Does Curiosity Shape Transferable Representations? Intrinsic Motivation as a Representation Learning Mechanism for Cross-Task Transfer in Procgen

Zhafira Fawnia* Harshit Aggarwal* Muhiim Ali* Eugenio Macias* William Yu*

Brown University

{zhafira_fawnia, harshit_aggarwal, muhiim_ali, eugenio_macias, william_yu}@brown.edu

Abstract

Intrinsic motivation methods such as ICM, RND, and CFN are widely used to encourage exploration in reinforcement learning, but their effect on *representation quality* and *cross-task transferability* remains poorly understood. We present a systematic empirical study of four intrinsic reward methods—ICM, RND, CFN, and NLL-Surprise—through a representational analysis of 507 trained checkpoints spanning 12 Procgen games, supplemented by additional cross-environment RND transfer sweeps, evaluating whether curiosity-driven exploration during source training produces representations that transfer better to related target tasks. We introduce a representational analysis framework comprising linear probes, effective rank, CKA similarity, and Grad-CAM saliency to characterise representation health independently of downstream task performance. Our findings show that intrinsic motivation's value is *conditional* on schedule, source–target pair, and baseline-representation quality rather than being uniformly beneficial: (1) ICM with a midtraining cutoff produces the strongest probe-based representation quality on the primary Maze→Heist pair, achieving 0.914 action probe accuracy versus 0.812 for vanilla PPO; (2) effective rank is a strong within-pair predictor of transferability ($r = 0.913$), but this correlation does not generalise across game pairs ($r = 0.093$); (3) the intrinsic reward weight η is the critical hyperparameter, with high values causing severe representation collapse; by contrast, the ICM forward/inverse balance β is comparatively insensitive in the moderate-η regime; (4) the cutoff mechanism—deactivating intrinsic rewards after early training—functions as a *rescue mechanism* for runs where the intrinsic signal has drifted: it improves ICM in the primary Maze→Heist representation comparison and produces the largest single-game rescue effect in the auxiliary 12-game Heist-probe analysis ($+0.211$ for Heist-source encoders), while also improving cross-environment RND transfer in the tested settings; (5) representation quality and downstream transfer performance are partially decoupled: ICM cutoff gives the strongest probe-based representation quality, while RND cutoff leads in zero-shot transfer and CFN has the highest fresh-head fine-tuning mean; (6) always-on intrinsic motivation hurts representation quality in the majority of 12 tested source games, with vanilla PPO producing the best representations in 8 of 12, and intrinsic motivation provides representational benefit primarily where the baseline is weakly specified (correlation between vanilla probe accuracy and intrinsic-method-vs-vanilla gap: $r = -0.41$ to -0.88); and (7) NLL-Surprise is uniquely fragile, collapsing representations even at moderate η. These results reframe intrinsic motivation as a *scheduled representation-learning intervention* whose value is conditional rather than universal: intrinsic rewards can help when introduced early and removed before they dominate, but always-on curiosity often degrades representation quality. We do not provide a calibrated decision rule for when to apply intrinsic motivation; the strongest predictor we identify is the negative correlation between baseline representation quality and intrinsic-method gap, suggesting intrinsic motivation is most useful when the baseline encoder is weakly specified.

1 Introduction

A central question in both reinforcement learning (RL) and cognitive science is what enables behaviour to *generalise*. Strong performance on one task does not imply flexible adaptation when goals, dynamics, or rewards change. Lake et al. [9] argue that human-like intelligence requires *building reusable internal representations of the world that support transfer across related settings*—not merely the ability to optimise reward in any one environment. By this standard, an agent that masters a single task is not yet intelligent in the sense Lake et al. intends; intelligence requires that the representations underlying that mastery generalise.

Curiosity-driven learning is relevant to this question because biological learners do not rely solely on external reward: they also seek novelty, surprise, and predictable structure [13]. Oudeyer's *intrinsic motivation* framework proposes that this drive is what produces the broad, reusable internal models that Lake et al.'s account of intelligence requires—intrinsically motivated experience generates richer training data than externally directed behaviour, and that richer experience is what produces transferable structure. The Intrinsic Curiosity Module (ICM) [14] captures this idea computationally by rewarding transitions that are hard to predict in a learned feature space.

This paper takes Oudeyer's claim as an empirical hypothesis and tests it. **Does intrinsic motivation during source training actually produce representations that satisfy Lake et al's reusability criterion—measurable as transfer to a re-**

*Equal contribution.

lated target task? We treat transfer not merely as an engineering benchmark, but as an operational test of whether intrinsic motivation produces the kind of reusable representation that cognitive theories require for intelligence. Our findings are mixed: intrinsic motivation, properly scheduled, can produce more transferable representations than vanilla PPO on specific source–target pairs (most clearly Maze→Heist and Plunder→Starpilot), but no single intrinsic-method schedule consistently outperforms vanilla across our 12-game survey. The IMPALA encoder architecture and PPO baseline appear to do most of the representation-learning work; intrinsic motivation operates as a *conditional refinement*, dependent on schedule, source-game match, and baseline-representation quality.

We conduct this study across 507 checkpoints spanning 12 Procgen source games, four intrinsic motivation methods (ICM, RND, CFN, NLL-Surprise), four intrinsic reward weights (η), cutoff versus always-on schedules, and multiple random seeds. Our representational analysis framework—linear probes, effective rank, CKA, and Grad-CAM saliency—measures transferability independently of downstream performance and without requiring a trained target-task agent. The picture that emerges is nuanced: **representation health** is a strong within-pair predictor of transfer success (r = 0.913 on Maze→Heist) but does not generalise as a universal cross-game diagnostic (r = 0.093 across 12 games); always-on intrinsic motivation *hurts* representations in 8 of 12 source games; and intrinsic motivation's representational benefit is concentrated where the baseline is weakly specified.

Our main contributions are:

1. **A representational analysis framework for transfer** that measures encoder transferability independently of trained target-task agents, applied across 12 Procgen games and four intrinsic motivation methods.
2. **A scoping result for representation-quality diagnostics.** Effective rank predicts transferability with strikingly high reliability *within* a fixed source–target pair ($r = 0.913$), but is essentially uncorrelated *across* game pairs ($r = 0.093$). We propose this distinction as a methodological standard: representation-quality metrics should be reported with the source–target scope under which they are validated.
3. **Identification of the cutoff mechanism as a rescue, not a universal improvement.** Always-on curiosity leaves vanilla PPO as the strongest representational baseline in 8 of 12 source games. The cutoff schedule can reduce this damage and is beneficial in the tested cross-environment RND settings, especially for Heist-source encoders; for ICM across 12 source games, however, cutoff matches always-on on average and helps mainly where always-on was already hurting.
4. **Baseline-quality dependence of intrinsic motivation.** Intrinsic motivation's value is strongly negatively correlated with vanilla's own representation quality ($r = -0.41$ to -0.88 across methods): it provides representational diversity when the baseline is weak, and noise when the baseline is already strong.
5. **Characterisation of the η sensitivity landscape**, showing a method-dependent transition from healthy to collapsed representations, with β (forward/inverse balance) comparatively insensitive in the moderate-η regime.

Scope. Throughout this paper, "transfer performance" refers to the comparison between intrinsic-motivation-trained and vanilla PPO source agents under fixed transfer protocols (zero-shot, fresh-head fine-tune, unfreeze-1 fine-tune). We do *not* systematically benchmark these against from-scratch target training, and preliminary evidence (Section 8, limitation 7) suggests that transfer can underperform from-scratch training on some pairs—for example, Miner from scratch reaches $\sim$ 100% win rate, while Maze→Miner transfer with 25M fine-tuning steps reaches only $\sim$ 11%. Our claims are therefore about the *relative* representational quality of intrinsic-motivation-trained encoders, not about whether transferring is better than not transferring. Establishing the latter is essential future work and a precondition for any practical application of these findings.

2 Related Work

Intrinsic motivation in RL. Intrinsic reward methods incentivise exploration by augmenting the extrinsic reward signal with a bonus derived from prediction error, novelty, or state visitation counts. ICM [14] uses forward-model prediction error in a learned feature space, with an inverse model to filter out action-irrelevant features. RND [3] measures the discrepancy between a fixed random target network and a trained predictor, providing a simpler novelty signal without dynamics modelling. CFN [10] estimates pseudocounts via coin-flip classification accuracy, offering a count-based alternative that is robust to the *noisy-TV problem* [4]—the failure mode in which curiosity-based methods become trapped maximising reward against stochastic but task-irrelevant transitions, since every layout in a procedurally generated environment looks novel to a forward model even when the layout difference is irrelevant to the task. NLL-Surprise [1] uses the negative log-likelihood of a Gaussian dynamics model, capturing aleatoric uncertainty through learned variance.

Exploration on Procgen. Several recent works study exploration on Procgen specifically. ExpGen [26] trains exploration-oriented policies that achieve strong within-game generalisation across procedurally generated levels. EDE [7] leverages epistemic uncertainty for exploration on Procgen and reports strong results on hard-exploration games. DEIR [24] proposes discriminative-model-based episodic intrinsic rewards and evaluates on Procgen. These works study Procgen exploration as a *single-game* generalisation problem (train and test on the same game, different levels), which is distinct from our *cross-game transfer* setting (train on one game, transfer to a structurally different game). To our knowledge, this is among the first systematic studies of cross-game transfer under intrinsic motivation in Procgen.

Coverage-based vs. understanding-based curiosity. A conceptual distinction in the curiosity literature is between methods that drive *coverage* (visit unfamiliar states; e.g. RND, CFN) and methods that drive *understanding* (predict dynam-

ics, naturally decay when learned; e.g. IMGEP / Learning Progress [13], world-model disagreement methods such as Plan2Explore [20]). All four methods we study (ICM, RND, CFN, NLL) sit on the coverage side of this taxonomy—even ICM's forward model in our setup generates a *novelty* signal, not a *learning-progress* signal. Whether understanding-based methods produce qualitatively different transfer behaviour is an open question; we report preliminary results from a complementary set of experiments on Chaser→Heist with three understanding-based methods in Appendix B.

Plasticity loss in deep RL. Recent work has documented that deep RL agents progressively lose representational capacity during training—a phenomenon variously called *plasticity loss* [11], *primacy bias* [12], or *dormant neurons* [21]. Proposed mechanisms include feature-rank collapse under repeated optimisation against narrow target distributions, and the accumulation of dead units (typically defined as units whose output magnitude or per-state standard deviation falls below a threshold). Our representational diagnostics (effective rank, dead-neuron fraction) directly instantiate this literature's metrics: a neuron is *dead* in our analysis if its standard deviation across the expert evaluation set falls below 10^{-5}, in line with Sokar et al. [21]. Our finding that always-on intrinsic reward exacerbates capacity loss in 8 of 12 source games is consistent with the prediction that uncontrolled training objectives accelerate plasticity loss in this setting.

Transfer learning in RL. Transfer learning seeks to leverage knowledge from a source task to accelerate learning or improve performance on a target task [23, 25]. In the Procgen benchmark [5], agents must generalise across procedurally generated levels that share visual style and action space but differ in layout and dynamics. Prior work has studied transfer via progressive neural networks [17], policy distillation [18], and representation learning [22]. However, the effect of the *exploration strategy* during source training on downstream transferability has received limited attention.

Representation analysis. Linear probing [2] is a standard diagnostic for representation quality: a linear classifier trained on frozen features measures how much task-relevant information is linearly accessible. Effective rank [16], computed from the entropy of normalised singular values, quantifies the dimensionality utilisation of a representation. Centered Kernel Alignment (CKA) [8] compares representational geometry across networks by measuring the similarity of their kernel matrices.

3 Methods

3.1 Agent Architecture

All agents use an IMPALA-style CNN encoder [6] followed by actor-critic heads. The encoder consists of three convolutional sequences, each containing a convolution, max-pooling, and two residual blocks with ReLU activations. The three sequences use channel widths $16, 32, 32$, reducing the spatial resolution from 64×64 to 8×8. The resulting $32 \cdot 8 \cdot 8 = 2048$ activations are flattened and projected to a 256-dimensional hidden representation $\mathbf{h} \in \mathbb{R}^{256}$, from which the actor head produces logits over 15 discrete actions and the critic head produces a scalar value estimate. Policy optimisation uses PPO [19] with standard hyperparameters: learning rate 5×10^{-4}, discount factor $\gamma = 0.999$, GAE parameter $\lambda = 0.95$, clip coefficient $\epsilon = 0.2$, entropy coefficient 0.01, and value function coefficient 0.5. We emphasise that "vanilla" PPO in our setting is *not exploration-free*: it uses entropy-regularised exploration via the 0.01 entropy coefficient. Our intrinsic-motivation methods are tested for whether they provide additional representational benefit *beyond* this baseline policy-gradient exploration, not whether they enable exploration where vanilla cannot.

3.2 Intrinsic Motivation Methods

All intrinsic reward methods share a unified interface: given a transition (s_t, a_t, s_{t+1}), each method computes a scalar intrinsic reward r_t^i. The total reward is

$$r_t = r_t^e + \eta \cdot \tilde{r}_t^i, \tag{1}$$

where r_t^e is the extrinsic reward, $\tilde{r}_t^i$ is the RMS-normalised intrinsic reward, and $\eta \in \{0.001, 0.005, 0.01, 0.05\}$ controls the intrinsic reward strength. RMS normalisation divides r_t^i by a running estimate of $\sqrt{\mathrm{Var}(r^i)}$, ensuring that the intrinsic signal remains at unit scale regardless of the method.

3.2.1 ICM (Intrinsic Curiosity Module)

ICM [14] learns a feature encoder ϕ, an inverse dynamics model g, and a forward dynamics model f. The intrinsic reward is the forward prediction error:

$$r_t^i = \|\hat{\phi}(s_{t+1}) - \phi(s_{t+1})\|_2^2, \tag{2}$$

where $\hat{\phi}(s_{t+1}) = f(\phi(s_t), a_t)$. The training loss is $\mathcal{L} = \beta\mathcal{L}_{\text{forward}} + (1-\beta)\mathcal{L}_{\text{inverse}}$, with $\beta = 0.2$. The inverse model encourages the ICM feature space to focus on action-relevant structure, while the forward model provides the novelty signal.

Here ϕ denotes the ICM module's internal feature encoder, which is separate from the PPO agent encoder whose 256-dimensional hidden representation is later probed. The ICM loss updates the curiosity module, while its effect on the PPO representation is indirect: the intrinsic reward changes the policy-gradient training signal received by the agent.

3.2.2 RND (Random Network Distillation)

RND [3] uses a fixed, randomly initialised target network f_{target} and a trainable predictor f_{pred}:

$$r_t^i = \|f_{\text{pred}}(s_t) - f_{\text{target}}(s_t)\|_2^2. \tag{3}$$

Novel states yield high prediction error because the predictor has not been trained on similar inputs. Unlike ICM, RND requires no dynamics model and depends only on observation novelty.

3.2.3 CFN (Coin Flip Networks)

CFN [10] estimates pseudocounts via $K = 32$ independent binary classifiers. For each state, random binary labels are

generated by projecting features through a fixed random matrix, and classifier accuracy serves as a proxy for visitation count:

$$\hat{N}(s) = \frac{\bar{a}^2}{(1-\bar{a})^2 + \varepsilon}, \quad r_t^i = \frac{1}{\sqrt{\hat{N}(s_t)}}, \tag{4}$$

where $\bar{a}$ is the mean classification accuracy across the K heads. Frequently visited states yield consistent classification, hence high pseudocounts and low intrinsic reward.

3.2.4 NLL-Surprise

NLL-Surprise [1] models transitions with a Gaussian dynamics model that outputs mean $\boldsymbol{\mu}$ and log-variance $\log \boldsymbol{\sigma}^2$ for the next-state features:

$$r_t^i = -\log p_\theta(\phi(s_{t+1}) \mid \phi(s_t), a_t) \tag{5}$$

$$= \tfrac{1}{2} \sum_d \left(\log \sigma_d^2 + \frac{(\phi_d - \mu_d)^2}{\sigma_d^2} \right) + C, \tag{6}$$

where C absorbs constants independent of θ. Unlike ICM's MSE-based error, NLL captures aleatoric uncertainty through the learned variance, which in principle allows the model to downweight inherently stochastic transitions.

3.3 Cutoff Mechanism

Procedurally generated environments pose a unique challenge for intrinsic motivation: because every episode presents novel level layouts, the curiosity signal never naturally decays. We introduce a **cutoff schedule** that deactivates all intrinsic rewards after a fixed number of training steps (10M of 25M total), creating an explore-then-exploit curriculum. After the cutoff, only extrinsic reward drives policy optimisation, allowing the agent to consolidate task-relevant features without the distraction of persistent novelty bonuses.

3.4 Environments

We use the Procgen benchmark [5] because its games share a common visual style (64×64 RGB) and discrete action space (15 actions), allowing cross-game transfer without modality or action-space mismatch. Its procedural generation also requires agents to learn level-invariant features rather than memorising fixed layouts, making it a natural testbed for representation transfer. The procedural generation does have one consequence specifically for intrinsic motivation: because every episode presents new layouts, the curiosity signal from forward-model error never naturally decays—motivating the cutoff mechanism described in Section 3.3.

Our primary transfer pair is **Maze** (source) → **Heist** (target): Maze emphasises navigation through corridors, while Heist requires key collection and door interaction, testing whether navigation-oriented representations generalise to object-interaction tasks. We additionally evaluate cross-environment transfer from Heist and Miner as source environments. To test the generality of our findings beyond a single game pair, we conduct a broad 12-game study spanning Bigfish, Chaser, Climber, Coinrun, Dodgeball, Heist, Jumper, Leaper, Maze, Ninja, Plunder, and Starpilot, probing each game's agents against its own expert. All experiments use `easy` distribution mode, 64 parallel environments, and 200 training levels, with the exception of the original cross-environment RND experiments (Maze/Heist/Miner source, Section 6.8) which use `hard` distribution mode. We therefore treat those hard-mode results as evidence about the *direction* of the cutoff effect within RND. To test whether this direction holds under the same distribution mode as the remaining experiments, we additionally run a separate easy-mode cross-environment RND sweep covering the same three source environments and a broader η range; the resulting transfer win rates are reported in Section 6.8.1.

Game-set sizes. The paper uses two distinct game-set sizes that readers should not conflate. The **12-game representational survey** (Section 6.7) trains agents on each of 12 source games and probes each against its *own* expert dataset; this is our breadth test for whether the Maze→Heist representational findings generalise. The **14 cross-game transfer pairs** reported in Appendix A are a separate downstream-performance benchmark in which a source-game encoder is evaluated under zero-shot and fine-tune protocols on a structurally different target game; the 14 pairs draw from the 12 source games together with a small number of additional target environments (Fruitbot, Miner, Bossfight). The two analyses use overlapping but distinct game subsets and answer complementary questions—representational health within a game vs. transfer behaviour across game pairs.

4 Evaluation Framework

4.1 Transfer Protocols

We evaluate transfer through three protocols:

1. **Zero-shot**: All weights frozen; evaluated on 2000 Heist episodes with deterministic actions (argmax over policy logits).
2. **Fresh-head finetune**: Encoder frozen; actor and critic heads reinitialised and trained on Heist for 5M steps with PPO.
3. **Unfreeze-1 finetune**: Encoder frozen; existing heads unfrozen with a critic-only warmup (10 iterations), then 5M steps on Heist.

Results are reported as mean reward with win rate and standard deviation across 3 seeds. Where available, we compare against a **from-scratch baseline**: vanilla PPO initialised randomly and trained directly on Heist for 5M steps.

4.2 Representational Analysis

To characterise representation quality independently of downstream task performance, we develop a suite of representational diagnostics applied to frozen source-task encoders:

Expert agent and dataset. The expert used for action and value probe targets is the Heist-trained vanilla PPO checkpoint used to construct the shared evaluation dataset; the checkpoint path is stored in the dataset metadata and reused when computing CKA-to-expert diagnostics. From this checkpoint we

collect a fixed evaluation dataset of **215 episodes (207,985 state–action pairs)** on held-out Heist levels (levels 500–699, easy mode, $\gamma = 0.999$), ensuring no level overlap with our 200-level training distribution. Each state is annotated with the expert argmax action (over 15 policy logits), expert value estimate (V-head output), full logits, reward-to-go, episode index, and step index. This frozen dataset serves as the shared reference against which all source-task encoders are probed. For the 12-game survey, we use the analogous held-out expert dataset for each game's own expert.

Linear probes. We train single-layer linear probes on frozen 256-dimensional hidden features extracted by the agent encoder. The action probe is a 15-way linear classifier trained to predict the expert agent's argmax action, and the value probe is a linear regressor trained to predict the expert's value estimate $V(s)$. Each probe is trained for 200 epochs with Adam (lr $= 10^{-3}$, weight decay 10^{-4}) using one random 80/20 train/test split and no cross-validation. The action probe reports held-out classification accuracy; the value probe reports

$$R^2 = 1 - \frac{\mathrm{MSE}}{\mathrm{Var}(y)}.$$

As a robustness diagnostic, the analysis pipeline also trains a one-hidden-layer MLP probe with hidden width 256 under the same optimiser settings. We report linear-probe results as the primary metric because they measure whether target-relevant information is directly linearly accessible from the frozen representation.

High action probe accuracy indicates that the encoder represents the probe states in a way that supports linear readout of the expert's action—a direct measure of feature transferability in the corresponding probe setting. High value R^2 indicates that expert state-value information is linearly accessible, reflecting whether the representation encodes task-relevant state quality for the corresponding probe task.

Effective rank. The encoder produces a 256-dimensional feature vector $\mathbf{h} \in \mathbb{R}^{256}$ for each observation, but not all dimensions necessarily carry useful information: some may always output zero (dead neurons), while others may be highly correlated (redundant). Effective rank quantifies how many dimensions are *actually used*.

Given the $N \times 256$ feature matrix X, we first mean-centre the features,

$$\widetilde{X} = X - \mathbf{1}\bar{x}^\top,$$

and compute singular values $\sigma_1 \geq \cdots \geq \sigma_{256}$ of $\widetilde{X}$. We normalise the singular values as

$$p_i = \frac{\sigma_i}{\sum_j \sigma_j},$$

and define entropy effective rank as

$$r_{\text{eff}}(X) = \exp(H(p)) = \exp\left(-\sum_i p_i \log p_i\right).$$

Thus $r_{\text{eff}} = 256$ corresponds to uniform use of all feature dimensions, while $r_{\text{eff}} \approx 1$ indicates collapse onto a single dominant direction. In our experiments, healthy representations (e.g. ICM cutoff) achieve $r_{\text{eff}} \approx 70$, while collapsed representations (e.g. NLL at high η) fall to $r_{\text{eff}} \approx 4$, with over 85% of neurons completely dead. We also report *dead neuron fraction*: the proportion of features whose per-state standard deviation falls below 10^{-5}, following Sokar et al. [21].

For variance-concentration diagnostics, we separately use the normalised PCA variance fractions

$$q_i = \frac{\sigma_i^2}{\sum_j \sigma_j^2},$$

so that k_{95} is the smallest k satisfying $\sum_{i=1}^{k} q_i \geq 0.95$.

CKA (Centered Kernel Alignment). For two feature matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{N \times D}$ computed from the same observations, linear CKA [8] measures representational similarity:

$$\mathrm{CKA}(\mathbf{X}, \mathbf{Y}) = \frac{\|\mathbf{Y}^\top \mathbf{X}\|_F^2}{\|\mathbf{X}^\top \mathbf{X}\|_F \cdot \|\mathbf{Y}^\top \mathbf{Y}\|_F}, \tag{7}$$

after mean-centering both matrices. CKA $= 1$ indicates identical representational geometry (up to linear transform); CKA $= 0$ indicates orthogonal representations. We compute pairwise CKA across all checkpoints to reveal method-family clustering and identify collapsed representations.

5 Experimental Setup

5.1 Training Configuration

Source-task agents are trained for 25M timesteps (64 parallel environments, 256-step rollouts). For the primary Maze→Heist, eta-sweep, and ICM-ablation experiments, agents are trained on Maze; for the cross-environment RND sweep, agents are trained on Maze, Heist, or Miner; and for the 12-game survey, each agent is trained on its own source game. Intrinsic reward methods use a separate optimiser (Adam, lr $= 10^{-3}$) with the same observation stream. Cutoff variants deactivate intrinsic rewards and freeze the intrinsic model after 10M steps.

5.2 Experiment Phases

We conduct three experiment phases totalling approximately 700 GPU-hours on NVIDIA RTX 3090 and A100 GPUs:

1. **Initial eta sweep** (single seed): 20 conditions—4 methods $\times$ 4 η values, plus cutoff variants (ICM, RND, NLL at $\eta = 0.01$) and a vanilla baseline—to identify candidate η ranges per method.
2. **NLL rerun** (single seed): Verification of NLL-Surprise results after observing extreme variance between runs.
3. **Multi-seed sweep** (3 seeds): 10 best conditions $\times$ 3 seeds = 30 runs to establish statistical reliability.

5.3 Representational Analysis Configuration

The representational analysis pipeline processes checkpoints against expert datasets, extracting 256-dimensional features for action and value probes, effective rank, dead neuron statistics, and pairwise CKA. Data sources span five experiment sets totalling 507 checkpoints:

- **Primary Maze→Heist sweep**: 9 method variants × 3 seeds, Maze → Heist, easy distribution mode, probed against a shared Heist expert (215 episodes, held-out levels 500+).
- **Eta sweep analysis set**: 4 methods × 4 η values × 3 seeds + cutoffs = 59 checkpoints, easy distribution mode, probed against Heist expert.
- **ICM ablation**: ICM β/η sweep with 2 seeds each, easy distribution mode, probed against Heist expert (114 checkpoints across $\beta \in \{0.2, 0.5, 0.8\}$ and $\eta \in \{0.005, 0.01, 0.05\}$, plus cutoff variants).
- **Cross-environment RND sweep**: RND from Maze/Heist/Miner → Heist (49 checkpoints, always-on and cutoff variants, including sequential training). This is the only *representational-analysis* experiment set using `hard` distribution mode, making cross-mode comparisons with the other sets approximate.
- **12-game survey**: 258 checkpoints across 12 Procgen games × {ICM, RND, CFN, vanilla} × $\eta \in \{0.005, 0.01\}$ × 3–5 seeds, easy distribution mode, each game probed against its own expert. This tests whether intrinsic motivation benefits generalise beyond the Maze→Heist pair.

The additional easy-mode cross-environment RND win-rate sweep in Section 6.8.1 is used for downstream transfer validation and is not included in the 507-checkpoint representational-analysis count.

6 Results

6.1 Representational Quality by Method

What "representational quality" means here. We operationalise representational quality through four complementary metrics, each probing a different aspect of whether the frozen encoder has learned something useful for the target task:

- **Action probe accuracy** (top panel, Figure 1) asks: *can a linear classifier read the expert's action choice directly from the features?* This measures whether task-relevant *behavioural* information is linearly accessible without any further adaptation. A value of 1.0 means the encoder perfectly separates states that require different actions; chance is $1/15 \approx 0.067$.
- **Value probe** R^2 (middle panel) asks: *can a linear regressor recover the expert's value estimate $V(s)$ from the features?* This measures whether the encoder captures *state quality*—distinguishing promising from unpromising states. $R^2 = 1$ indicates perfect recovery; $R^2 = 0$ indicates no information beyond the mean; negative R^2 indicates the features actively mislead the regressor.
- **Effective rank** (bottom panel) asks: *how many of the 256 feature dimensions carry independent information?* This measures the encoder's dimensional *capacity utilisation*. Low effective rank indicates representation collapse (redundant or dead dimensions); high effective rank indicates distributed, high-capacity features.
- **Dead neuron fraction** (Table 1) is the proportion of feature dimensions whose per-state standard deviation across the expert dataset falls below 10^{-5}. This is the direct measure of capacity collapse, complementary to effective rank—high effective rank with high dead-neuron fraction is impossible, but low effective rank can occur with intermediate dead-neuron levels when surviving units are highly correlated.

Table 1: Representational analysis: linear probe accuracy, value R^2, effective rank, and dead neuron fraction for frozen Maze encoders probed on Heist expert labels. Mean ± std across 3 seeds. Best per column **bolded**.

Method	Action Acc.	Value R^2	Eff. Rank	Dead %
Vanilla	.812 ± .125	.500 ± .214	53.9 ± 18.8	50.9 ± 16.6
ICM	.825 ± .039	.520 ± .035	62.7 ± 8.9	42.7 ± 10.8
ICM cutoff	**.914 ± .023**	**.623 ± .066**	**70.6 ± 2.9**	**33.7 ± 5.9**
RND	.828 ± .064	.570 ± .093	66.6 ± 7.3	41.0 ± 7.5
RND cutoff	.846 ± .087	.547 ± .145	59.7 ± 9.6	45.3 ± 8.8
CFN	.856 ± .055	.551 ± .044	57.4 ± 11.5	47.5 ± 10.8
CFN cutoff	.855 ± .086	.621 ± .120	59.7 ± 13.1	47.7 ± 15.2
NLL	.204 ± .078	.073 ± .103	4.1 ± 4.6	85.7 ± 13.2
NLL cutoff	.419 ± .324	−.494 ± .855	13.2 ± 12.0	66.5 ± 37.0

Together, these metrics distinguish several failure modes: an encoder can have high rank but low probe accuracy (high-dimensional but uninformative, as with CFN), or high probe accuracy but low rank (narrow but task-aligned), or both collapse together (NLL). A representation is "high quality" only when all four are simultaneously consistent: independent information (high rank, low dead fraction) organised around task-relevant structure (high action accuracy and value R^2).

The CFN counter-example makes this concrete. Consider CFN in the broader 12-game survey (Section 6.7): CFN maintains high effective rank (57–80 across games, often exceeding vanilla) yet achieves the worst action probe accuracy of any method tested (0.505 mean, vs. 0.771 for vanilla). CFN spreads its representational capacity across many dimensions, but those dimensions do not linearly encode the expert's action choices. This is the opposite failure mode from NLL: NLL collapses to a useless one-dimensional representation, while CFN maintains a high-dimensional *but unstructured* representation. Neither probe accuracy nor effective rank alone would flag both of these failures. Only the combination—high rank *paired with* high probe accuracy and high value R^2—corresponds to representations we observe to transfer well.

Figure 1 presents the primary method comparison from the multi-seed Maze → Heist experiments (3 seeds each). Table 1 provides exact statistics.

ICM with cutoff is strongest in the primary Maze→Heist comparison. ICM cutoff achieves the highest action probe accuracy (0.914 ± 0.023), value R^2 (0.623 ± 0.066), and effective rank (70.6 ± 2.9), with the lowest dead neuron fraction (33.7%). This represents a 12.6 percentage-point improvement in action accuracy over vanilla PPO (0.812 ± 0.125) with substantially reduced variance, indicating both better and more consistent representations.

All methods except NLL improve upon or match vanilla in this primary comparison. ICM, RND, and CFN (both always-on and cutoff variants) achieve action probe accuracies

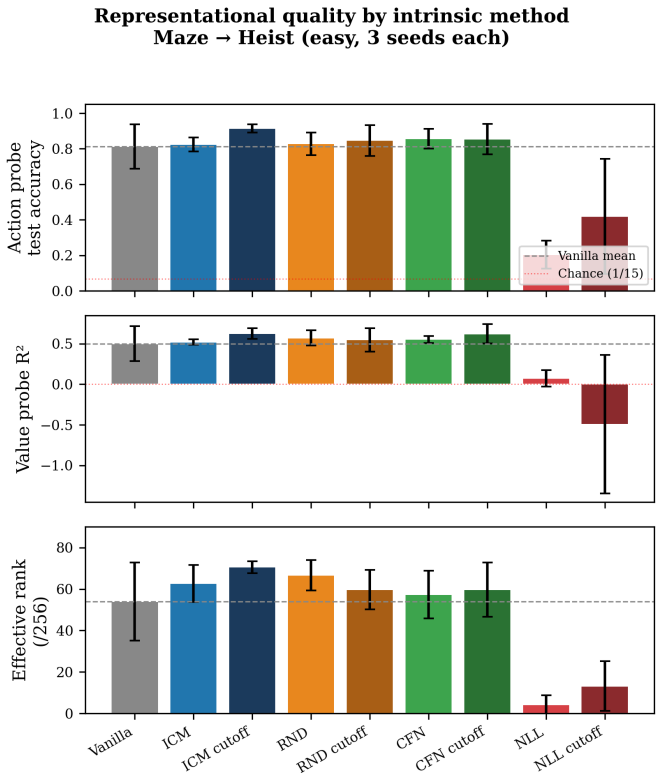

Figure 1: Representational quality by intrinsic method (Maze → Heist, 3 seeds). **Top**: Action probe test accuracy (15-class linear classifier on frozen features). **Middle**: Value probe R^2 (linear regression on expert $V(s)$). **Bottom**: Effective rank of the 256-dimensional representation. Dashed grey line: vanilla PPO baseline. ICM with cutoff achieves the strongest values across all three metrics in this comparison. NLL-Surprise shows severe representation collapse. Dead-neuron fraction (4th metric) is reported in Table 1 only; it is highly correlated with effective rank ($r = -0.854$, see Figure 2).

in the 0.825–0.856 range, all modestly above vanilla. The cutoff variants show no systematic advantage for RND and CFN at this η value.

NLL-Surprise causes severe representation collapse. NLL always-on achieves only 0.204 ± 0.078 action accuracy—barely above chance ($1/15 \approx 0.067$)—with an effective rank of 4.1 (out of 256) and 85.7% dead neurons. Even with cutoff, recovery is limited: 0.419 ± 0.324 accuracy with very high variance, indicating that the damage from NLL's training signal persists even after deactivation.

6.2 Effective Rank Predicts Transferability Within a Fixed Pair

Figure 2 shows the relationship between effective rank and action probe accuracy across all 160 checkpoints pooled from four experiment sets.

The correlation between effective rank and action probe accuracy is strong ($r = 0.913$), showing that effective rank tracks probe-based transferability within this fixed Maze→Heist setting. This relationship is near-linear across the full range of representation health, from collapsed NLL

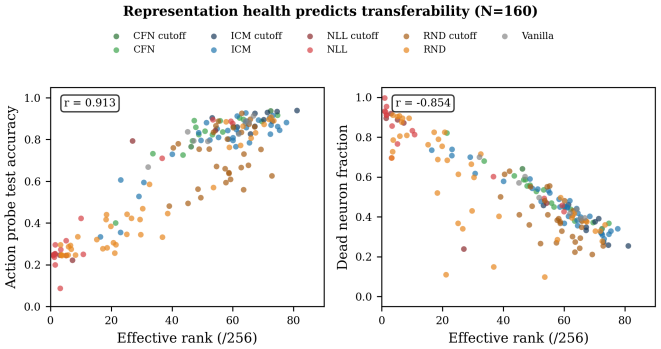

Figure 2: Representation health predicts transferability ($N = 160$ checkpoints). **Left**: Effective rank vs. action probe accuracy ($r = 0.913$). **Right**: Effective rank vs. dead neuron fraction ($r = -0.854$). The NLL-Surprise cluster (bottom-left/top-right) represents collapsed representations; healthy methods form a continuum in the upper-right/lower-left.

Table 2: Variance concentration statistics (Maze→Heist, 3 seeds). **Top-1**: variance fraction in the largest PC. **Top-10**: fraction in the first 10 PCs. k_{95}: dimensions needed for 95% variance. Healthy reps have low Top-1 and high k_{95}; the largest k_{95} (best capacity utilisation) is bolded.

Method	Top-1	Top-10	k_{95}
ICM cutoff	0.223	0.701	**40.3**
RND	0.247	0.715	38.7
ICM	0.176	0.713	36.3
CFN cutoff	0.177	0.722	35.3
RND cutoff	0.193	0.704	35.3
CFN	0.207	0.753	32.3
Vanilla	0.210	0.726	32.3
NLL cutoff	0.833	0.982	5.3
NLL	0.955	0.998	1.5

representations (rank < 10, accuracy < 0.2) to the healthiest ICM cutoff representations (rank > 70, accuracy > 0.9).

The negative correlation with dead neuron fraction ($r = -0.854$) confirms a consistent failure mode: representations with many dead neurons have low effective rank and poor transferability. This suggests that **effective rank can serve as a practical label-free diagnostic for representation collapse within a fixed or comparable transfer setting**. As shown in Section 6.7, however, it should not be interpreted as an absolute cross-game measure of transfer readiness.

6.3 Variance Concentration: A Complementary Collapse Diagnostic

Effective rank summarises the distribution of variance across all 256 dimensions, but examining the *top principal components* provides a sharper view of collapse severity. Table 2 reports three complementary statistics: top-1 variance fraction (share of total variance in the dominant dimension), top-10 variance fraction, and k_{95} (number of dimensions required to capture 95% of total variance).

NLL collapse is extreme. NLL-always-on concentrates 95.5% of total feature variance in a single dimension, with

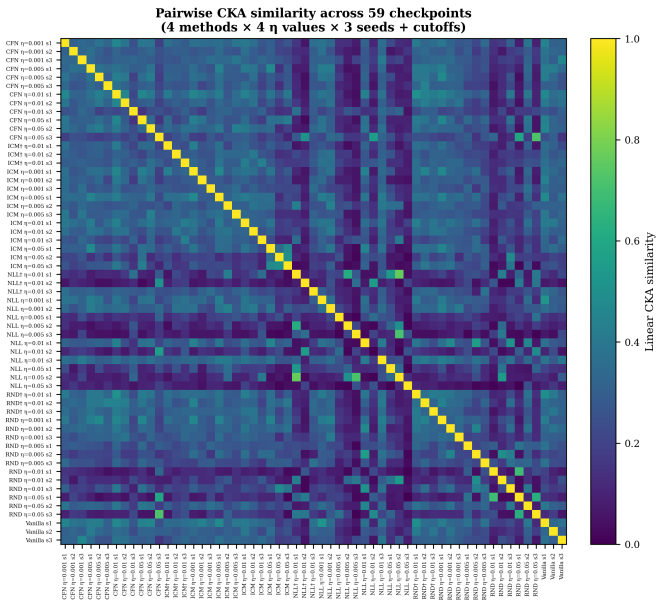

Figure 3: Pairwise CKA similarity across 59 checkpoints (4 methods $\times$ 4 η values $\times$ 3 seeds + cutoff variants + vanilla baselines). Block-diagonal structure indicates that same-method agents learn more similar representations regardless of η or seed. NLL rows/columns are notably dark, confirming that collapsed representations are structurally dissimilar to all healthy representations.

$k_{95} = 1.5$—the representation is essentially one-dimensional. NLL cutoff recovers modestly to $k_{95} = 5.3$, confirming partial but incomplete recovery.

Healthy methods distribute variance across $\sim$ 30–40 dimensions. All non-NLL methods have k_{95} in the range 32–40, indicating that roughly 30–40 independent feature directions capture most of the representational information. ICM cutoff leads with $k_{95} = 40.3$, consistent with its higher effective rank. In this setting, this suggests a rough collapse diagnostic: representations with $k_{95} < 10$ are associated with failed transfer probes, while $k_{95} > 30$ is typically observed among useful representations.

6.4 Representational Similarity Structure

Figure 3 shows the pairwise CKA similarity matrix across 59 checkpoints from the eta sweep.

Three structural patterns emerge:

Method-family clustering. Each intrinsic method (CFN, ICM, RND) forms a visible block of elevated CKA ($\sim$ 0.5–0.7) along the diagonal, indicating that same-method agents converge on similar representational geometry despite different η values and random seeds. This suggests that the training signal, not stochastic variation, is the primary determinant of representation structure.

NLL is a qualitative outlier. The NLL block shows near-zero CKA with all other methods, confirming that its collapsed representations occupy a fundamentally different region of representation space. NLL is not merely low-performing; it produces a qualitatively different representation that is structurally dissimilar to all healthy methods.

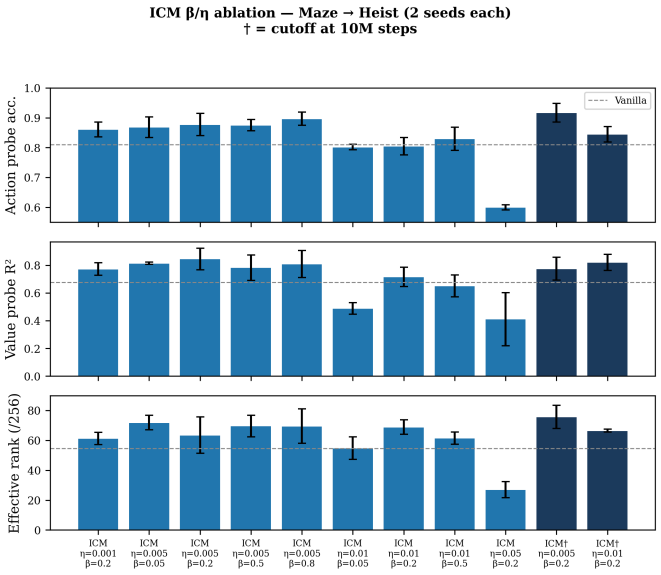

Figure 4: ICM β/η ablation (Maze $\rightarrow$ Heist, 2 seeds each). † indicates cutoff at 10M steps. Dashed line: vanilla baseline. β (forward/inverse balance) has minimal effect within the moderate-η regime, while $\eta = 0.05$ causes sharp degradation. ICM† with $\eta = 0.005$ achieves the best overall action accuracy ($\sim$ 0.93).

Cross-method similarity. ICM, RND, CFN, and vanilla share moderate CKA ($\sim$ 0.3–0.5), suggesting partial convergence on overlapping feature spaces. At low η, intrinsic methods produce representations close to vanilla, consistent with the intrinsic signal being a weak perturbation.

6.5 ICM Hyperparameter Sensitivity

Figure 4 presents the β/η ablation for ICM, isolating the effect of the forward/inverse loss balance and intrinsic reward weight.

η is the critical hyperparameter. Within the moderate range ($\eta \in [0.001, 0.005]$), all β values from 0.05 to 0.8 yield action accuracy in the 0.86–0.90 range, with value R^2 of 0.78–0.83 and effective rank of 60–72. This remarkable robustness to β indicates that the balance between forward and inverse model losses is not a sensitive design choice—a notable contrast with prior pretraining-regime work [e.g., 22] where the inverse-model contribution is critical, suggesting that the alongside-PPO online regime tolerates a wider β range than the frozen-encoder pretraining regime.

$\eta = 0.05$ causes collapse. At this value, action accuracy drops to $\sim$ 0.60, value R^2 falls to $\sim$ 0.40, and effective rank crashes to $\sim$ 27—well below vanilla. This is the "too much curiosity" failure mode: the intrinsic signal dominates extrinsic reward, driving the agent toward surprising but task-irrelevant transitions.

Cutoff rescues moderate-η runs. ICM cutoff at $\eta = 0.005$ achieves the best overall result ($\sim$ 0.93 action accuracy, $\sim$ 75 effective rank), exceeding both always-on ICM and vanilla by a substantial margin. The cutoff allows the agent to benefit from early curiosity-driven exploration while consolidating on task reward during later training.

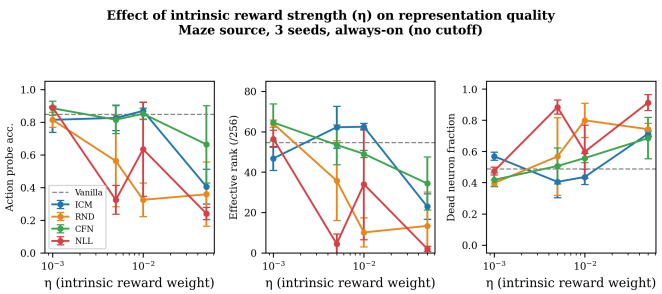

Figure 5: Effect of intrinsic reward strength (η) on representation quality (Maze source, 3 seeds, always-on). **Left**: Action probe accuracy. **Center**: Effective rank. **Right**: Dead neuron fraction. A sharp transition occurs around $\eta \in [0.01, 0.05]$: NLL and RND collapse while ICM remains relatively robust. Dashed line: vanilla baseline.

6.6 Effect of Intrinsic Reward Strength

Figure 5 characterises the η sensitivity landscape across all four methods using the multi-seed sweep data (3 seeds, always-on, no cutoff).

Sharp transition around $\eta \in [0.01, 0.05]$. All methods perform at or above vanilla at low η (≤ 0.005), but a clear divergence occurs at higher values. NLL shows severe collapse at $\eta = 0.05$ (near-zero accuracy, near-zero rank, $\sim 100\%$ dead neurons). RND also degrades substantially. CFN shows moderate sensitivity.

ICM is the most η-robust method in this sweep. ICM maintains reasonable representation quality even at $\eta = 0.05$. A plausible explanation is that ICM's intrinsic signal is action-conditioned and computed in a feature space shaped partly by inverse dynamics, making the resulting reward less prone to purely observation-novelty drift than RND or NLL. Because the ICM encoder is separate from the PPO encoder we later probe, this should be interpreted as an indirect training-signal effect rather than a direct architectural constraint on the probed representation.

The degradation is abrupt, not gradual. The transition from healthy to collapsed representations occurs over less than one order of magnitude in η, suggesting a sharp transition in optimisation behaviour as the intrinsic objective begins to dominate the extrinsic PPO signal rather than a smooth degradation.

6.7 Broad Game Survey: Always-On Curiosity Usually Hurts

To test whether our Maze→Heist findings generalise, we evaluate 258 always-on checkpoints across 12 Procgen games, each probed against its own expert. Table 3 summarises action probe accuracy by game and method.

Vanilla is the best method in 8 of 12 games. When intrinsic reward is left on for the full 25M steps, it degrades representations more often than it helps. The mean action accuracy across all games is highest for vanilla (0.771), followed by ICM (0.757), RND (0.636), and CFN (0.505).

CFN consistently hurts. CFN underperforms vanilla in all 12 games, with a mean deficit of -0.266. Despite maintaining high effective rank (often > 75), CFN produces representations with poor linear decodability, further confirming that high rank alone is not sufficient for transfer quality.

Table 3: Action probe accuracy across 12 Procgen games (mean over 3–5 seeds, always-on, $\eta \in \{0.005, 0.01\}$). Each game is probed against its own expert. Best method per game **bolded**. Vanilla is best in 8 of 12 games.

Game	Vanilla	ICM	RND	CFN
Plunder	**0.935**	0.905	0.639	0.582
Bigfish	**0.905**	0.889	0.733	0.563
Starpilot	**0.901**	0.868	0.827	0.517
Chaser	0.899	**0.914**	0.878	0.553
Climber	**0.890**	0.793	0.613	0.542
Dodgeball	0.865	**0.875**	0.821	0.567
Heist	**0.807**	0.660	0.345	0.535
Ninja	**0.732**	0.557	0.518	0.409
Jumper	**0.626**	0.550	0.468	0.382
Maze	0.599	**0.869**	0.537	0.556
Leaper	0.585	0.563	**0.805**	0.377
Coinrun	0.512	**0.636**	0.445	0.477
Mean	**0.771**	0.757	0.636	0.505

ICM helps selectively where vanilla representations are weakest. ICM's largest gains over vanilla occur on Maze ($+0.270$) and Coinrun ($+0.124$)—the two source games where vanilla itself produces the weakest in-game probe accuracy. On games where vanilla already achieves > 0.85 accuracy, ICM offers no improvement or slightly degrades performance. Quantitatively, the correlation between vanilla's action accuracy and ICM's advantage over vanilla is $r = -0.411$ across the 12 games, indicating that ICM helps more when the baseline representation is weaker. The same pattern holds even more strongly for CFN ($r = -0.878$) and RND ($r = -0.414$): intrinsic motivation adds useful representation diversity when the baseline representation is under-specified, but introduces noise when the baseline already captures the relevant features.

The effective rank correlation does not generalise across games. Within the Maze→Heist pair, effective rank predicts action accuracy with $r = 0.913$ (Section 6.2). Across all 258 checkpoints in the 12-game survey, this correlation drops to $r = 0.093$—essentially zero. This is because different games have different intrinsic difficulty for linear probing: a high-rank Coinrun representation (rank ~ 65) still achieves lower accuracy than a low-rank Plunder representation (rank ~ 80), because the tasks differ in complexity. Effective rank remains a reliable diagnostic *within* a fixed game pair, but it cannot be compared across game pairs as an absolute measure of transferability.

6.8 Cross-Environment Transfer and Cutoff as a Rescue Mechanism

These experiments use Procgen's `hard` distribution mode (see Section 3.4); cross-mode comparisons with the easy-mode results in Section 6.1–Section 6.7 are approximate.

Figure 6 evaluates cross-environment transfer using RND agents trained on Maze, Heist, and Miner, all probed against

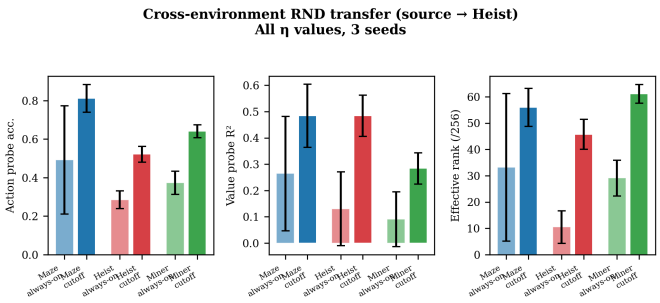

Figure 6: Cross-environment RND transfer (source → Heist). All η values, 3 seeds. Cutoff variants (dark bars) consistently outperform always-on (light bars) across all source environments. Maze with cutoff is the best source ($\sim$ 0.81 action accuracy); Miner with cutoff is also competitive ($\sim$ 0.63), demonstrating viable cross-environment transfer.

Table 4: RND cutoff effect across source environments (all → Heist). Mean across η values and 3 seeds. Cutoff at 10M of 25M steps.

Source	Variant	Action Acc.	Eff. Rank	Dead %
Maze	Always-on	0.493	33.2	59.0%
	Cutoff	**0.811**	**56.0**	**44.1%**
Heist	Always-on	0.285	10.5	87.4%
	Cutoff	**0.521**	**45.7**	**43.8%**
Miner	Always-on	0.373	29.1	37.6%
	Cutoff	**0.641**	**61.1**	**31.2%**

the Heist expert dataset. This experiment tests whether the cutoff benefit observed on Maze→Heist generalises across source environments.

Cutoff is essential across all source environments. Table 4 quantifies the cutoff effect. In every source environment, the cutoff variant approximately doubles action probe accuracy and effective rank while halving dead neuron fraction. This is the most consistent finding across all experiments: the persistent intrinsic signal degrades representations regardless of the source task.

Source environment matters. Maze with cutoff achieves the highest action accuracy (0.811) and the most task-relevant features for Heist. Miner with cutoff is competitive (0.641), suggesting that cross-environment transfer is viable when representations are healthy. Surprisingly, Heist (same-environment) with always-on RND performs worst (0.285), indicating that the intrinsic signal appears to drift toward less task-relevant representations even in same-task settings during extended training—the explore-then-exploit schedule is beneficial in this RND setting despite source-target similarity.

Connecting the 12-game and cross-environment findings. The 12-game survey (Section 6.7) shows that always-on intrinsic motivation hurts in most games. The cross-environment RND results show that cutoff rescues representation quality across all three tested source environments. Because this experiment uses `hard` distribution mode, we interpret this as directional evidence for the cutoff effect within RND rather than as a direct magnitude comparison with the

Table 5: Cross-environment RND transfer win rates to Heist target (`easy` mode, $\eta \in \{0.001, 0.01, 0.05\}$, 3 seeds, N=9 per cell). **ZS**: zero-shot win rate. **FH**: fresh-head fine-tune win rate. Cutoff at 10M of 25M steps. Cutoff outperforms always-on in every cell.

Source	Variant	ZS Win Rate	FH Win Rate
Maze	Always-on	2.2%	4.9%
	Cutoff	**2.6%**	**6.5%**
Heist	Always-on	2.8%	5.6%
	Cutoff	**7.9%**	**10.1%**
Miner	Always-on	1.3%	4.4%
	Cutoff	**2.8%**	**4.9%**

easy-mode Maze→Heist experiments. Together, these findings suggest that **always-on curiosity is a risky default**: intrinsic motivation should be treated as an early-training intervention, not a persistent objective.

6.8.1 Cross-Environment RND Transfer Win Rates: Easy Mode vs. Hard Mode

The preceding analysis (Table 4, Figure 6) characterises the cutoff effect through *representation quality* metrics—probe accuracy, effective rank, dead neuron fraction—measured on `hard`-mode checkpoints. To complement those diagnostics with actual downstream performance and verify that the directional finding holds at the same distribution mode as the rest of the paper, we ran a separate `easy`-mode cross-environment RND sweep ($\eta \in \{0.001, 0.01, 0.05\}$, 3 seeds per configuration, N=9 per cell after aggregating across η values) and measured **transfer win rates**: zero-shot (ZS) and fresh-head fine-tune (FH) performance on the target environment.

Table 5 reports the easy-mode win rates. The cutoff variant outperforms always-on in every cell under both protocols, replicating the directional finding from the hard-mode probe analysis with an independent sweep at a different distribution mode and a broader η range.

Largest easy-mode gain: Heist source. The ZS win rate for Heist source with cutoff (7.9%) is 2.8× higher than always-on (2.8%), and FH cutoff (10.1%) nearly doubles always-on (5.6%). This mirrors the hard-mode probe finding that Heist-source representations are the most sensitive to always-on intrinsic drift.

A descriptive hard-mode vs. easy-mode comparison is reported in Appendix C; because the sweeps differ in η range and distribution mode, we treat it only as configuration-sensitivity evidence.

6.9 ICM Cutoff Across 12 Games: A "Rescue" Mechanism

To test whether the cutoff benefit observed on Maze→Heist and in the cross-environment RND experiments generalises across games for ICM, we analyse 72 additional ICM cutoff checkpoints (12 games × 2 η values × 3 seeds) against the same Heist expert dataset. Table 6 compares ICM cutoff to both always-on ICM and vanilla across all 12 games.

Cutoff's benefit is game-specific, not universal. On av-

Table 6: ICM cutoff vs. always-on ICM vs. vanilla action probe accuracy for 12 source-game encoders evaluated on a fixed Heist expert probe dataset (3 seeds per η value, 2 η values pooled). Δ_{cut}: cutoff minus always-on. Cutoff wins in 5 of 12 games, with the largest gains on games where always-on ICM was already hurting.

Game	Vanilla	ICM	ICM†	Δ_{cut}
Heist	0.807	0.660	**0.871**	**+0.211**
Climber	0.890	0.793	0.838	+0.045
Ninja	0.732	0.557	0.595	+0.038
Maze	0.599	0.869	**0.898**	+0.029
Dodgeball	0.865	0.875	**0.884**	+0.010
Plunder	0.935	0.905	0.905	0.000
Jumper	0.626	0.550	0.539	−0.011
Bigfish	0.905	0.889	0.854	−0.035
Leaper	0.585	0.563	0.515	−0.048
Chaser	0.899	0.914	0.862	−0.052
Starpilot	0.901	0.868	0.803	−0.065
Coinrun	0.512	0.636	0.512	−0.124
Mean	**0.771**	0.757	0.756	−0.001

erage across 12 games, ICM cutoff performs essentially identically to always-on ICM (0.756 vs. 0.757). This qualifies the cross-environment RND pattern in Table 4: for ICM in this 12-game Heist-probe analysis, cutoff is a wash on average. However, the per-game variation is large and systematic.

Cutoff is a "rescue" mechanism. Ranking games by always-on ICM's deficit relative to vanilla reveals the pattern clearly. On games where always-on ICM was already hurting representations (Heist: -0.147, Ninja: -0.174, Climber: -0.097), cutoff rescues most of the lost performance. The Heist result is especially clear: always-on ICM reaches only 0.660 action accuracy—well below vanilla's 0.807—but ICM cutoff recovers to 0.871, exceeding vanilla by $+0.064$ and improving effective rank by $+14.3$. Conversely, on games where always-on ICM already produced healthy representations (Plunder, Bigfish, Starpilot, Chaser), cutoff produces small but consistent degradations, suggesting that cutting off a productive intrinsic signal mid-training wastes the diversity benefit.

Combined ICM story across 12 games. Using a best-of-ICM-variants metric (max over always-on and cutoff per game), some form of ICM matches or exceeds vanilla in 7 of 12 games—but no single ICM schedule dominates: always-on alone beats vanilla in only 4 of 12. The right intrinsic schedule is game-dependent, and the right schedule for a given source game requires per-game tuning. In this auxiliary Heist-probe analysis, cutoff behaves as a rescue mechanism for ICM rather than as a universal improvement.

Implication for Maze→Heist. The $+0.211$ Heist-source cutoff gain is the largest single-game rescue effect in this 12-game Heist-probe analysis. It supports the broader mechanism that cutoff can prevent always-on ICM from drifting away from task-relevant structure. This is consistent with, but does not by itself explain, the primary Maze→Heist result: the $+0.211$ value belongs to this auxiliary 12-game analysis (all 12 source-game encoders evaluated against a fixed Heist expert dataset), rather than arising directly from the Maze-source primary experiment reported in Table 1.

Table 7: Maze → Heist transfer performance across 3 seeds (mean ± std). **ZS**: zero-shot win rate. **FH**: fresh-head finetune win rate. **U1**: unfreeze-1 finetune win rate. Best per column **bolded**. †: cutoff at 10M of 25M steps.

Condition	ZS	FH	U1
Vanilla	$12.8 \pm 2.6\%$	$35.2 \pm 5.9\%$	$12.5 \pm 6.1\%$
ICM η=0.005	$7.3 \pm 1.0\%$	$33.3 \pm 9.5\%$	$\mathbf{27.3 \pm 15.5}\%$
ICM† η=0.005	$7.3 \pm 1.2\%$	$31.7 \pm 10.1\%$	$19.0 \pm 8.8\%$
RND η=0.001	$7.7 \pm 2.9\%$	$33.8 \pm 6.6\%$	$17.0 \pm 1.8\%$
RND† η=0.001	$\mathbf{17.5 \pm 2.6}\%$	$31.7 \pm 12.0\%$	$23.0 \pm 10.0\%$
CFN η=0.005	$14.7 \pm 4.7\%$	$\mathbf{37.3 \pm 16.8}\%$	$21.5 \pm 5.6\%$
CFN† η=0.005	$8.5 \pm 5.3\%$	$26.8 \pm 1.9\%$	$18.7 \pm 9.5\%$
NLL η=0.001	$8.8 \pm 9.3\%$	$13.0 \pm 15.3\%$	$7.7 \pm 8.3\%$

6.10 Transfer Performance

Table 7 presents the multi-seed transfer results (zero-shot, fresh-head finetune, unfreeze-1 finetune) for the best-performing conditions.

RND cutoff is strongest zero-shot method in this comparison. RND† at $\eta = 0.001$ achieves $17.5 \pm 2.6\%$ zero-shot win rate—37% above vanilla (12.8%) with equally low variance. The cutoff is important in this comparison: always-on RND at the same η achieves only 7.7%.

CFN 0.005 has the highest fresh-head finetune mean (37.3%), but also the highest variance among the reported fresh-head results (std = 16.8%), driven by one strong seed.

Fresh-head finetune equalises methods. When a new policy head is trained on Heist, the advantage of intrinsic motivation largely disappears: vanilla (35.2%) is competitive with all methods. This suggests that the IMPALA encoder architecture itself—not the exploration signal—is the primary driver of finetune transfer quality.

Unfreeze-1 favours ICM. When existing heads are unfrozen, ICM (27.3%) substantially outperforms vanilla (12.5%), suggesting that ICM's source policy heads encode more transferable action structure than vanilla's.

Representation quality and transfer performance are partially decoupled. ICM cutoff produces the best *representations* (highest probe accuracy and rank) but not the best *transfer performance* (RND cutoff leads zero-shot; CFN leads finetune). This suggests that linear probe accuracy is an informative but incomplete proxy for transfer: downstream performance also depends on policy-head compatibility, action-distribution alignment, and the specific adaptation protocol.

6.11 Seed Stability and Reliability

Beyond mean performance, practical transfer requires representations that are *reliable* across random seeds. Table 8 reports the coefficient of variation (CV = std/mean) for action probe accuracy across 3 seeds on Maze→Heist.

ICM cutoff is the most stable method in this comparison,

Table 8: Seed-to-seed stability of action probe accuracy (Maze→Heist, 3 seeds). **CV**: coefficient of variation (std/mean, lower = more stable). Sorted by stability.

Method	Mean	Std	CV
ICM cutoff	0.914	0.023	**2.5%**
ICM	0.825	0.039	4.7%
CFN	0.856	0.055	6.4%
RND	0.828	0.064	7.7%
CFN cutoff	0.855	0.086	10.1%
RND cutoff	0.846	0.087	10.3%
Vanilla	0.812	0.125	15.4%
NLL	0.204	0.078	38.2%
NLL cutoff	0.419	0.324	77.3%

not just the best on average. Its CV of 2.5% is much lower than vanilla PPO's 15.4%, meaning the gap between best and worst seed is substantially smaller. In these runs, ICM cutoff produces high-quality representations more consistently; vanilla PPO occasionally produces a mediocre one (the 0.125 std spans roughly 0.69–0.94 accuracy across seeds).

NLL remains highly seed-sensitive. NLL cutoff's 77.3% CV reflects seed-to-seed variability ranging from near-chance (0.095) to modest (0.743) performance—individual NLL checkpoints should therefore be interpreted cautiously without multiple seeds.

7 Additional Representation and Transfer Diagnostics

To complement the quantitative results in Section 6, we examine the geometry and spatial selectivity of the learned encoder representations. Rather than introducing a separate evaluation protocol, this section uses the same frozen IMPALA encoder features and expert-probe datasets described in Section 4.2 and Section 5, together with Grad-CAM and bottleneck-feature PCA visualizations. These analyses are intended as qualitative diagnostics: they help explain when high probe accuracy corresponds to spatially grounded features and when apparent representation quality may instead reflect non-spatial or collapsed structure.

7.1 Cross-Environment Probe Visualization

Unlike the 12-game survey in Section 6.7, where each game is probed against its own expert, this visualization uses a fixed Heist expert probe dataset for all source-game encoders. Thus, Figure 7 should be interpreted as a cross-environment Heist-probe diagnostic rather than as a within-game representation benchmark.

Figure 7 reports linear probe accuracy, value R^2, and effective rank for every (source game, method) cell. Three patterns emerge.

Vanilla PPO is a strong representational baseline. Across 12 games, vanilla encoders reach 0.80–0.94 action probe accuracy, exceeding always-on CFN (0.46–0.85), RND (0.46–0.91), and ICM (0.51–0.93) on most environments. Effective rank tells the same story: vanilla averages 53/256, while CFN, RND, and always-on ICM collapse to 17–38 on navigation games.

ICM cutoff is competitive with vanilla in this Heist-probe visualization. ICM cutoff is strongest on Maze, where it substantially exceeds vanilla in this fixed Heist-probe diagnostic (0.90 vs. 0.60), and remains close to vanilla on Plunder (0.91 vs. 0.94). On Bigfish, always-on ICM is closer to vanilla (0.89 vs. 0.91), while cutoff falls slightly lower (0.85). We treat this as qualitative support rather than a separate aggregate benchmark—consistent with Section 6.9, cutoff appears most useful when always-on intrinsic motivation has already degraded the representation, and does not universally improve every game.

Persistent intrinsic reward can damage representations. On Heist, Ninja, Climber, and Jumper, switching from vanilla to always-on CFN or RND drops action probe accuracy by 0.20–0.40 in this Heist-probe visualization. This pattern is consistent with the broader 12-game survey in Section 6.7, where always-on intrinsic reward often underperforms vanilla. One possible explanation is a noisy-TV-like effect: prolonged intrinsic reward in procedurally varied environments may amplify non-task-relevant variance and destabilize the encoder, consistent with the broader plasticity-loss literature [11, 21].

7.2 Grad-CAM

Grad-CAM on $V(s)$ provides a qualitative visualization that complements the probe-based analysis (Figure 11). The visualizations are computed on source-game observations and should be interpreted as source-task saliency diagnostics rather than direct Heist-transfer explanations. Healthy encoders produce localized hotspots on task-relevant spatial structures—such as corridors in navigation settings and object regions when such objects are present—while collapsed encoders (NLL $\eta = 0.05$) produce diffuse, observation-independent activation along image edges, indicating that the V-function gradient has lost spatial selectivity. Encoders trained on visually distant games (Bigfish, Starpilot) show diffuse heatmaps on maze observations even when probe accuracy is moderate, suggesting that high probe accuracy can arise from non-spatial features and is not by itself evidence of grounded spatial structure.

PCA of bottleneck features. To interpret the structure implied by the visualization of Grad-CAM, we apply PCA to the 256-dimensional IMPALA bottleneck activations pooled across all 59 Maze checkpoints (59 models × 2,000 held-out observations = 118,000 data points). This is related in spirit to EigenCAM-style analyses, but we use it only as a bottleneck-feature geometry diagnostic, not as a saliency or attribution method. The first three principal components explain only 11.1%, 7.1%, and 5.9% of variance respectively, reflecting the high intrinsic dimensionality of healthy representations. KMeans clustering with silhouette scoring reveals $k = 3$ as the natural partition ($s = 0.879$), yet two methods break away as singleton clusters even at $k = 8$: RND $\eta = 0.05$ occupies an extreme positive lobe along PC1 (centroid $\approx +12$, within-method spread $\sigma_{\mathrm{PC1}} = 17.5$), and NLL-Surprise cutoff is dis-

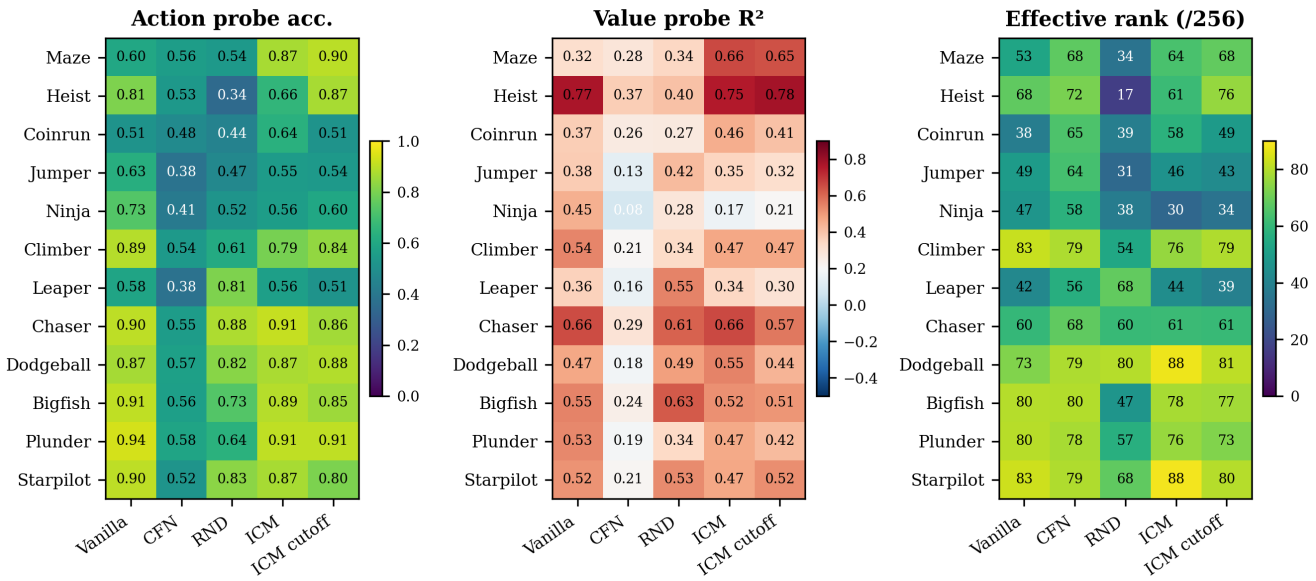

Figure 7: Cross-environment Heist-probe visualization: 12 source games × 5 method families, mean across seeds (ICM cutoff: 6 runs). Unlike the within-game 12-game survey in Section 6.7, all encoders here are evaluated using a fixed Heist expert probe dataset. The figure is therefore used as a qualitative cross-environment diagnostic rather than as the main aggregate benchmark.

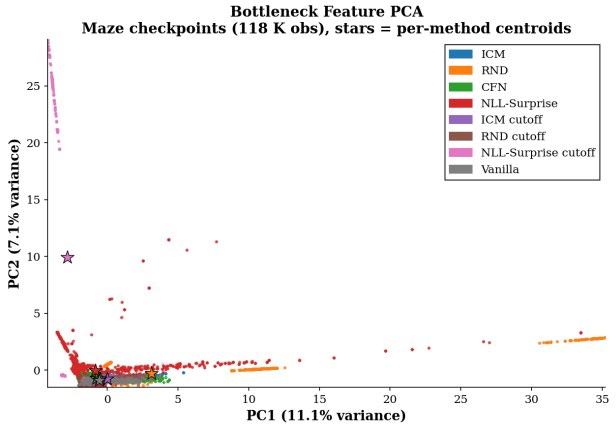

Figure 8: PC1 vs. PC2 of 256-dim IMPALA bottleneck features (118K observations, 59 Maze checkpoints).

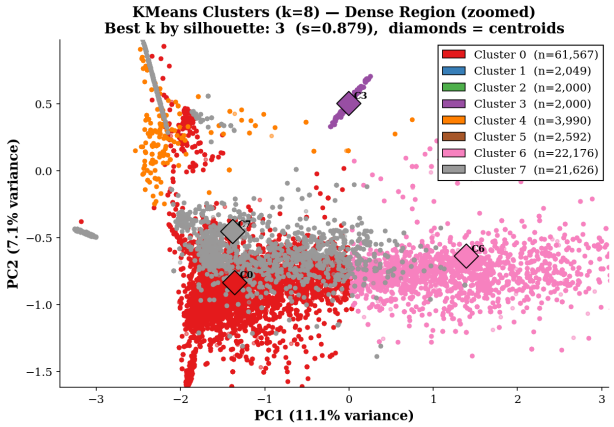

Figure 9: KMeans clusters ($k = 8$) with centroids (diamonds). RND $\eta = 0.05$ and NLL-Surprise cutoff form singleton clusters far from the dense central region; the remaining methods cluster tightly near the origin.

placed along PC2 (centroid $\approx +10$, $\sigma_{\mathrm{PC2}} = 15.5$). Figure 8 shows the raw PC1–PC2 distribution and Figure 9 shows the corresponding KMeans cluster assignments; the per-cluster method composition is reported in Figure 10. All remaining methods—vanilla, ICM (all η), CFN (all η), and low-η RND—cluster tightly near the origin ($\sigma_{\mathrm{PC1}} \lesssim 2$), consistent with their similar probe accuracy and effective-rank scores. This geometric picture reinforces the qualitative finding that the two outlier methods are those whose saliency maps are either diffuse (NLL-Surprise) or edge-biased (high-η RND), suggesting that PC displacement in bottleneck-feature space and loss of spatial selectivity in ∇V may be related signatures of representational collapse.

7.3 Implications for the Transfer Story

Three takeaways tie this analysis back to the transfer results in Tables 7–9.

(1) Cutoff should be interpreted as a rescue mechanism rather than a universal improvement: it is most helpful when always-on intrinsic reward has degraded the representation. This is consistent with the source-eval observations in the 12-game survey: always-on intrinsic reward degrades representations on 9 of 12 source games, and ICM cutoff preserves both source policy quality and representational quality where always-on has begun to drift.

(2) Effective rank is a useful early-stopping or run-screening

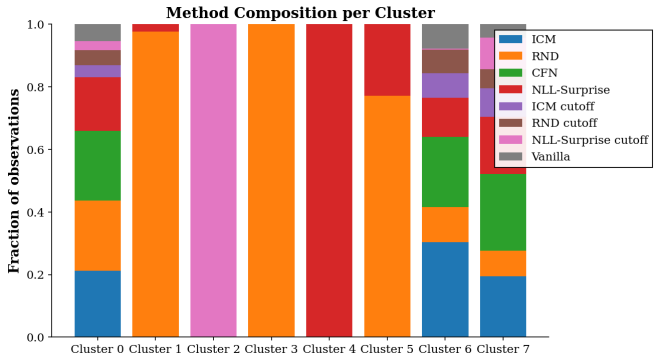

Figure 10: Method composition of each cluster as a stacked bar. Singleton clusters are dominated by single methods (RND $\eta = 0.05$ and NLL-Surprise cutoff), while the dense central clusters are method-mixed.

signal within comparable settings. Multi-seed runs whose effective rank drops below ~20/256 by mid-training reliably produce zero or near-zero zero-shot transfer; pruning these runs would reduce the variance reported in Table 7. As shown in Section 6.7, however, this should not be treated as an absolute cross-game transfer metric.

(3) Vanilla PPO learns surprisingly transferable IMPALA features. The strong vanilla baseline in Figure 7 aligns with the multi-seed result that vanilla matches or beats every intrinsic method on fresh-head fine-tune (Table 7). This suggests that the IMPALA architecture is doing most of the representation-learning work, and intrinsic motivation operates as a *conditional refinement* on top of it: under the right schedule, it can exceed vanilla on specific source–target pairs (notably Maze→Heist, where ICM cutoff exceeds vanilla by $+0.064$ on Heist-source probes, and Plunder→Starpilot, where ICM achieves 67.3% fine-tune win rate vs. vanilla's 41.2%); on a randomly chosen game, however, it typically only matches vanilla, and on the wrong schedule it actively degrades. The contribution of intrinsic motivation to representation learning is therefore real but conditional, dependent on schedule, source-game match, and baseline-representation quality.

7.4 Auxiliary Source–Target Performance Decoupling Analysis

As an auxiliary analysis, we also examine a smaller Maze→Heist clip-normalisation experiment to illustrate that source-task win rate and target-task transfer can be decoupled. Because this experiment uses a different normalization and fine-tuning protocol from the main transfer results in Section 6.10, we treat it as supporting evidence rather than as part of the primary benchmark.

A useful auxiliary pattern emerges when we compare each agent's *source* win rate against its *target* win rate under zero-shot and fine-tuning protocols. If transfer performance were determined primarily by source-task mastery, we would expect source WR to correlate positively with target WR. Instead, we observe a substantial decoupling: target win rates (especially after fine-tuning) can exceed source win rates in this auxiliary setting, and zero-shot performance is nearly always *below* source WR. Table 9 presents this comparison for the auxiliary clip-normalisation experiments (Maze→Heist, $\lambda = 0.05$, 3 seeds), which provide a compact illustration of the effect.

Table 9: Source WR vs. zero-shot and fine-tuned target WR (Maze→Heist, clip normalisation, $\lambda = 0.05$, 3 seeds). **Src**: held-out source win rate. **ZS**: zero-shot target. **1L/2L**: 1- and 2-layer fine-tune. †: cutoff at 10M.

Condition	Src WR	ZS	1L FT	2L FT
Vanilla	17.8%	10.3%	13.3%	**61.7%**
ICM always-on	1.8%	**21.0%**	30.0%	46.7%
ICM†	**15.7%**	15.0%	**35.0%**	36.7%
Target-to-source ratio (fine-tune WR / source WR):				
Vanilla			2L FT: 3.5×	
ICM always-on			1L FT: 16.7×	2L FT: 25.9×
ICM†			1L FT: 2.2×	

Fine-tuned target WR can exceed source WR in this auxiliary setting. The most extreme case is ICM always-on: its intrinsic signal degrades the source policy to 1.8% Maze win rate, yet after just 5M fine-tuning steps it reaches 30% on Heist with 1-layer unfreeze and 46.7% with 2-layer unfreeze. Vanilla PPO shows the same pattern less strongly: 17.8% source → 61.7% 2-layer fine-tune ($3.5\times$).

Zero-shot target WR is typically below source WR. Across conditions, zero-shot Heist performance ranges from 10–21%, comparable to or below the vanilla source WR of 17.8%. This is the *opposite* direction of the fine-tune effect: without adaptation, the pretrained policy carries limited task-specific knowledge across the domain gap.

Source WR predicts neither zero-shot nor fine-tuned performance. ICM always-on has the lowest source WR (1.8%) but the highest zero-shot WR (21.0%) and the highest 2-layer fine-tune WR excluding vanilla (46.7%). ICM cutoff has $9\times$ higher source WR than ICM always-on (15.7% vs. 1.8%) but slightly lower 2-layer fine-tune (36.7% vs. 46.7%). There is no consistent relationship between source mastery and transfer success.

Three interpretations, one caveat. The target-exceeding-source pattern may reflect (a) *Heist being intrinsically easier than Maze*—both games share visual style and action space, but Heist's key-and-door objective has shorter solution paths than Maze's corridor navigation; (b) *fine-tuning doing most of the work*—5M target-task steps may be sufficient to learn Heist from almost any reasonable encoder, consistent with our finding that the IMPALA architecture itself is the primary driver of fine-tune transfer quality (Section 6.10); or (c) *the source encoder containing transferable structural features that the source policy has not learned to act on, but that fine-tuning unlocks via a fresh head*. Under reading (c), source WR is a measure of source-policy quality, not source-encoder quality; the encoder's value is realised only when paired with a target-task action distribution. This is consistent with the partial-decoupling result of Section 6.10: representational health is a precondition for transfer, but actualising it requires adaptation. The caveat is that fine-tune WR *exceeding source WR* does not imply fine-tune WR *exceeding from-scratch WR*: transferring may still be worse than training directly on the target (see Sec-

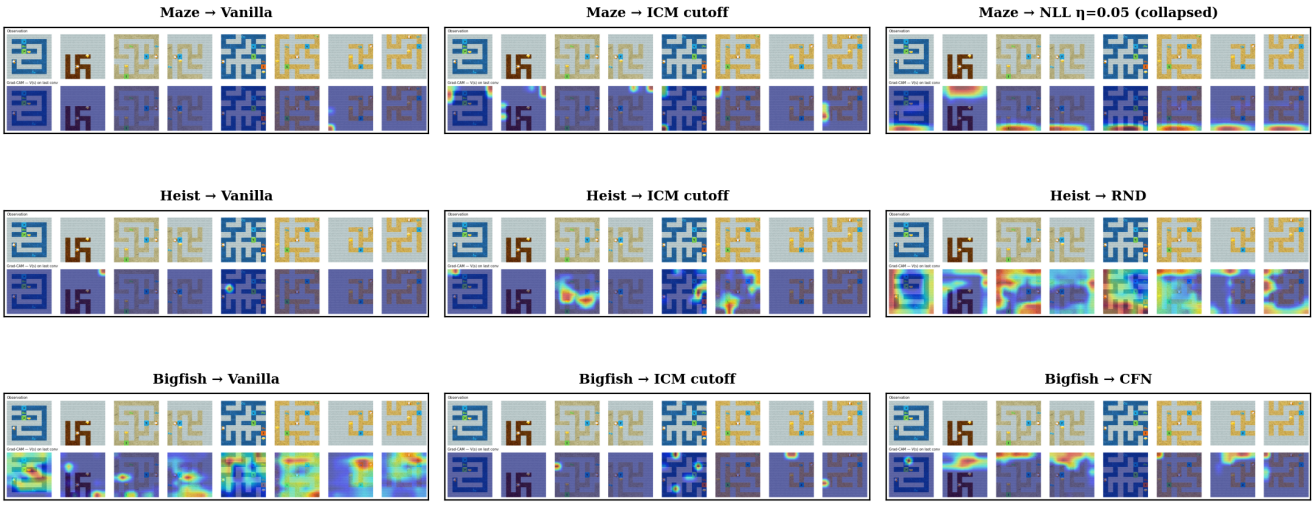

Figure 11: Grad-CAM on $V(s)$ for 3 source games (rows) $\times$ 3 representation qualities (columns). Visualizations are computed on source-game observations (row 1: Maze; row 2: Bigfish; row 3: Starpilot) and should be interpreted as source-task saliency diagnostics, not direct Heist-transfer explanations. Healthy encoders attend to task-relevant spatial regions; collapsed encoders show diffuse, observation-independent attention; cross-domain encoders (Bigfish, Starpilot rows) show diffuse attention even when probe accuracy is moderate.

tion 8, Limitations).

8 Discussion

8.1 Curiosity as Representation Learning

Our results support reframing intrinsic motivation from an exploration heuristic to a *representation learning intervention*. The primary effect of curiosity during source training is not improved exploration per se—indeed, source-task win rates are often comparable or lower for ICM agents—but rather the construction of representations with higher effective rank, fewer dead neurons, and greater linear decodability of target-task-relevant features.

These findings have a specific implication for the cognitive-science framing in Section 1. Oudeyer & Kaplan's claim is that intrinsic motivation produces the broad, reusable internal models that intelligence requires. Our findings refine this: *the intrinsic-motivation signal alone is not sufficient.* The IMPALA encoder and PPO baseline already produce reusable structure (vanilla wins 8 of 12 games on representation quality); intrinsic motivation operates as a *conditional refinement* on top of this base. By Lake et al.'s reusability criterion, vanilla PPO with the right architecture is doing more of the intelligence-relevant work than the curiosity literature typically credits. Intrinsic motivation's contribution is to *preserve representations against degradation* on most games, and to *augment them* on the specific subset where the baseline is weakly specified. This is a more conservative position than Oudeyer's original framing but is consistent with the broader RL-and-cognition view that representations emerge from the joint product of architecture, data, and objective, with no single component being the sole driver of generality.

Representational geometry and generalization. The bottleneck-feature PCA provides a geometric characterization of which methods fail to transfer. Methods displaced from the healthy cluster in PC space—RND $\eta = 0.05$ at PC1$\approx +12$, NLL-Surprise cutoff at PC2$\approx +10$ (Figure 9)—are those with the worst probe accuracy on Heist (Table 1). The converse is equally informative: the six methods that concentrate near the origin are geometrically indistinguishable despite spanning four algorithms and four η values. This indicates that the IMPALA architecture imposes a strong prior on the encoder's representational trajectory—absent a dominant training signal, encoders converge to the same region of feature space regardless of which intrinsic reward is applied. The geometry of collapse is discrete rather than gradual: methods either stay near the healthy cluster or are substantially displaced, suggesting a threshold beyond which the intrinsic signal overrides architectural regularization. The variance-concentration results provide a complementary view. Rigotti et al. [15] showed that the number of task-relevant features a linear readout can extract scales with representational dimensionality, and that collapse to near-linear codes is the primary capacity failure mode. Consistent with this, healthy representations retain $k_{95} \approx 30$–40 effective dimensions; NLL-always-on collapses to $k_{95} \approx 1.5$ with 85.7% dead neurons, eliminating the capacity for flexible linear readout. Dimensionality is necessary but not sufficient—CFN maintains high effective rank yet achieves poor probe accuracy (Section 6.1)—confirming that task-relevant structure, not

dimensionality alone, determines cross-task decodability.

8.2 Why Cutoff Works

The cutoff mechanism is one of the most important practical findings. In procedurally generated environments, the intrinsic signal may not naturally saturate because each episode presents novel level layouts. Without cutoff, the persistent intrinsic signal can increasingly compete with extrinsic reward, gradually pushing representations toward features that predict novelty rather than task success—a form of representation drift consistent with the broader plasticity-loss literature [11, 12]. The cutoff creates a two-phase curriculum: early curiosity-driven exploration diversifies the representation, and subsequent extrinsic-only training consolidates task-relevant features.

This framing helps explain why cutoff tends to matter most when the intrinsic signal is strong and why it is beneficial in the tested cross-environment RND setting (Table 4). However, the cutoff benefit is method-dependent: at $\eta = 0.01$ on Maze→Heist, cutoff strongly rescues RND ($+0.513$ action accuracy), modestly improves ICM ($+0.030$), but actually worsens NLL (-0.187). Cutoff can only help if the early-training intrinsic signal produces useful exploration or useful representation diversity. ICM and RND appear to provide useful early-training intrinsic signals in these settings, whereas NLL's early signal is already damaging, so the damage persists even after deactivation. The 12-game survey further supports the view that always-on curiosity is a risky default—vanilla PPO outperforms always-on intrinsic methods in most games—making proper scheduling important rather than optional.

8.3 The η Collapse Regime

The abrupt collapse at $\eta \approx 0.01$–0.05 for NLL and RND, but not ICM, reveals an important distinction. ICM's intrinsic signal is action-conditioned and shaped by inverse dynamics, making it less prone to purely observation-novelty drift than RND or NLL. NLL and RND lack this structure: their intrinsic signals can reinforce features that optimise the intrinsic objective (predicting the random target network, or minimising NLL) without regard for behavioural relevance, leading to representational collapse when the intrinsic signal dominates. Because the ICM feature encoder is separate from the PPO encoder we probe, this should be read as an indirect effect—the action-conditioned reward signal shapes the PPO training objective—rather than a direct architectural constraint on the probed representation.

8.4 Effective Rank: A Scoped Diagnostic

Within a fixed game pair, the strong correlation ($r = 0.913$) between effective rank and action probe accuracy makes effective rank a practical diagnostic for transfer readiness. Unlike probe accuracy, which requires expert target-task labels, effective rank can be computed from the source encoder alone using any set of observations. This opens the possibility of *online monitoring* during source training: an agent whose effective rank falls below a threshold could trigger early stopping or η reduction, preventing the representation collapse that we observe at high η.

However, the 12-game survey (Section 6.7) reveals an important caveat: across game pairs, this correlation vanishes ($r = 0.093$). A high-rank representation from Coinrun is not comparable to a high-rank representation from Plunder, because the underlying task complexity differs. Furthermore, CFN maintains high effective rank across most games while consistently underperforming vanilla in probe accuracy—demonstrating that rank measures dimensional diversity, not semantic quality. Effective rank should therefore be used as a *within-pair* diagnostic for collapse detection, not as a universal cross-game transferability score. More broadly, linear probes measure whether target-relevant information is accessible in frozen features; they do not fully determine policy transfer, which also depends on source-policy action distributions, head compatibility, and adaptation protocol.

8.5 Limitations

Our study has several limitations.

First, **the cutoff mechanism has not been tested across all method–game-pair combinations.** Within Maze→Heist, cutoff has been tested for ICM, RND, and NLL: it dramatically helps RND ($+0.513$ action accuracy at η=0.01), modestly helps ICM ($+0.030$), but actually hurts NLL (-0.187), whose representations are too damaged by early training for cutoff to rescue. Cross-environment cutoff data (Maze/Heist/Miner → Heist, Table 4) exists only for RND. Whether cutoff benefits for ICM generalise across source environments, and whether CFN benefits from cutoff at all, remain open questions.

Second, **the cutoff mechanism is not universally beneficial for ICM.** Our 12-game cutoff analysis (Section 6.9) reveals that ICM cutoff improves action probe accuracy vs. always-on ICM in only 5 of 12 games and matches vanilla on average. The benefit is concentrated on games where always-on ICM was hurting (Heist: $+0.211$, Climber: $+0.045$, Ninja: $+0.038$); on games where always-on ICM was already productive, cutoff slightly degrades performance. This contradicts the stronger claim—based on the cross-environment RND experiments—that cutoff is universally beneficial. A more accurate framing is that cutoff is a *rescue mechanism* for runs where the intrinsic signal has drifted into hurting the representation, rather than an unconditional improvement.

Third, **effective rank does not generalise as a cross-game diagnostic.** The within-pair correlation ($r = 0.913$) vanishes across game pairs ($r = 0.093$), and CFN demonstrates that high rank does not guarantee good probe accuracy. This limits the practical applicability of effective rank as a universal transfer readiness metric.

Fourth, our multi-seed experiments use 3 seeds, which limits statistical power for conditions with high variance (notably CFN and NLL). The cutoff schedule is also fixed at 10M/25M steps; systematic study of cutoff timing and decay schedules may yield further improvements.

Fifth, the partial decoupling between representation quality and transfer performance (Section 6.10) suggests that our representational diagnostics capture necessary but not sufficient

conditions. Additional factors—policy head structure, action distribution similarity between source and target, and the specific adaptation protocol—warrant investigation.

Sixth, **distribution mode is not held constant across all experiment sets.** Four of our five representational-analysis experiment sets (primary Maze→Heist, η sweep, ICM ablation, and the 12-game survey) use Procgen's `easy` distribution mode, while the cross-environment RND experiments (Section 6.8) use `hard` mode. A separate easy-mode cross-environment RND sweep is reported in Section 6.8.1 for downstream win-rate validation, but it is not included in the 507-checkpoint representational-analysis total. This mismatch means cross-environment comparisons—particularly claims that cutoff benefits generalise from Maze→Heist (easy) to Maze/Heist/Miner→Heist (hard)—must be treated as approximate rather than fully controlled. `Hard` mode increases procedural variation and level difficulty, which could shift both baseline representation quality and the effect size of any intervention. We partially address this in Section 6.8.1 by running an easy-mode cross-environment RND sweep, which confirms the directional cutoff finding but reveals that absolute win rates are substantially lower in easy mode than hard mode for Maze source ($5.4\times$ lower ZS), while Miner source shows near-parity across modes. Full characterisation would require a factorial study isolating distribution mode from η range and target-environment differences.

Seventh, **our transfer protocols are not consistently benchmarked against from-scratch training on the target**. Preliminary from-scratch comparisons suggest that transferred backbones do not always outperform random initialisation. For example, Miner from scratch reaches $\sim 100\%$ win rate, while Maze→Miner transfer with 25M fine-tuning steps reaches only $\sim 11\%$. A small hard-mode Maze→Heist comparison showed a similar directional concern, but used non-identical train/test level sets, so we treat it only as motivation for a systematic from-scratch baseline study. A systematic from-scratch baseline comparison is essential future work for claiming practical transfer utility, and this caveat applies to all representation-learning claims in this paper.

Eighth, **the fine-tuning step budget has not been varied in our primary experiments.** A supplementary sweep testing 15 M versus 25 M fine-tuning steps across three cross-environment pairs (Maze→Miner, Heist→Miner, Miner→Maze; RND / RND-cutoff; easy distribution mode; 3 seeds each) found no consistent improvement from the larger budget: mean test win rates were 12.1% vs. 11.1% (Maze→Miner), 11.5% vs. 11.9% (Heist→Miner), and 8.6% vs. 8.7% (Miner→Maze). This is consistent with the possibility that representation quality, rather than fine-tuning duration alone, limits adaptation in these auxiliary settings. The sweep covers only the 15–25 M range; whether our primary 5 M fine-tuning budget lies below the saturation point on the Maze→Heist pair remains untested.

9 Conclusion

We have presented a systematic study of intrinsic motivation as a representation-learning mechanism for cross-task transfer in Procgen. Five findings stand out.

First, intrinsic motivation reframes as a scheduled representation-learning intervention, not a persistent exploration objective. Across 507 checkpoints, the determining factor is not method alone, but the interaction among method, schedule, source game, and intrinsic reward strength: always-on schedules often damage representations, leaving vanilla PPO as the strongest baseline in 8 of 12 source games, while cutoff schedules either rescue these or leave healthy ones largely intact. Intrinsic motivation, when correctly scheduled, can produce more transferable representations than vanilla PPO on specific source–target pairs (most clearly Maze→Heist and Plunder→Starpilot)—but no single intrinsic-method schedule consistently outperforms vanilla across our 12-game survey. Curiosity's contribution is therefore real but conditional.

Second, the effect of intrinsic motivation depends on baseline representation quality. Across the 12-game survey, the gap between intrinsic-method and vanilla representation quality correlates negatively with vanilla's own representation quality ($r = -0.41$ to -0.88). Intrinsic motivation provides representational diversity when vanilla is weak, and noise when vanilla is already strong. This re-frames the question "does intrinsic motivation help transfer?" as "for which source games is vanilla insufficient, and does intrinsic motivation fill the gap?"

Third, cross-game transfer in Procgen is fundamentally difficult, and the wins are concentrated. Across 14 game pairs (Appendix A), cross-game zero-shot transfer is highly pair-dependent: many mechanically mismatched pairs remain near zero, while a smaller number of structurally compatible or degenerate pairs transfer substantially better. No method achieves consistent wins across the broad pair set. Where intrinsic motivation does help, the gain is concentrated on specific pairs with structural compatibility—most clearly Plunder→Starpilot (ICM fine-tune 67.3% vs. vanilla 41.2%) and our primary Maze→Heist pair. Source–target match and baseline-representation quality jointly determine success.

Fourth, representation health and transfer performance are scope-dependent and partially decoupled. Effective rank predicts transferability with $r = 0.913$ within Maze→Heist but $r = 0.093$ across game pairs—establishing both its utility as a within-pair early-warning diagnostic and its limit as a cross-game absolute score. Separately, the methods producing the best representations are not consistently the methods producing the best transfer (RND-cut zero-shot, CFN fine-tune, ICM-cut representation quality). Researchers using rank, dormancy, or similar diagnostics should validate within-scope before extrapolating, and should not treat representation quality alone as a sufficient predictor of downstream transfer.

Fifth, curiosity is not a single design choice but a stack of conditional ones. Method (ICM vs. RND vs. CFN vs. NLL), schedule (always-on vs. cutoff), strength (η), forward/inverse

loss balance (β, for ICM), and source–game match all interact. ICM's robustness to high η is consistent with inverse-model stabilisation; NLL's collapse without such a constraint suggests this architectural difference is the critical factor. β is comparatively insensitive in the moderate-η regime, indicating that the intrinsic reward weight, not the loss balance, is the critical tuning axis. Deploying any of these methods in a new domain requires re-tuning across the full conditional stack, not just selecting a method and an η.

Open questions. Whether transfer from any of these encoders beats from-scratch training on the target (Section 8, limitation 7)—a precondition for practical deployment—and whether the patterns we identify hold for understanding-based curiosity methods (Appendix B) are the next steps for this line of work. Returning to the cognitive-science framing of Section 1: our findings suggest that the question "what produces reusable representations?" is more fruitfully decomposed into "what architecture supports them, what training signal preserves them, and under what conditions does intrinsic motivation augment them?" than into the simpler "is intrinsic motivation necessary for intelligence?" framing of older work.

References

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

[2] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *ICLR Workshop*, 2017.

[3] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *ICLR*, 2019.

[4] Yuri Burda, Harrison Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A Efros. Large-scale study of curiosity-driven learning. In *ICLR*, 2019.

[5] Karl Cobbe, Chris Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning. In *ICML*, 2020.

[6] Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane Legg, and Koray Kavukcuoglu. IMPALA: Scalable distributed deep-RL with importance weighted actor-learner architectures. In *ICML*, 2018.

[7] Yiding Jiang, J. Zico Kolter, and Roberta Raileanu. On the importance of exploration for generalization in reinforcement learning. In *ICML*, 2023.

[8] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *ICML*, 2019.

[9] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40, 2017.

[10] Sam Lobel, Akhil Bagaria, and George Konidaris. Flipping coins to estimate pseudocounts for exploration in reinforcement learning. In *ICML*, 2023.

[11] Clare Lyle, Zeyu Zheng, Evgenii Nikishin, Bernardo Avila Pires, Razvan Pascanu, and Will Dabney. Understanding plasticity in neural networks. In *ICML*, 2023.

[12] Evgenii Nikishin, Max Schwarzer, Pierluca D'Oro, Pierre-Luc Bacon, and Aaron Courville. The primacy bias in deep reinforcement learning. In *ICML*, 2022.

[13] Pierre-Yves Oudeyer and Frederic Kaplan. What is intrinsic motivation? A typology of computational approaches. *Frontiers in Neurorobotics*, 1:6, 2007.

[14] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *ICML*, 2017.

[15] Mattia Rigotti, Omri Barak, Melissa R Warden, Xiao-Jing Wang, Nathaniel D Daw, Earl K Miller, and Stefano Fusi. The importance of mixed selectivity in complex cognitive tasks. *Nature*, 497(7451):585–590, 2013.

[16] Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *European Signal Processing Conference (EUSIPCO)*, 2007.

[17] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

[18] Andrei A Rusu, Sergio Gomez Colmenarejo, Caglar Gulcehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy distillation. In *ICLR*, 2016.

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

[20] Ramanan Sekar, Oleh Rybkin, Kostas Daniilidis, Pieter Abbeel, Danijar Hafner, and Deepak Pathak. Planning to explore via self-supervised world models. In *ICML*, 2020.

[21] Ghada Sokar, Rishabh Agarwal, Pablo Samuel Castro, and Utku Evci. The dormant neuron phenomenon in deep reinforcement learning. In *ICML*, 2023.

[22] Adam Stooke, Kimin Lee, Pieter Abbeel, and Michael Laskin. Decoupling representation learning from reinforcement learning. In *ICML*, 2021.

[23] Matthew E Taylor and Peter Stone. Transfer learning for reinforcement learning domains: A survey. *JMLR*, 10:1633–1685, 2009.

[24] Shanchuan Wan, Yujin Tang, Yingtao Tian, and Tomoyuki Kaneko. DEIR: Efficient and robust exploration through discriminative-model-based episodic intrinsic rewards. In *IJCAI*, 2023.

[25] Zhuangdi Zhu, Kaixiang Lin, Anil K Jain, and Jiayu Zhou. Transfer learning in deep reinforcement learning: A survey. *IEEE TPAMI*, 45(11):13344–13362, 2023.

[26] Ev Zisselman, Itai Lavie, Daniel Soudry, and Aviv Tamar. Explore to generalize in zero-shot RL. In

NeurIPS, 2023.

A Cross-Game-Pair Transfer Results

To evaluate whether the effects observed on Maze→Heist generalise, we test zero-shot and fine-tuned transfer across 14 game pairs spanning diverse mechanical relationships: navigation (Maze↔Chaser, Heist→Maze), platforming (Coinrun↔Jumper, Climber↔Ninja), combat (Dodgeball→Starpilot, Starpilot→Bossfight), and resource collection (Maze→Miner, Plunder→Starpilot). All agents are trained for 25M steps on the source game with $\eta = 0.005$ (always-on, no cutoff) and evaluated on the target. Results are reported as mean ± std across 3 seeds.

A.1 Zero-Shot Transfer

Table 10 presents zero-shot transfer across all 14 game pairs. Cross-game zero-shot transfer is generally very difficult: win rates rarely exceed 5%, reflecting the substantial mechanical divergence between Procgen games despite their shared visual style and action space.

Vanilla wins the majority of zero-shot pairs. Vanilla achieves the best zero-shot mean reward in 8 of 14 game pairs, ICM in 3, CFN in 2, and RND in 1. This is consistent with the 12-game representational survey (Section 6.7): always-on intrinsic motivation does not reliably produce representations that zero-shot transfer better than vanilla.

Some pairs transfer well, most do not. Maze→Chaser achieves 100% win rate for all methods (Chaser's win condition is trivially easy to satisfy), and Dodgeball→Starpilot and Plunder→Starpilot achieve $> 30\%$ win rates, likely because combat-oriented source games learn spatial attention patterns useful for Starpilot. Most other pairs achieve $< 5\%$ win rate, confirming that cross-game transfer in Procgen is fundamentally difficult.

Transfer is asymmetric. Heist→Maze (3.8% vanilla win rate) is substantially easier than Maze→Heist (1.5%), and Ninja→Climber (2.7%) is easier than Climber→Ninja (2.1%). The source game's complexity and the overlap of required skills determine transfer difficulty in a direction-dependent way.

A.2 Fine-Tuned Transfer

Table 11 presents fine-tuned transfer, where the source encoder is frozen and a fresh policy head is trained on the target for 5M steps.

Fine-tuning reduces the vanilla advantage. With fine-tuning, ICM becomes competitive or superior in more pairs: ICM achieves the best fine-tuned reward in 5 of 14 pairs (vs. 3 zero-shot), and produces the standout result on Plunder→Starpilot (67.3% vs. 41.2% vanilla—a 63% relative improvement with low variance). This suggests ICM's representations, while not always immediately usable, encode features that a fresh policy head can exploit more effectively.

CFN shows extreme pair-dependence. CFN achieves the best fine-tune result on 3 pairs (Ninja→Climber: 19.8%, Jumper→Coinrun: 43.2%, Chaser→Maze: 9.5%) but catastrophically fails on others (Dodgeball→Starpilot: 0.0%, Plunder→Starpilot: 0.0%). This bimodal behaviour—either excellent or completely broken—makes CFN unreliable as a general-purpose method despite occasional strong performance.

No single method dominates fine-tuning. The best method varies by game pair in an unpredictable way: ICM excels on Maze→Heist and Plunder→Starpilot, CFN on Ninja→Climber and Jumper→Coinrun, RND on Leaper→Jumper, and vanilla on Dodgeball→Starpilot and Climber→Ninja. This reinforces the conclusion from the main text: the benefit of intrinsic motivation is game-pair-dependent, and no method is universally superior.

ICM's fine-tune advantage is strongest on the hardest pairs. On pairs where vanilla fine-tuning achieves $< 15\%$ win rate (Maze→Heist, Maze→Miner, Coinrun→Jumper), ICM consistently matches or exceeds vanilla. On easier pairs where vanilla already achieves $> 40\%$ (Dodgeball→Starpilot, Plunder→Starpilot), the picture is mixed. This parallels the 12-game representational finding (Section 6.7) that ICM helps most when vanilla representations are under-specified.

A.3 Which Game Pairs Transfer Best?

Taken together, Table 3 (representational quality per source game), Table 10 (zero-shot transfer), and Table 11 (fine-tuned transfer) let us identify game pairs where transfer works best along different axes.

Source games that produce the healthiest representations (same-game probing)

Ranking the 12 source games by vanilla action probe accuracy *probed against that same game's own expert* reveals three tiers (Table 12). Note that this is an *in-game* representational quality metric (source = target for probing), distinct from the cross-game transfer metrics in Table 10 and Table 11:

Games with spatially rich, reward-dense gameplay (Plunder, Bigfish, Starpilot) produce the highest-quality encoders. Games with sparse or narrow objectives (Maze, Coinrun) produce the weakest.

Game pairs with the strongest fine-tuned transfer

Ranking the 14 pairs by best fine-tune win rate across methods (Table 13):

In-game quality does not guarantee transfer quality. Plunder is tier-1 in-game (probe acc 0.935) *and* produces the top ICM transfer result (Plunder → Starpilot: 67.3% fine-tune WR). But Bigfish is also tier-1 in-game (probe acc 0.905) and only achieves modest transfer (26.2% on Bigfish → Fruitbot). Conversely, Maze is tier-weak in-game (0.599) yet serves as the source for our primary transfer pair (Maze → Heist, 6.7% ICM fine-tune), showing that useful transfer is possible even from representationally-weak sources. The relationship between same-game probe accuracy and cross-game transfer is therefore nonlinear—*mechanical overlap between source and target matters more than raw source-encoder quality*.

Table 10: Zero-shot transfer performance across 14 game pairs ($\eta = 0.005$, always-on, 3 seeds). Best mean reward per pair **bolded**. Note: Maze→Chaser shows identical 100% win rates across methods because Chaser's win condition is trivially satisfied; this row is excluded from the per-pair "best method" analysis.

Game Pair	Metric	Vanilla	ICM	RND	CFN
Bigfish → Fruitbot	Mean Reward	-2.60 ± 0.25	-2.19 ± 0.45	-1.85 ± 0.32	$\mathbf{-1.31 \pm 0.14}$
	Win Rate	$25.2 \pm 1.9\%$	$22.5 \pm 4.7\%$	$\mathbf{26.7 \pm 2.3\%}$	$21.7 \pm 4.5\%$
Chaser → Maze	Mean Reward	$\mathbf{0.41 \pm 0.29}$	0.24 ± 0.27	0.05 ± 0.04	0.29 ± 0.41
	Win Rate	$\mathbf{4.1 \pm 2.9\%}$	$2.4 \pm 2.7\%$	$0.5 \pm 0.4\%$	$2.9 \pm 4.1\%$
Climber → Ninja	Mean Reward	$\mathbf{0.21 \pm 0.20}$	0.15 ± 0.14	0.18 ± 0.16	0.20 ± 0.29
	Win Rate	$\mathbf{2.1 \pm 2.0\%}$	$1.5 \pm 1.4\%$	$1.8 \pm 1.6\%$	$2.0 \pm 2.9\%$
Coinrun → Jumper	Mean Reward	0.75 ± 0.33	$\mathbf{0.79 \pm 0.36}$	0.66 ± 0.36	0.61 ± 0.45
	Win Rate	$7.6 \pm 3.3\%$	$\mathbf{7.9 \pm 3.6\%}$	$6.6 \pm 3.6\%$	$6.1 \pm 4.5\%$
Dodgeball → Starpilot	Mean Reward	$\mathbf{1.15 \pm 0.23}$	0.58 ± 0.26	0.76 ± 0.12	0.21 ± 0.30
	Win Rate	$\mathbf{44.0 \pm 6.6\%}$	$26.6 \pm 9.2\%$	$32.2 \pm 7.6\%$	$10.9 \pm 15.2\%$
Heist → Maze	Mean Reward	$\mathbf{0.38 \pm 0.13}$	0.18 ± 0.25	0.00 ± 0.00	0.25 ± 0.36
	Win Rate	$\mathbf{3.8 \pm 1.3\%}$	$1.8 \pm 2.5\%$	$0.0 \pm 0.0\%$	$2.6 \pm 3.6\%$
Jumper → Coinrun	Mean Reward	$\mathbf{1.06 \pm 1.27}$	0.19 ± 0.27	0.98 ± 1.38	0.00 ± 0.00
	Win Rate	$\mathbf{10.6 \pm 12.7\%}$	$1.9 \pm 2.7\%$	$9.8 \pm 13.8\%$	$0.0 \pm 0.0\%$
Leaper → Jumper	Mean Reward	0.40 ± 0.22	0.25 ± 0.09	0.17 ± 0.08	$\mathbf{0.57 \pm 0.47}$
	Win Rate	$4.0 \pm 2.2\%$	$2.5 \pm 0.9\%$	$1.7 \pm 0.8\%$	$\mathbf{5.7 \pm 4.7\%}$
Maze → Chaser†	Mean Reward	0.20 ± 0.05	$\mathbf{0.20 \pm 0.07}$	0.14 ± 0.09	0.15 ± 0.06
	Win Rate	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$
Maze → Heist	Mean Reward	0.15 ± 0.05	0.22 ± 0.17	0.07 ± 0.05	$\mathbf{0.23 \pm 0.15}$
	Win Rate	$1.5 \pm 0.5\%$	$2.2 \pm 1.7\%$	$0.7 \pm 0.5\%$	$\mathbf{2.3 \pm 1.5\%}$
Maze → Miner	Mean Reward	0.08 ± 0.01	$\mathbf{0.14 \pm 0.03}$	0.05 ± 0.03	0.08 ± 0.03
	Win Rate	$8.2 \pm 1.0\%$	$\mathbf{12.6 \pm 2.6\%}$	$4.8 \pm 3.5\%$	$7.5 \pm 2.6\%$
Ninja → Climber	Mean Reward	0.12 ± 0.09	$\mathbf{0.19 \pm 0.06}$	0.02 ± 0.03	0.00 ± 0.00
	Win Rate	$2.7 \pm 1.4\%$	$\mathbf{4.3 \pm 0.4\%}$	$0.2 \pm 0.3\%$	$0.0 \pm 0.0\%$
Plunder → Starpilot	Mean Reward	$\mathbf{0.95 \pm 0.27}$	0.89 ± 0.55	0.79 ± 0.54	0.23 ± 0.32
	Win Rate	$\mathbf{44.6 \pm 8.3\%}$	$39.3 \pm 16.8\%$	$35.2 \pm 22.5\%$	$11.7 \pm 16.2\%$
Starpilot → Bossfight	Mean Reward	$\mathbf{0.02 \pm 0.02}$	0.01 ± 0.00	0.00 ± 0.00	0.01 ± 0.01
	Win Rate	$0.3 \pm 0.3\%$	$\mathbf{0.5 \pm 0.4\%}$	$0.4 \pm 0.1\%$	$\mathbf{0.5 \pm 0.8\%}$

†Win rate is uninformative for this pair: see caption.

Pairs where intrinsic motivation provides the clearest benefit

The two most informative success cases for the intrinsic-motivation-helps-transfer hypothesis are:

- **Plunder → Starpilot**: ICM achieves $67.3 \pm 3.4\%$ fine-tune win rate vs. $41.2 \pm 12.0\%$ for vanilla—a 63% relative improvement with *lower* variance. This is the cleanest single-pair demonstration that intrinsic motivation can substantially outperform vanilla PPO on transfer, even with always-on curiosity.
- **Coinrun → Jumper**: ICM achieves $17.0 \pm 1.8\%$ vs. $12.0 \pm 5.6\%$ for vanilla—a smaller absolute gain but with $3\times$ lower variance. Notable because the source game (Coinrun) is in the "weak" representational tier, illustrating that intrinsic motivation helps most where vanilla representations are under-specified.

Pairs where transfer fundamentally fails

Several pairs achieve near-zero zero-shot win rates ($<5\%$) regardless of method: Climber → Ninja (2.1%), Maze → Heist (1.5%), Starpilot → Bossfight (0.3%), Ninja → Climber (2.7%). These pairs share a property: the target game requires mechanical competencies (e.g., Bossfight's enemy patterns, Heist's key-and-door objectives) that are not rehearsed in the source game. No method we tested overcomes this gap zero-shot, though fine-tuning partially recovers performance on most.

Recommendations

Based on our findings, we recommend the following pair-selection heuristics for future work on curiosity-driven transfer:

1. **Test on Plunder → Starpilot** as a positive benchmark for intrinsic motivation—it is the cleanest demonstration that ICM can dominate vanilla.

Table 11: Fine-tuned transfer performance across 14 game pairs (fresh-head, $\eta = 0.005$, always-on, 3 seeds). Best mean reward per pair **bolded**. Maze→Chaser win rates are uninformative for the same reason as in Table 10.

Game Pair	Metric	Vanilla	ICM	RND	CFN
Bigfish → Fruitbot	Mean Reward	-1.04 ± 0.19	-1.32 ± 0.32	-1.39 ± 0.17	$\mathbf{-0.90 \pm 0.06}$
	Win Rate	$\mathbf{26.2 \pm 1.3}\%$	$25.5 \pm 1.5\%$	$25.1 \pm 1.7\%$	$24.0 \pm 0.3\%$
Chaser → Maze	Mean Reward	0.91 ± 0.01	0.93 ± 0.05	0.93 ± 0.04	$\mathbf{0.95 \pm 0.04}$
	Win Rate	$9.1 \pm 0.1\%$	$9.3 \pm 0.5\%$	$9.3 \pm 0.4\%$	$\mathbf{9.5 \pm 0.4}\%$
Climber → Ninja	Mean Reward	$\mathbf{1.62 \pm 0.68}$	1.35 ± 0.40	1.20 ± 0.41	0.73 ± 0.04
	Win Rate	$\mathbf{16.2 \pm 6.8}\%$	$13.5 \pm 4.0\%$	$12.0 \pm 4.1\%$	$7.3 \pm 0.4\%$
Coinrun → Jumper	Mean Reward	1.20 ± 0.56	$\mathbf{1.71 \pm 0.18}$	1.25 ± 0.34	1.07 ± 0.36
	Win Rate	$12.0 \pm 5.6\%$	$\mathbf{17.0 \pm 1.8}\%$	$12.5 \pm 3.4\%$	$10.7 \pm 3.6\%$
Dodgeball → Starpilot	Mean Reward	$\mathbf{2.12 \pm 0.15}$	1.76 ± 0.10	1.40 ± 0.10	0.00 ± 0.00
	Win Rate	$\mathbf{68.3 \pm 4.9}\%$	$61.1 \pm 6.5\%$	$54.3 \pm 2.6\%$	$0.0 \pm 0.0\%$
Heist → Maze	Mean Reward	0.91 ± 0.02	$\mathbf{0.95 \pm 0.01}$	0.87 ± 0.04	0.94 ± 0.03
	Win Rate	$9.1 \pm 0.2\%$	$\mathbf{9.5 \pm 0.1}\%$	$8.7 \pm 0.4\%$	$9.4 \pm 0.3\%$
Jumper → Coinrun	Mean Reward	3.07 ± 2.51	3.24 ± 2.37	4.08 ± 2.92	$\mathbf{4.32 \pm 3.05}$
	Win Rate	$30.7 \pm 25.1\%$	$32.4 \pm 23.7\%$	$40.8 \pm 29.2\%$	$\mathbf{43.2 \pm 30.5}\%$
Leaper → Jumper	Mean Reward	1.72 ± 0.11	1.42 ± 0.67	$\mathbf{1.79 \pm 0.49}$	0.84 ± 0.04
	Win Rate	$17.2 \pm 1.1\%$	$14.2 \pm 6.7\%$	$\mathbf{17.9 \pm 4.9}\%$	$8.4 \pm 0.4\%$
Maze → Chaser[†]	Mean Reward	0.19 ± 0.10	$\mathbf{0.26 \pm 0.14}$	0.07 ± 0.04	0.14 ± 0.12
	Win Rate	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$
Maze → Heist	Mean Reward	0.62 ± 0.01	$\mathbf{0.67 \pm 0.15}$	0.47 ± 0.10	0.41 ± 0.02
	Win Rate	$6.2 \pm 0.1\%$	$\mathbf{6.7 \pm 1.5}\%$	$4.7 \pm 1.0\%$	$4.1 \pm 0.2\%$
Maze → Miner	Mean Reward	$\mathbf{0.14 \pm 0.05}$	0.13 ± 0.02	0.10 ± 0.01	0.11 ± 0.01
	Win Rate	$\mathbf{12.6 \pm 3.6}\%$	$11.9 \pm 1.3\%$	$10.1 \pm 1.2\%$	$10.7 \pm 1.1\%$
Ninja → Climber	Mean Reward	0.63 ± 0.15	0.92 ± 0.25	0.63 ± 0.18	$\mathbf{1.09 \pm 0.06}$
	Win Rate	$12.4 \pm 1.1\%$	$16.3 \pm 2.4\%$	$13.2 \pm 3.7\%$	$\mathbf{19.8 \pm 0.9}\%$
Plunder → Starpilot	Mean Reward	1.03 ± 0.55	$\mathbf{2.29 \pm 0.35}$	0.55 ± 0.42	0.00 ± 0.00
	Win Rate	$41.2 \pm 12.0\%$	$\mathbf{67.3 \pm 3.4}\%$	$23.6 \pm 16.9\%$	$0.0 \pm 0.0\%$
Starpilot → Bossfight	Mean Reward	$\mathbf{0.52 \pm 0.02}$	0.44 ± 0.08	0.47 ± 0.04	0.47 ± 0.06
	Win Rate	$14.5 \pm 1.4\%$	$\mathbf{14.6 \pm 0.5}\%$	$13.6 \pm 0.8\%$	$13.7 \pm 1.1\%$

[†]Win rate is uninformative for this pair: see caption.

Table 12: Source-game tiers by in-game vanilla action probe accuracy (each game probed against its own expert).

Tier	Source games	Probe acc.
Excellent	Plunder, Bigfish, Starpilot, Climber, Chaser	0.89–0.94
Good	Dodgeball, Heist	0.81–0.87
Moderate	Ninja, Jumper	0.63–0.73
Weak	Coinrun, Leaper, Maze	0.51–0.60

2. **Avoid claiming generality from easy pairs** (e.g., Dodgeball → Starpilot, Maze → Chaser) where the transfer is trivially achieved by any method.
3. **Use weak-source pairs** (Coinrun-based, Maze-based) as stress tests: these are where intrinsic motivation's benefit is most visible, but also where from-scratch baselines are competitive.
4. **Prioritise fine-tuning over zero-shot evaluation** when comparing intrinsic methods—the zero-shot signal is weak and noisy in Procgen, while fine-tuned differences are larger and more robust.

Table 13: Top fine-tune transfer pairs ranked by best across methods. Maze→Chaser is excluded because its 100% win rate is trivially attained (see Table 10).

Rank	Pair	Best FT WR	Winning method
1	Dodgeball → Starpilot	68.3%	Vanilla
2	Plunder → Starpilot	**67.3%**	**ICM** (+26pp)
3	Jumper → Coinrun	43.2%	CFN (high variance)
4	Coinrun → Jumper	17.0%	ICM (+5pp)
5	Leaper → Jumper	17.9%	RND
6	Ninja → Climber	19.8%	CFN
7	Climber → Ninja	16.2%	Vanilla

B Understanding-Based Curiosity: A Preliminary Cross-Pair Test

The four intrinsic methods studied in the main paper (ICM, RND, CFN, NLL-Surprise) all sit on the *coverage* side of the curiosity taxonomy (Section 2): they reward states that are unfamiliar to the agent's predictor, but they do not naturally stop

when the dynamics are learned. A complementary strand of curiosity research focuses on *understanding-based* methods—IMGEP / Learning Progress, GRIMGEP, and world-model-disagreement methods such as Plan2Explore—which derive their reward from progress on a learned dynamics model and naturally decay as the model converges.

Whether understanding-based methods produce qualitatively different transfer behaviour from coverage-based methods is a meaningful open question, and one we initially set out to answer. This appendix reports preliminary results from that effort. We adapted three understanding-based methods to our PPO+IMPALA pipeline:

- **IMGEP** (Learning Progress over a co-trained CNN latent, $K = 32$ regions),
- **GRIMGEP** (frozen-encoder clustering basis, $K = 12$ regions),
- **WME** (World Model Ensemble: 5 forward-prediction heads sharing one ImpalaCNN, disagreement-as-reward),

each in always-on and 10M-step-cutoff variants. We chose **Chaser → Heist** as the source–target pair on the grounds of shared causal structure: both games require collecting an enabler object (Chaser's pellet/star, Heist's key) before using it on a target entity (eat enemy, open lock). Each method × cutoff condition was run with 3 seeds for 25M source-task steps, with the same backbone, optimiser, and $\eta = 0.005$ used for the main-paper ICM/RND experiments.

Results. After stripping seeds with catastrophic encoder collapse (defined as effective rank below 5 by mid-training), no understanding-based method significantly outperformed vanilla PPO on Chaser → Heist transfer. Stripping was necessary: 2 of 18 understanding-method seeds collapsed to effective rank $\approx$ 1 with >60% layer-4 dormancy, while the remaining 14 seeds maintained healthy representations (rank 100–185, dormancy 13–18%). On the healthy seeds only, all six understanding-method conditions performed substantially worse than vanilla PPO on both zero-shot transfer and fresh-head fine-tune. The understanding-method numbers for this preliminary sweep are reported in Table 14; we report them only relative to one another rather than against a vanilla baseline value, because the auxiliary sweep used non-identical training settings (bfloat16 mixed precision and minibatch 8 vs. minibatch 4 in some main-paper sets), which makes a numerical vanilla comparison from this sweep an apples-to-oranges comparison with the main-paper Maze-source numbers. The directional finding—no understanding-based method matched the vanilla baselines reported elsewhere in the paper—is robust to that confound.

The pattern persisted after stripping collapsed seeds and persisted with the cutoff variants. Diagnosing this required re-examining what each method was actually optimising:

- For **IMGEP**, learning progress over a co-trained CNN latent appears to be a degenerate signal: the latent shifts continuously under PPO gradient updates, so per-region learning progress is dominated by encoder drift rather than dynamics improvement. This is consistent with prior work using *frozen* offline encoders for IMGEP latents; we did not find a published precedent for IMGEP over a co-trained CNN.
- For **GRIMGEP**, even with a frozen IDM-warmed clustering basis, the LP-reward gradient flowing through the policy and dynamics encoder caused representation drift on a timescale similar to IMGEP. $K = 12$ was an attempt to reduce noise relative to $K = 32$, but did not change the qualitative outcome.
- For **WME** (the best of the three), the disagreement signal does decay as the ensemble converges, in line with theory. But the resulting source-task win rates were low (10–15% on Chaser, vs. vanilla's higher rate), suggesting the agent never reliably learned the collect-then-use chain—possibly because the temporal credit-assignment window for the chain ($\sim$3 seconds in Chaser) is shorter than the timescale on which disagreement decays. Without source-task mastery, transfer is moot.

Table 14: Chaser → Heist understanding-based curiosity results (healthy-seed means after stripping collapsed seeds, n=14 of 18). Vanilla baseline numbers from this auxiliary sweep are not reported because of the training-setting mismatch noted in the prose; instead, see Table 7 and Table 11 for vanilla numbers under the main-paper protocol.

Method	Zero-shot	Fresh-head FT
GRIMGEP	0.030	0.064
GRIMGEP-cut	0.007	0.050
IMGEP	0.012	0.048
IMGEP-cut	0.019	0.058
WME	0.009	0.057
WME-cut	0.019	0.065

Interpretation. Three readings are consistent with these results:

1. **Implementation-specific failure.** Each of the three methods has a known sensitive design choice (IMGEP's encoder, GRIMGEP's clustering basis, WME's ensemble bagging). Better-tuned versions of each might perform differently. We did not exhaust the design space.
2. **Source-task mismatch.** Chaser's collect-then-use chain may be too short-horizon for understanding-based methods to gain traction. A source task with longer-horizon causal structure might produce different results.
3. **The IMPALA-architecture story extends.** Our main paper finds that the IMPALA architecture, not the intrinsic signal, is the primary representational driver (Section 7). The Chaser → Heist results are consistent with this: even understanding-based curiosity, with its different theoretical motivation, fails to improve on vanilla PPO. This reading is the strongest connection to the main paper's central finding.

We do not claim these preliminary results refute understanding-based curiosity as a transfer mechanism; the implementations were not exhaustively tuned, the source–target pair was a single choice, and the seed counts ($n = 3$, with collapse stripping further reducing n) limit statistical power. We report them because they *constrain* the space of plausible transfer mechanisms: at minimum, the simplest

implementations of three understanding-based methods do not, in our setting, improve on vanilla. Combined with our coverage-based findings (Table 3, Table 6, and the 14-pair Appendix A survey), the picture is one in which the IMPALA + PPO baseline is a strong default that intrinsic methods of either flavour rarely beat—and sometimes degrade.

Methodological notes. Source training: 25M steps, 64 parallel environments, $\eta = 0.005$, 200 levels, easy mode. Same backbone (IMPALA CNN) and optimiser as the main paper. Cutoff variants: intrinsic reward and intrinsic-model gradient frozen at 10M steps; PPO continues on extrinsic-only reward to 25M. Encoder-collapse stripping: seeds with effective rank < 5 by step 12.5M (halfway through training) excluded from method-level means. Settings differ from main-paper experiments in two respects: bfloat16 mixed precision (vs. fp16 in some main-paper sets) and minibatch 8 (vs. minibatch 4 in some main-paper sets); these differences may contribute to the gap with main-paper vanilla numbers and constitute a known confound in cross-set comparison.

C Hard-Mode vs. Easy-Mode Cross-Environment RND Comparison

Table 15 provides a side-by-side view of cross-environment RND transfer win rates under `hard` mode (the original sweep underlying the representation-quality analysis in Section 6.8) and `easy` mode (the supplementary win-rate sweep reported in Section 6.8.1). The two sweeps differ in η range and distribution mode and use different fine-tuning step budgets; we therefore present this table as configuration-sensitivity evidence rather than as a controlled cross-mode comparison.

Table 15: Cross-environment RND transfer win rates under `hard` vs. `easy` distribution mode (target: Heist; 3 seeds per cell; cutoff at 10M of 25M steps). Hard-mode numbers come from the original cross-environment sweep underlying Table 4; easy-mode numbers are aggregated from the supplementary sweep in Section 6.8.1 ($\eta \in \{0.001, 0.01, 0.05\}$, N=9 per cell). Cutoff outperforms always-on in every cell of both modes.

		Hard mode		Easy mode	
Source	**Variant**	**ZS**	**FH**	**ZS**	**FH**
Maze	Always-on	11.9%	33.8%	2.2%	4.9%
	Cutoff	**14.0%**	**37.3%**	**2.6%**	**6.5%**
Heist	Always-on	14.1%	32.3%	2.8%	5.6%
	Cutoff	**17.5%**	**38.4%**	**7.9%**	**10.1%**
Miner	Always-on	1.7%	4.3%	1.3%	4.4%
	Cutoff	**2.9%**	**5.0%**	**2.8%**	**4.9%**

Two patterns are visible. First, the directional cutoff effect—cutoff $>$ always-on—holds in every cell of both modes, replicating the main finding of Section 6.8 under independent settings. Second, absolute win rates differ markedly between modes for Maze and Heist sources (e.g. Maze ZS 11.9% hard vs. 2.2% easy; Heist ZS 14.1% hard vs. 2.8% easy), while Miner source shows near-parity across modes. The mode-mismatch is largest for navigation-style sources, suggesting that the difficulty regime interacts with source-game structure to determine absolute transfer magnitude—but not direction. As discussed in Section 8, limitation 6, fully isolating distribution mode from η range and protocol differences would require a factorial sweep we did not run.