Steering Robustness into World Action Models via Mechanistic Interpretability and Optimal Control

Julian Skifstad*, Jihoon Hong*, Qiyue Dai, Alice Chan, and Glen Chou
Georgia Institute of Technology, Atlanta, GA.
Email: {jskifstad3, jhong392, qdai41, ichan30, chou}@gatech.edu
*Equal contribution.

***Abstract*—World Action Models (WAMs) enable semantically- and physically-informed control but are brittle under distribution shift. In this work, we study how robustness-relevant perturbations are represented in WAM activation space, and show that some WAM architectures encode robustness-critical features as low-dimensional, approximately linear directions. Building on this, we introduce World-Action Linear Quadratic Regulator (WA-LQR), a minimally-invasive reduced-order LQR controller with online feedback leveraging empirically validated local linearity in WAM activation dynamics. Through mechanistic analysis, we predict strong steerability in Cosmos-Policy and DiT4DiT but weak steerability in LingBot-VA, which is validated in steering evaluations. On Cosmos-Policy and DiT4DiT, WA-LQR generalizes contrastive directions to new tasks and improves robustness to camera, gripper, and visual-noise perturbations over unsteered and prompt steering baselines.**

I. Introduction

Foundation models have advanced robot learning through policies that generalize across objects, tasks, and environments. While VLA models map observations and instructions directly to actions, World Action Models (WAMs) additionally couple action prediction with future-state modeling through video-generative backbones [17, 18, 42, 41, 44]. This makes WAMs a promising route toward coherent robot policies, but they remain brittle under out-of-distribution shifts such as camera changes, robot initial-state perturbations, and visual corruption [44]. Since these nuisance factors are unavoidable in deployment, improving robustness without exhaustive new data or slow retraining is essential.

In this paper, we ask whether the internal activations of WAMs reveal why they fail under such perturbations, and whether these representations can be used to improve robustness without finetuning. We study perturbations to camera position, initial gripper position, and Gaussian image noise. Using mechanistic interpretability (MI), we compare activations from nominal and perturbed rollouts to test for simple geometric structure in activation space. We find that this structure is architecture-dependent: Cosmos-Policy [18] and DiT4DiT [24] exhibit low-dimensional linear separability for several perturbations, while LingBot-VA [20] shows substantially weaker separability. This suggests that some WAMs contain steerable representations of robustness-critical features, while others may not.

These observations motivate training-free WAM steering, where contrastive examples define latent directions that separate nominal from perturbed behavior. We use these directions for open-loop and closed-loop robustness interventions. In particular, we introduce *World-Action Linear Quadratic Regulator* (WA-LQR), a reduced-order optimal-control method that projects activations into a low-dimensional contrastive subspace and uses local linear dynamics to synthesize a closed-loop LQR steering controller. Unlike open-loop activation addition, WA-LQR adapts online, steering only when activations deviate from the target feature strength while penalizing large perturbations. We evaluate our mechanistic predictions and steering method on Cosmos-Policy, DiT4DiT, and LingBot-VA. Our analysis predicts strong steerability for Cosmos-Policy and DiT4DiT, and weak steerability for LingBot-VA, which intervention results validate. On Cosmos-Policy (and DiT4DiT, to a lesser degree), contrastive directions transfer across LIBERO [22] tasks, and WA-LQR improves robustness to camera, gripper, and visual-noise perturbations over non-steered, prompt-steering, and open-loop steering baselines. These results show that MI can diagnose WAM robustness and guide inference-time interventions. Our contributions are:

- We conduct a mechanistic analysis of WAM activations under robustness-relevant perturbations, identifying features that exhibit low-dimensional linear separability.
- We use these insights to construct contrastive activation directions for training-free WAM steering. We also show that local linearity in reduced WAM activation space enables efficient synthesis of closed-loop steering controllers via WA-LQR.
- We evaluate WA-LQR on robustness benchmarks, showing that MI-based steerability predictions align with intervention outcomes and that WA-LQR improves robustness across multiple perturbation types.

II. Related Work

WAMs: Recently, WAMs emerged as a promising framework for robot learning by coupling action generation with future-state prediction, often through adapting video generative backbones to robotic data [42, 18, 41, 38]. Existing WAMs include inverse-dynamics-style models, where predicted futures are decoded into actions [4, 40, 13, 15, 21], and unified video-action models, where future states and actions are generated together [18, 42, 41]. We study both families in this work.

Robustness of Action Models: Despite progress in general-purpose robot policies, robustness remains a major deployment obstacle. WAMs aim to improve robustness with

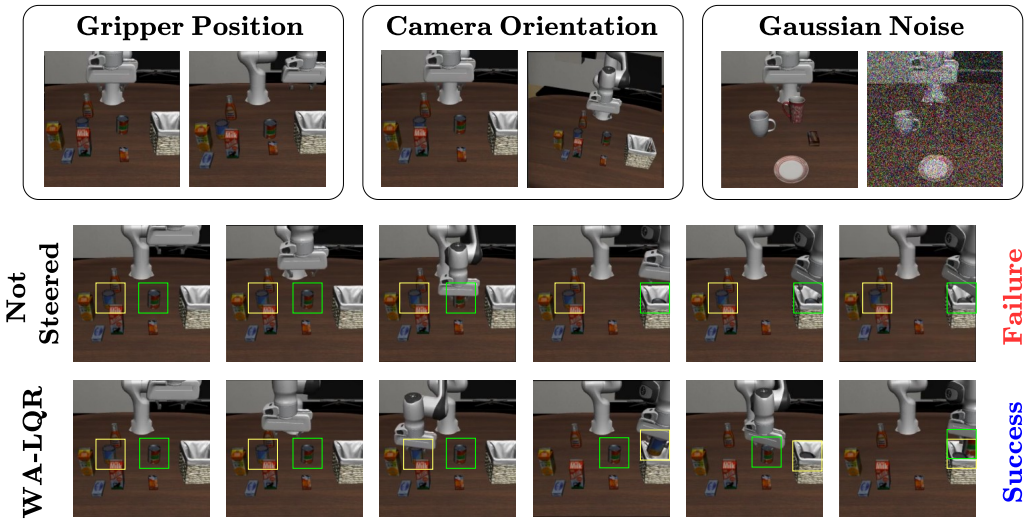

Fig. 1. WA-LQR enhances robustness of World Action Models to perturbations such as changes in gripper position, camera orientation, or corruption of sensors with Gaussian noise.

future-state modeling and video-based priors [13, 18, 21, 42, 41, 43], but still fail under shifts such as camera viewpoint and robot initial-state changes [44]. Prior studies with VLAs suggest that robustness gains often require data diversity, RL post-training, or robustness-oriented fine-tuning, but acquiring vast robotics training data remains costly [45, 23, 36, 8]. This motivates inference-time methods that improve robustness without exhaustive perturbation data or costly retraining.

MI: MI seeks internal representations that mediate model behavior [2, 33]. Relatively little work applies MI to robotics foundation models, where existing studies identify steerable activation directions [10, 35], formalize feature observability and controllability [3], or finetune task-relevant attention heads [27] in VLAs.

Activation Steering: Activation steering refers to inference-time hidden-state modifications that change behavior without retraining [6, 29, 26, 46, 19]. Recent work models activations as dynamical systems and uses feedback for more targeted steering, with applications in LLMs, ViTs, and T2V models [28, 34, 32, 7, 5, 12] Here, we study WAMs, whose DiT-style video backbones jointly encode future visual states, action dynamics, and control outputs. We show that WAM steerability is architecture-dependent and develop feedback-based steering that treats WAM inference as a reduced-order dynamical system.

III. Preliminaries and Problem Setup

Linear Quadratic Regulator (LQR): LQR [11] constructs a controller minimizing a quadratic state-control cost (1a) for linear time-varying dynamics (1b) Given a set of nominal setpoints $\{(\bar{z}_k, \bar{u}_k)\}_{k=1,\cdots,H}$, LQR can be formulated to penalize deviations $\delta z_k := z_k - \bar{z}_k$ and $\delta u_k := u_k - \bar{u}_k$ from the setpoints. The resulting optimization can be formulated as

$$\min_{\{\delta u_k\}_{k=1}^{H-1}} \sum_{k=1}^{H-1} \left(\delta z_k^\top Q_k \delta z_k + \delta u_k^\top R_k \delta u_k\right) + \delta z_H^\top Q_H \delta z_H \tag{1a}$$

$$\text{s.t.} \quad \delta z_{k+1} = A_k \delta z_k + B_k \delta u_k, \quad k = 1, \ldots, H-1, \tag{1b}$$

where $Q_k \succeq 0$ penalizes state error and $R_k \succ 0$ penalizes control effort. The optimal policy admits a closed form tracking controller $u_k^* := \bar{u}_k - K_k \delta z_k$.

WAM Architectures: While the two types of WAMs vary in architecture, they both build on Diffusion Transformer (DiT) based video generation models [30]. Starting from a random latent action representation $\hat{x}_T \sim \mathcal{N}(0, I)$, these models are used to gradually denoise it to a clean x_{out} by

$$\hat{x}_{t-1} = \texttt{STEP}(\hat{x}_t, t, M(\hat{x}_t, t, h)), \qquad x_{\text{out}} = \texttt{DEC}(\hat{x}_1), \tag{2}$$

where h is the embedding of the task prompt p, M is the model, STEP is a choice of ODE solver, and DEC decodes latent actions. Typically, M is a sequence of L DiT layers $\phi^{(l)}$ for $l = 0, \cdots, L-1$:

$$\begin{aligned} x_{0,t} := \texttt{Tok}(\hat{x}_t), \qquad x_{l+1,t} = \phi^{(l)}(x_{l,t}, t, h), \\ M(\hat{x}_t, t, h) := \texttt{Detok}(x_{L,t}). \end{aligned} \tag{3}$$

After block L, the WAM output is detokenized and passed to the model scheduler (2). In this work, we perform activation steering by adding control inputs in the DiT block residual stream. Let $u_{l,t}$ denote these perturbations. The steered dynamics are

$$x_{l+1,t} = f_{l,t}(x_{l,t}, u_{l,t}) := \phi^{(l)}(x_{l,t}, t, h) + u_{l,t}. \tag{4}$$

Problem Setup: We seek to identify robustness-relevant features (ξ) in WAM activation space and use these features for training-free robustness improvement via activation steering.

Task 1: Feature discovery in WAM activation space (Sec. IV). Given a WAM, for each layer l, we seek linear directions which separate the low-dimensional projection of contrastive activation pairs (ξ^+, ξ^-).

Task 2: Training-free robustness steering (Sec. V). Using the discovered features, construct an inference-time intervention that modifies activations to suppress nuisance-related features and thereby improve robustness.

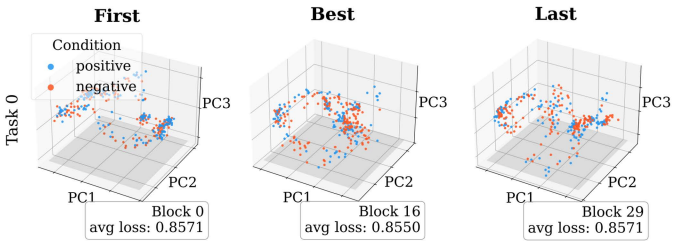

Fig. 2. Activations from camera-perturbed and clean LingBot-VA inputs at the first, best intermediate, and final transformer block residual streams. We observe much weaker separability than in the Cosmos-Policy setting.

IV. A Mechanistic Study for Interpreting WAMs

A central assumption in the activation steering literature is that model activation spaces exhibit interpretable geometric structure, across domains including LLMs, VLAs, and video generation models [37, 31, 34, 12, 10]. Specifically, most steering algorithms assume linear semantic feature directions, as determined by the emergent linear separability of latent activations corresponding to semantically contrastive inputs [1, 25]. However, it remains unclear whether analogous geometric structure exists in WAMs, despite their reliance on steerable foundation-model backbones [10, 14].

a) Setup: We study the emergent geometry in robustness features for WAMs, specifically Cosmos-Policy 2B, DiT4DiT, and LingBot-VA [18, 24, 20]. We consider contrastive datasets related to three key sources of sensitivity in WAM manipulation tasks: perturbation to initial gripper position, initial camera position, and corruption of camera inputs with Gaussian noise. We collect activations corresponding to each dataset through a full model forward pass. That is, for prompts $p_+ \in \mathcal{D}_+$, $p_- \in \mathcal{D}_-$ where, e.g., $\mathcal{D}_+ = \{\texttt{Clean inputs}\}$, $\mathcal{D}_- = \{\texttt{Noised inputs}\}$, we collect DiT activations $x_{t,\ell}^{p_+}$ and $x_{t,\ell}^{p_-}$ for all t, ℓ. To ensure computational feasibility, we consider an average pooling over the token position, resulting in a summarized activation $\bar{x} \in \mathbb{R}^D$ for all samples.

Given contrastive sets of activations $\{\bar{x}_i^+\}_i$ and $\{\bar{x}_j^-\}_j$, we define a *contrastive direction* as $d_k = \bar{x}_k^+ - \bar{x}_k^-$. Inspired by Marks and Tegmark [25], we consider a simple geometric evaluation of the separability of contrastive datasets via principal component analysis (PCA) conducted on the set of contrastive directions, $\{d_k\}_k$. With the objective of enabling linear feature-based steering, we seek to identify linear separability of contrastive datasets. Note that in the high dimensionality D of the original mean-pooled system, N samples with $N \ll D$ are trivially linearly separable, rendering analysis in the full dimension system uninformative. Hence, we consider a low-dimensional approximation.

Projecting the contrastive activations $\bar{x}^+$ and $\bar{x}^-$ onto the top 3 PC's, we fit a linear support vector machine (SVM) to the contrastive activations in three dimensions, and measure the classification loss as the average hinge loss per sample. Intuitively, this measures how well the 3-dimensional linear SVM distinguishes the two datasets, where perfect classification yields a loss of 0, and random classification yields a loss of 1 (see App. XV-A for details).

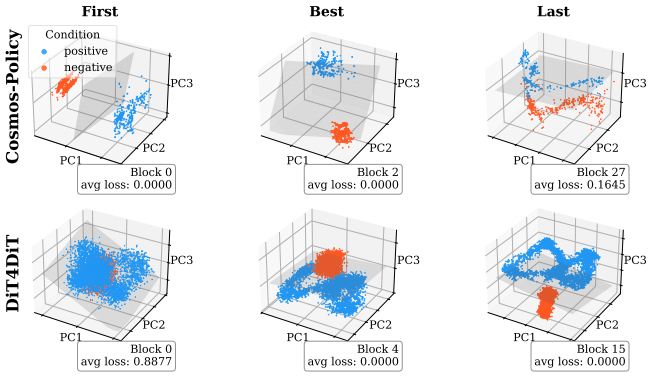

Fig. 3. Activations corresponding to *noise-perturbed* and clean inputs on the first, best intermediate, and final DiT block residual stream for Cosmos-Policy and DiT4DiT on LIBERO-10 task 0. Activations are projected onto the top 3 principal components, with the reported SVM (gray) and hinge loss.

b) Results: We perform this analysis across the perturbations for all 10 tasks of the LIBERO-10 dataset [22], and visualize the results Fig. 2-3 (further evaluations are provided in Ap. XVI). Notably, we find that the separability of different features is highly task and model-dependent, indicating that the latent representation corresponding to the same perturbation is not always shared across different scenes and tasks. A key observation of this study is that *there exist combinations of tasks which share feature representations* corresponding to the same input perturbation (see App. XVI).

V. Activation Steering for Robustifying WAMs

Motivated by the MI results of Sec. IV, we overview our steering method. We first construct contrastive vectors isolating the desired robustness feature and discuss a simple open-loop means of steering WAMs with the contrastive vectors (Sec. V-A). We also propose a reduced-order LQR-based steering approach (Sec. V-B) to enable scalable control-theoretic steering for WAMs.

A. Open-Loop Activation Addition from Contrastive Vectors

We first present the simplest contrastive steering method: *activation addition* (ActAdd) [37]. A steering vector is formed by averaging contrastive directions (App. X) over pairs and action-chunk positions. During inference, ActAdd steers by adding $a_{l,t}$ with strength γ:

$$x_{l,t,\tau} \leftarrow x_{l,t,\tau} + \gamma a_{l,t}, \qquad \forall \tau \in [H_a]. \tag{5}$$

The same direction is applied across all action-chunk timesteps, with γ setting steering strength. ActAdd is training-free and weight-preserving, but open-loop: it ignores the current activation, WAM dynamics, and whether the feature is already at the desired strength, which can cause oversteering and action degradation.

B. Steering Latent World Activation Dynamics via WA-LQR

WA-LQR generalizes ActAdd with feedback control, adapting the T2V steering method of [12] to WAMs. Instead of adding a fixed $\gamma a_{l,t}$, it projects contrastive vectors into a latent feature space, measures deviation from a latent setpoint,

and computes a minimum-cost LQR intervention. Thus, it preserves ActAdd's training-free nature while adapting interventions to the realized WAM activation.

a) Dimensionality Reduction: As in T2V models [12], full activation-space LQR is infeasible due to the high dimensionality of WAM activations. We therefore assume (and validate in App. VI) that robustness-relevant variation is concentrated in a low-dimensional contrastive subspace. For each layer-denoising pair (l, t), we stack contrastive directions across prompt pairs and action-chunk timesteps $\tau \in [H_a]$, and apply streaming randomized SVD [9] to obtain a projection matrix $P_{l,t} \in \mathbb{R}^{D_z \times D_x}$, where $D_z \ll D_x$. We then define the low-dimensional latent activation

$$z_{l,t,\tau} := P_{l,t} x_{l,t,\tau} \in \mathbb{R}^{D_z}. \tag{6}$$

We model the dynamics in this low dimensional space as

$$\delta z_{l+1,t,\tau} \approx \widetilde{A}_{l,t} \delta z_{l,t,\tau} + \widetilde{B}_{l,t} \delta u_{l,t,\tau}, \qquad l = 0, \ldots, L-1, \tag{7}$$

where $\widetilde{A}_{l,t}, \widetilde{B}_{l,t}$ are low dimensional projections of the Jacobians of the controlled block dynamics in (4) evaluated along a nominal trajectory, $A_{l,t} := \frac{\partial f_{l,t}}{\partial x}\big|_{\bar{x}_{l,t}, \bar{u}_{l,t}}$, $B_{l,t} := \frac{\partial f_{l,t}}{\partial u}\big|_{\bar{x}_{l,t}, \bar{u}_{l,t}}$.

b) Defining Feature Setpoints: Within the latent space, we define feature setpoints for robustness-relevant WAM features. Let $e^z_{l,t}$ denote a chunk-averaged latent contrastive direction and define $v^z_{l,t} := e^z_{l,t} / \|e^z_{l,t}\|_2$. The *feature strength* for a realized latent activation $z_{l,t,\tau}$ is $\beta^z_{l,t,\tau} := (v^z_{l,t})^\top z_{l,t,\tau}$. We set the desired feature strength as $\beta^{z,*}_{l,t} := \lambda \|e^z_{l,t}\|_2$, where λ controls steering strength. The feature tracking error for action-chunk timestep τ is

$$\alpha_{l,t,\tau} := \beta^{z,*}_{l,t} - (v^z_{l,t})^\top z_{l,t,\tau}, \text{ with } \delta z_{l,t,\tau} := -\alpha_{l,t,\tau} v^z_{l,t}. \tag{8}$$

c) Reaching Feature Setpoints via WA-LQR: Using the latent dynamics in (7), we compute LQR controllers that steer WAM activations toward the feature setpoints. For each denoising timestep t, we solve an independent L-step LQR over the transformer blocks:

$$\begin{aligned} \min_{\{\delta u_{l,t}\}_{l=0}^{L-1}} & \sum_{l=0}^{L-1} \left(\bar{\delta z}_{l,t}^\top Q_{l,t} \bar{\delta z}_{l,t} + \delta u_{l,t}^\top R^{\text{chunk}}_{l,t} \delta u_{l,t} \right) \\ & \qquad + \bar{\delta z}_{L,t}^\top Q_{L,t} \bar{\delta z}_{L,t} \\ \text{s.t.} \quad & \bar{\delta z}_{l+1,t} = \widetilde{A}_{l,t} \bar{\delta z}_{l,t} + \widetilde{B}_{l,t} \delta u_{l,t}, \quad l = 0, \ldots, L-1. \end{aligned} \tag{9}$$

Here $\bar{\delta z}_{l,t}$ is the chunk-aggregated latent error $\bar{\delta z}_{l,t} := \frac{1}{H_a} \sum_{\tau=1}^{H_a} \delta z_{l,t,\tau}$. The control penalty incorporates an action-decay schedule over robot action chunk indices (see App. IX for details).

At inference time, WA-LQR adapts to the realized latent activation at each layer and denoising timestep:

$$x_{l,t,\tau} \leftarrow x_{l,t,\tau} + P_{l,t}^\top u^*_{l,t}, \qquad u^*_{l,t} := -K_{l,t} \bar{\delta z}_{l,t}. \tag{10}$$

VI. Results

We leverage the mechanistic insights from Sec. IV to inform steering objectives designed to improve robustness to OOD perturbations in Cosmos-Policy 2B [18], LingBot-VA [20], and DiT4DiT [24] using WA-LQR (Sec. V-B) and ActAdd [37], and compare to the original model. Here, we seek to improve success rate under perturbations to gripper position, applied to LIBERO-10 evaluation tasks [22]. Additional results, including different perturbations, are included in the appendix. Following the results from Sec. IV, we evaluate over a group of tasks which share perturbation representation per model.

TABLE I

LIBERO-10 [22] success rates on Cosmos-Policy 2B [18], DiT4DiT [24], and LingBot-VA [20] under various perturbation. Task $i \to j$ denotes all $P_{l,t}, \tilde{A}_{l,t}, \tilde{B}_{l,t}$ matrices, $e^z_{l,t}$ vectors are computed with task i, and used to steer task j.

	Tasks	No Steering	ActAdd	WA-LQR
Cosmos-Policy (Gripper pos.)	$1 \to 1$	46.7%±0.091	**66.7%±0.086**	56.7%±0.090
	$1 \to 2$	83.3%±0.068	80.0%±0.073	**100.0%±0.00**
	$1 \to 3$	46.7%±0.091	56.7%±0.090	**60.0%±0.089**
	$1 \to 7$	80.0%±0.073	60.0%±0.089	**83.3%±0.068**
	$1 \to 9$	50.0%±0.091	53.3%±0.091	**56.7%±0.090**
	Average	61.3%±0.040	63.3%±0.039	**71.3%±0.037**
DiT4DiT (Noise)	$1 \to 1$	20.0%±0.073	**63.3%±0.088**	43.3%±0.090
	$1 \to 2$	10.0%±0.055	26.7%±0.081	**30.0%±0.084**
	$1 \to 0$	16.7%±0.068	40.0%±0.089	**73.3%±0.081**
	Average	15.6%±0.038	43.3%±0.052	**48.9%±0.053**
LingBot-VA (Camera pos.)	$0 \to 0$	65.0%±0.107	35.0%±0.107	70.0%±0.103
	$0 \to 2$	55.0%±0.111	45.0%±0.111	45.0%±0.111
	$0 \to 4$	15.0%±0.080	10.0%±0.067	30.0%±0.103
	$0 \to 5$	70.0%±0.103	75.0%±0.097	60.0%±0.110
	$0 \to 9$	35.0%±0.107	40.0%±0.110	50.0%±0.112
	Average	48.0%±0.050	41.0%±0.049	51.0%±0.050

The steering results for all models and task groupings are summarized in Tab. I. On Cosmos-Policy and DiT4DiT, which exhibit stronger feature separation, both WA-LQR and ActAdd consistently improve upon the unsteered model, with WA-LQR generally achieving the best performance. In contrast, steering is less effective on LingBot-VA, where feature alignment is weaker. These findings suggest that the mechanistic insights developed in Sec. IV are predictive of steering effectiveness (see App. XIV).

VII. Discussion, Limitations, and Conclusion

We investigate mechanistic interpretability and activation steering in robotics World Action Models (WAMs). Our analysis reveals a clear linear structure in feature-relevant activations for OOD perturbations in Cosmos-Policy [18] and DiT4DiT, while LingBot-VA [20] lacks comparable structure. Motivated by this observation, we introduce WA-LQR, an activation steering framework that exploits locally linear dynamics within a feature-relevant activation subspace to improve OOD robustness without finetuning.

Limitations: A key limitation of WA-LQR is its limited generality. To our knowledge, there is currently no interpretable method for predicting when tasks or environments share transferable representations, requiring mechanistic analysis on a per-setting basis. We also observe strong dependence on model architecture. Nonetheless, our results highlight the importance of understanding how steerability emerges in robotics foundation models and suggest a promising direction for improving robustness through mechanistic control of hidden model representations.

References

[1] Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37: 136037–136083, 2024.

[2] Leonard Bereska and Stratis Gavves. Mechanistic interpretability for AI safety - a review. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=ePUVetPKu6. Survey Certification, Expert Certification.

[3] Hugo Buurmeijer, Carmen Amo Alonso, Aiden Swann, and Marco Pavone. Observing and controlling features in vision-language-action models. *arXiv preprint arXiv:2603.05487*, 2026.

[4] Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. *Advances in neural information processing systems*, 36:9156–9172, 2023.

[5] Yigit Ekin and Yossi Gandelsman. The unreasonable effectiveness of text embedding interpolation for continuous image steering. *arXiv preprint arXiv:2603.17998*, 2026.

[6] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. (arXiv:2209.10652), 2022. doi: 10.48550/arXiv.2209.10652. URL http://arxiv.org/abs/2209.10652. arXiv:2209.10652.

[7] Simone Facchiano, Stefano Saravalle, Matteo Migliarini, Edoardo De Matteis, Alessio Sampieri, Andrea Pilzer, Emanuele Rodolà, Indro Spinelli, Luca Franco, and Fabio Galasso. Video unlearning via low-rank refusal vector. *arXiv preprint arXiv:2506.07891*, 2025.

[8] Yanjiang Guo, Jianke Zhang, Xiaoyu Chen, Xiang Ji, Yen-Jen Wang, Yucheng Hu, and Jianyu Chen. Improving vision-language-action model with online reinforcement learning. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 15665–15672. IEEE, 2025.

[9] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

[10] Bear Häon, Kaylene Caswell Stocking, Ian Chuang, and Claire Tomlin. Mechanistic interpretability for steering vision-language-action models. In Joseph Lim, Shuran Song, and Hae-Won Park, editors, *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 2743–2762. PMLR, 27–30 Sep 2025. URL https://proceedings.mlr.press/v305/haon25a.html.

[11] Joao P Hespanha. *Linear systems theory*. Princeton university press, 2018.

[12] Jihoon Hong, Alice Chan, Qiyue Dai, Julian Skifstad, and Glen Chou. Activation steering of video generation models via reduced-order linear optimal control. 2026.

[13] Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video prediction policy: A generalist robot policy with predictive visual representations. *arXiv preprint arXiv:2412.14803*, 2024.

[14] Chengyue Huang, Mellon M. Zhang, Robert Azarcon, Glen Chou, and Zsolt Kira. Maps: Preserving vision-language representations via module-wise proximity scheduling for better vision-language-action generalization. *arXiv preprint arXiv:2511.19878*, November 2025. doi: 10.48550/arXiv.2511.19878. URL http://arxiv.org/abs/2511.19878.

[15] Yueru Jia, Jiaming Liu, Shengbang Liu, Rui Zhou, Wanhe Yu, Yuyang Yan, Xiaowei Chi, Yandong Guo, Boxin Shi, and Shanghang Zhang. Video2act: A dual-system video diffusion policy with robotic spatio-motional modeling. *arXiv preprint arXiv:2512.03044*, 2025.

[16] V. Kecman. *Support Vector Machines – An Introduction*, page 1–47. Springer, Berlin, Heidelberg, 2005. ISBN 9783540323846. doi: 10.1007/10984697_1. URL https://doi.org/10.1007/10984697_1.

[17] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[18] Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, et al. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026.

[19] Bruce W. Lee, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Erik Miehling, Pierre Dognin, Manish Nagireddy, and Amit Dhurandhar. Programming refusal with conditional activation steering. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Oi47wc10sm.

[20] Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, Yujun Shen, and Yinghao Xu. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026.

[21] Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, et al. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026.

[22] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances*

in Neural Information Processing Systems, 36:44776–44791, 2023.

[23] Jijia Liu, Feng Gao, Bingwen Wei, Xinlei Chen, Qingmin Liao, Yi Wu, Chao Yu, and Yu Wang. What can rl bring to vla generalization? an empirical study. *Advances in Neural Information Processing Systems*, 38:97121–97151, 2026.

[24] Teli Ma, Jia Zheng, Zifan Wang, Chunli Jiang, Andy Cui, Junwei Liang, and Shuo Yang. Dit4dit: Jointly modeling video dynamics and actions for generalizable robot control. *arXiv preprint arXiv:2603.10448*, 2026.

[25] Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*, August 2024. doi: 10.48550/arXiv.2310.06824. URL http://arxiv.org/abs/2310.06824.

[26] Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=aajyHYjjsk.

[27] Chancharik Mitra, Yusen Luo, Raj Saravanan, Dantong Niu, Anirudh Pai, Jesse Thomason, Trevor Darrell, Abrar Anwar, Deva Ramanan, and Roei Herzig. Mechanistic finetuning of vision-language-action models via few-shot demonstrations. *arXiv preprint arXiv:2511.22697*, 2025.

[28] Dung V. Nguyen, Hieu M. Vu, Nhi Y. Pham, Lei Zhang, and Tan M. Nguyen. Activation steering with a feedback controller. (arXiv:2510.04309), October 2025. doi: 10.48550/arXiv.2510.04309. URL http://arxiv.org/abs/2510.04309. arXiv:2510.04309.

[29] Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023.

[30] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[31] Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15504–15522, 2024.

[32] Pau Rodriguez, Michal Klein, Eleonora Gualdoni, Valentino Maiorca, Arno Blaas, Luca Zappella, Marco Cuturi, and Xavier Suau. Lineas: End-to-end learning of activation steering with a distributional loss. *arXiv preprint arXiv:2503.10679*, 2025.

[33] Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeff Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Bloom, et al. Open problems in mechanistic interpretability. *arXiv preprint arXiv:2501.16496*, 2025.

[34] Julian Skifstad, Xinyue Annie Yang, and Glen Chou. Local linearity of llms enables activation steering via model-based linear optimal control. *arXiv preprint arXiv:2604.19018*, 2026.

[35] Aiden Swann, Lachlain McGranahan, Hugo Buurmeijer, Monroe Kennedy III, and Mac Schwager. Sparse autoencoders reveal interpretable and steerable features in vla models. *arXiv preprint arXiv:2603.19183*, 2026.

[36] Shuhan Tan, Kairan Dou, Yue Zhao, and Philipp Krähenbühl. Interactive post-training for vision-language-action models. *arXiv preprint arXiv:2505.17016*, 2025.

[37] Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Ulisse Mini, and Monte MacDiarmid. Activation addition: Steering language models without optimization. 2024.

[38] Siyin Wang, Junhao Shi, Zhaoyang Fu, Xinzhe He, Feihong Liu, Chenchen Yang, Yikang Zhou, Zhaoye Fei, Jingjing Gong, Jinlan Fu, et al. World action models: The next frontier in embodied ai. *arXiv preprint arXiv:2605.12090*, 2026.

[39] Zixuan Wang, Yuxin Chen, Yuqi Liu, Jinhui Ye, Pengguang Chen, Changsheng Lu, Shu Liu, Bei Yu, and Jiaya Jia. Vp-vla: Visual prompting as an interface for vision-language-action models. *arXiv preprint arXiv:2603.22003*, May 2026. doi: 10.48550/arXiv.2603.22003. URL http://arxiv.org/abs/2603.22003. arXiv:2603.22003.

[40] Youpeng Wen, Junfan Lin, Yi Zhu, Jianhua Han, Hang Xu, Shen Zhao, and Xiaodan Liang. Vidman: Exploiting implicit dynamics from video diffusion model for effective robot manipulation. *Advances in Neural Information Processing Systems*, 37:41051–41075, 2024.

[41] Angen Ye, Boyuan Wang, Chaojun Ni, Guan Huang, Guosheng Zhao, Hao Li, Hengtao Li, Jie Li, Jindi Lv, Jingyu Liu, et al. Gigaworld-policy: An efficient action-centered world–action model. *arXiv preprint arXiv:2603.17240*, 2026.

[42] Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

[43] Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026.

[44] Zhanguang Zhang, Zhiyuan Li, Behnam Rahmati, Rui Heng Yang, Yintao Ma, Amir Rasouli, Sajjad Pakdamansavoji, Yangzheng Wu, Lingfeng Zhang, Tongtong Cao, et al. Do world action models generalize better than vlas? a robustness study. *arXiv preprint arXiv:2603.22078*, 2026.

[45] Jiaming Zhou, Ke Ye, Jiayi Liu, Teli Ma, Zifan Wang, Ronghe Qiu, Kun-Yu Lin, Zhilin Zhao, and Junwei Liang. Exploring the limits of vision-language-action manipulations in cross-task generalization, 2025. URL

https://arxiv.org/abs/2505.15660.

[46] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023.

Appendices

In the following, we provide an overview of our appendices. In App. VIII, we empirically verify the low-dimensional, locally linear approximation adopted in Sec. V. In Sec. IX, we provide additional details regarding our WA-LQR formulation. In App. X, we describe how contrastive vectors are constructed for Gaussian noise, camera perturbations, and gripper-position perturbations, including the definitions of the desirable and undesirable activation sets used for steering. In App. XII, we provide supplemental experimental results for LingBot-VA, including additional WA-LQR evaluations on camera orientation, initial gripper-position, and Gaussian-noise perturbations in both the action and video modules. In App. XIII, we provide supplemental experimental results on DiT4DiT, including a mechanistic analysis of feature separability and an evaluation of WA-LQR for improving robustness under Gaussian-noise and gripper-position perturbations. In App. XIV, we visualize the correlation between the mechanistic interpretability results and performance. In App. XV, we provide additional experimental details for the mechanistic study and for the Cosmos-Policy and LingBot-VA evaluations, including activation collection, dimensionality reduction, SVM fitting, and LQR steering implementation details. In App. XVI, we provide the complete low-dimensional separability results across all LIBERO-10 tasks for Cosmos-Policy and LingBot-VA, including both task-level and pairwise separability plots. In App. XVII, we provide full model separation plots across layers for Cosmos-Policy and DiT4DiT.

VIII. Activation Properties

We first empirically verify the assumptions that enable LQR-based steering of latent activations. Fig. 4 demonstrates that the variance of contrastive vectors projected to a 64-dimensional latent subspace, $P_{l,t}(x_{l,t,\tau}^{(\xi_n^+)} - x_{l,t,\tau}^{(\xi_n^-)})$, is mostly captured by the top few dimensions. This is caused by rapid decay of singular values, which suggests that the influence of perturbations such as the change in camera orientation on activations manifest in a significantly lower dimensional subspace compared to raw activations.

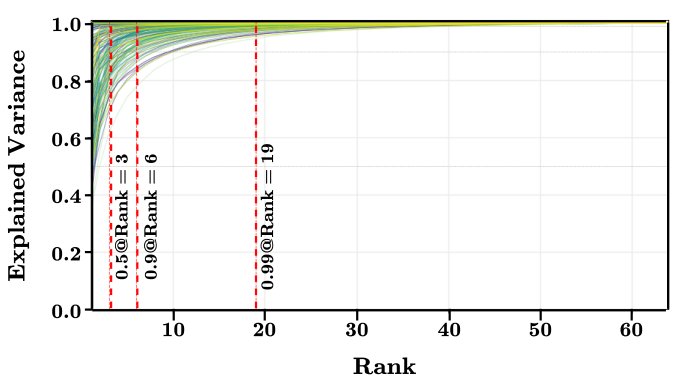

Fig. 4. Cumulative variance of latent contrastive vectors for camera orientation perturbation on Cosmos-Policy 2B [18] explained by the top-k singular vectors.

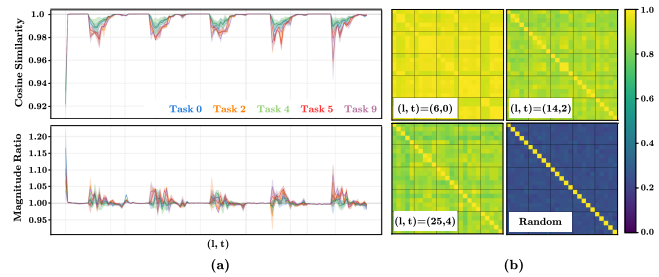

Fig. 5. (a) The cosine similarity and magnitude ratio between linear approximation using $\tilde{A}_{l,t}$ and actual latent activation, under change in camera orientation. (b) Overlap between subspaces spanned by 16 top right singular vectors of $\tilde{A}_{l,t}$ matrices, obtained from 25 random inputs across 5 tasks.

In addition, we observe that WAMs are surprisingly locally linear, which allows them to be effectively steered using LQR. Fig. 5 (a) shows the cosine similarity and the magnitude ratio between $P_{l+1,t}\phi^{(l)}\left(x_{l,t} + P_{l,t}^T\epsilon, t, h\right)$ and $P_{l+1,t}\phi^{(l)}\left(x_{l,t}, t, h\right) + \tilde{A}_{l,t}\epsilon$ for random perturbation ϵ whose norm is proportional to that of $\|x_{l,t}\|$. Both cosine similarities and magnitude ratio falling closely to 1 throughout the inference process demonstrates that the latent dynamics can be well represented by first order approximation. Furthermore, we find that the matrices $\tilde{A}_{l,t}$ are highly similar across different inputs. Motivated by the concentration of variance on the top few singular vectors, we visualize in Fig. 5 (b) the overlap between the subspaces spanned by the top 16 right singular vectors of $\tilde{A}_{l,t}$ over 25 random inputs from 5 different tasks, measured by the mean squared cosine of principal angles (as inspired by a similar metric in [34]). While not shown here, the left singular vectors also demonstrate significant overlap, which allows reuse of $\tilde{A}_{l,t}$ computed from a single input for steering model behavior on other tasks.

IX. Extended WA-LQR Details

For clarity, we restate the latent-space LQR problem presented in Sec. V. We first compute a chunk-averaged latent contrastive direction

$$e_{l,t}^z := \frac{1}{NH_a}\sum_{n=1}^{N}\sum_{\tau=1}^{H_a} P_{l,t}(x_{l,t,\tau}^{(\xi_n^+)} - x_{l,t,\tau}^{(\xi_n^-)}), \tag{11}$$

and define $v_{l,t}^z := \frac{e_{l,t}^z}{\|e_{l,t}^z\|_2}$. The resulting LQR problem is:

$$\begin{aligned} \min_{\{\delta u_{l,t}\}_{l=0}^{L-1}} & \sum_{l=0}^{L-1}\left(\bar{\delta z}_{l,t}^\top Q_{l,t}\bar{\delta z}_{l,t} + \delta u_{l,t}^\top R_{l,t}^{\text{chunk}}\delta u_{l,t}\right) \\ & + \bar{\delta z}_{L,t}^\top Q_{L,t}\bar{\delta z}_{L,t} \\ \text{s.t.} \quad & \bar{\delta z}_{l+1,t} = \widetilde{A}_{l,t}\bar{\delta z}_{l,t} + \widetilde{B}_{l,t}\delta u_{l,t}, \quad l = 0,\dots,L-1. \end{aligned} \tag{12}$$

Here $\bar{\delta z}_{l,t}$ is the chunk-aggregated latent error $\bar{\delta z}_{l,t} := \frac{1}{H_a}\sum_{\tau=1}^{H_a}\delta z_{l,t,\tau} = -(\frac{1}{H_a}\sum_{\tau=1}^{H_a}\alpha_{l,t,\tau})v_{l,t}^z$. The control penalty incorporates an action-decay schedule over robot action chunk indices. Let $\tau \in \{0,\dots,H_a-1\}$ index the action chunks executed over time. For chunk τ, we set $r(\tau) := \min(R_{\text{final}}, R_{\text{init}}\exp(\tau/\tau_R))$, where R_{init} is the initial steering penalty, R_{final} is a large saturation value, and τ_R controls the decay rate. The LQR control-cost matrix at

chunk τ is then $R_{l,t}^{\text{chunk}}(\tau) := r(\tau) I_{d_u}$, which is directly used in LQR (12). Since $r(\tau)$ increases with τ and saturates at R_{final}, steering is strongest for early chunks and gradually decays as the robot proceeds. A larger τ_R slows this growth, causing more chunks to be steered before saturation. In our experiments, R_{final} is chosen large enough that saturated chunks receive negligible steering.

For each denoising timestep t, solving (12) yields gains $K_{l,t} \in \mathbb{R}^{d_u \times d_z}$. These gains are used only within the corresponding denoising pass. At inference time, the controller is

$$u_{l,t}^* := \bar{u}_{l,t} - K_{l,t}\bar{\delta z}_{l,t} = \bar{u}_{l,t} + K_{l,t}\left(\frac{1}{H_a}\sum_{\tau=1}^{H_a} \alpha_{l,t,\tau}\right) v_{l,t}^z. \tag{13}$$

When $\bar{u}_{l,t} = 0$, the intervention magnitude is proportional to the online feature-tracking error averaged across the robot action chunk. The same policy $K_{l,t}$ is applied across all action-chunk timesteps, rather than using separate chunk-specific steering rules. The full WA-LQR procedure chains the T per-timestep controllers through WAM inference. For denoising timestep t, we apply the L feedback gains $\{K_{0,t}, \ldots, K_{L-1,t}\}$ inside the DiT blocks, as in (4). The steered block output $x_{L,t}$ is then passed to the scheduler, which produces $\hat{x}_{t-1}$ and initializes the next denoising pass, yielding T chained L-step LQR controllers.

X. Contrastive Vectors

To construct contrastive vectors we define a *desirable* or target set of inputs $\mathcal{D}_+$, with rollouts emblematic of behavior we seek to induce via steering, and and *undesirable* set of inputs $\mathcal{D}_-$, which are perturbed by some nuisance and result in failed rollouts. Given paired inputs (ξ_n^+, ξ_n^-), define *contrastive activation directions* by subtracting hidden states from the two forward passes. For layer l, denoising timestep t, and action-chunk timestep $\tau \in [H_a]$, this gives

$$d_{l,t,\tau}^{(n)} := x_{l,t,\tau}^{(\xi_n^+)} - x_{l,t,\tau}^{(\xi_n^-)}. \tag{14}$$

For Gaussian noise, we consider with noised and clean camera images, i.e., $\mathcal{D}_+ = \{\texttt{Clean inputs}\}$ and $\mathcal{D}_- = \{\texttt{Noised inputs}\}$. For pairs of inputs in $\xi^+ \in \mathcal{D}_+$ and $\xi^- \in \mathcal{D}_-$, we perform a model forward pass on each input and collect their corresponding activations x^+, x^-.

Similarly, the camera perturbation contrastive input sets are defined as $\mathcal{D}_+ = \{\texttt{No perturbation}\}$ and $\mathcal{D}_- = \{\texttt{Perturbed camera view}\}$. The corresponding activations x^+, x^- are collected from a forward pass of the model.

For gripper position perturbation, the definition of $\mathcal{D}_+$ and $\mathcal{D}_-$ differs slightly. Rather than defining the sets as unperturbed and perturbed, respectively, we let $\mathcal{D}_+$ be inputs corresponding to successful rollouts and $\mathcal{D}_-$ be inputs corresponding to unsuccessful inputs, both under gripper perturbation. This is to account for an observed variable sensitivity to different perturbations, i.e., directly including perturbed inputs in the negative dataset would result in many successful rollouts in the negative activations, resulting intuitively in "steering away" from desired behavior. Note, however, that we do not observe a meaningful difference in the mechanistic analysis when making the distinction between successful and unsuccessful rollouts in our dataset configuration (see Ap. XV-A).

Given sets of contrastive vectors $\{x_i^+\}$ and $\{x_j^-\}$, we compute the contrastive direction simply as the difference between pairs of positive and negative activations, with different pooling and processing as described in Sec. V.

TABLE II
LIBERO-10 [22] success rates on Cosmos-Policy 2B [18] under perturbations. Task $i \to j$ denotes all $P_{l,t}, \tilde{A}_{l,t}, \tilde{B}_{l,t}$ matrices, $e_{l,t}^z$ vectors are computed with task i, and used to steer task j.

	Tasks	No Steering	Prompt Steering	ActAdd [37]	WA LQR
Camera Orientation	$0 \to 0$	33.3%	40.0%	43.3%	**60.0%**
	$0 \to 2$	63.3%	60.0%	**73.3%**	70.0%
	$0 \to 4$	10.0%	6.7%	**16.7%**	13.3%
	$0 \to 5$	56.7%	70.0%	56.7%	**83.3%**
	$0 \to 9$	66.7%	56.7%	56.7%	**70.0%**
	Average	46.0%	46.7%	49.3%	**59.3%**
Gripper Position	$1 \to 1$	46.7%	46.7%	**66.7%**	56.7%
	$1 \to 2$	83.3%	83.3%	80.0%	**100.0%**
	$1 \to 3$	46.7%	53.3%	56.7%	**60.0%**
	$1 \to 7$	80.0%	63.3%	60.0%	**83.3%**
	$1 \to 9$	50.0%	**56.7%**	53.3%	**56.7%**
	Average	61.3%	60.7%	63.3%	**71.3%**
Gaussian Noise	$6 \to 0$	3.3%	0.0%	**43.3%**	26.7%
	$6 \to 1$	53.3%	0.0%	**70.0%**	66.7%
	$6 \to 4$	10.0%	0.0%	**80.0%**	43.3%
	$6 \to 6$	36.7%	3.3%	**76.7%**	73.3%
	$6 \to 7$	30.0%	0.0%	**66.7%**	43.3%
	Average	26.7%	0.7%	**67.3%**	50.7%

XI. Supplemental Experiments on Cosmos-Policy

We present the complete steering results on Cosmos-Policy, omitted from the main body for space. The results are presented in Tab. II. As an additional baseline method which does not involve finetuning, we compare to chain-of-thought prompting, inspired by [39].

XII. Supplemental Experimental Results on LingBot-VA

Although we show some results of WA-LQR on LingBot in Table I, we conducted a more comprehensive evaluation of WA-LQR applied in both the action module and the video module separately. Specifically, relative to Table III, we further evaluate on Gaussian noise perturbations, on more task transfers, and on steering in the video module (denoted "(Video)" in Table III). To allow a fair comparison with Cosmos, we evaluate our method on LingBot-VA on the same set of tasks and perturbations as shown in Table III. We did not observe significant effectiveness of steering, and this is supported by the result of our mechanistic analysis on the extracted activations from LingBot-VA's action and video modules, where it is shown that there is poor linear separability (with high numerical classification losses, especially relative to Cosmos), as presented in in Fig 17, 18, and 19.

TABLE III

LIBERO-10 [22] SUCCESS RATES ON LINGBOT-VA [20]. TASK $i \rightarrow$ TASK j DENOTES ALL $P_{l,t}, \tilde{A}_{l,t}, \tilde{B}_{l,t}$ MATRICES, $e^{z}_{l,t}$ VECTORS ARE COMPUTED WITH TASK i, AND USED TO STEER TASK j. 20 TRIALS/TASK.

Perturbation	Tasks	No Steering	Ours (Action)	Ours (Video)
Camera Orientation	Task 0 → Task 0	65.0%	70.0%	65.0%
	Task 0 → Task 2	55.0%	45.0%	55.0%
	Task 0 → Task 4	15.0%	30.0%	10.0%
	Task 0 → Task 5	70.0%	60.0%	65.0%
	Task 0 → Task 9	35.0%	50.0%	60.0%
	Average	48.0%	51.0%	51.0%
Initial Gripper Position	Task 1 → Task 1	65.0%	75.0%	70.0%
	Task 1 → Task 2	70.0%	80.0%	70.0%
	Task 1 → Task 3	70.0%	75.0%	85.0%
	Task 1 → Task 7	80.0%	80.0%	80.0%
	Task 1 → Task 9	75.0%	65.0%	70.0%
	Average	72.0%	75.0%	75.0%
Camera Gaussian Noise	Task 6 → Task 0	35.0%	20.0%	65.0%
	Task 6 → Task 1	90.0%	95.0%	20.0%
	Task 6 → Task 4	75.0%	75.0%	100.0%
	Task 6 → Task 6	85.0%	70.0%	75.0%
	Task 6 → Task 7	10.0%	20.0%	20.0%
	Average	59.0%	56.0%	56.0%

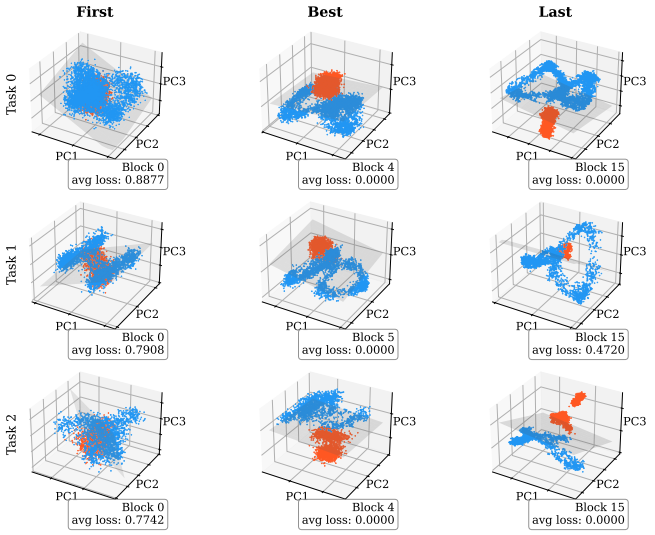

Fig. 6. Feature separability: DiT4DiT Gaussian noise corruption for tasks 0-2.

XIII. SUPPLEMENTAL EXPERIMENTAL RESULTS ON DIT4DIT

To further demonstrate the efficacy of our mechanistic analysis for predicting steering performance (via feature separability) and the efficacy of our WA-LQR steering approach for tasks with separable features, we evaluate our pipeline on another WAM, namely the inverse-dynamics mixture of transformers (MoT) model architecture in DiT4DiT [24] (available here). We first perform a mechanistic analysis on feature separability for gripper position and Gaussian noise perturbations (App. XIII-A), and then evaluate the performance of WA-LQR for steering these features (App. XIII-B).

A. Mechanistic Analysis

Similar to the procedure for analyzing the feature separability of LingBot-VA in Sec. IV and Ap. XV-A, we analyze activations extracted from the action-generation DiT. Here, we consider two perturbation types: Gaussian noise image corruption, and perturbation to initial gripper position. We summarize the results in Fig. 6-8. Unlike LingBot-VA, we observe significant separability in the Gaussian noise feature, particularly in later layers (Fig. 6, which achieves zero separation loss at later layers, and 8, which has a similarly low loss). However, unlike Cosmos-Policy, we do not find separability in the gripper position perturbation (has around 0.6 average separation loss).

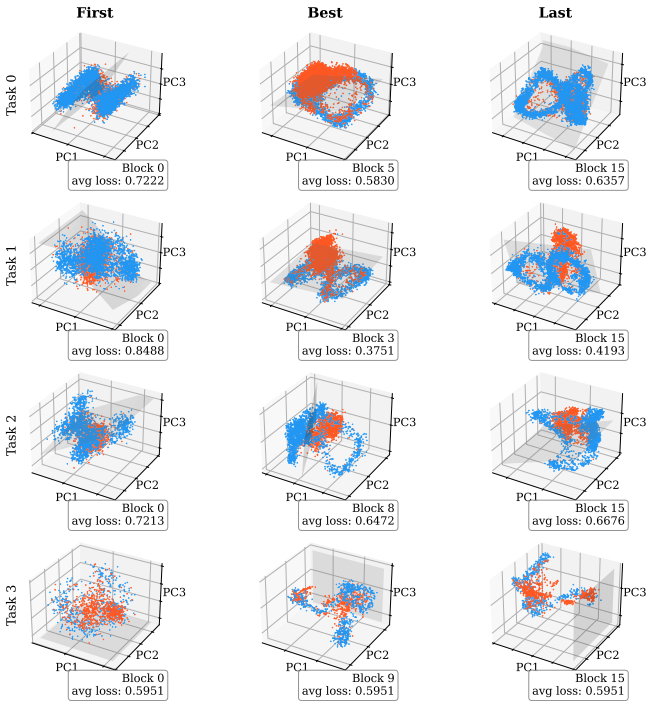

Fig. 7. Feature separability: DiT4DiT gripper perturbation for tasks 0-3.

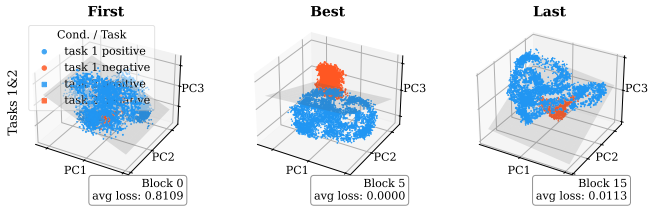

Fig. 8. Pairwise Gaussian noise corruption feature separation on DiT4DiT.

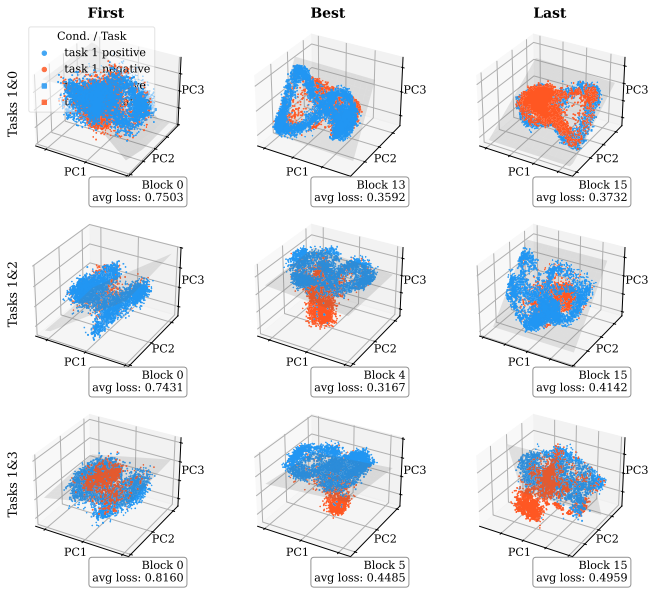

Fig. 9. Pairwise gripper perturbation feature separation on DiT4DiT.

B. Evaluating WA-LQR for Improving Robustness

From our mechanistic analysis, we identify Gaussian noise as a linearly separable perturbation feature, as well as identify a small grouping of tasks which share similar representation. To further validate the generality of our approach, we evaluate WA-LQR (Sec. V-B), steering on all timesteps of the action generation model. We seek to improve robustness via steering on the features corresponding to Gaussian noise and gripper position perturbations. We present these results in Tab. IV. Similar to Cosmos-Policy (Tab. I), we see improvements in success rate (21.7% on average) for Gaussian noise, where we observe significant feature separability. For gripper perturbation, we do not see as meaningful of a steering improvement, which is supported by and consistent with the mechanistic results of Fig. 7 and also with our steering results in LingBot-VA (Tab. III).

TABLE IV
LIBERO-10 [22] SUCCESS RATES ON DIT4DIT [24]. TASK $i \rightarrow$ TASK j DENOTES ALL $P_{l,t}$, $\tilde{A}_{l,t}$, $\tilde{B}_{l,t}$ MATRICES AND $e^z_{l,t}$ VECTORS ARE COMPUTED WITH TASK i, AND USED TO STEER TASK j.

Perturbation	Tasks	No Steering	Ours (Action)
Initial Gripper Position	Task 0 → Task 0	53.3%	**66.7%**
	Task 0 → Task 1	69.6%	**70.0%**
	Task 0 → Task 2	70.0%	**73.3%**
	Task 0 → Task 3	70.0%	**76.7%**
	Average	65.7%	**71.7%**
Camera Gaussian Noise	Task 1 → Task 1	20.0%	**43.3%**
	Task 1 → Task 2	10.0%	**30.0%**
	Task 1 → Task 0	16.7%	**73.3%**
	Average	15.6%	**48.9%**

XIV. Correlation Between Interpretability Results and Steering Performance

We summarize the relationship between separability and steering performance in Fig. 10. We observe a clear negative correlation between hinge loss and robustness improvements through steering.

XV. Experimental Details

A. Mechanistic Study

To analyze the geometry of each model's latent activations, we operate directly in the full activation space, using mean-pooled activations across token positions as described in Sec. IV. For Cosmos-Policy, which utilizes a unified single-backbone architecture, we consider the activations across all latent frames. For LingBot-VA and DiT4DiT, which adopt inverse-dynamics style architectures with distinct DiTs for different modalities, we consider the activations from the action-generation DiT. This is consistent with our steering formulation (Ap. XV-B-XV-C).

To collect contrastive examples for each perturbation type, we conduct a similar procedure the contrastive vector collection process in Ap. X. That is, we consider successful vs. unsuccessful rollout as $\mathcal{D}_+$ and $\mathcal{D}_-$, respectively, under camera or gripper position perturbation. For Gaussian noise, we let $\mathcal{D}_+ = \{\texttt{Clean inputs}\}$, and $\mathcal{D}_- = \{\texttt{Noised inputs}\}$.

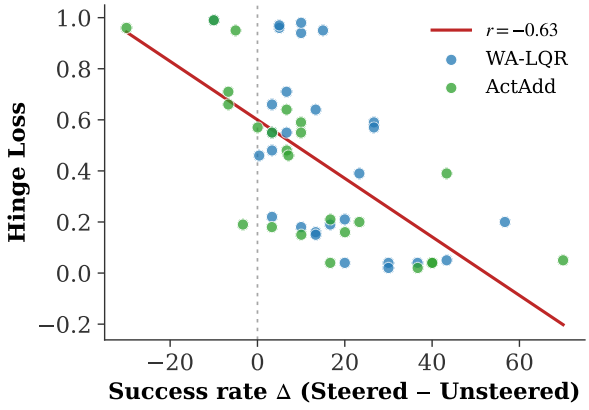

Fig. 10. Hinge loss vs. steering performance across all tasks and models, with reported line of best fit (red) and correlation coefficient.

We also evaluated the gripper and camera perturbation in terms of $\mathcal{D}_+ = \{\texttt{Without perturbation}\}$, and $\mathcal{D}_- = \{\texttt{With perturbation}\}$, but did not observe a meaningful difference.

Given $\mathcal{D}_+$ and $\mathcal{D}_-$, we compute contrastive directions, and conduct Principal Component Analysis on the set of contrastive directions, as described in Sec. IV. To fit the support vector machine (SVM), we initialize the separating $\{x|\hat{w}^T x + \hat{b} = 0\}$ hyperplane as the zero vector and perform gradient descent with the loss function:

$$\texttt{loss}_{\texttt{SVM}}(\hat{w}, \hat{b}) = (1/2)\hat{w}^\top \hat{w} + C \cdot \sum_{i \in [N]} \max(0, 1 - y_i(\hat{w}^T \bar{x}_i + \hat{b})), \tag{15}$$

where C is some constant (in our experiments, we set $C = 10$). Note the second term resembles the second term in Eq. 15 resembles the hinge loss in Eq. 16, evaluated on the intermediate hyperplane parameterized by $\hat{w}, \hat{b}$. The SVM is always computed in the three dimensional subspace defined by the top three principal components of the contrastive directions. We then report the hinge classification loss:

$$\texttt{loss}(\mathcal{D}_+, \mathcal{D}_-) = (1/N) \textstyle\sum_{i \in [N]} \max(0, 1 - y_i(w^T \bar{x}_i + b)), \tag{16}$$

where $\{x|w^T x + b = 0\}$ denotes the SVM hyperplane and $y_i \in \{-1, 1\}$ [16].

B. Cosmos-Policy Evaluations on LIBERO-10

Cosmos Policy is a diffusion-based world-action model with diffusion transformer backbone where it jointly denoise robot action chunk and future state predictions (future proprioception, wrist image, and third-person image). In our method, steering is applied at all latent outputs except for the action chunks. To construct contrastive vectors, activations are collected from all 28 transformer blocks across 5 denoising timesteps for matched pairs of clean and perturbed observations of the same task and scene.

For each layer and timestep, a randomized SVD is run on the paired matrix to produce a rank-64 basis to reduce dimensionality. The compact subspace is used to find the linearized dynamics that approximates how activations propagate from one layer to the next. The LQR gains are precomputed via backward Riccati recursion, where the control cost grows exponentially with the action chunk index. A LQR hyperparameter search is conducted before settling on the best-performing parameters for all other tasks of the same scene. During inference time, the projected activation error relative to the nominal trajectory is used to inject a correction through the precomputed gains.

C. LingBot-VA Evaluations on LIBERO-10

LingBot-VA is a mixture-of-transformers (MoT) based world-action model. It first uses a video module to predict future visual frames, which are then passed to a lightweight action module with a smaller hidden dimension to generate robot actions. In our experiments, we apply our method to either the video module or the action module while keeping the other component unchanged.

For both modules, we collect activations from the outputs of all 30 transformer blocks. Since the video and action modules use different denoising schedules, we select module-specific denoising timesteps. For the action module, we use timesteps $(t \in 0, 10, 20, 30, 40)$, For video module, we use timesteps $(t \in 0, 4, 9, 14, 19)$. These timesteps are chosen to cover the corresponding denoising trajectory of each module.

To reduce dimensionality, we partition the 30 transformer layers into three groups: layers 0–9, 10–19, and 20–29. For each layer partition and denoising timestep, we pool activation differences of contrastive pairs from all layers within the corresponding partition and compute a shared rank-64 SVD basis. The produced compact subspace for each layer group and timestep pair is used by our LQR injector during evaluation. At evaluation time, the LQR injector follows the same layer partitioning and timestep mapping. For each activation, we project it onto the corresponding low-dimensional SVD subspace, compute the LQR correction in this reduced space, and map the correction back to the full activation space before applying it to the model.

XVI. Complete Low-Dimensional Separability Results

We report the feature separation for three considered robustness features: perturbation to initial gripper position, initial camera position, and corruption of camera inputs with Gaussian noise, for all tasks in the LIBERO-10 dataset [22]. The results are summarized in Figs. 11-16. As reported in Sec. IV, we observe linear separation across tasks and perturbations in Cosmos-Policy (Figs. 11-13, but do not observe such separation in LingBot-VA (Figs. 14-19). For Cosmos-Policy, where we observe meaningful separation, we also present pairwise separability plots (Fig. 20-22), generated by aggregating the activations for pairs of tasks and otherwise following the same procedure as before.

XVII. Full Model Separation

For completeness, we report the full mechanistic separation plots across all layers, shown in Figs. 23-32.

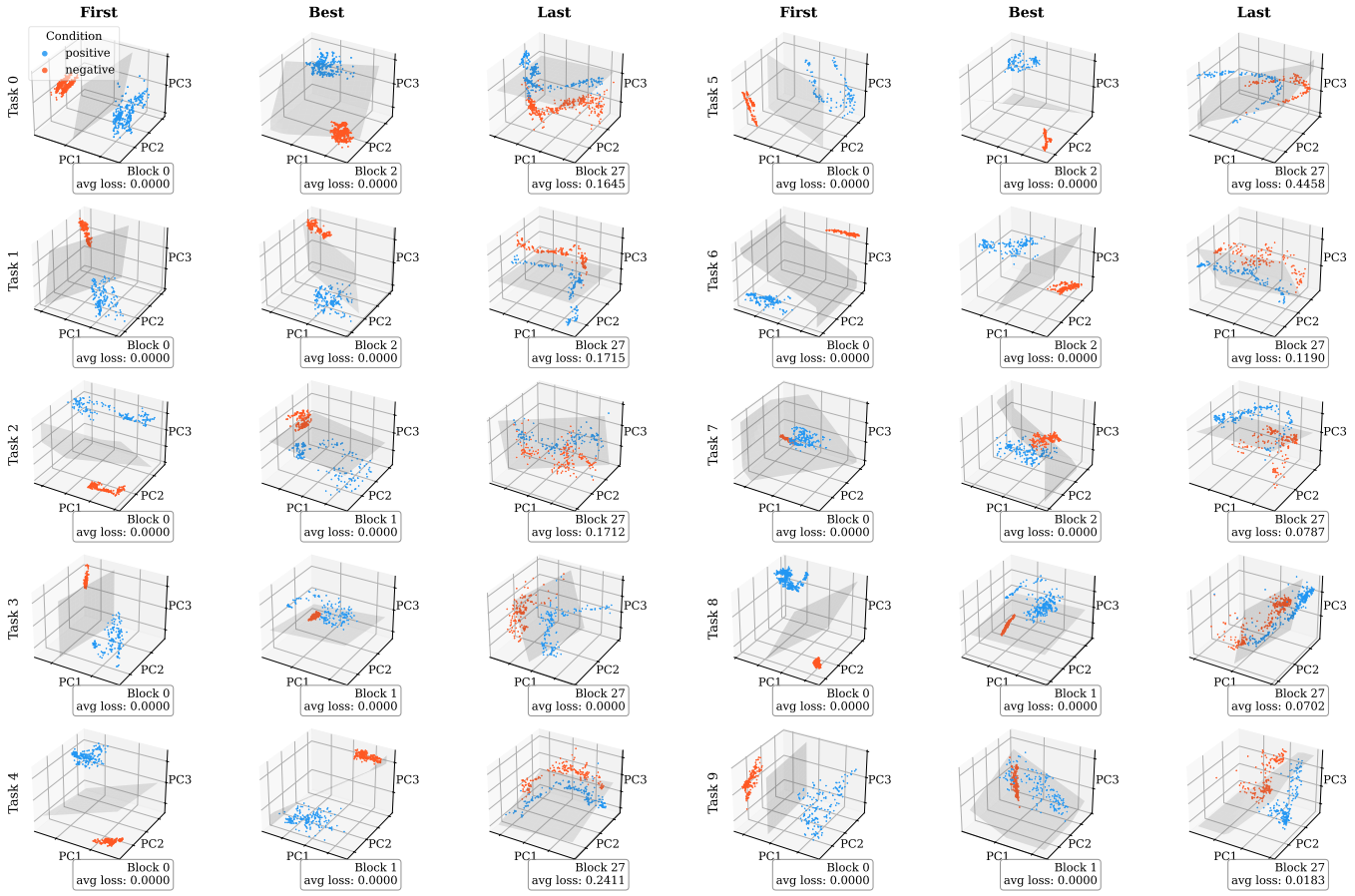

Fig. 11. Cosmos-Policy noise corruption separation for all LIBERO-10 tasks

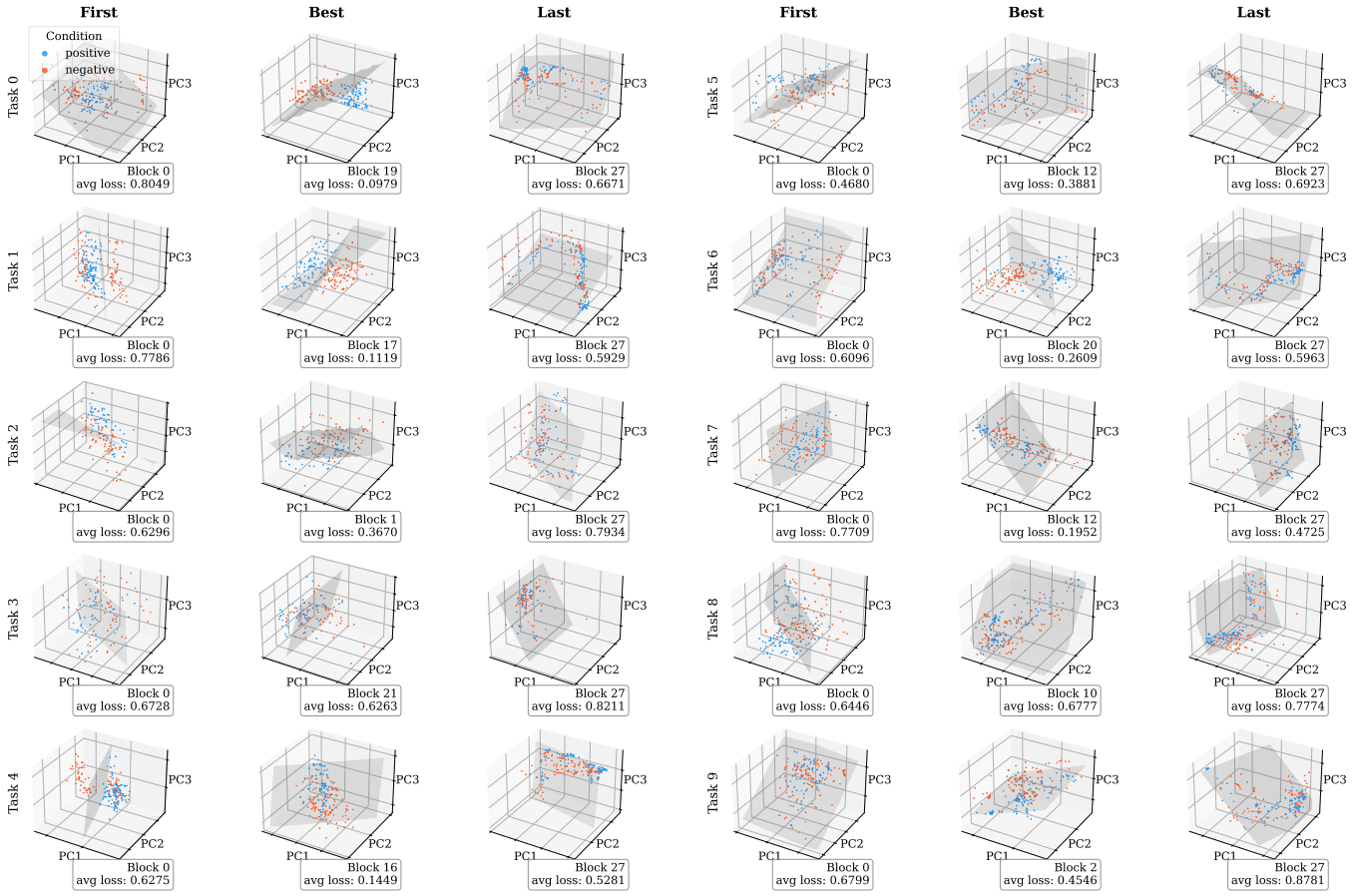

Fig. 12. Cosmos-Policy camera perturbation separation for all LIBERO-10 tasks.

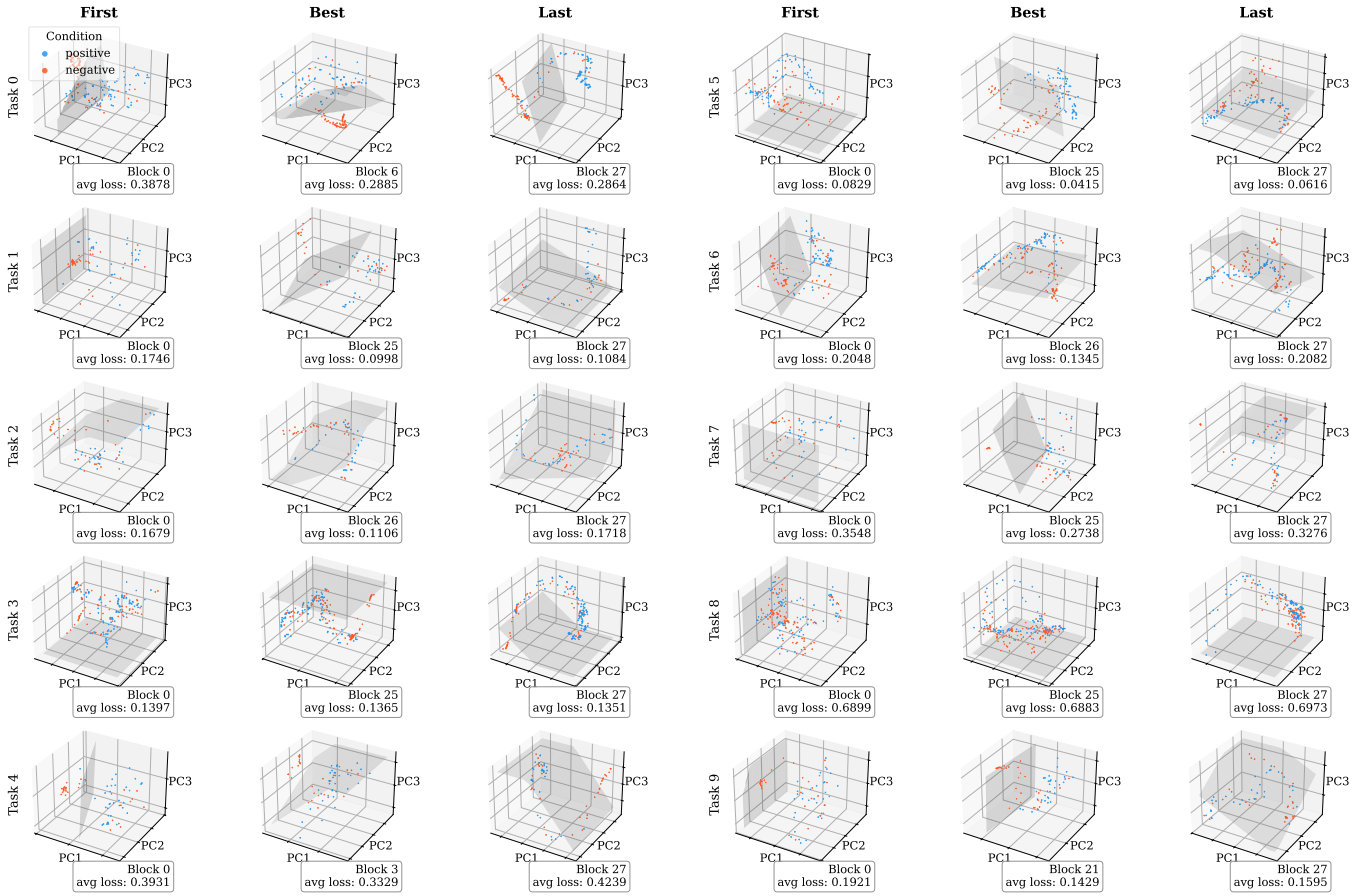

Fig. 13. Cosmos-Policy gripper perturbation separation for all LIBERO-10 tasks.

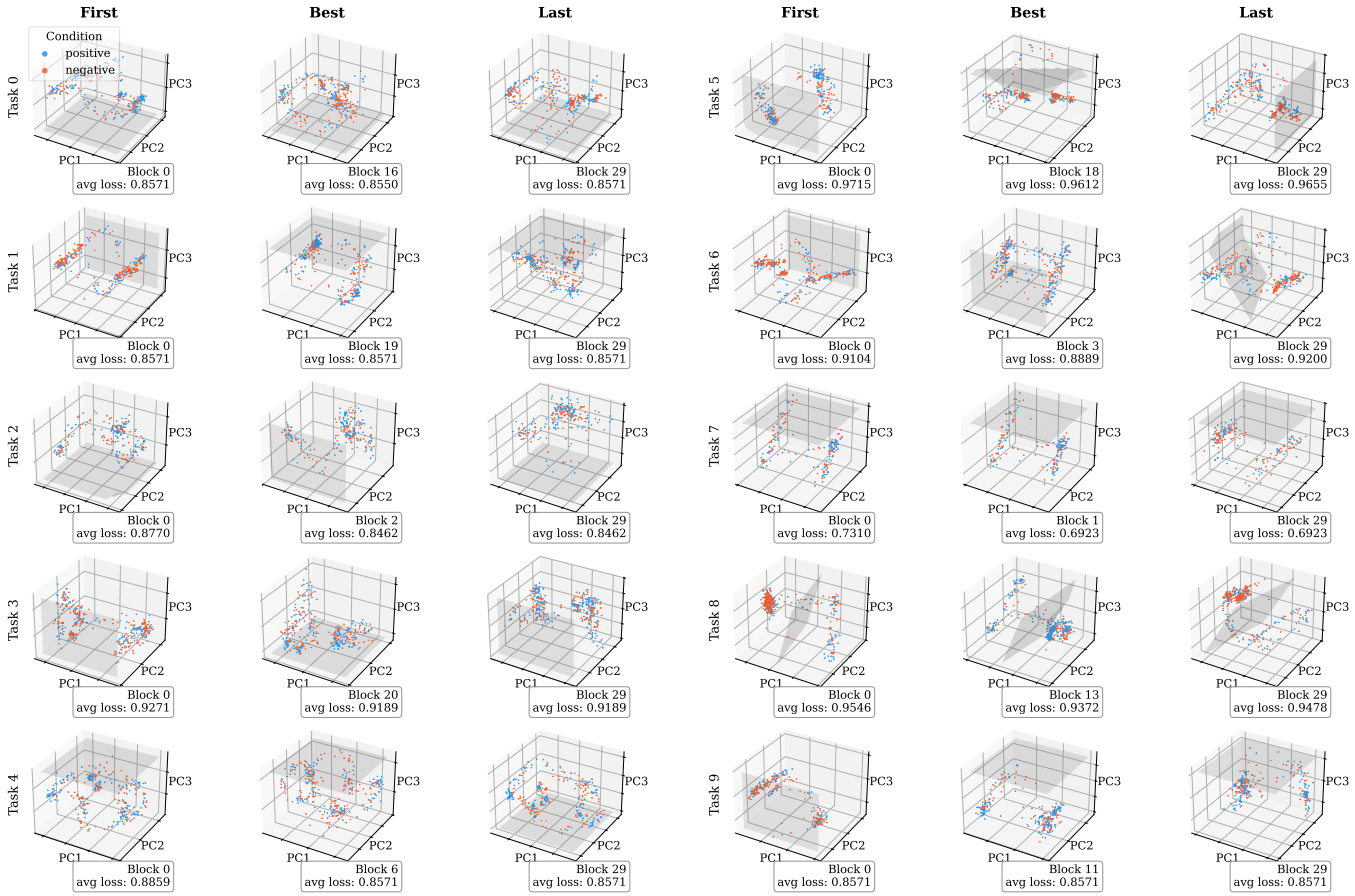

Fig. 14. LingBot-VA camera perturbation Separation for all LIBERO-10 tasks (Action Module).

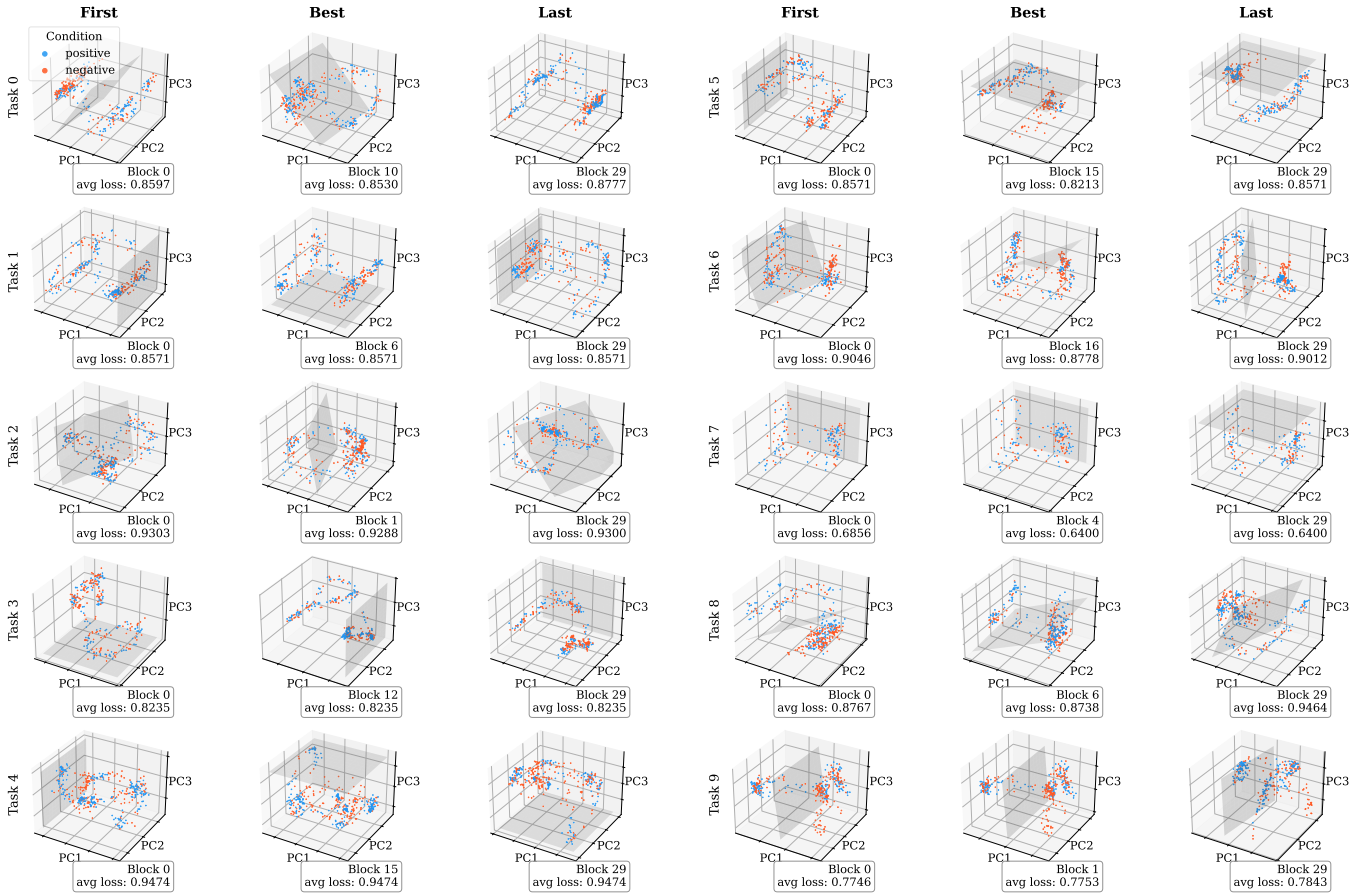

Fig. 15. LingBot-VA gripper perturbation Separation for all LIBERO-10 tasks (Action Module).

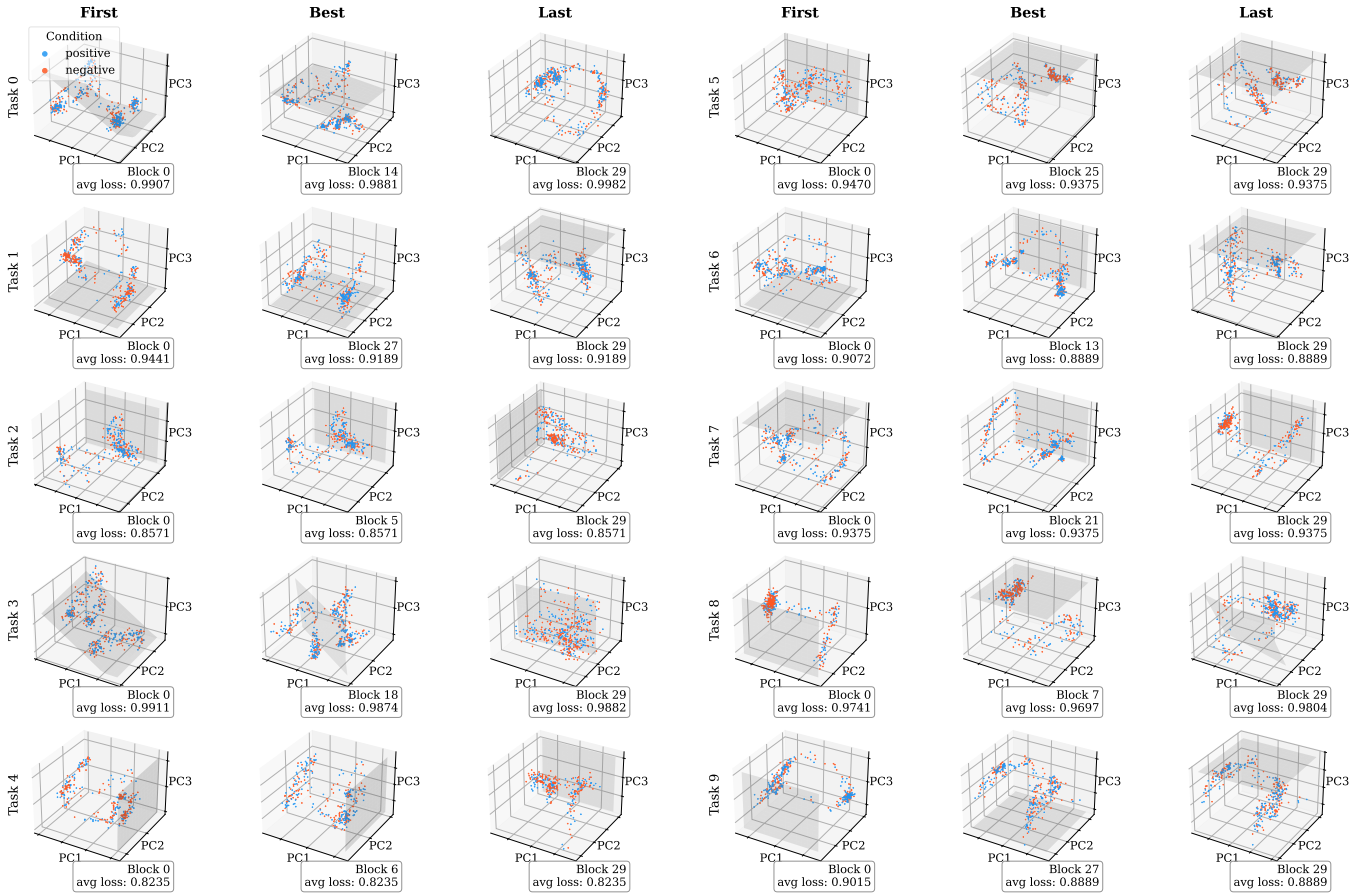

Fig. 16. LingBot-VA noise corruption separation for all LIBERO-10 tasks (Action Module).

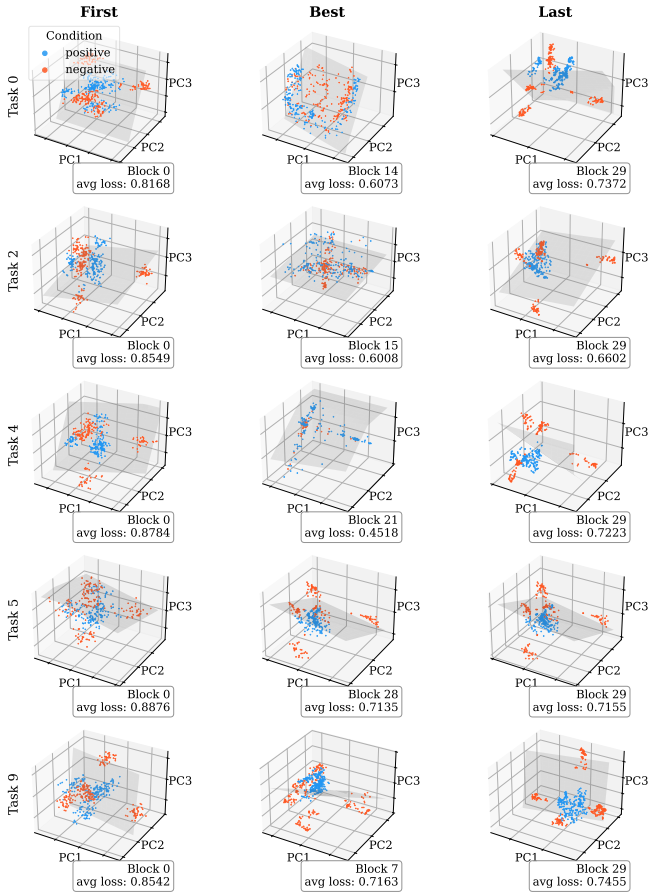

Fig. 17. LingBot-VA Camera Orientation Perturbation Separation for LIBERO-10 task 0, 2, 4, 5, 9 (Video Module).

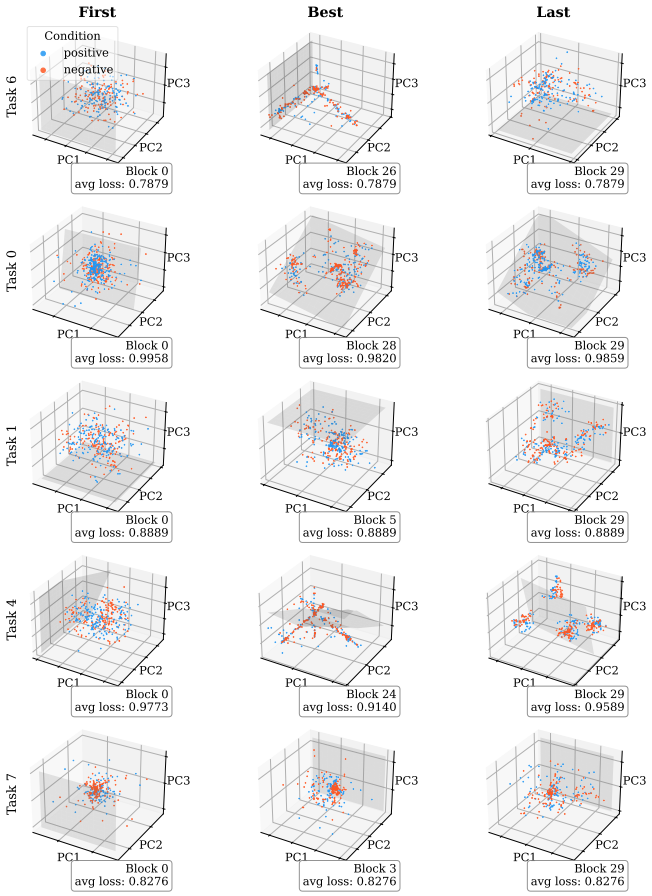

Fig. 18. LingBot-VA noise corruption separation for LIBERO-10 task 0, 1, 4, 6, 7 (Video Module).

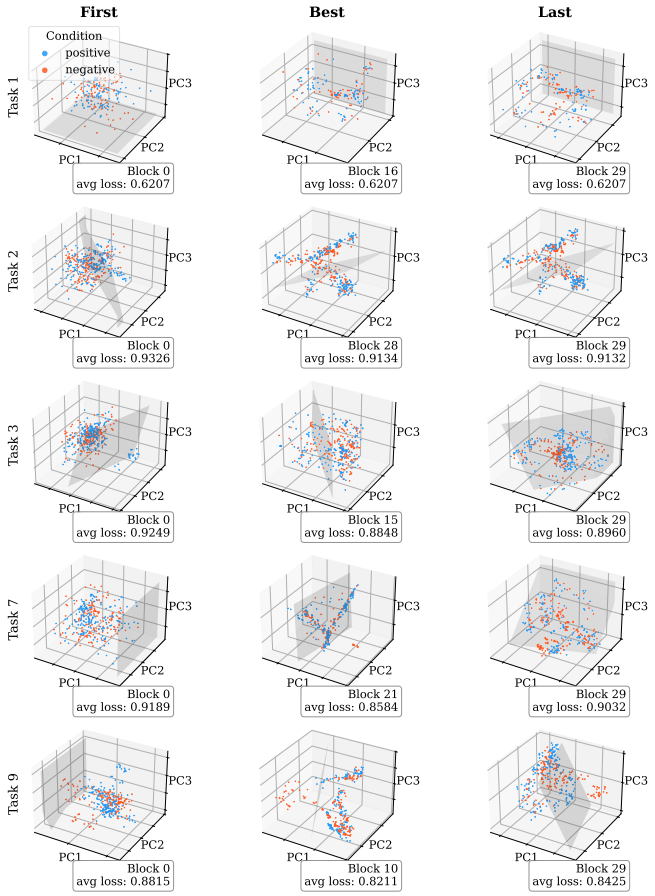

Fig. 19. LingBot-VA Gripper Perturbation Separation for LIBERO-10 task 1, 2, 3, 7, 9 (Video Module).

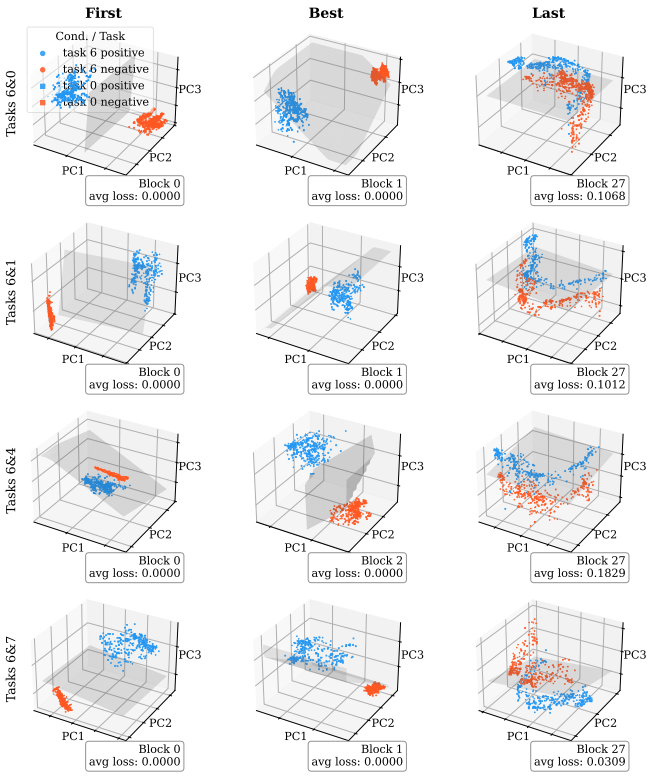

Fig. 20. Pairwise separability for Cosmos-Policy with Gaussian noise corruption.

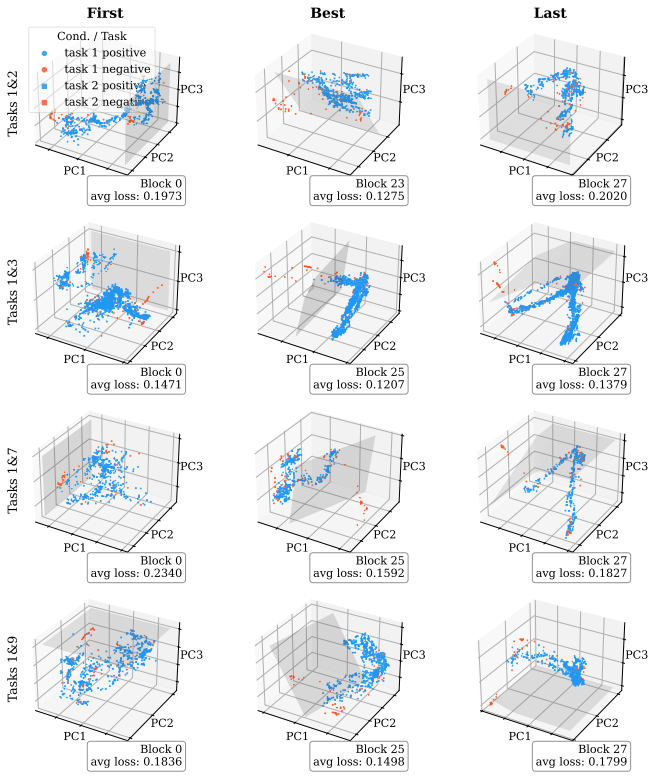

Fig. 21. Pairwise separability for Cosmos-Policy with gripper position perturbation.

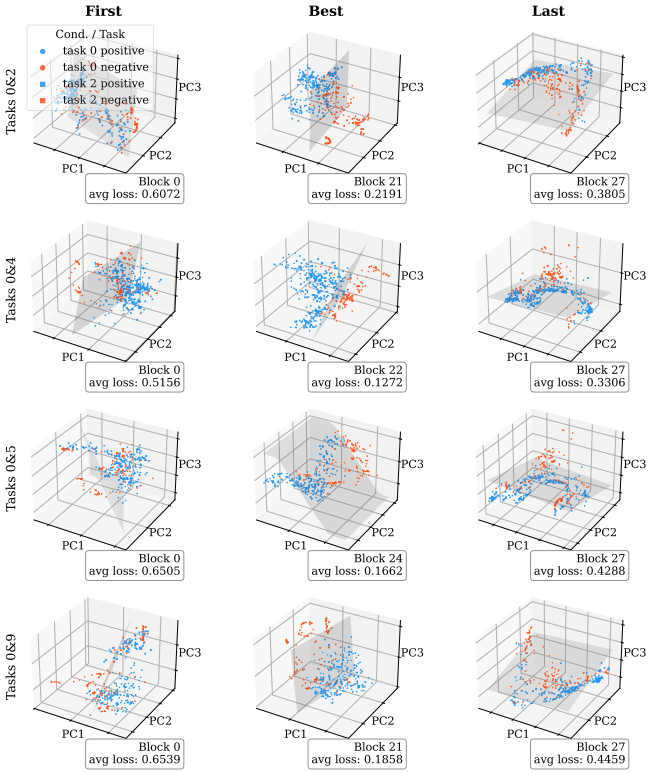

Fig. 22. Pairwise separability for Cosmos-Policy with camera position perturbation.

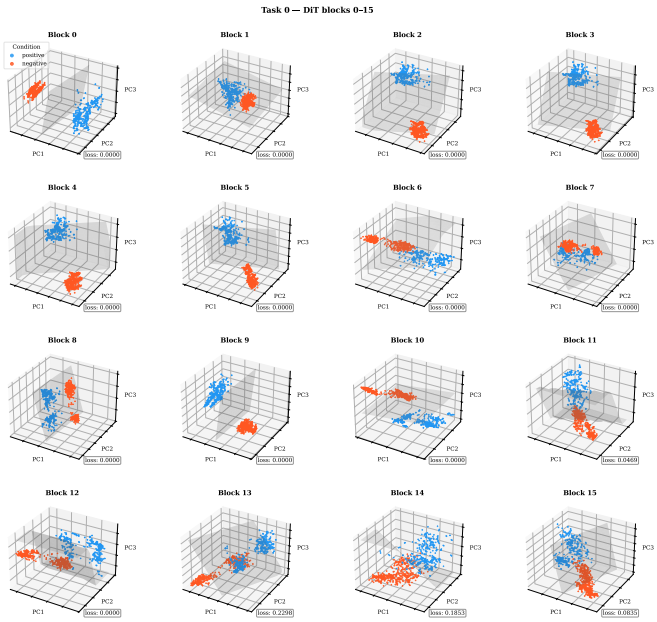

Fig. 23. Cosmos-Policy Task 0 noise corruption

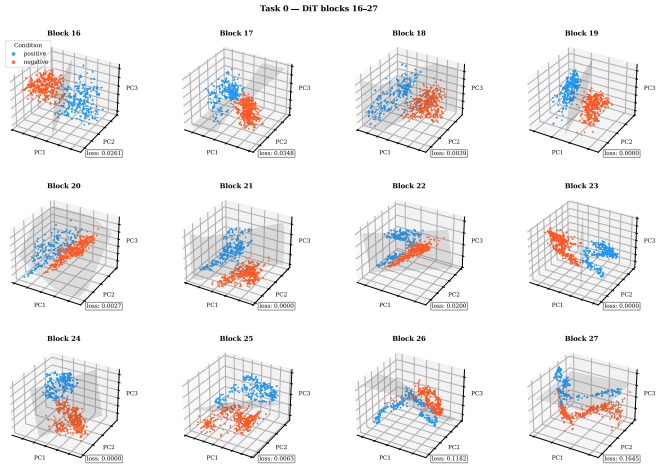

Fig. 24. Cosmos-Policy Task 0 noise corruption

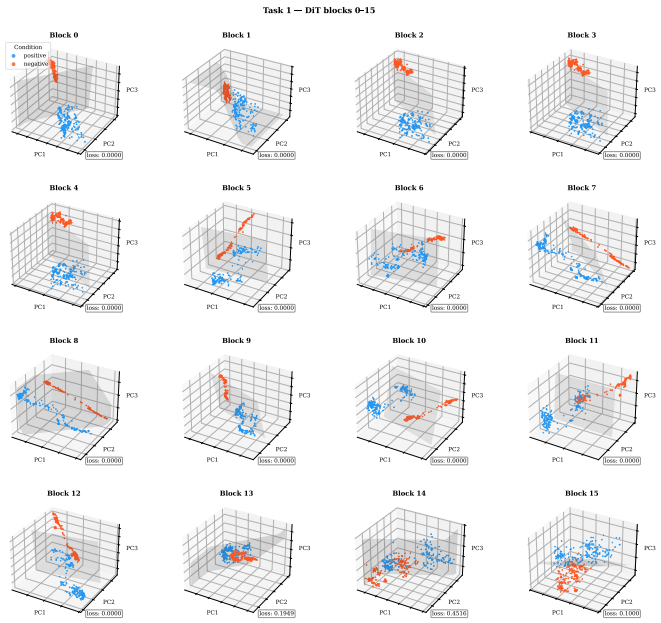

Fig. 25. Cosmos-Policy Task 1 noise corruption

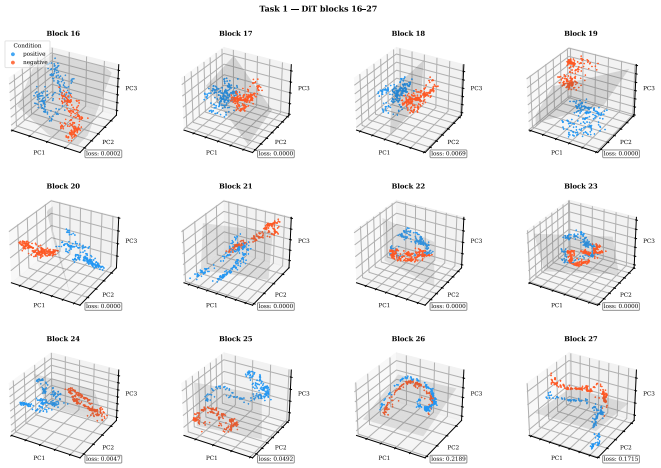

Fig. 26. Cosmos-Policy Task 1 noise corruption

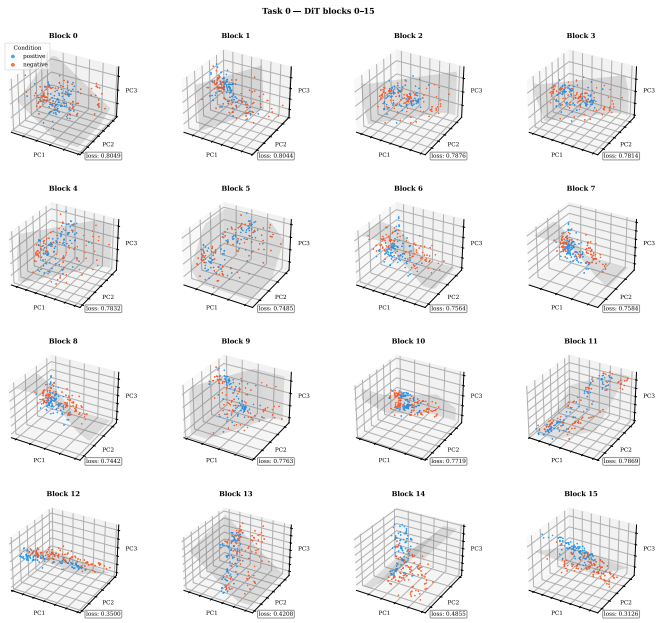

Fig. 27. Cosmos-Policy Task 0 Camera Perturbation

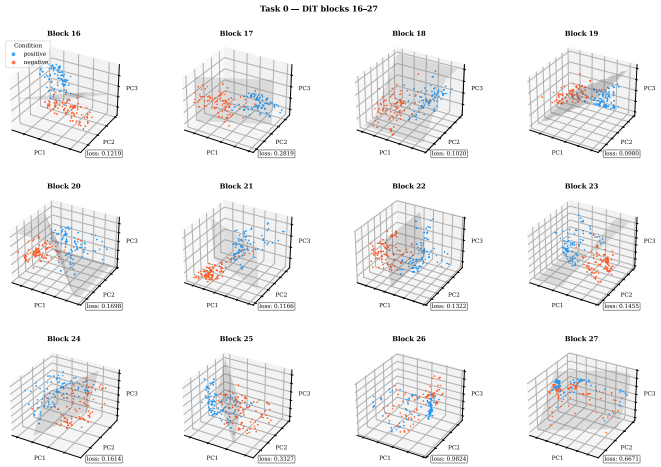

Fig. 28. Cosmos-Policy Task 0 Camera Perturbation

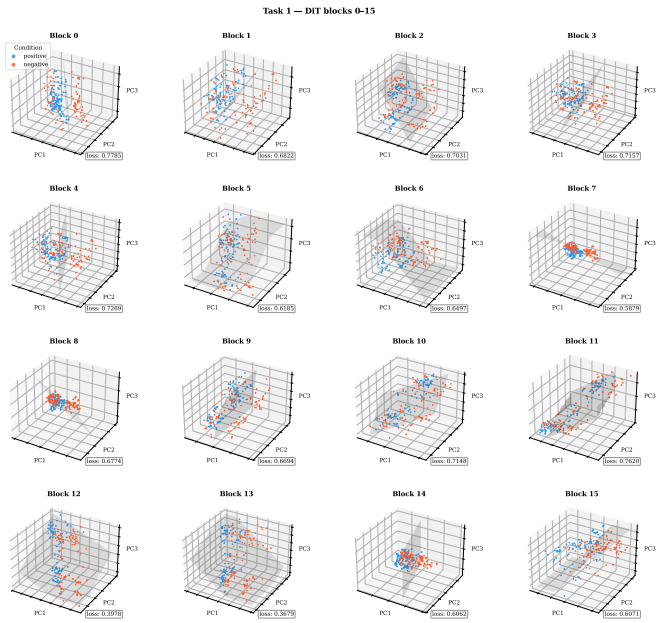

Fig. 29. Cosmos-Policy Task 1 Camera Perturbation

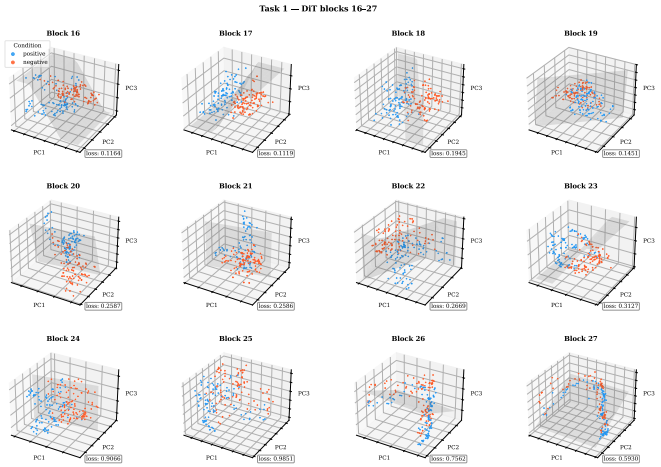

Fig. 30. Cosmos-Policy Task 1 Camera Perturbation

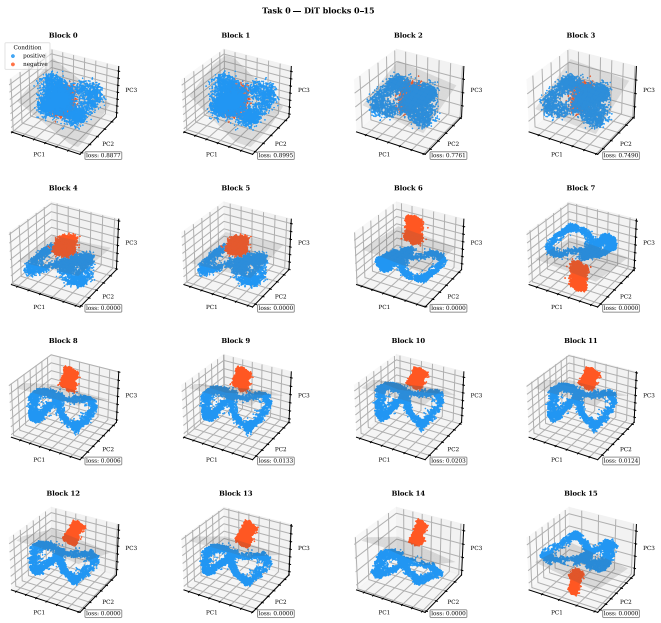

Fig. 31. DiT4DiT Task 0 noise corruption

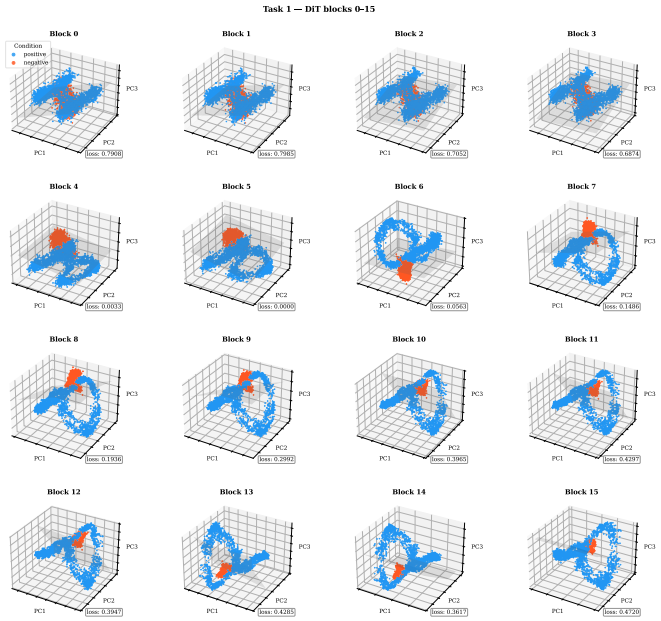

Fig. 32. DiT4DiT Task 1 noise corruption