Robustness of Deep Q-Networks to Visual Noise: A Brain-Inspired Evaluation on Atari Environments

Saba Noorghasemi
Department of Computer Science
Saarland University
Campus, 66123 Saarbrücken, Germany
sano00004@stud.uni-saarland.de

Abstract

The human perceptual system is remarkably effective at decision making under noisy and incomplete sensory input. Motivated by the representational findings of Cross et al. (2021), we evaluate the robustness of modern Deep Q-Network (DQN) agents to realistic visual corruptions in the Arcade Learning Environment (ALE). Using pretrained Stable-Baselines3 DQNs for Breakout, Space Invaders, and Enduro, we inject six noise types (gaussian, salt&pepper, occlusion, blur, frame-drop, pixelation) at six severity levels (0–5) *after* Atari preprocessing and frame stacking, and run 10 evaluation episodes per condition. We report mean return, standard deviation, and percent drop relative to a clean baseline. Across games, global distortions (pixelation, blur) are most destructive; partial corruptions (occlusion) and temporal loss (frame-drop) are less harmful. Enduro exhibits the greatest robustness with near-linear performance degradation as severity increases; Breakout collapses under nearly all noise; Space Invaders is intermediate but highly sensitive to pixelation/blur. These results highlight substantial gaps between biological nuisance-invariance and the visual robustness of standard deep RL policies.

1 Introduction

Perception in the real world is imperfect: humans routinely act under sensory noise, missing information, and occlusions. Recent work has shown striking correspondences between deep reinforcement learning (RL) representations and neural activity measured with fMRI during Atari gameplay Cross et al. (2021). However, the *robustness* of deep RL policies to perceptual corruptions remains underexplored.

We revisit Atari DQNs through the lens of brain-inspired robustness. Specifically, we assess whether vision-based DQN agents—used as computational proxies in Cross et al. (2021)—exhibit performance stability under corruptions that mimic noisy perception (e.g., blur, pixelation, occlusion). We evaluate three ALE games with pretrained Stable-Baselines3 (SB3) DQNs and quantify the effect of six corruption types across six severities. Our findings reveal consistent failure modes under global distortions and point to a gap between biological nuisance-invariance and standard deep RL policies.

Contributions. (i) A systematic, controlled robustness evaluation of pretrained SB3 DQNs across six perceptual corruptions and six severities on three Atari games. (ii) Empirical evidence that global spatial distortions (pixelation, blur) are universally destructive, while occlusion and frame-drop are comparatively benign. (iii) Cross-game analysis showing Enduro $>$ Space Invaders $>$ Breakout in robustness, with interpretable patterns versus severity.

Preprint. Under review.

2 Related Work

Deep Q-Networks achieve human-level control on Atari (Mnih et al., 2015). Cross-disciplinary work has connected DQN internal representations to human behavior and brain activity (Cross et al., 2021). Meanwhile, robust representation learning and domain randomization have improved transfer and invariance in RL (Higgins et al., 2017; Srinivas et al., 2020). Our focus is complementary: we quantify robustness of off-the-shelf vision-based DQNs under explicit test-time corruptions, without retraining.

3 Methods

Environments and agents. We evaluate SB3 DQNs (CNN policies, pretrained and fetched from Hugging Face) for: `BreakoutNoFrameskip-v4`, `EnduroNoFrameskip-v4`, `SpaceInvadersNoFrameskip-v4`. Model identifiers:

- `sb3/dqn-BreakoutNoFrameskip-v4`
- `sb3/dqn-EnduroNoFrameskip-v4`
- `sb3/dqn-SpaceInvadersNoFrameskip-v4`

Agents are used *as-is* (no fine-tuning).

Observation pipeline. We follow the standard ALE pipeline: `AtariPreprocessing` (grayscale, 84×84, frame-skip 4, life loss as terminal) and `FrameStack(4)`, then apply corruptions to the stacked observation. This matches the training-time input format and isolates corruption effects at evaluation.

Corruption types and severities. We consider six corruption types: **gaussian**, **saltpepper**, **occlusion**, **blur**, **framedrop**, **pixelation**. Each is evaluated at integer severities $\{0, 1, 2, 3, 4, 5\}$ (where 0 is clean). Exact parameterizations are in Appendix A.

Evaluation protocol. For each (game, noise, severity) triple, we run 10 episodes with fixed seeds. Metrics: *mean episodic return*, *standard deviation*, and *percent drop* w.r.t. the clean baseline (severity 0).

4 Results

We summarize cross-game patterns, then game-specific behavior. All figures are generated from the CSV outputs of the evaluation suite.

4.1 Cross-game patterns

Ordering by destructiveness (percent drop; highest to lowest).

- **Breakout:** Pixelation, Blur, Salt&Pepper, Gaussian $>$ Occlusion $>$ Frame-drop.
- **Space Invaders:** Pixelation $>$ Blur $>$ Salt&Pepper $>$ Gaussian $>$ Occlusion $>$ Frame-drop.
- **Enduro:** Pixelation $>$ (Blur, Salt&Pepper) $>$ Occlusion $>$ Frame-drop $>$ Gaussian.

Key insights.

1. **Blur is more harmful than expected.** Large drops across all games (comparable to pixelation), suggesting poor tolerance to global low-pass distortions.
2. **Occlusion is milder than global distortions.** Partial visibility preserves sufficient context; Space Invaders even shows a mid-severity performance spike.
3. **Frame-drop is surprisingly benign.** Agents often interpolate temporally; mean returns remain relatively high, though variance can be large (instability).

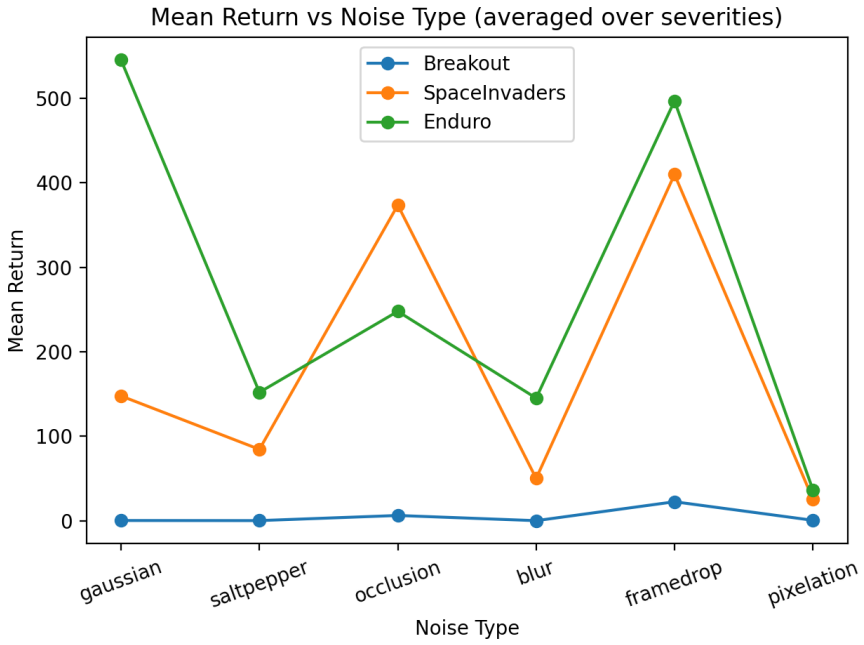
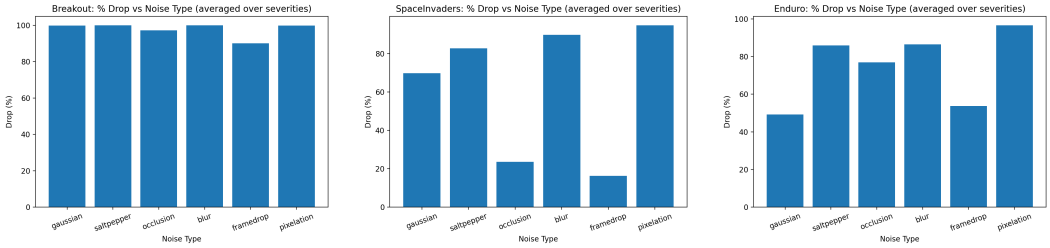


Figure 1: Mean return per noise type (averaged across severities). Enduro maintains the highest returns overall; Breakout collapses; Space Invaders is intermediate with large drops for pixelation/blur.



(a) Breakout (b) Space Invaders (c) Enduro

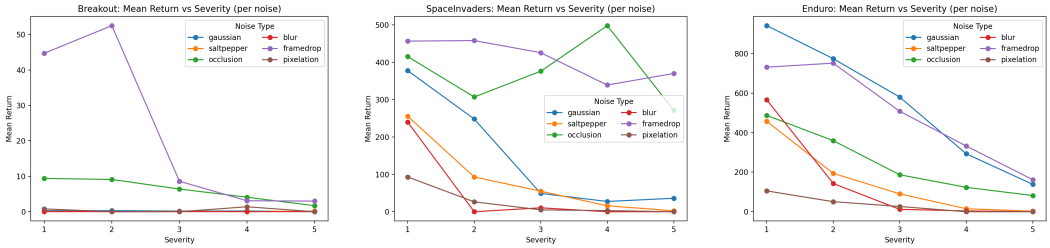
Figure 2: Percent drop in return by noise type (averaged across severities).

4. **Gaussian < pixelation/blur in harm.** Additive noise perturbs detail but preserves global structure; CNN features remain partially useful.
5. **Pixelation is universally catastrophic and consistent.** Coarse downsampling removes spatial cues, yielding low mean and low variance (stable failure).

4.2 Per-game behavior and variance

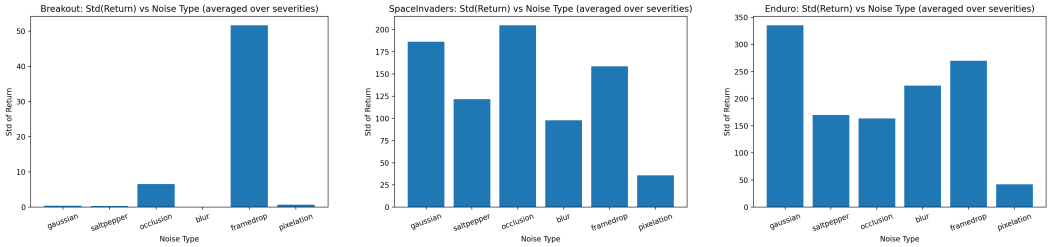
Breakout (least robust). Mean returns are near-zero across most corruptions; even the best case (frame-drop) reaches only modest scores. Standard deviation is near-zero (consistently failing) except for frame-drop, where high variance indicates occasional recoveries but unstable behavior. Overall: sharp collapse even at low severities.

Space Invaders (moderately robust, inconsistent). Performs relatively well under frame-drop and occlusion; collapses under pixelation and blur. Variance is high for gaussian, occlusion, and frame-drop (run-to-run instability); low for pixelation (consistent failure).



(a) Breakout (b) Space Invaders (c) Enduro

Figure 3: Mean return vs. severity (aggregated across noise types). Enduro degrades nearly linearly; Space Invaders is mostly linear but shows a non-monotonic occlusion spike at severities 3–4; Breakout collapses early.



(a) Breakout (b) Space Invaders (c) Enduro

Figure 4: Standard deviation of return vs. noise type (averaged over severity). High variance under gaussian/occlusion/frame-drop indicates unstable partial robustness; low variance under pixelation indicates consistent failure.

Enduro (most robust). Highest mean returns across noises; degrades smoothly with severity (nearly linear). Gaussian and frame-drop are best tolerated; pixelation causes stable failure (low mean, low variance). High variance under gaussian/blur/frame-drop indicates partial adaptability.

5 Discussion

Our results align with neuroscience observations that effective state-space representations should be *nuisance-invariant* (Cross et al., 2021). Standard DQNs trained on clean pixels do not learn invariances to global distortions (blur, pixelation); instead, they rely heavily on precise spatial detail. In contrast, occlusion—which preserves some spatial layout—is often manageable, and temporal continuity helps mitigate frame-drop.

Why is blur so destructive? Low-pass filtering removes edges and degrades motion/temporal energy that CNNs implicitly rely on for control. Without explicit robustness objectives or augmentation, features fail to generalize.

Limitations. We evaluate pretrained models only (no retraining or data augmentation), three games, and 10 episodes per condition. We do not compare to human behavioral data.

Future work. (i) Train with corruption augmentations and domain randomization; (ii) contrastive state-representation learning for invariances; (iii) recurrent or belief-state agents; (iv) robustness metrics beyond return (e.g., action disagreement).

6 Conclusion

Pretrained Atari DQNs are fragile to perceptual corruptions, especially global spatial distortions. Enduro is comparatively robust with smooth degradation; Breakout collapses even at mild noise;

Space Invaders is intermediate but highly sensitive to pixelation/blur. Bridging the gap to biological robustness likely requires explicit nuisance-invariance in learned state representations.

Artifacts. Code, pretrained weights, CSVs, and plots will be released at: `https://github.com/SabaNrg/Atari-Project`.

A Implementation details

A.1 Observation pipeline and noise placement

We construct the environment with `AtariPreprocessing` and `FrameStack(4)`, then apply a chosen corruption to the stacked observation (postprocessing). This preserves the input format the DQN was trained on.

A.2 Corruption parameterization (by severity)

We use discrete severities $s \in \{0, 1, 2, 3, 4, 5\}$; s=0 is clean. Example mappings (tune to match your code release):

- **Gaussian:** $\sigma \in \{0.0,\ 0.05,\ 0.10,\ 0.15,\ 0.20,\ 0.25\}$ (on normalized $[0, 1]$ tensors).
- **Salt&Pepper:** corruption probability $p \in \{0.0,\ 0.01,\ 0.02,\ 0.04,\ 0.08,\ 0.12\}$.
- **Blur:** kernel radius or σ increasing with s (e.g., Gaussian blur $\sigma \in \{0, 1, 2, 3, 4, 5\}$).
- **Pixelation:** downsample to $\{84,\ 56,\ 42,\ 28,\ 21,\ 14\}$ then upsample to 84×84.
- **Occlusion:** number/size of random rectangles grows with s (e.g., area ratio $\{0,\ 0.05,\ 0.10,\ 0.15,\ 0.20,\ 0.30\}$).
- **Frame-drop:** replace every k-th frame with previous; k decreases with s (more drops).

Exact values will be documented in the released code.

B Pseudocode

B.1 Fetching pretrained SB3 models (`fetch_sb3_models.py`)

Algorithm 1 Fetch SB3 models from Hugging Face

Input: model map $\mathcal{M}$: env-id $\rightarrow$ (repo-id, filename)
for each (env-id, (repo, fname)) $\in \mathcal{M}$ **do**
    use HF Hub API to download artifact `fname` from `repo`
    save to local path (e.g., `models/env-id/`)
end for
Output: local directory with pretrained `.zip` files

B.2 Evaluation with corruptions (`run_model.py`)

Algorithm 2 Evaluate pretrained DQN under noise

Input: env-id, model path, noise type set $\mathcal{N}$, severities $\mathcal{S} = \{0, \ldots, 5\}$, episodes E=10, seed
build base env with `AtariPreprocessing`; apply `FrameStack(4)`
for each noise $n \in \mathcal{N}$ **do**
  for each severity $s \in \mathcal{S}$ **do**
    wrap env with corruption (n, s) *after* frame stack
    load pretrained SB3 DQN; set seeds; reset env
    for episode $= 1$ to E:
      run greedy policy (or deterministic eval) until terminal
      accumulate reward
    compute mean return, std, and percent drop vs s=0
    append one row to CSV: (env-id, model, noise, severity, seed, mean, std, percent_drop)
  end for
end for
Output: a per-env CSV (or multiple CSVs) with all conditions

B.3 Plotting (`plots.py`)

Algorithm 3 Aggregate CSVs and render figures

Input: CSV files per env; output directory
load and concatenate tables; compute grouped statistics
render: (i) mean vs noise (avg over severities), (ii) percent drop vs noise, (iii) mean vs severity per env, (iv) std vs noise
save figures as: `plot1_mean_vs_noise_all_sev.png`, `plot2_{Game}_mean_vs_severity_all_noise.png`, `plot3_{Game}_percent_drop_all_sev.png`, `plot4_{Game}_std_vs_noise_all_sev.png`

C CSV schema and data availability

Each row records: `env_id`, `model_id`, `noise_type`, `severity`, `seed`, `episode_returns` (optional), `mean_return`, `std_return`, `percent_drop`. We provide one CSV per environment (31 tables per model were used to generate Figures 1–4). Full tables and code are linked in the project repository.

D Additional qualitative observations

- **Non-monotonicity under occlusion (Space Invaders).** A spike at severities 3–4 suggests exploitable regularities in mask placement or agent behavior.
- **Variance as robustness signal.** High std for gaussian/occlusion/frame-drop reflects partial, unstable compensation; low std for pixelation reflects uniform failure.

Acknowledgments

- We thank Cross et al. for releasing insights and inspiration on brain–DQN correspondences.
- The author used OpenAI's ChatGPT to refine the text for grammar and clarity; all content and conclusions were generated and verified by the author.

References

Logan Cross, Jeff Cockburn, Yisong Yue, and John P. O'Doherty. Using deep reinforcement learning to reveal how the brain encodes abstract state-space representations in high-dimensional environments. *Neuron*, 109(4):724–738, 2021.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Marc G. Veness, et al. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015.

Irina Higgins, Arka Pal, Andrei A. Rusu, et al. DARLA: Improving zero-shot transfer in reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

Aravind Srinivas, Michael Laskin, and Pieter Abbeel. CURL: Contrastive unsupervised representations for reinforcement learning. In *Proceedings of ICML*, 2020.