in section 5), the policy is deterministic by taking actions as the mean of the distribution. Actions take the form of continuous real-valued torques on $[-1, 1]$. The intrinsic rewards are normalized by dividing by a running estimate of the standard deviation of the sum of discounted rewards with discount factor $\gamma = 0.99$, which then supervises the value network MLP with two hidden layers [128, 64] to estimate the expected discounted return using this same discount factor. Both policy and value networks compute on hidden states from separate LSTM blocks (Figure 2A) using a shared embedding from the sensory feature extractors. Image embeddings are obtained from 64x64 pixel observations passed through a three-layer ResNet resulting in outputs with a spatial resolution of 16x16 (closely matching the visual acuity of larval zebrafish at the final layer). Image embeddings were flattened and concatenated with proprioceptive features including joint positions and rotational velocities and their embeddings from a two-layer MLP with hidden size [64, 64].

Intrinsic Drive Module (IDM) The IDM architecture details vary depending on which intrinsic drive it implements, and is described on a case-by-case basis in the sections below. Here, we describe the commonalities between intrinsic drives, which include the optimizer, forward dynamics loss, general forward model architecture, and memory buffer parameters. Each forward dynamics model across intrinsic drives is implemented as a two-layer MLP with hidden sizes [512, 512], trained to predict the true observation one time-step into the future from the current observation using an MSE loss (where inference is done in feature-space), and optimized using Adam [68] with learning rate $\alpha = 0.001$. The IDM maintains a memory buffer of the last 100 observation embeddings from each environment in the vector and trains it's constituent networks on this buffer every 20 steps. The IDM is trained for $1e^5$ steps before the intrinsic rewards are observed by the agent.

3M-Progress The ethological memory is created (with the default configuration outlined above) by training a swimmer agent on a simple navigation task in a head-free version of the experimental protocol outlined in section 3. This environment is chosen to provide the forward model with a similar visual input space as the head-fixed version while maintaining the ethology of unconstrained swimming in which the agent experiences naturalistic fluid forces and positional displacement in response to swim commands. The task is implemented as shaped reward proportional to the distance of the agent from a target location that in front of the agent, such that the swim-to-target behavior results in stabilizing the constant backwards flow of the high-contrast grating. The episode is long

enough that optimizing this reward allows the agent to become passive was the target is reached. Together, our setup allows the agent to experience state-action-state triplets (current state, current action, and resulting state) associated with active and passive behaviors. This provides a close correspondence between sensory-motor coupling in our virtual ethological environment and the closed-loop experimental condition in Mu et al. [2], where larval zebrafish swim against the passive backwards flow of high-contrast gratings motivated by positional homeostasis. Although our agent is motivated to swim by a different signal than its biological twin (i.e., moving towards a target location rather than a homeostatic drive that resists displacement from an opposing current), the design of the virtual environment renders the sensory-motor stream experienced by the internal world model equivalent between scenarios, since in both cases optic flow is counteracted by forward swim motion. Because this sensory-motor stream and it's resulting world-model alone define the ethological memory, the discrepancy in the signal that drove behavior has no bearing on training the new policy and value network in the open-loop environment (Figures 1D, 2B).

In the open-loop environment, the agent randomly initializes a new world-model memory that is trained online with the default configuration outlined in the IDM section. The ethological world-model memory is loaded from earliest checkpoint where the agent achieved optimal swim-to-target behavior and it's weights are frozen. The model-memory-mismatch is computed as the MSE between the predictions from each memory, filtered by an exponential moving average with timescale $\gamma = 0.99$, and the difference filtered and unfiltered predictions are passed through an L_1 activation.

Random Network Distillation (RND) [4] For RND both target and predictor networks are implemented using the default configuration in the IDM section. The predictor network is trained to predict random feature projections from the target as described in section 3.

Intrinsic Curiosity Module (ICM) [3] In addition to a forward model implemented using the default configuration in the IDM section, the ICM implements an inverse dynamics model as an MLP with an identical architecture and optimization routine to train a one-layer MLP on top of sensory features. The forward and inverse networks are cotrained by minimizing a joint objective $\beta L_F + (1-\beta)L_I$, where L_F and L_I are the forward and inverse MSE loss functions, respectively. In our experiments, we set $\beta = 0.2$.

Disagreement [5] We use $N = 3$ randomly initialized independent forward models using the default configuration described in the IDM section. Disagreement is computed as the mean variance across feature dimensions.

γ-Progress [6] Using a randomly initialized forward model implemented using the default configuration in the IDM section, the trailing memory is created by copying these initial weights and updating them using an exponential moving average with timescale γ. In all our experiments, we use $\gamma = 0.99$.

F Inter-Subject Noise Correction Derivation

Herein we describe how the metric $\mathcal{M}$ should correct for noise if there is trial-to-trial variability. This is unified and adapted from Nayebi* et al. [62], Nayebi et al. [63, 69]. If you prefer to skip the derivation, for common choices of metric $\mathcal{M}$, such as Pearson correlation, RSA, and especially any metric that satisfies transitive closure [70], one will need to correct by the square root of the product of the mapping consistency and internal consistency of the units, in order to properly approximate the true value of $\mathcal{M}$ in the limit of infinite trials.

To make this correction explicit, suppose we have neural responses from two animals (or subjects) A and B. Let t_i^p be the vector of true responses (either at a given time bin or averaged across a set of time bins) of animal $p \in \mathcal{A} = \{\mathsf{A}, \mathsf{B}, \dots\}$ on stimulus set $i \in \{\mathsf{train}, \mathsf{test}\}$. Of course, we only receive noisy observations of t_i^p, so let $\mathrm{s}_{j,i}^p$ be the jth set of n trials of t_i^p. Finally, let $M(x;y)_i$ be the predictions of a mapping M (e.g., PLS, or any type of regression) when trained on input x to match output y and tested on stimulus set i. For example, $M\left(\mathrm{t}_{\mathsf{train}}^{\mathsf{A}}; \mathrm{t}_{\mathsf{train}}^{\mathsf{B}}\right)_{\mathsf{test}}$ is the prediction of mapping M on the test set stimuli trained to match the true neural responses of animal B given, as input, the true neural responses of animal A on the train set stimuli. Similarly, $M\left(\mathrm{s}_{1,\mathsf{train}}^{\mathsf{A}}; \mathrm{s}_{1,\mathsf{train}}^{\mathsf{B}}\right)_{\mathsf{test}}$ is the prediction of mapping M on the test set stimuli trained to match the trial-average of noisy sample 1

on the train set stimuli of animal B given, as input, the trial-average of noisy sample 1 on the train set stimuli of animal A. Then we have that:

$$\mathcal{M}_{\text{true}} := \left\langle \mathsf{Corr}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}}; \mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}}, \mathrm{t}^{\mathsf{B}}_{\mathsf{test}}\right)\right\rangle$$

$$\sim \widehat{\mathcal{M}}_{\text{est}} := \left\langle \frac{\overbrace{\mathsf{Corr}\big(M(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}}; \mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}})_{\mathsf{test}},\ \mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\big)}^{\text{predictivity}}}{\sqrt{\underbrace{\widetilde{\mathsf{Corr}}\big(M(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}}; \mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}})_{\mathsf{test}},\ M(\mathrm{s}^{\mathsf{A}}_{2,\mathsf{train}}; \mathrm{s}^{\mathsf{B}}_{2,\mathsf{train}})_{\mathsf{test}}\big)}_{\text{mapping consistency}} \times \underbrace{\widetilde{\mathsf{Corr}}\big(\mathrm{s}^{\mathsf{B}}_{1,\mathsf{test}},\ \mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\big)}_{\text{internal consistency}}}} \right\rangle, \tag{4}$$

where the average $\langle\cdot\rangle$ is taken over bootstrapped split-half trials and train-test splits, and $\mathsf{Corr}(\cdot,\cdot)$ denotes the Pearson correlation of the two quantities. $\widetilde{\mathsf{Corr}}(\cdot,\cdot)$ denotes the Spearman-Brown corrected value of the original quantity (since it is computed on split-halves of the trials, unlike the numerator, which is evaluated on the full trial set). The analogous correction for RSA holds, where the RDM/RSM of the responses is instead used for s, and M is the identity mapping, $M(x;\cdot)_{\mathsf{test}} \equiv x_{\mathsf{test}}$. When constructing $\widehat{\mathcal{M}}_{\text{est}}$ for model-brain mappings (rather than brain-brain mappings), we just replace A with the model responses, which are deterministic.

The above correction in (4) is fully implemented in the `brainmodel_utils` package (`https://github.com/neuroagents-lab/brainmodel_utils`), specifically in the `get_linregress_consistency` function. This function can be imported as follows:

```
from brainmodel_utils.metrics.consistency import get_linregress_consistency
```

The `r_xy_n_sb` value returned by this function corresponds to the ratio in (4). Refer to the README and the function docstring for usage details across a range of linearly regressed and non-regressed (e.g. RSA) metrics.

F.1 Single Subject Pair

Suppose we have neural responses from two animals (or subjects) A and B. Let t^p_i be the vector of true responses (either at a given time bin or averaged across a set of time bins) of animal $p \in \mathcal{A} = \{\mathsf{A}, \mathsf{B}, \dots\}$ on stimulus set $i \in \{\mathsf{train}, \mathsf{test}\}$. Of course, we only receive noisy observations of t^p_i, so let $\mathrm{s}^p_{j,i}$ be the jth set of n trials of t^p_i. Finally, let $M(x;y)_i$ be the predictions of a mapping M (e.g., PLS) when trained on input x to match output y and tested on stimulus set i. For example, $M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}}; \mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}}$ is the prediction of mapping M on the test set stimuli trained to match the true neural responses of animal B given, as input, the true neural responses of animal A on the train set stimuli. Similarly, $M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}}; \mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}}$ is the prediction of mapping M on the test set stimuli trained to match the trial-average of noisy sample 1 on the train set stimuli of animal B given, as input, the trial-average of noisy sample 1 on the train set stimuli of animal A.

With these definitions in hand, the inter-animal mapping consistency from animal A to animal B corresponds to the following "true" quantity to be estimated by $\widehat{\mathcal{M}}_{\text{est}}$ in the limit of infinite trials:

$$\mathcal{M}_{\text{true}} := \mathsf{Corr}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}}; \mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}}, \mathrm{t}^{\mathsf{B}}_{\mathsf{test}}\right), \tag{5}$$

where $\mathsf{Corr}(\cdot,\cdot)$ is the Pearson correlation across a stimulus set. In what follows, we will argue that Eq (5) can be approximated with the following ratio of measurable quantities, where we split in half and average the noisy trial observations, indexed by 1 and by 2:

$$\mathcal{M}_{\text{true}} := \mathsf{Corr}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}}; \mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}}, \mathrm{t}^{\mathsf{B}}_{\mathsf{test}}\right)$$

$$\sim \widehat{\mathcal{M}}_{\text{est}} := \frac{\mathsf{Corr}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}}; \mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}}, \mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)}{\sqrt{\mathsf{Corr}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}}; \mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}}, M\left(\mathrm{s}^{\mathsf{A}}_{2,\mathsf{train}}; \mathrm{s}^{\mathsf{B}}_{2,\mathsf{train}}\right)_{\mathsf{test}}\right) \times \mathsf{Corr}\left(\mathrm{s}^{\mathsf{B}}_{1,\mathsf{test}}, \mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)}}. \tag{6}$$

In words, the inter-animal consistency (i.e., the quantity on the left side of Eq (6)) corresponds to the predictivity of the mapping on the test set stimuli from animal A to animal B on two different (averaged) halves of noisy trials (i.e., the numerator on the right side of Eq (6)), corrected by the

square root of the mapping reliability on animal A's responses to the test set stimuli on two different halves of noisy trials multiplied by the internal consistency of animal B.

We justify the approximation in Eq (6) by gradually replacing the true quantities (t) by their measurable estimates (s), starting from the original quantity in Eq (5). First, we make the approximation that:

$$\mathsf{Corr}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}};\mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)\sim\mathsf{Corr}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}};\mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}},\mathrm{t}^{\mathsf{B}}_{\mathsf{test}}\right)\times\mathsf{Corr}\left(\mathrm{t}^{\mathsf{B}}_{\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right), \quad (7)$$

by the transitivity of very positive correlations. Namely, in scenarios where correlations are very close to 1, a form of transitivity holds, meaning if variable A is highly correlated with variable B, and variable B with variable C, then variable A is also highly correlated with variable C. This is the desired situation, as low or negative correlations indicate neurons that are not self-consistent. Moreover, calculating certain metrics in these cases can result in undefined values due to operations like taking the square root of a negative number. Assuming high correlations is reasonable, especially when the number of stimuli is large. Next, by transitivity and normality assumptions in the structure of the noisy estimates and since the number of trials (n) between the two sets is the same, we have that:

$$\begin{aligned}\mathsf{Corr}\left(\mathrm{s}^{\mathsf{B}}_{1,\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)&\sim\mathsf{Corr}\left(\mathrm{s}^{\mathsf{B}}_{1,\mathsf{test}},\mathrm{t}^{\mathsf{B}}_{\mathsf{test}}\right)\times\mathsf{Corr}\left(\mathrm{t}^{\mathsf{B}}_{\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)\\&\sim\mathsf{Corr}\left(\mathrm{t}^{\mathsf{B}}_{\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)^2. \quad (8)\end{aligned}$$

In words, Eq (8) states that the correlation between the average of two sets of noisy observations of n trials each is approximately the square of the correlation between the true value and average of one set of n noisy trials. Therefore, combining Eq (7) and Eq (8), it follows that:

$$\mathsf{Corr}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}};\mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}},\mathrm{t}^{\mathsf{B}}_{\mathsf{test}}\right)\sim\frac{\mathsf{Corr}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}};\mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)}{\sqrt{\mathsf{Corr}\left(\mathrm{s}^{\mathsf{B}}_{1,\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)}}. \quad (9)$$

From the right side of Eq (9), we can see that we have removed $\mathrm{t}^{\mathsf{B}}_{\mathsf{test}}$, but we still need to remove the $M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}};\mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}}$ term, as this term still contains unmeasurable (i.e., true) quantities. We apply the same two steps, described above, by analogy, though these approximations may not always be true (they are, however, true for Gaussian noise):

$$\begin{aligned}\mathsf{Corr}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)&\sim\mathsf{Corr}\left(\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}},M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}};\mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}}\right)\\&\quad\times\mathsf{Corr}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}};\mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}},M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}}\right)\\\mathsf{Corr}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}},M\left(\mathrm{s}^{\mathsf{A}}_{2,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{2,\mathsf{train}}\right)_{\mathsf{test}}\right)&\\\sim\mathsf{Corr}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}},M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}};\mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}}\right)^2,&\end{aligned}$$

which taken together implies the following:

$$\mathsf{Corr}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}};\mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)\sim\frac{\mathsf{Corr}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)}{\sqrt{\mathsf{Corr}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}},M\left(\mathrm{s}^{\mathsf{A}}_{2,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{2,\mathsf{train}}\right)_{\mathsf{test}}\right)}}. \quad (10)$$

Eq (9) and Eq (10) together imply the final estimated quantity given in Eq (6).

F.2 Multiple Subject Pairs

For multiple animals, we consider the average of the true quantity for each target in B in Eq (5) across source animals A in the ordered pair (A,B) of animals A and B:

$$\begin{aligned}\mathcal{M}_{\text{true}}&:=\left\langle\mathsf{Corr}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}};\mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}},\mathrm{t}^{\mathsf{B}}_{\mathsf{test}}\right)\right\rangle_{\mathsf{A}\in\mathcal{A}:(\mathsf{A},\mathsf{B})\in\mathcal{A}\times\mathcal{A}}\\\sim\widehat{\mathcal{M}}_{\text{est}}&:=\left\langle\frac{\mathsf{Corr}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)}{\sqrt{\widetilde{\mathsf{Corr}}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}},M\left(\mathrm{s}^{\mathsf{A}}_{2,\mathsf{train}};\mathrm{s}^{\mathsf{B}}_{2,\mathsf{train}}\right)_{\mathsf{test}}\right)\times\widetilde{\mathsf{Corr}}\left(\mathrm{s}^{\mathsf{B}}_{1,\mathsf{test}},\mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)}}\right\rangle_{\mathsf{A}\in\mathcal{A}:(\mathsf{A},\mathsf{B})\in\mathcal{A}\times\mathcal{A}}.\end{aligned}$$

We also bootstrap across trials, and have multiple train/test splits, in which case the average on the right hand side of the equation includes averages across these as well.

Note that each neuron in our analysis will have this single average value associated with it when *it* was a target animal (B), averaged over source animals/subsampled source neurons, bootstrapped trials, and train/test splits. This yields a vector of these average values, which we can take median and standard error of the mean (s.e.m.) over, as we do with standard explained variance metrics.

F.3 RSA

We can extend the above derivations to other commonly used metrics for comparing representations that involve correlation. Since $\mathsf{RSA}(x, y) \coloneqq \mathsf{Corr}(\mathsf{RDM}(x), \mathsf{RDM}(y))$, then the corresponding quantity in Eq (6) analogously (by transitivity of maximally positive correlations) becomes:

$$\mathcal{M}_{\text{true}} \coloneqq \left\langle \mathsf{RSA}\left(M\left(\mathrm{t}^{\mathsf{A}}_{\mathsf{train}}; \mathrm{t}^{\mathsf{B}}_{\mathsf{train}}\right)_{\mathsf{test}}, \mathrm{t}^{\mathsf{B}}_{\mathsf{test}}\right)\right\rangle_{\mathsf{A}\in\mathcal{A}:(\mathsf{A},\mathsf{B})\in\mathcal{A}\times\mathcal{A}}$$
$$\sim \widehat{\mathcal{M}}_{\text{est}} \coloneqq \left\langle \frac{\mathsf{RSA}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}}; \mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}}, \mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)}{\sqrt{\widetilde{\mathsf{RSA}}\left(M\left(\mathrm{s}^{\mathsf{A}}_{1,\mathsf{train}}; \mathrm{s}^{\mathsf{B}}_{1,\mathsf{train}}\right)_{\mathsf{test}}, M\left(\mathrm{s}^{\mathsf{A}}_{2,\mathsf{train}}; \mathrm{s}^{\mathsf{B}}_{2,\mathsf{train}}\right)_{\mathsf{test}}\right) \times \widetilde{\mathsf{RSA}}\left(\mathrm{s}^{\mathsf{B}}_{1,\mathsf{test}}, \mathrm{s}^{\mathsf{B}}_{2,\mathsf{test}}\right)}}\right\rangle_{\mathsf{A}\in\mathcal{A}:(\mathsf{A},\mathsf{B})\in\mathcal{A}\times\mathcal{A}} . \tag{11}$$

Note that in this case, each *animal* (rather than neuron) in our analysis will have this single average value associated with it when *it* was a target animal (B) (since RSA is computed over images and neurons), where the average is over source animals/subsampled source neurons, bootstrapped trials, and train/test splits. This yields a vector of these average values, which we can take median and s.e.m. over, across animals $\mathsf{B} \in \mathcal{A}$.

For RSA, we can use the identity mapping (since RSA is computed over neurons as well, the number of neurons between source and target animal can be different to compare them with the identity mapping). As parameters are not fit, we can choose $\mathsf{train} = \mathsf{test}$, so that Eq (11) becomes:

$$\mathcal{M}_{\text{true}} \coloneqq \left\langle \mathsf{RSA}\left(\mathrm{t}^{\mathsf{A}}, \mathrm{t}^{\mathsf{B}}\right)\right\rangle_{\mathsf{A}\in\mathcal{A}:(\mathsf{A},\mathsf{B})\in\mathcal{A}\times\mathcal{A}} \sim \widehat{\mathcal{M}}_{\text{est}} \coloneqq \left\langle \frac{\mathsf{RSA}\left(\mathrm{s}^{\mathsf{A}}_1, \mathrm{s}^{\mathsf{B}}_2\right)}{\sqrt{\widetilde{\mathsf{RSA}}\left(\mathrm{s}^{\mathsf{A}}_1, \mathrm{s}^{\mathsf{A}}_2\right) \times \widetilde{\mathsf{RSA}}\left(\mathrm{s}^{\mathsf{B}}_1, \mathrm{s}^{\mathsf{B}}_2\right)}}\right\rangle_{\mathsf{A}\in\mathcal{A}:(\mathsf{A},\mathsf{B})\in\mathcal{A}\times\mathcal{A}} . \tag{12}$$

F.4 Pooled Source Animal

Often times, we may not have enough neurons per animal to ensure that the estimated inter-animal consistency in our data closely matches the "true" inter-animal consistency. In order to address this issue, we holdout one animal at a time and compare it to the pseudo-population aggregated across units from the remaining animals, as opposed to computing the consistencies in a pairwise fashion. Thus, B is still the target heldout animal as in the pairwise case, but now the average over A is over a sole "pooled" source animal constructed from the pseudo-population of the remaining animals.

Pooling data across subjects to create larger pseudopopulations is a common practice [**?**], and helps researchers better isolate core representational principles that are conserved across individuals when data collection modalities limit the number of collected neurons per session.

F.5 Spearman-Brown Correction

The Spearman-Brown correction can be applied to each of the terms in the denominator individually, as they are each correlations of observations from half the trials of the *same* underlying process to itself (unlike the numerator). Namely,

$$\widetilde{\mathsf{Corr}}\left(X, Y\right) \coloneqq \frac{2\,\mathsf{Corr}\left(X, Y\right)}{1 + \mathsf{Corr}\left(X, Y\right)}.$$

Analogously, since $\mathsf{RSA}(X, Y) \coloneqq \mathsf{Corr}(\mathsf{RDM}(x), \mathsf{RDM}(y))$, then we define

$$\begin{aligned}\widetilde{\mathsf{RSA}}\,(X, Y) &\coloneqq \widetilde{\mathsf{Corr}}(\mathsf{RDM}(x), \mathsf{RDM}(y)) \\ &= \frac{2\,\mathsf{RSA}\,(X, Y)}{1 + \mathsf{RSA}\,(X, Y)}.\end{aligned}$$

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We confirm that the claims made in the abstract and introduction do not exaggerate or fabricate the scope of our contributions and results.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: We confirm that we include a paragraph on limitations in the conclusions section.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: We do not provide any theoretical results in our paper.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We fully detail the architectures, environment, and algorithms used, as well as the data used for model-brain alignment comparisons.

   Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We will provide full access to our code, as well as links to access the larval zebrafish data.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide a complete characterization of all the methods in our paper, with more details in the supplemental materials.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report error bars on all our plots when applicable.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We include details on the computational resources we used.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We strickly adhere to the NeurIPS code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Our paper addressed fundamental scientific questions, and is not directly aimed towards societal impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

Justification: Our paper does not have high risk for misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We make sure to properly credit everybody.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not provide any new assests.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: We do not conduct any crowdsourcing experiments.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: Our paper does not use participants.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: We do not use LLMs for the mentioned purposes.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.