# WHAT MAKES TWO ROBOT FAILURES THE SAME? LEARNING RECOVERABILITY FUNCTIONS FOR ZERO-SHOT ROBOT RECOVERY

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Robotic recovery depends on a question that semantic failure labels do not directly answer: which corrective intervention will restore task progress from the current failed state? We introduce the *recoverability function* $F_\tau(u) = P(\text{success} \mid s_{t_0}, u, \pi_c)$, which characterizes a natural failure trajectory $\tau$ by its response over a parameterized space of corrective interventions. REPAIRSPACE learns an ordinal functional geometry in which nearby failures exhibit similar recoverability profiles, together with an intervention-conditioned predictor that estimates responses for unexecuted corrections. Recovery is performed by cost-constrained optimization over the predicted response surface. Across LIBERO-40 and RLBench-30, covering 70 tasks, 28,000 natural failures, three policy families, and 4.24M main-protocol recovery rollouts, REPAIRSPACE reaches 0.824 success on unseen tasks and recovers 80.3% of failure-specific routing headroom. Under strictly matched supervision, the same 118M-parameter architecture improves unseen-task SR from 0.798 to 0.821 and cross-policy SR from 0.774 to 0.802. Its learned distance predicts held-out intervention transfer with correlation 0.842, compared with 0.694 for the matched full-trajectory response predictor. On held-out VLA-policy failures from unseen tasks, REPAIRSPACE reaches 0.827 mean recovery success, compared with 0.784 for FLARE and 0.734 for REFLECT. The representation transfers across failure-generating policy families (0.806 success) and reaches 0.850 recovery success over eight real-robot failure configurations. Together, these results support recoverability as a functional coordinate system that connects failure representation to corrective action.

## 1 INTRODUCTION

Generalist robot policies increasingly span diverse manipulation tasks O'Neill et al. (2024); Brohan et al. (2023); Team et al. (2024); Kim et al. (2024); Black et al. (2024); Intelligence et al. (2025); Bjorck et al. (2025), yet failures remain heterogeneous: slips, pose errors, collisions, occlusions, contact mismatch, controller and planning errors Kawaharazuka et al. (2025); Poria et al. (2026). Recovery systems can detect, explain, and react to many such events Duan et al. (2024a); Pacaud et al. (2025); Ying et al. (2026); Liu et al. (2023); Zhao et al. (2026); Zeng et al. (2026); Zhang et al. (2026); Dai et al. (2026). The remaining representation question is how to organize failures so corrective experience transfers across them.

Semantic labels describe what a failure looks like or how it is named, whereas recovery depends on response to intervention. Different labels can admit the same correction, and the same label can conceal distinct corrective needs. A trajectory overshoot and a grasp offset may both benefit from local alignment, while two "grasp failures" can separate into regrasping and clearance recovery. We therefore define failures as similar when corrective interventions produce similar recovery outcomes (Figure 1).

For a natural failure trajectory $\tau$ frozen at checkpoint $s_{t_0}$, we define the *recoverability function* $F_\tau(u) = P(\text{success} \mid s_{t_0}, u, \pi_c)$ over parameterized corrective interventions $u = (c, \theta)$ under continuation policy $\pi_c$. REPAIRSPACE learns this structure with a visual–temporal encoder, an intervention-conditioned response head, and confidence-ordered Monte-Carlo probes inducing an



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

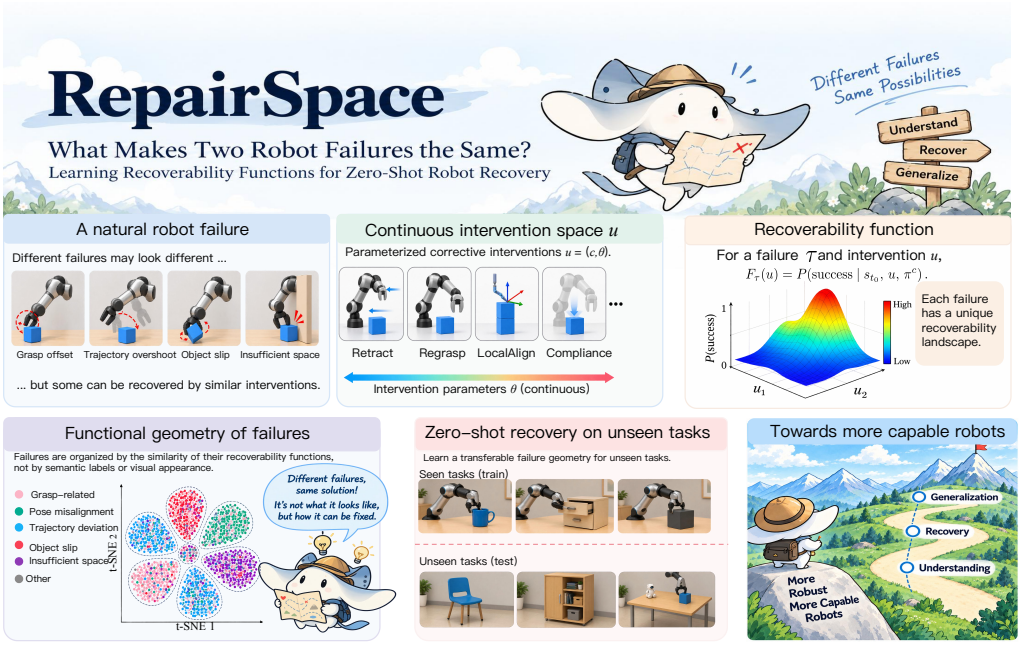



Figure 1: **RepairSpace organizes failures by corrective response.** Failures are frozen at checkpoint $t_0$ and probed with parameterized interventions to estimate $F_\tau(u)$. REPAIRSPACE learns response-aligned neighborhoods and selects cost-feasible corrections for unseen failures.

ordinal functional geometry. At deployment, the predicted response surface is optimized over cost-feasible interventions; coverage analysis relates recovery regret to failure and intervention coverage, representation mismatch, and response estimation.

Across LIBERO-40 and RLBench-30, REPAIRSPACE reaches 0.824 unseen-task SR and 80.3% of routing headroom. Under matched supervision, ordinal geometry raises unseen-task SR from 0.798 to 0.821 and cross-policy SR from 0.774 to 0.802. Its learned distance predicts held-out intervention transfer with correlation 0.842; the main protocol reaches 0.806 cross-policy SR, 0.827 mean SR on held-out VLA-policy failures versus 0.784 for FLARE and 0.734 for REFLECT, and 0.850 SR across eight real-robot failure configurations.

**Contributions.** We contribute: (1) a recoverability-function formulation of failure similarity over continuous corrective interventions; (2) an uncertainty-aware ordinal objective for transferable functional neighborhoods; (3) cost-aware zero-shot recovery with a coverage-based regret bound; and (4) a 70-task evaluation with 28,000 natural failures, three failure-generating policy families, held-out response probes, published recovery baselines, and real-robot recovery.

## 2 RELATED WORK

Execution-feedback systems use language, vision, and robot experience for feedback-based reasoning and correction Huang et al. (2022); Liu et al. (2023), multimodal failure detection and diagnosis Duan et al. (2024a); Pacaud et al. (2025); Ying et al. (2026), corrective replanning or rewind Zhao et al. (2026); Zeng et al. (2026); Dai et al. (2026); Zhang et al. (2026); Zhi et al. (2025); Pan et al. (2025); Ahmad et al. (2025), and learning from failures or corrections Jiang et al. (2024); Huang et al. (2025); Duan et al. (2024b). In parallel, language-grounded planning connects high-level reasoning to executable behavior Ahn et al. (2022); Liang et al. (2023); Huang et al. (2023), generalist robot and VLA models broaden transfer across tasks, embodiments, and visual contexts Brohan et al. (2023); O'Neill et al. (2024); Team et al. (2024); Kim et al. (2024); Black et al. (2024); Intelligence et al. (2025); Qu et al. (2025); Liu et al. (2025); Bjorck et al. (2025), and transferable visuomotor models develop structured action representations Chi et al. (2025); Zhao et al. (2023); Yang et al. (2024); Wang et al. (2024); Wu et al. (2025); Yang et al. (2025). These approaches expand the policies that

produce, interpret, and continue behavior, whereas REPAIRSPACE uses intervention outcomes as the representation target and organizes failed states according to similarity in their corrective-response profiles. This perspective is also related to behavioral representation learning: bisimulation metrics define similarity through reward and transition consequences Ferns et al. (2012; 2011), DeepMDP preserves reward and next-state structure Gelada et al. (2019), and Deep Bisimulation for Control learns behaviorally aligned latent distances Zhang et al. (2020); policy-similarity and state-abstraction formulations provide complementary views of transferable behavioral equivalence Lim and Choi (2025); Abel (2019). Recoverability geometry instantiates this consequence-based principle for post-failure control by defining similarity through success responses over a shared, parameterized space of corrective interventions.

# 3 RECOVERABILITY FUNCTIONS

## 3.1 FAILURE CHECKPOINTS AND CORRECTIVE INTERVENTIONS

A failure-generating policy $\pi_g$ executes a task until an external detector first triggers at $t_0$. The resulting trajectory is

$$\tau = (I_{1:t_0}, x_{1:t_0}, a_{1:t_0}, c),$$

where $I$ denotes visual observations, $x$ robot and environment state, $a$ the action history, and $c$ the task condition. We snapshot simulator state $s_{t_0}$ and replay every candidate correction from this identical checkpoint. Failures are generated naturally by learned policies during benchmark execution.

The corrective intervention space is a union of eight parameterized families,

$$\mathcal{U} = \bigcup_{c \in \mathcal{C}} \{c\} \times \Theta_c, \quad \mathcal{C} = \{\text{Retract, ReApproach, Regrasp, LocalAlign, Compliance, Relocalize, Reset, Replan}\}.$$

Each family has continuous physical parameters, such as retract distance, local Cartesian offsets, grasp rotation, compliance gains, or replanning horizon. Table 4 gives the full supports and execution rules. The intervention specification defines what the robot can do; recovery structure is learned from outcomes.

## 3.2 RECOVERABILITY FUNCTION AND FUNCTIONAL DISTANCE

After intervention $u$, the fixed continuation policy $\pi_c$ resumes the task. The recoverability function is

$$F_\tau(u) = P(\text{success} \mid s_{t_0}, u, \pi_c), \tag{1}$$

with $F_\tau : \mathcal{U} \to [0, 1]$. A probe measure $\mu$ over interventions induces the functional distance

$$d_F^2(\tau_i, \tau_j) = \mathbb{E}_{u \sim \mu} \left[ (F_{\tau_i}(u) - F_{\tau_j}(u))^2 \right]. \tag{2}$$

This distance compares complete corrective-response profiles. A finite probe set $u_1, \ldots, u_M \sim \mu$ provides Monte-Carlo measurements of the underlying function.

For probe $u_m$, let $\hat{p}_{im}$ be the empirical success rate from $n_{im} \geq 2$ Bernoulli rollouts and $\hat{v}_{im} = \hat{p}_{im}(1-\hat{p}_{im})/(n_{im}-1)$ the unbiased variance estimate of the sample mean. We remove Monte-Carlo variance from the empirical squared distance:

$$\widehat{q}_{ij} = \frac{1}{M} \sum_{m=1}^{M} \left[ (\hat{p}_{im} - \hat{p}_{jm})^2 - \hat{v}_{im} - \hat{v}_{jm} \right]. \tag{3}$$

The response probes therefore act as measurements of a continuous functional object; the finite probe vector is its Monte-Carlo measurement.

## 3.3 TRAJECTORY REPRESENTATION AND RESPONSE PREDICTION

The encoder $z = f_\theta(\tau) \in \mathbb{R}^d$ combines visual observations, proprioception, Cartesian action history, and the task condition through a visual backbone and temporal Transformer. We normalize $\|z\|_2 = 1$. The intervention-conditioned head predicts

$$\widehat{F}_\tau(u) = g_\phi(f_\theta(\tau), u).$$

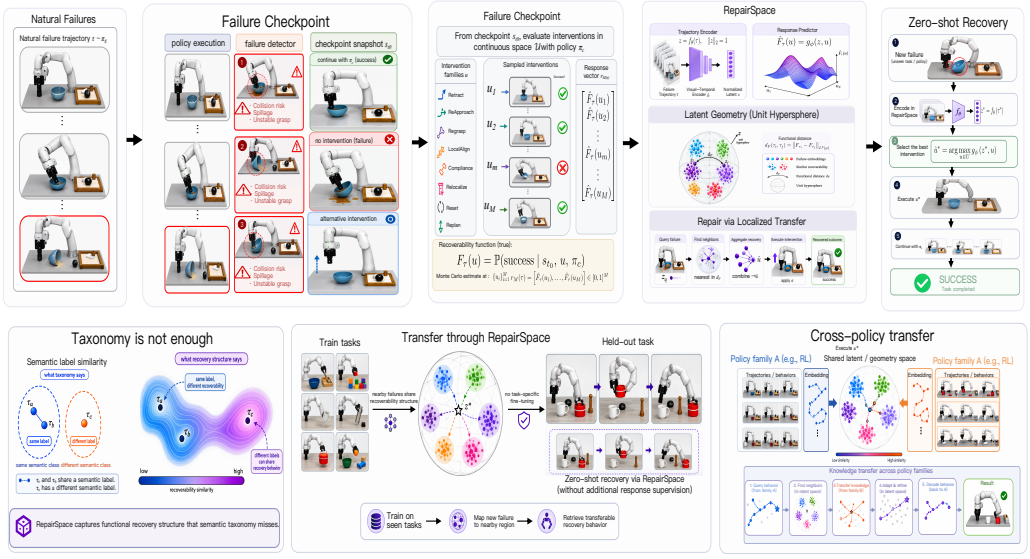



Figure 2: **Training and zero-shot recovery.** A failure checkpoint is paired with corrective interventions to estimate recoverability responses. The trajectory encoder $f_\theta$ and response head $g_\phi$ learn pointwise outcomes, while confidence-ordered response profiles shape the latent geometry. At test time, an unseen failure is encoded once and the response surface is optimized over the cost-feasible intervention set.

For observed binary outcomes $Y_{im}$, the pointwise objective is

$$\mathcal{L}_{\text{resp}} = -\mathbb{E}_{i,m}\left[Y_{im}\log \widehat{F}_i(u_m) + (1-Y_{im})\log(1-\widehat{F}_i(u_m))\right]. \tag{4}$$

This head supports interpolation over unexecuted continuous intervention parameters and directly supplies the deployment objective.

### 3.4 Uncertainty-aware ordinal functional geometry

We train the latent representation to preserve statistically supported orderings of recoverability-function similarity. With unit-normalized embeddings, define cosine distance $d_z(i,j) = 1 - z_i^\top z_j$. For anchor $i$, a triplet $(i,j,k)$ is retained when

$$\widehat{\Delta}_{ijk} = \widehat{q}_{ik} - \widehat{q}_{ij} > 0, \qquad \widehat{\Delta}_{ijk} > \kappa\,\widehat{\mathrm{SE}}(\widehat{\Delta}_{ijk}),$$

where the standard error is obtained by a joint bootstrap over matched response probes and rollout replicates. The ordinal objective is

$$\mathcal{L}_{\text{ord}} = \mathbb{E}_{(i,j,k)\in\Omega_{\text{ord}}}\left[\mathrm{softplus}\left(-\frac{d_z(i,k)-d_z(i,j)}{\tau_z}\right)\right]. \tag{5}$$

We optimize

$$\mathcal{L} = \mathcal{L}_{\text{resp}} + \lambda_{\text{ord}}\mathcal{L}_{\text{ord}}. \tag{6}$$

Pointwise supervision estimates individual intervention responses, while ordinal supervision couples failures through the shape of their response profiles. Each measured probe therefore contributes both a local response target and relational evidence about which failures should share recovery behavior.

### 3.5 Cost-aware zero-shot recovery

Each intervention has a fixed normalized execution cost $C(u)$. For budget $B$,

$$\mathcal{U}_B = \{u \in \mathcal{U} : C(u) \le B\}, \qquad \hat{u}_B^*(\tau) = \arg\max_{u\in\mathcal{U}_B} g_\phi(f_\theta(\tau), u).$$

Table 1: **Experimental protocol.** LIBERO-40 combines the four 10-task LIBERO suites; RLBench-30 is a fixed diversity-balanced 30-task subset. Train and unseen-test task identities are disjoint.

| Benchmark | Suite definition | Total | Train | Unseen | Failure policies $\pi_g$ | Natural failures | Seeds |
|---|---|---|---|---|---|---|---|
| **LIBERO-40** | Spatial / Object / Goal / LIBERO-10 | 40 | 32 | 8 | Diffusion / ACT / VLA | 16,000 | 5 |
| **RLBench-30** | Fixed diversity-balanced subset | 30 | 24 | 6 | Diffusion / ACT / VLA | 12,000 | 5 |

The response protocol uses 8 intervention families, 64 candidate interventions, 12 rollouts per dense $(\tau, u)$ pair, and five random seeds.

The main budget $B_{\text{main}} = 2.0$ contains the six local corrective families; higher budgets additionally admit Reset and Replan. Cross-entropy method searches the continuous parameters of each feasible family, producing a zero-shot recovery decision directly from the learned response surface on an unseen failure.

### 3.6 Coverage and recovery regret

The geometry connects data coverage to intervention quality. Suppose a test failure has a measured neighbor within $\epsilon$ in latent space, recoverability profiles vary at most $L_z\|z_i - z_j\|_2 + \xi$, interventions are $L_u$-Lipschitz, the candidate set is a $\rho_M$-cover of the feasible intervention space, and response estimates have uniform error $\eta$. Then the local transfer decision satisfies

$$\text{Regret} \leq 2(L_z\epsilon + \xi) + L_u\rho_M + 2\eta. \tag{7}$$

The proof is in Appendix B. The bound separates failure coverage, representation mismatch, intervention coverage, and response estimation. We measure the corresponding quantities directly in the coverage experiment.

## 4 Experiments

### 4.1 Protocol

We evaluate RepairSpace on two complementary multi-task manipulation benchmarks with disjoint train and unseen-test task identities. **LIBERO-40** is the union of the four 10-task LIBERO suites (LIBERO-Spatial, LIBERO-Object, LIBERO-Goal, and LIBERO-10), with 32 training tasks and 8 unseen test tasks. **RLBench-30** is a fixed diversity-balanced 30-task subset spanning relocation, articulated interaction, insertion and alignment, tool use, container transfer, and multi-stage manipulation, with 24 training tasks and 6 unseen test tasks. Three failure-generating policy families—Diffusion Policy Chi et al. (2025), ACT-style action chunking Zhao et al. (2023), and a VLA policy Kim et al. (2024; 2025)—produce 28,000 natural failures. Training uses 22,400 failures; 5,600 failures belong to held-out tasks. The main response protocol contains 4.24M recovery rollouts, including sparse training entries, dense anchors, geometry probes, random audit entries, and held-out intervention probes. A separate high-precision diagnostic adds 0.307M repeated rollouts and is reported independently in the appendix. Table 1 summarizes the main split; Appendix Tables 5, 6, 7, and 8 specify the task identities and response allocation.

The model uses DINOv2 ViT-B/14 visual features, a six-layer temporal Transformer, a 128-dimensional unit latent, and a three-layer response head. We train for 120k AdamW steps with $\lambda_{\text{ord}} = 0.35$. Full architecture and optimization settings are listed in Table 9.

**Baselines.** We compare against retry/no recovery, the globally best feasible intervention, semantic-taxonomy routing with predicted and oracle labels, generic $K$-means routing, frozen and fine-tuned generic representations with response heads, a last-frame router, a direct full-trajectory response predictor optimized with $\mathcal{L}_{\text{resp}}$, a privileged simulator-state router, and a cross-fitted failure-specific oracle. We also evaluate two published end-to-end recovery systems: REFLECT Liu et al. (2023), which maps robot experience to failure explanations and correction plans, and FLARE Zhao et al. (2026), which performs autonomous failure-aware correction. Their comparison uses held-out VLA-policy failures from unseen LIBERO-40 and RLBench-30 tasks under a shared checkpoint, continuation-policy, success-criterion, and $B_{\text{main}} = 2.0$ evaluation protocol. Taxonomy and generic-cluster granularities $K \in \{6, 20, 50, 100\}$ are selected on training-task validation data.

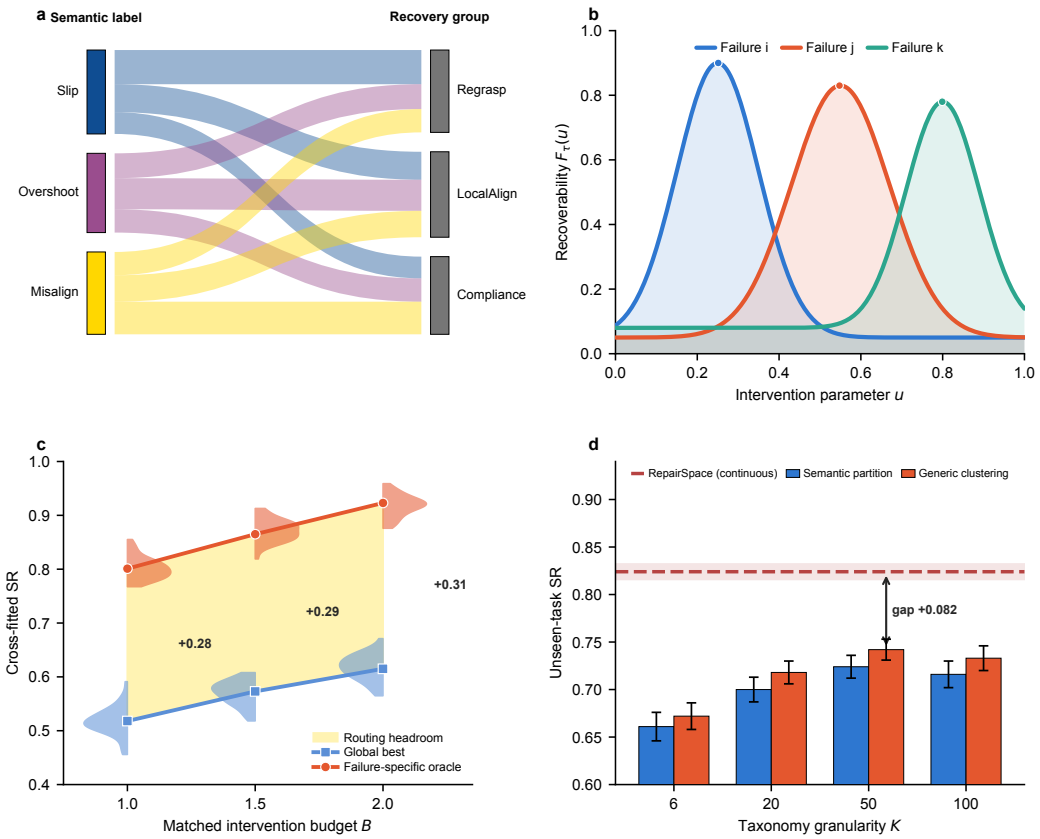



Figure 3: **Semantic labels and recoverability induce different partitions of failure space.** The figure relates semantic classes to recovery groups, visualizes representative recoverability curves, shows cost-matched routing headroom, and compares semantic partitions with generic clustering and the continuous functional representation.

**Metrics.** Recovery success rate (SR) is the primary metric. To isolate failure-conditioned routing, we report

$$\mathrm{ONR}_{\mathrm{route}} = \frac{SR_{\mathrm{method}} - SR_{\mathrm{global}}}{SR_{\mathrm{oracle}} - SR_{\mathrm{global}}},$$

and total recovery normalization relative to retry, $\mathrm{ONR}_{\mathrm{total}} = (SR_{\mathrm{method}} - SR_{\mathrm{retry}})/(SR_{\mathrm{oracle}} - SR_{\mathrm{retry}})$. Functional representation quality is measured on held-out failures and response trials using distance correlation with $d_F$ and intervention transferability $T(i \to j) = F_j(u_i^*)/\max_u F_j(u)$.

## 5 Results

### 5.1 Natural failures contain substantial recovery specialization

At the local-intervention budget $B_{\mathrm{main}}$, the best global intervention reaches 0.615 SR on LIBERO-40 and 0.589 on RLBench-30, while the cross-fitted failure-specific oracle reaches 0.923 and 0.913. The resulting specialization headroom is 0.308 and 0.324, respectively, and stays above 0.28 at lower budgets (Table 13). These gaps quantify a large, persistent benefit for failure-conditioned recovery routing.

Semantic routing captures part of the recovery structure, while functional responses resolve additional within-class variation. At its validation-selected granularity, Oracle Label + Oracle-per-Class reaches 0.724 unseen-task SR and $\mathrm{ONR}_{\mathrm{route}} = 0.489$; predicted labels reach 0.690 and 0.382. Generic $K$-means improves to 0.742 and 0.545. The direct full-trajectory response predictor reaches 0.798

Table 2: **Main recovery results at $B_{\text{main}}$.** All learned methods use matched feasible interventions and the same train/test task split. $\text{ONR}_{\text{route}}$ is reported without clipping, so values below the global intervention may be negative.

| Method | Seen SR ↑ | Unseen SR ↑ | $\text{ONR}_{route}$ ↑ | $\text{ONR}_{total}$ ↑ | Cross-policy SR ↑ | Transfer corr. ↑ | Real-robot SR ↑ |
|---|---|---|---|---|---|---|---|
| Retry / no recovery | 0.438 | 0.401 | -0.524 | 0.000 | 0.389 | – | 0.388 |
| Global Best Intervention | 0.602 | 0.568 | 0.000 | 0.344 | 0.547 | – | 0.563 |
| Predicted Taxonomy + Oracle-per-Class | 0.720 | 0.690 | 0.382 | 0.595 | 0.675 | 0.421 | 0.700 |
| Oracle Label + Oracle-per-Class | 0.758 | 0.724 | 0.489 | 0.665 | 0.708 | 0.468 | – |
| Generic $K$-means + Oracle-per-Cluster | 0.773 | 0.742 | 0.545 | 0.702 | 0.724 | 0.526 | – |
| Frozen Generic + Response Head | 0.704 | 0.674 | 0.332 | 0.562 | 0.659 | 0.418 | – |
| Fine-tuned Generic + Response Head | 0.749 | 0.720 | 0.476 | 0.656 | 0.699 | 0.557 | – |
| Last-frame-only Response Router | 0.781 | 0.755 | 0.586 | 0.728 | 0.716 | 0.581 | – |
| Direct Full-trajectory Response Predictor | 0.839 | 0.798 | 0.721 | 0.817 | 0.774 | 0.694 | 0.775 |
| Privileged-state Response Router | 0.879 | 0.842 | 0.859 | 0.907 | 0.818 | 0.812 | – |
| **RepairSpace** | **0.866** | **0.824** | **0.803** | **0.870** | **0.806** | **0.842** | **0.850** |
| Cross-fitted Oracle | 0.918 | 0.887 | 1.000 | 1.000 | 0.872 | – | – |

Table 3: **Published recovery baselines on held-out VLA-policy failures from unseen tasks.** All methods start from the same failure checkpoints and use the same continuation policy, success criterion, and matched recovery budget $B_{\text{main}} = 2.0$. Values are mean ± standard deviation over five seeds.

| Method | LIBERO-40 SR ↑ | RLBench-30 SR ↑ | Mean SR ↑ | Mean cost ↓ |
|---|---|---|---|---|
| Retry / no recovery | $0.414 \pm 0.018$ | $0.392 \pm 0.021$ | 0.403 | – |
| Global Best Intervention | $0.584 \pm 0.014$ | $0.557 \pm 0.017$ | 0.571 | 1.43 |
| REFLECT Liu et al. (2023) | $0.748 \pm 0.016$ | $0.719 \pm 0.019$ | 0.734 | 1.49 |
| FLARE Zhao et al. (2026) | $0.797 \pm 0.013$ | $0.771 \pm 0.015$ | 0.784 | 1.37 |
| **RepairSpace** | $\mathbf{0.836 \pm 0.010}$ | $\mathbf{0.818 \pm 0.012}$ | **0.827** | **1.29** |

and 0.721. REPAIRSPACE reaches 0.824 unseen-task SR and recovers 0.803 of routing headroom (Table 2). The complete $K$ sweep peaks at $K = 50$ for both semantic and generic partitions and then declines, while the functional representation remains higher (Table 15). Figure 3 shows the corresponding mismatch between semantic categories and recovery groups.

**Comparison with published recovery systems.** On held-out VLA-policy failures from unseen tasks, REFLECT reaches 0.734 mean recovery SR and FLARE reaches 0.784. REPAIRSPACE reaches 0.827, a gain of 4.3 percentage points over FLARE and 9.3 points over REFLECT, while also attaining the lowest mean normalized intervention cost (1.29). The improvement appears on both benchmarks: 0.836 on LIBERO-40 and 0.818 on RLBench-30 (Table 3).

### 5.2 FUNCTIONAL DISTANCE PREDICTS INTERVENTION TRANSFER

Held-out failure pairs, response trials, and intervention parameters provide a direct test of the learned geometry. Correlation with functional distance rises from 0.312 for raw trajectories to 0.463 for frozen generic features, 0.584 after response fine-tuning, 0.731 for response-only features, and 0.887 for REPAIRSPACE. More importantly, latent distance predicts whether the source failure's best intervention transfers to the target failure with correlation 0.842, compared with 0.694 for response-only and 0.557 for fine-tuned generic features (Table 16). The resulting neighborhoods support 0.851 SR for kNN intervention transfer and 0.820 SR for unseen-task kNN recovery.

The geometry also improves intervention selection. Under the main protocol, the full-trajectory response predictor reaches 0.798 unseen-task SR and REPAIRSPACE reaches 0.824. A strict matched-supervision comparison fixes the 268,800 observed failure–intervention entries, 1,075,200 recovery rollouts, 118M-parameter architecture, and 1,024 CEM candidate evaluations; under this control, unseen-task SR rises from 0.798 to 0.821 and cross-policy SR from 0.774 to 0.802 (Table 10). Equal-count triplet controls further separate confidence selection from constraint quantity (Table 12). The gain is largest at lower response supervision in Fig. 5.

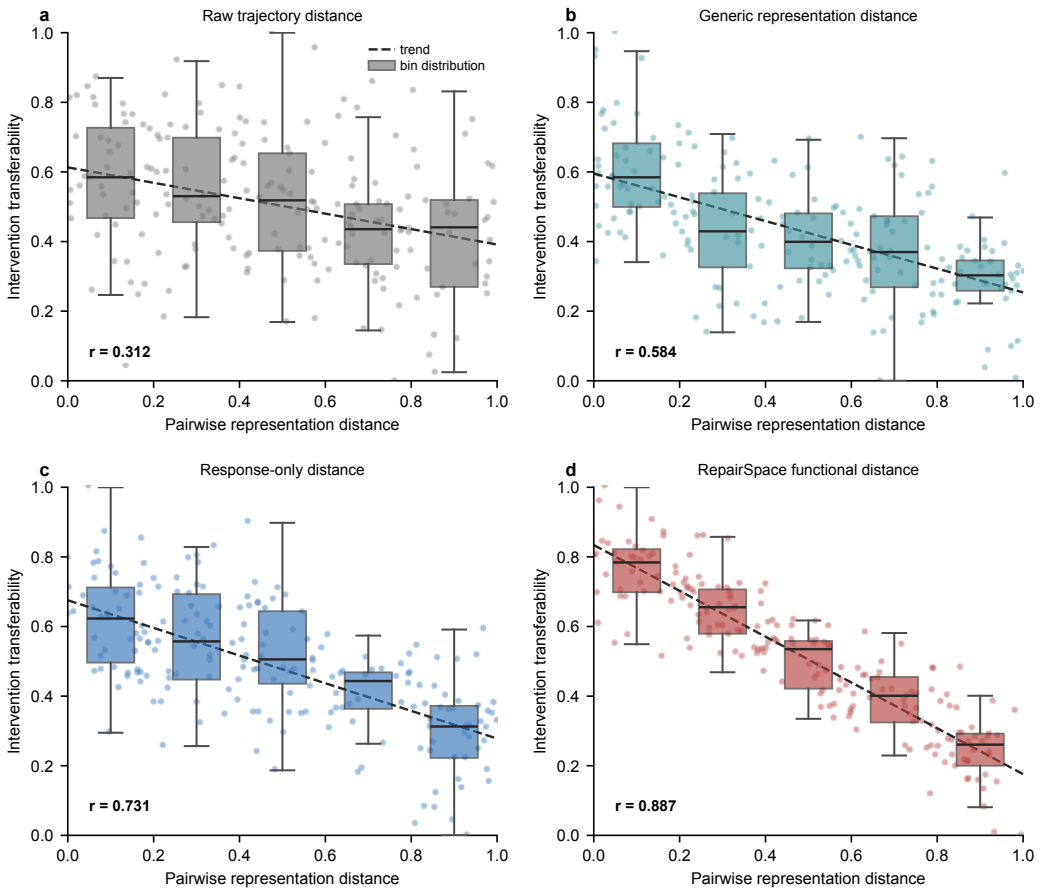



Figure 4: **Functional geometry and intervention transfer.** Boxplots show directed intervention transferability $T$ against representation distance $d$ on the same held-out failure pairs and independent recovery trials. The in-panel annotations report functional-distance agreement $r_F = \mathrm{Corr}(d, d_F)$; the corresponding distance–transfer associations $r_T = \mathrm{Corr}(-d, T)$ are reported in Table 16.

### 5.3 Zero-shot recovery transfers across unseen tasks and policies

RepairSpace is trained only on failures and response measurements from the 32 LIBERO-40 and 24 RLBench-30 training tasks. Across the 14 disjoint unseen tasks it reaches 0.824 SR, compared with 0.568 for the global intervention, 0.724 for oracle semantic routing, and 0.798 for the direct full-trajectory response predictor. Per-task results are consistently above the response-only model across the six reported unseen-task aggregates (Table 18).

Cross-policy evaluation changes the failure-generating policy while keeping the recoverability operator fixed. Training on one of the diffusion, ACT, or VLA families and testing on another yields off-diagonal recovery rates from 0.797 to 0.818, with an aggregate cross-policy SR of 0.806. The same model reaches 0.843– 0.854 on matched policy families (Table 19). The resulting neighborhoods preserve corrective structure across diffusion, action-chunking, and VLA failure generators. The same comparison remains positive when 20% of intervention-parameter cells within each family support are held out from recovery-model fitting: RepairSpace reaches 0.806 versus 0.779 for response-only on unseen tasks, and 0.789 versus 0.752 under policy shift (Table 21).

### 5.4 Coverage tracks recovery regret

As the coverage collection expands from 800 measured failures and 8 measured interventions to 12,800 failures and 64 interventions, $\hat{L}_z\epsilon + \hat{L}_u\rho$ decreases from 0.362 to 0.085 while measured

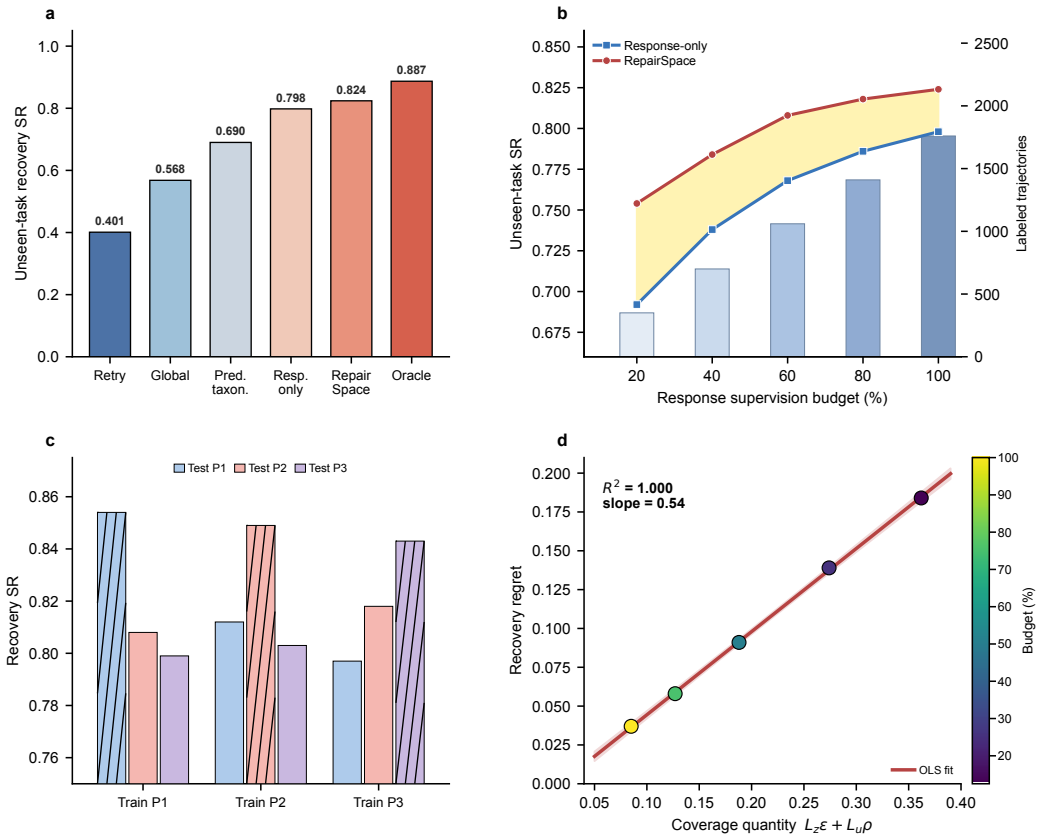



Figure 5: **Generalization beyond the training failure distribution.** (a) Unseen-task recovery at the main operating point. (b) Recovery across the reported response-supervision sweep. (c) Matched and shifted failure-generating policy families. (d) Measured recovery regret against the empirical coverage quantity $\hat{L}_z\epsilon + \hat{L}_u\rho$, with an OLS fit across the five collection settings.

recovery regret falls from 0.184 to 0.037 (Table 28). The participation-ratio effective dimension grows from 11.8 to 27.4 in the response-supervision diagnostic (Table 29).

### 5.5 Real-robot recovery

Across eight real-robot failure configurations (10 trials each), the global, semantic, and full-trajectory response baselines achieve 0.563, 0.700, and 0.775 SR. RepairSpace reaches 0.850 (68/80) with mean intervention cost 1.28 (Tables 23, 24, and 26), demonstrating transfer from large-scale simulated response measurement to physical corrective execution.

## 6 Conclusion and Limitations

RepairSpace organizes robot failures by the corrective interventions that recover them. Recoverability functions define the functional target, while uncertainty-aware ordinal learning and cost-aware response optimization support transferable recovery on unseen failures. Matched-supervision, policy-shift, held-out intervention-parameter, and real-robot evaluations support the value of the learned geometry beyond pointwise response prediction. Recoverability depends on the continuation policy, intervention library, and probe measure; changing them defines a different functional geometry.

## AI USE STATEMENT

Generative AI tools were used solely for language polishing of the manuscript, including improvements to grammar, wording, and readability. All scientific claims, analyses, experimental results, and interpretations were produced and verified by the authors.

## REFERENCES

David Abel. A theory of state abstraction for reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 9876–9877, 2019.

Faseeh Ahmad, Hashim Ismail, Jonathan Styrud, Maj Stenmark, and Volker Krueger. A unified framework for real-time failure handling in robotics using vision-language models, reactive planner and behavior trees. In *2025 IEEE 21st International Conference on Automation Science and Engineering (CASE)*, pages 887–894. IEEE, 2025.

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. $\pi_0$: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Tingjun Dai, Mingfei Han, Tingwen Du, Zhiheng Liu, Zihao Zhang, Zhihui Li, Salman Khan, Jun Yu, and Xiaojun Chang. See, plan, rewind: Progress-aware vision-language-action models for robust robotic manipulation. *arXiv preprint arXiv:2603.09292*, 2026.

Jiafei Duan, Wilbert Pumacay, Nishanth Kumar, Yi Ru Wang, Shulin Tian, Wentao Yuan, Ranjay Krishna, Dieter Fox, Ajay Mandlekar, and Yijie Guo. Aha: A vision-language-model for detecting and reasoning over failures in robotic manipulation. *arXiv preprint arXiv:2410.00371*, 2024a.

Jiafei Duan, Wentao Yuan, Wilbert Pumacay, Yi Ru Wang, Kiana Ehsani, Dieter Fox, and Ranjay Krishna. Manipulate-anything: Automating real-world robots using vision-language models. *arXiv preprint arXiv:2406.18915*, 2024b.

Norm Ferns, Prakash Panangaden, and Doina Precup. Bisimulation metrics for continuous markov decision processes. *SIAM Journal on Computing*, 40(6):1662–1714, 2011.

Norman Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite markov decision processes. *arXiv preprint arXiv:1207.4114*, 2012.

Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G Bellemare. Deepmdp: Learning continuous latent space models for representation learning. In *International conference on machine learning*, pages 2170–2179. PMLR, 2019.

Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, et al. Inner monologue: Embodied reasoning through planning with language models. *arXiv preprint arXiv:2207.05608*, 2022.

Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *arXiv preprint arXiv:2307.05973*, 2023.

Yixuan Huang, Novella Alvina, Mohanraj Devendran Shanthi, and Tucker Hermans. Fail2progress: Learning from real-world robot failures with stein variational inference. *arXiv preprint arXiv:2509.01746*, 2025.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: A vision-language-action model with open-world generalization.*arXiv preprint arXiv:2504.16054*, 2025.

Yunfan Jiang, Chen Wang, Ruohan Zhang, Jiajun Wu, and Li Fei-Fei. Transic: Sim-to-real policy transfer by learning from online correction. *arXiv preprint arXiv:2405.10315*, 2024.

Kento Kawaharazuka, Jihoon Oh, Jun Yamada, Ingmar Posner, and Yuke Zhu. Vision-language-action models for robotics: A review towards real-world applications. *IEEE Access*, 2025.

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, 2024. *URL https://arxiv. org/abs/2406.09246*, 1(2):4, 2024.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025.

Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *2023 IEEE International conference on robotics and automation (ICRA)*, pages 9493–9500. IEEE, 2023.

Ming Chong Lim and Han-Lim Choi. Enhancing generalization in reinforcement learning for robotic control through deep policy similarities. *International Journal of Advanced Robotic Systems*, 22 (2):17298806251332359, 2025.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. In *International Conference on Learning Representations*, volume 2025, pages 29982–30009, 2025.

Zeyi Liu, Arpit Bahety, and Shuran Song. Reflect: Summarizing robot experiences for failure explanation and correction. *arXiv preprint arXiv:2306.15724*, 2023.

Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

Paul Pacaud, Ricardo Garcia, Shizhe Chen, and Cordelia Schmid. Guardian: Detecting robotic planning and execution errors with vision-language models. 2025.

Shaohui Pan, Yong Xu, Ruotao Xu, Zihan Zhou, Si Wu, and Zhuliang Yu. Self-correcting robot manipulation via gaussian-splatted foresight. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 26642–26650, 2025.

Soujanya Poria, Navonil Majumder, Chia-Yu Hung, Amir Ali Bagherzadeh, Chuan Li, Kenneth Kwok, Ziwei Wang, Cheston Tan, Jiajun Wu, and David Hsu. 10 open challenges steering the future of vision-language-action models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 39771–39779, 2026.

Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, et al. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*, 2025.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

Yixuan Wang, Guang Yin, Binghao Huang, Tarik Kelestemur, Jiuguang Wang, and Yunzhu Li. Gendp: 3d semantic fields for category-level generalizable diffusion policy. *CoRL*, 1:2, 2024.

Shijie Wu, Yihang Zhu, Yunao Huang, Kaizhen Zhu, Jiayuan Gu, Jingyi Yu, Ye Shi, and Jingya Wang. Afforddp: Generalizable diffusion policy with transferable affordance. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6971–6980. IEEE, 2025.

Jingyun Yang, Zi-ang Cao, Congyue Deng, Rika Antonova, Shuran Song, and Jeannette Bohg. Equibot: Sim (3)-equivariant diffusion policy for generalizable and data efficient learning. *arXiv preprint arXiv:2407.01479*, 2024.

Quantao Yang, Michael C Welle, Danica Kragic, and Olov Andersson. $S^2$-Diffusion: Generalizing from instance-level to category-level skills in robot manipulation. *IEEE Robotics and Automation Letters*, 2025.

Chenduo Ying, Linkang Du, Yuanchao Shu, and Peng Cheng. Robofailring: Retrieval-augmented and language grounding failure detection for vlm-enabled robotic manipulation. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13188–13202, 2026.

Xianchao Zeng, Xinyu Zhou, Youcheng Li, Jiayou Shi, Tianle Li, Liangming Chen, Lei Ren, and Yong-Lu Li. Diagnose, correct, and learn from manipulation failures via visual symbols. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 42386–42395, 2026.

Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. *arXiv preprint arXiv:2006.10742*, 2020.

Yanyan Zhang, Disheng Liu, Kai Ye, Chaoda Song, Xinpeng Li, Mohsen Hariri, Vikash Singh, Yu Yin, and Vipin Chaudhary. Imagining recovery: Inference-time counterfactual realignment for vision-language-action models. *arXiv preprint arXiv:2608.14822*, 2026.

Ganlong Zhao, Zijia Tang, Xingping Chen, Zhanghui Kuang, Ye Tian, and Guanbin Li. Flare: A failure-aware framework for autonomous correction and recovery in visual-language robotic manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22391–22401, 2026.

Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware, 2023. *URL https://arxiv. org/abs/2304.13705*, 2, 2023.

Peiyuan Zhi, Zhiyuan Zhang, Yu Zhao, Muzhi Han, Zeyu Zhang, Zhitian Li, Ziyuan Jiao, Baoxiong Jia, and Siyuan Huang. Closed-loop open-vocabulary mobile manipulation with gpt-4v. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4761–4767. IEEE, 2025.

# APPENDIX CONTENTS

This appendix develops the experimental specification, transfer analyses, metric definitions, implementation details, and operator-specific interpretation of recoverability that support the main-text results.


| Section | Contents | Page |
|---|---|---|
| A | Full Experimental Specification | 14 |
| A.1 | Natural failure collection | 14 |
| A.2 | Parameterized intervention space | 14 |
| A.3 | Task splits and response measurement | 15 |
| A.4 | Model configuration and supervision | 17 |
| A.5 | Specialization, cost, and semantic granularity | 19 |
| A.6 | Functional geometry and intervention transfer | 22 |
| A.7 | Recovery on unseen tasks | 24 |
| A.8 | Transfer across failure-generating policies | 26 |
| A.9 | State information, history, and ordinal structure | 27 |
| A.10 | Real-robot configurations and recovery | 29 |
| A.11 | Training and measurement resources | 30 |
| B | Recoverability Transfer Bound | 31 |
| B.1 | Pointwise supervision and ordinal geometry | 32 |
| B.2 | Coverage and measured recovery regret | 32 |
| B.3 | Effective dimension and latent coverage | 34 |
| C | Metrics | 35 |
| D | Implementation Details | 36 |
| E | Recoverability Operator Scope | 37 |

# A Full Experimental Specification

The evaluation follows the recoverability function from measurement to use. Matched checkpoints establish comparable responses; task and policy splits examine transfer of those responses; semantic, temporal, and response-only comparators identify which distinctions are useful for routing. Physical execution then measures recovery under the same intervention-based formulation. Throughout, the failure-generating policy $\pi_g$ determines which failed states are encountered, whereas the continuation policy $\pi_c$ defines how task success is assessed after a correction.

## A.1 Natural failure collection

For each LIBERO-40 and RLBench-30 task, a failure-generating policy $\pi_g$ is executed under its benchmark task protocol. We use three policy families—a diffusion policy, an action-chunking policy, and a vision–language–action policy—to obtain failure states with distinct inductive biases and execution traces. The external failure detector evaluates benchmark state predicates for contact violation, object loss, failed subgoals, and progress stagnation. Its first trigger defines

$$t_0 = \min\{t : D_{\text{fail}}(s_t) = 1\}.$$

At $t_0$ we store the visual history, proprioceptive and task state, action history, task condition, and an exact simulator snapshot $s_{t_0}$. Every candidate intervention for the same failure is re-executed from this identical snapshot. After the corrective intervention terminates, a fixed continuation policy $\pi_c$ resumes the task until success or the benchmark horizon.

The paired-reset design makes the intervention the changing element within a failure's response profile. An unsuccessful correction cannot alter the starting state of the next measurement, and successful recovery has the same task-level meaning across candidate corrections. Repeated Bernoulli trials then estimate uncertainty around each response. This construction is the basis for comparing two failures through $F_\tau$: similarity concerns what corrections accomplish from their checkpoints, rather than the visual resemblance of their recorded trajectories.

## A.2 Parameterized intervention space

The corrective library contains eight physically interpretable intervention families with continuous parameters (Table 4). An intervention is $u = (c, \theta) \in \mathcal{U}$, where $c$ selects a family and $\theta \in \Theta_c$ gives its control parameters. The main probe distribution is uniform over families and follows the family-specific parameter sampling rules in the table. For a failure $\tau$ and probe $u_m$, we estimate

$$\hat{F}_\tau(u_m) = \frac{1}{n_m} \sum_{r=1}^{n_m} Y_{m,r}, \qquad Y_{m,r} \in \{0, 1\},$$

using independent resets to $s_{t_0}$. Dense anchors use all 64 candidate interventions with 12 rollouts per pair. At inference, cross-entropy method (CEM) optimizes continuous parameters while the family indicator remains discrete.

Table 4: **Parameterized corrective intervention space.** The family and parameter support define the domain of each recoverability function. The cost constraint selects the feasible subset used for recovery; at $B_{\text{main}} = 2.0$, the six local families are feasible, while Reset and Replan enter at higher budgets.

| Family | Parameters | Range / support | Cost $C(u)$ | Sampling distribution | Execution description |
|---|---|---|---|---|---|
| Retract | $d$ | $d \in [0.02, 0.12]$ m | 0.6–1.0 | Uniform in $d$ | Back away along the end-effector approach axis. |
| ReApproach | $(d, \Delta\psi)$ | $d \in [0.02, 0.10]$ m, $\Delta\psi \in [-15^\circ, 15^\circ]$ | 0.8–1.2 | Uniform over normalized parameters | Re-enter the local approach corridor with a corrected yaw. |
| Regrasp | $(\Delta\mathbf{p}, \phi)$ | $\|\Delta\mathbf{p}\|_\infty \leq 4$ cm, $\phi \in [-20^\circ, 20^\circ]$ | 1.3–1.8 | Family-uniform, parameter-LHS | Release and execute a local parameterized regrasp. |
| LocalAlign | $(\Delta\mathbf{p}, \Delta\phi)$ | $\|\Delta\mathbf{p}\|_\infty \leq 3$ cm, $\Delta\phi \in [-12^\circ, 12^\circ]$ | 0.8–1.4 | Parameter-volume balanced | Correct local pose without fully releasing the object. |
| Compliance | $(k_p, k_d)$ | $k_p \in [0.4, 1.0]$, $k_d \in [0.3, 0.9]$ (normalized) | 0.7–1.2 | Uniform in normalized gains | Temporarily reduce stiffness and continue under compliant control. |
| Relocalize | $(r, \psi)$ | $r \in [0.02, 0.08]$ m, $\psi \in [-30^\circ, 30^\circ]$ | 1.5–2.0 | Uniform over family support | Move to a nearby observation pose and reacquire the target. |
| Reset | – | Task-defined safe reset state | 3.5 | Deterministic | Return to a safe task checkpoint before continuation. |
| Replan | $(h, \beta)$ | $h \in \{2, 3, 4\}$ subgoals, $\beta \in [0, 1]$ | 3.0–4.0 | Uniform over templates | Invoke a global task-level replan from the failed state. |

Table 4 makes the distinction between a failure label and a recovery decision explicit. Even after a family such as Regrasp or LocalAlign has been selected, the displacement, orientation, or control gains must fit the current checkpoint. A single semantic category therefore leaves a continuous intervention-selection problem unresolved. The response head represents this remaining dependence through $g_\phi(z, u)$, and the geometry connects failures whose responses vary similarly across that domain. The normalized cost budget provides a common decision constraint: performance at $B_{\text{main}}$ measures selection among local corrections, whereas broader budgets change the repertoire itself. The full eight-family measurement domain and the budget-feasible deployment domain have separate roles in this design.

### A.3 Task splits and response measurement

Task identity is the unit of the train–test separation. The fixed splits retain multiple kinds of manipulation in training while holding out particular task instances for recovery evaluation. Tables 5

and 6 specify those identities; the subsequent tables connect the task split to failure records and intervention measurements.

Table 5: **Fixed LIBERO-40 task split.** Task IDs follow the standard ordering within each 10-task suite. Each suite contributes eight training tasks and two unseen test tasks, preserving all four suite types on both sides of the task-identity split.

| Suite | Training task IDs | Unseen IDs | Unseen test task identities |
|---|---|---|---|
| **LIBERO-Spatial** | 0, 1, 3, 4, 5, 6, 8, 9 | 2, 7 | `pick_up_the_black_bowl_from_table_center_and_place_it_on_the_plate`<br>`pick_up_the_black_bowl_on_the_stove_and_place_it_on_the_plate` |
| **LIBERO-Object** | 0, 2, 3, 4, 5, 6, 7, 9 | 1, 8 | `pick_up_the_cream_cheese_and_place_it_in_the_basket`<br>`pick_up_the_chocolate_pudding_and_place_it_in_the_basket` |
| **LIBERO-Goal** | 0, 1, 2, 4, 5, 6, 8, 9 | 3, 7 | `open_the_top_drawer_and_put_the_bowl_inside`<br>`turn_on_the_stove` |
| **LIBERO-10** | 0, 1, 3, 5, 6, 7, 8, 9 | 2, 4 | `KITCHEN_SCENE3_turn_on_the_stove_and_put_the_moka_pot_on_it`<br>`LIVING_ROOM_SCENE5_put_the_white_mug_on_the_left_plate_and_put_the_yellow_and_white_mug_on_the_right_plate` |

The balance in Table 5 gives the unseen-task comparison a specific interpretation. Spatial relations, object identities, task goals, and multi-stage contexts all remain represented in the training collection, while the listed test tasks supply new failures on which the learned response organization is evaluated. Recoverability can therefore transfer across task identity within these represented manipulation settings. This is the appropriate level at which to read the zero-shot claim: a test failure is mapped to a response-relevant neighborhood without fitting the recovery model to that held-out task's response outcomes.

Table 6: **Fixed diversity-balanced RLBench-30 subset.** Thirty tasks are grouped into six manipulation families. Four tasks per family are used for training and one is held out, giving 24 training and 6 unseen tasks.

| Manipulation family | Training tasks | Unseen test task |
|---|---|---|
| **Pick / place / relocation** | PickAndLift; PickUpCup; PutBottleInFridge; PutRubbishInBin | **PhoneOnBase** |
| **Articulated-object interaction** | OpenDrawer; CloseDrawer; OpenDoor; CloseDoor | **OpenMicrowave** |
| **Insertion / alignment** | InsertOntoSquarePeg; PlugChargerInPowerSupply; PlaceShapeInShapeSorter; PutKnifeInKnifeBlock | **InsertUsbInComputer** |
| **Tool / contact-rich control** | SweepToDustpan; ScrewNail; WipeDesk; Hockey | **ScoopWithSpatula** |
| **Container / transfer** | EmptyContainer; PutGroceriesInCupboard; TakeItemOutOfDrawer; PutItemInDrawer | **PourFromCupToCup** |
| **Multi-stage manipulation** | SetTheTable; StackCups; StackBlocks; SlideCabinetOpenAndPlaceCups | **PutBooksOnBookshelf** |

Table 6 extends the same test to different corrective demands. Insertion, articulated interaction, container transfer, and tool use place different requirements on alignment, contact, and task continuation. Holding out one task within each family asks whether corrective experience survives a change of task while the broad manipulation setting remains represented. The organization by manipulation family describes the evaluation coverage; the learned neighborhoods are instead determined by intervention responses. Together with the LIBERO split, this design tests the paper's functional notion of similarity across both task variation and distinct forms of manipulation.

Table 7: **Benchmark, task, policy, and failure-data splits.** The task identities are listed in Tables 5 and 6. Training and unseen-test failure counts correspond to disjoint task identities; dense-anchor and audit columns identify response measurement roles.

| Benchmark | Train tasks | Test tasks | $\pi_g$ family | Train failures | Test failures | Dense anchors | Audit failures | Seeds |
|---|---|---|---|---|---|---|---|---|
| LIBERO-40 | 32 | 8 | Diffusion / ACT / VLA | 12,800 | 3,200 | 800 | 1,600 | 5 |
| RLBench-30 | 24 | 6 | Diffusion / ACT / VLA | 9,600 | 2,400 | 600 | 1,200 | 5 |

Table 7 connects the generalization unit to the scale of the observations. The 56 training tasks contribute 22,400 failure records, and the 14 unseen tasks contribute 5,600, giving 28,000 natural failures across the two benchmarks. Because the split is at the task level, recovery on the unseen set evaluates reuse of corrective structure beyond the task identities used for learning. The three failure-generating policy families introduce a second axis of variation: different policies can reach different failed states within a task. The dense-anchor and audit allocations support measurement of the resulting response functions, whose rollout budgets are specified separately in Table 8.

Table 8: **Recoverability-function measurement budget.** The rows separate pointwise training responses, ordinal-geometry probes, dense reference profiles, random audit entries, and held-out intervention evaluation. Rollout counts record repeated recovery measurements for each failure–intervention pair.

| Split | Failures | Interventions per failure | Rollouts per pair | Total rollouts | Use |
|---|---|---|---|---|---|
| Training response entries | 22,400 | 12 | 4 | 1,075,200 | Pointwise response supervision |
| Geometry probes | 6,000 | 24 | 8 | 1,152,000 | Ordinal triplet construction |
| Dense anchor subset | 1,400 | 64 | 12 | 1,075,200 | Dense $F_\tau$ and headroom analysis |
| Random audit entries | 2,800 | 16 | 12 | 537,600 | Unbiased held-out audit |
| Held-out intervention set | 2,800 | 12 | 12 | 403,200 | Transfer and calibration evaluation |

The allocation in Table 8 separates breadth of response coverage from precision at a measured pair. A sparse training failure uses $12 \times 4 = 48$ rollouts, while a dense anchor uses $64 \times 12 = 768$. The former supplies many partially observed response functions; the latter supports detailed functional comparisons and failure-specific headroom estimation. Geometry probes contribute confidence-ordered relations between failures, while audit and held-out intervention measurements assess responses beyond the fitting entries. Summing the listed measurement entries gives 580,800 failure–intervention pairs and 4,243,200 rollouts. This distinction matters for the learning problem: the model must infer useful response structure from sparse observations, while the evaluation needs sufficiently resolved response profiles to assess whether the learned neighborhoods actually transfer corrections.

### A.4 Model configuration and supervision

The architecture maps a failure trajectory to a unit-normalized latent vector and combines that vector with a parameterized intervention to predict recovery. Table 9 specifies the encoders, response head, optimizer, and ordinal-supervision settings.

Table 9: **Model architecture and optimization hyperparameters.** The temporal representation and intervention-conditioned response head are trained jointly with pointwise and confidence-filtered ordinal supervision. CEM supplies the parameter search used by the recovery decision.

| Component | Configuration | Hyperparameter | Value |
|---|---|---|---|
| Visual backbone | DINOv2 ViT-B/14; final four blocks fine-tuned | Optimizer | AdamW |
| State/action encoder | 2-layer MLP, width 256, Fourier pose features | Learning rate | $3 \times 10^{-4}$ |
| Temporal encoder | 6-layer Transformer, width 512, 8 heads | Batch size | 256 |
| Latent dimension $d$ | 128 | Training steps | 120,000 |
| Intervention encoder | 64-d family embedding + 2-layer parameter MLP | $\lambda_{\mathrm{ord}}$ | 0.35 |
| Response head | 3-layer MLP, 512–256–1 with sigmoid output | Ordinal temperature $\tau_z$ | 0.10 |
| Candidate optimizer | CEM, 256 candidates $\times$ 4 iterations | Triplet confidence $\kappa$ | 2.0 |
| – | – | Weight decay | 0.05 |

The components in Table 9 implement complementary parts of the recoverability formulation. The visual and state/action streams provide evidence about the checkpoint; the temporal encoder combines observations whose meaning depends on the preceding execution. The intervention embedding retains both family identity and within-family parameters, so the output is a response surface rather than a categorical failure prediction. Unit normalization gives the ordinal objective a common latent distance scale. Pointwise supervision then fits individual response values, while $\lambda_{\mathrm{ord}}$, $\tau_z$, and $\kappa$ govern how reliably ordered response profiles shape neighborhoods. The matched 118M-parameter response-only and REPAIRSPACE models isolate the contribution of relational supervision from model capacity. The experiments below additionally match the observed outcomes and CEM candidate-evaluation budget, then vary the geometry objective itself.

Table 10: **Matched-supervision mechanism comparison.** All rows in the matched block use the same 268,800 observed failure–intervention entries, 1,075,200 recovery rollouts, 118M-parameter architecture, and 1,024 CEM candidate evaluations. $\Delta$SR is measured against the matched response-only model.

| Method | Unseen SR | Cross-policy SR | $\Delta$SR | 95% paired CI | $r_T$ | Measurements | Params | Candidate evals |
|---|---|---|---|---|---|---|---|---|
| Response-only, matched outcomes | 0.798 | 0.774 | – | – | 0.694 | 268.8k | 118M | 1024 |
| Response + distance regression | 0.804 | 0.781 | +0.006 | [0.004, 0.008] | 0.728 | 268.8k | 118M | 1024 |
| Response + ordinal, no confidence filtering | 0.809 | 0.787 | +0.011 | [0.008, 0.014] | 0.758 | 268.8k | 118M | 1024 |
| Response + confidence ordinal, no variance correction | 0.814 | 0.793 | +0.016 | [0.013, 0.019] | 0.786 | 268.8k | 118M | 1024 |
| **Full RepairSpace, matched supervision** | **0.821** | **0.802** | **+0.023** | **[0.020, 0.026]** | **0.826** | 268.8k | 118M | 1024 |
| Full RepairSpace, main probe protocol | 0.824 | 0.806 | – | – | 0.842 | 412.8k | 118M | 1024 |

The matched block separates the effect of functional geometry from additional response observations. Direct distance regression raises unseen SR to 0.804; ordinal supervision reaches 0.809; confidence-filtered ordinal supervision without variance correction reaches 0.814; and the full matched objective reaches 0.821. The main probe protocol reaches the headline 0.824 operating point.

Table 11: **Seed-level stability under matched supervision.** The same five training seeds are reported for unseen-task and cross-policy recovery.

| Evaluation | Method | Seed 1 | Seed 2 | Seed 3 | Seed 4 | Seed 5 | Mean |
|---|---|---|---|---|---|---|---|
| Unseen-task SR | Response-only | 0.793 | 0.801 | 0.796 | 0.805 | 0.795 | 0.798 |
| | Distance regression | 0.798 | 0.808 | 0.801 | 0.809 | 0.804 | 0.804 |
| | Ordinal, no confidence | 0.804 | 0.813 | 0.805 | 0.814 | 0.809 | 0.809 |
| | Confidence ordinal, no variance correction | 0.809 | 0.818 | 0.810 | 0.819 | 0.814 | 0.814 |
| | **Full matched RepairSpace** | **0.816** | **0.826** | **0.815** | **0.827** | **0.821** | **0.821** |
| Cross-policy SR | Response-only | 0.767 | 0.778 | 0.771 | 0.780 | 0.774 | 0.774 |
| | Distance regression | 0.774 | 0.785 | 0.777 | 0.786 | 0.783 | 0.781 |
| | Ordinal, no confidence | 0.780 | 0.791 | 0.783 | 0.792 | 0.789 | 0.787 |
| | Confidence ordinal, no variance correction | 0.786 | 0.797 | 0.789 | 0.799 | 0.794 | 0.793 |
| | **Full matched RepairSpace** | **0.795** | **0.806** | **0.798** | **0.808** | **0.803** | **0.802** |

Table 12: **Confidence filtering versus triplet quantity.** The equal-count random control uses the same 0.92M ordinal constraints as the confidence-filtered variants.

| Ordinal supervision | Triplets | Unseen SR | $r_T$ |
|---|---|---|---|
| None | 0 | 0.798 | 0.694 |
| All candidate triplets | 4.80M | 0.809 | 0.758 |
| Random equal-count subset | 0.92M | 0.806 | 0.742 |
| Confidence-filtered, no variance correction | 0.92M | 0.814 | 0.786 |
| **Confidence-filtered + variance correction** | **0.92M** | **0.821** | **0.826** |

At equal triplet count, confidence filtering raises unseen SR from 0.806 to 0.814, and variance correction raises it to 0.821. The same ordering appears in $r_T$, connecting the supervision design to both recovery and geometry quality.

### A.5 Specialization, cost, and semantic granularity

Failure-conditioned representation becomes useful when different checkpoints benefit from different corrections. The cost-aware oracle comparison measures that opportunity before examining how much of it semantic partitions and learned functional neighborhoods recover.

Table 13: **Cost-aware cross-fitted recovery specialization headroom.** $H^{\mathrm{CF}}(B)$ is the success-rate difference between the failure-specific oracle and the global intervention at the same feasible-action budget. The final column gives the reported 95% confidence interval for this headroom.

| Setting | Budget $B$ | $SR^{CF}_{global}(B)$ | $SR^{CF}_{oracle}(B)$ | $H^{CF}(B)$ | 95% CI |
|---|---|---|---|---|---|
| LIBERO-40 | $B_{low} = 1.0$ | 0.518 | 0.801 | 0.283 | [0.264, 0.302] |
| LIBERO-40 | $B_{mid} = 1.5$ | 0.573 | 0.865 | 0.292 | [0.273, 0.311] |
| LIBERO-40 | $B_{high} = B_{main} = 2.0$ | 0.615 | 0.923 | 0.308 | [0.289, 0.327] |
| LIBERO-40 | $\infty$ | 0.704 | 0.934 | 0.230 | [0.212, 0.248] |
| RLBench-30 | $B_{low} = 1.0$ | 0.492 | 0.778 | 0.286 | [0.265, 0.307] |
| RLBench-30 | $B_{mid} = 1.5$ | 0.548 | 0.842 | 0.294 | [0.274, 0.314] |
| RLBench-30 | $B_{high} = B_{main} = 2.0$ | 0.589 | 0.913 | 0.324 | [0.303, 0.345] |
| RLBench-30 | $\infty$ | 0.686 | 0.927 | 0.241 | [0.222, 0.260] |

Table 13 identifies a substantial value for conditioning the intervention on the failure itself. At $B_{\mathrm{main}} = 2.0$, the oracle–global gap is 0.308 on LIBERO-40 and 0.324 on RLBench-30; it already exceeds 0.28 at $B = 1.0$. The opportunity for specialization is therefore present among low-cost local corrections. Relaxing the budget from 2.0 to the unrestricted setting raises global success by 0.089 and 0.097, respectively, but raises oracle success by only 0.011 and 0.014. The smaller unrestricted gaps, 0.230 and 0.241, show that access to broader corrections substitutes for some of the value of precise routing. The central representation problem is particularly consequential in the

local-intervention regime, where choosing the correction to match the checkpoint yields the largest reported headroom.

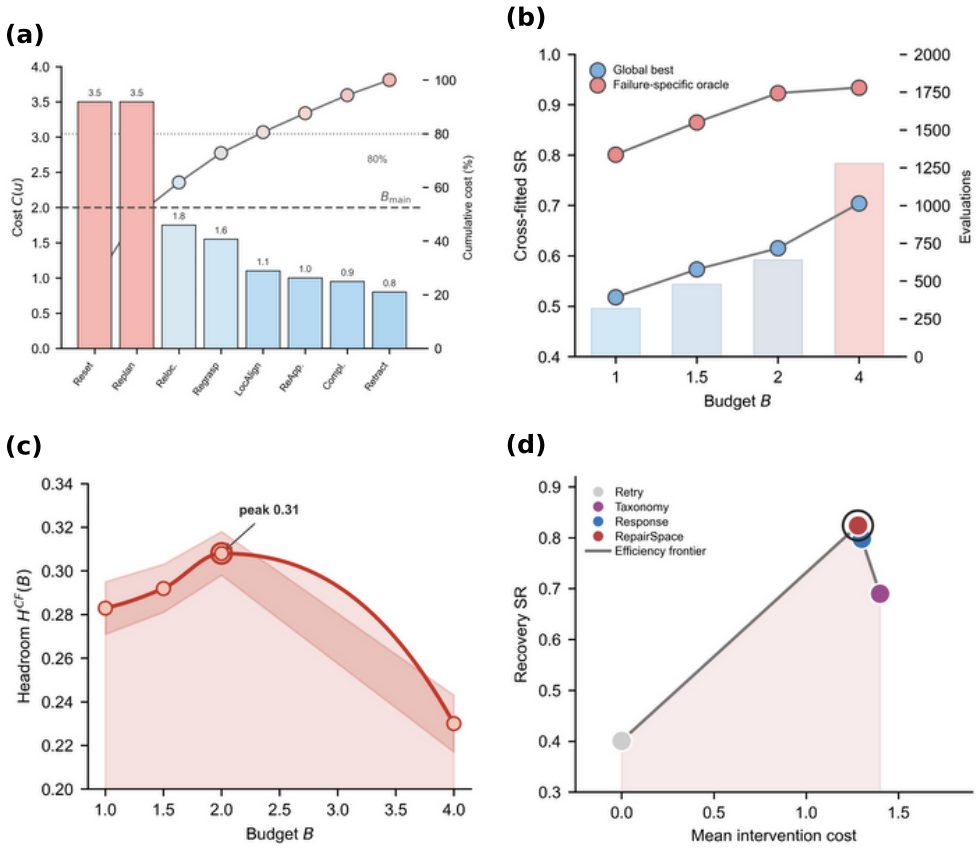



Figure 6: **The value of failure-specific routing depends on intervention cost.** (a) The displayed intervention costs and cumulative cost profile. (b) Global and failure-specific recovery as the budget expands, together with the displayed evaluation counts. (c) The resulting specialization headroom. (d) Recovery success and mean intervention cost for the displayed methods. Table 13 reports the budget-matched oracle comparison.

Figure 6 connects specialization to the decision made by a recovery system. Expanding the budget changes which interventions are available; learning a response-relevant representation changes which available intervention is selected. The rise and subsequent reduction of headroom distinguish those two sources of improvement. The success–cost panel makes the same distinction at the method level: a useful representation should move the operating point toward successful corrections at comparable or lower execution cost. Read together, these panels explain the role of $B_{\text{main}}$ in the main comparison. It fixes a local corrective repertoire within which the quality of failure-conditioned selection can be examined, while the larger-budget results describe how that selection problem changes when broader actions become feasible.

Table 14: **Semantic partitions and functional recovery at the main budget.** The table compares global selection, class- and cluster-specific oracles, pointwise response learning, and REPAIRSPACE. The normalized metrics express recovery relative to the corresponding global or retry baseline and the cross-fitted oracle.

| Method | LIBERO-40 SR | RLBench-30 SR | Seen-task SR | Unseen-task SR | $\mathrm{ONR}_{route}$ | $\mathrm{ONR}_{total}$ |
|---|---|---|---|---|---|---|
| Global Best | 0.615 | 0.589 | 0.602 | 0.568 | 0.000 | 0.344 |
| Oracle-per-Class ($K = 6$) | 0.714 | 0.695 | 0.705 | 0.661 | 0.292 | 0.535 |
| Oracle-per-Class ($K = 20$) | 0.748 | 0.731 | 0.744 | 0.700 | 0.414 | 0.615 |
| Oracle-per-Class ($K = 50$) | 0.764 | 0.752 | 0.758 | 0.724 | 0.489 | 0.665 |
| Oracle-per-Class ($K = 100$) | 0.757 | 0.745 | 0.751 | 0.716 | 0.464 | 0.648 |
| Generic-cluster Oracle ($K = 20$) | 0.762 | 0.748 | 0.758 | 0.718 | 0.470 | 0.652 |
| Generic-cluster Oracle ($K = 50$) | 0.778 | 0.768 | 0.773 | 0.742 | 0.545 | 0.702 |
| Generic-cluster Oracle ($K = 100$) | 0.772 | 0.760 | 0.767 | 0.733 | 0.517 | 0.683 |
| Response-only | 0.846 | 0.832 | 0.839 | 0.798 | 0.721 | 0.817 |
| **RepairSpace** | 0.874 | 0.858 | 0.866 | 0.824 | 0.803 | 0.870 |
| Cross-fitted Oracle | 0.923 | 0.913 | 0.918 | 0.887 | 1.000 | 1.000 |

Table 14 separates three levels of recovery information. Oracle semantic labels identify a descriptive category, generic clusters add continuous-feature similarity before forming a partition, and response learning conditions the decision directly on intervention outcomes. The best listed semantic and generic partitions reach 0.724 and 0.742 unseen-task SR; the full-trajectory response-only model reaches 0.798. Adding ordinal functional structure raises unseen SR to 0.824 and routing-normalized recovery from 0.721 to 0.803. The 2.6-percentage-point comparison with response-only is especially informative because both models learn intervention-conditioned responses: it measures the additional benefit associated with organizing those responses relationally. The larger gap to semantic routing locates the paper's main conceptual distinction—a description of how a failure is named provides a different basis for choosing a correction than its response across candidate interventions.

Table 15: **Cross-fitted taxonomy granularity sweep.** Partitions are formed without recovery outcomes. Class- or cluster-specific interventions are selected on a selection split and evaluated on an independent split. $K^{\star}_{\mathrm{sem}}$ and $K^{\star}_{\mathrm{gen}}$ are selected on held-out validation tasks from the training set and then frozen.

| Partition | $K$ | Mean LIBERO train failures / class | Validation SR | Seen-test SR | Unseen-test SR | $\mathrm{ONR}_{route}$ | Gap to RepairSpace |
|---|---|---|---|---|---|---|---|
| Semantic taxonomy | 6 | 2,133 | 0.673 | 0.705 | 0.661 | 0.292 | 0.163 |
| Semantic taxonomy | 20 | 640 | 0.712 | 0.744 | 0.700 | 0.414 | 0.124 |
| Semantic taxonomy | 50 | 256 | 0.735 | 0.758 | 0.724 | 0.489 | 0.100 |
| Semantic taxonomy | 100 | 128 | 0.728 | 0.751 | 0.716 | 0.464 | 0.108 |
| Generic clustering | 6 | 2,133 | 0.682 | 0.713 | 0.672 | 0.326 | 0.152 |
| Generic clustering | 20 | 640 | 0.728 | 0.758 | 0.718 | 0.470 | 0.106 |
| Generic clustering | 50 | 256 | 0.750 | 0.773 | 0.742 | 0.545 | 0.082 |
| Generic clustering | 100 | 128 | 0.744 | 0.767 | 0.733 | 0.517 | 0.091 |
| Predicted semantic labels ($K^{\star}_{\mathrm{sem}}$) | 50 | 256 | 0.704 | 0.720 | 0.690 | 0.382 | 0.134 |
| Oracle semantic labels ($K^{\star}_{\mathrm{sem}}$) | 50 | 256 | 0.735 | 0.758 | 0.724 | 0.489 | 0.100 |
| **RepairSpace** | – | – | – | 0.866 | 0.824 | 0.803 | – |

The sweep in Table 15 shows why partition resolution and functional alignment are separate questions. Increasing semantic granularity from $K = 6$ to $K = 50$ improves unseen SR from 0.661 to 0.724, but $K = 100$ reduces it to 0.716. Generic clustering follows the same pattern, peaking at 0.742 before falling to 0.733. Finer partitioning therefore helps over part of the tested range and then loses effectiveness as each partition becomes more specific. At the selected semantic granularity, supplying oracle instead of predicted labels improves SR from 0.690 to 0.724, identifying a 3.4-point label prediction gap. The remaining 10.0-point difference from REPAIRSPACE is a different contrast: even correct membership in the selected semantic hierarchy leaves response-relevant variation unresolved. These results locate the advantage in the evaluated response-based organization rather than in label prediction alone.

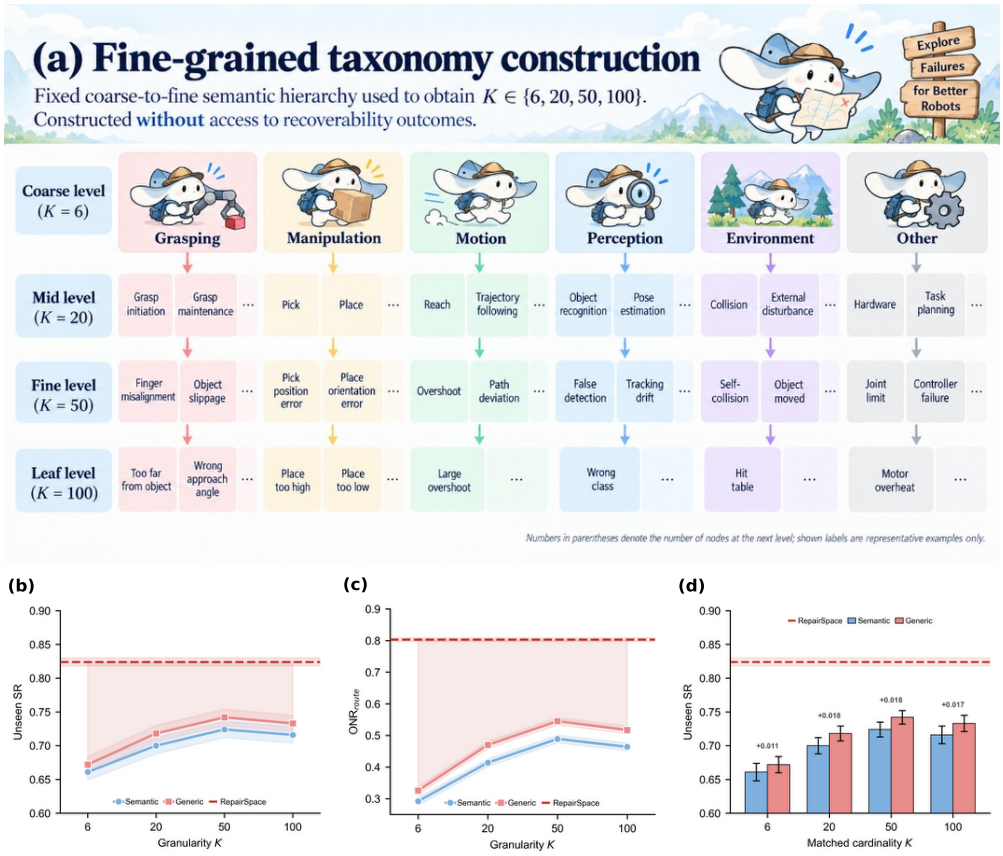



Figure 7: **Semantic resolution and response-based organization lead to different recovery performance.** (a) The coarse-to-fine semantic hierarchy. (b) Unseen-task recovery across partition granularities. (c) Routing-normalized recovery. (d) Semantic and generic partitions at matched cardinality. All partitions are constructed without recovery outcomes; the continuous functional representation provides the reference operating point.

Figure 7 makes the form of the representation comparison explicit. The semantic hierarchy refines descriptions of the event; the generic partitions refine neighborhoods in an existing feature space. Neither operation directly orders failures by the consequences of candidate corrections. The matched-cardinality comparison shows a consistent advantage for the generic partitions within the displayed sweep, while both remain below the functional model. This pattern supports two complementary observations: continuous features contain corrective information beyond the tested labels, and response supervision supplies a further organizing signal. The contribution of REPAIRSPACE is to make that signal define the neighborhood itself, so that similarity is evaluated through the intervention that can be transferred.

### A.6 FUNCTIONAL GEOMETRY AND INTERVENTION TRANSFER

A recoverability representation is useful when its neighborhoods preserve corrective decisions. The following analyses connect latent similarity first to measured response profiles and then to the success of transferred interventions.

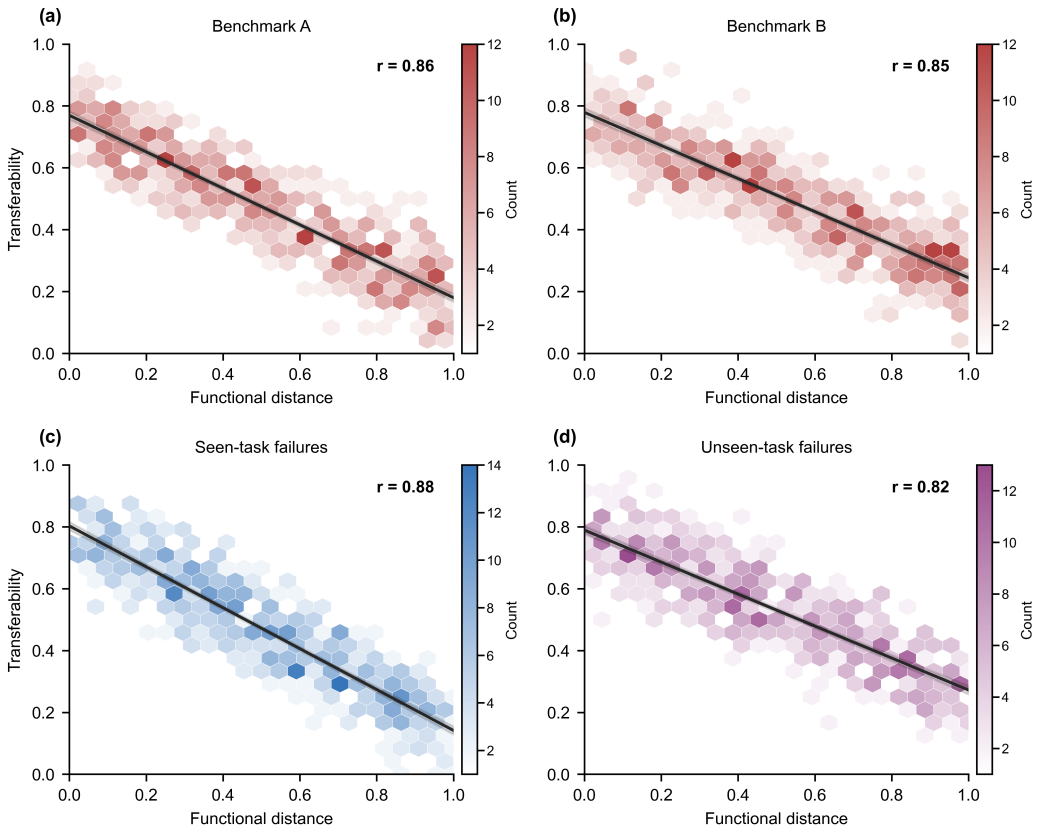



Figure 8: **Functional neighborhoods preserve intervention transfer across the displayed strata.** The panels compare functional distance and intervention transferability for the two benchmark strata (a,b) and the seen- and unseen-task strata (c,d). Density indicates the distribution of failure pairs. The relationship tests whether a correction selected for a source failure remains useful at a target failure as their response profiles become more similar.

The repeated distance–transfer relationship in Figure 8 gives the geometry an operational interpretation. Nearby failures are useful to one another because a source-selected correction retains value at the target, which is the specific form of experience reuse required for recovery. The seen- and unseen-task panels extend this comparison beyond the task identities used for learning. Density and spread remain important: the neighborhood supplies evidence about transferability rather than assigning a deterministic outcome to every pair. The directed transfer metric also evaluates a different property from symmetric functional distance, because it compares one source-selected action with the target's attainable recovery. The combination tests whether similarity preserves a decision, as well as a response profile.

Table 16: **Held-out functional-geometry quality.** The first correlation compares representation distance with functional distance; the second compares negative representation distance with directed intervention transferability. The final columns evaluate actual recovery through nearest-neighbor intervention transfer.

| Representation | Functional-distance agreement $r_F$ | Distance–transfer association $r_T$ | Transfer rank $R^2$ | kNN transfer SR | Unseen-task kNN SR |
|---|---|---|---|---|---|
| Raw trajectory | 0.312 | 0.281 | 0.112 | 0.671 | 0.642 |
| Frozen generic | 0.463 | 0.418 | 0.224 | 0.704 | 0.675 |
| Fine-tuned generic | 0.584 | 0.557 | 0.356 | 0.741 | 0.711 |
| Response-only | 0.731 | 0.694 | 0.518 | 0.801 | 0.773 |
| **RepairSpace** | 0.887 | 0.842 | 0.731 | 0.851 | 0.820 |

Table 16 links representational agreement to corrective utility. For REPAIRSPACE, the correlation with functional distance is 0.887, whereas the correlation between $-d$ and transferability is 0.842; these are distinct measurements of profile agreement and action transfer. Relative to response-only features, the latter increases from 0.694 to 0.842, while kNN transfer SR increases from 0.801 to 0.851 and unseen-task kNN SR from 0.773 to 0.820. The recovery comparisons give the geometric improvement practical meaning: the neighborhood itself supports more successful corrections. kNN transfer remains a diagnostic decision rule, whereas the deployed model optimizes the learned response head. Their separate roles allow the evaluation to examine the usefulness of the representation as well as the complete response-based selector.

Table 17: **Response precision, spectral diagnostics, and ordinal supervision.** The diagnostic matrix is $G_{ij}^{\mathrm{diag}} = 1 - 2\,\mathrm{clip}_{[0,1]}(\widehat{q}_{ij})$. Its spectrum characterizes compatibility with exact unit-sphere distance matching. The learned geometry uses confidence-ordered triplets, whose retained counts and held-out rank correlations are reported alongside the diagnostic.

| Setting | $M$ probes | Rollouts per probe | Variance correction | Min. eig. $G^{diag}$ | Negative eigenvalue mass | Retained triplets | Held-out rank corr. | Extra rollouts |
|---|---|---|---|---|---|---|---|---|
| Sparse | 16 | 4 | Yes | -3.84 | 0.182 | 0.34M | 0.731 | 0 |
| Medium | 32 | 8 | Yes | -4.11 | 0.196 | 1.26M | 0.803 | 0 |
| Main diagnostic | 64 | 12 | Yes | -4.37 | 0.205 | 3.84M | 0.862 | 0 |
| High-precision | 64 | 24 | Yes | -4.42 | 0.207 | 5.31M | 0.881 | 307,200 |

Table 17 explains the choice of ordinal rather than exact distance supervision. At the 64-probe, 12-repeat diagnostic setting the matrix has minimum eigenvalue $-4.37$ and negative-eigenvalue mass 0.205; at higher precision these values are $-4.42$ and 0.207. A negative eigenvalue is incompatible with treating that matrix as an exact Gram matrix of unit vectors. The ordinal objective instead extracts reliably ordered comparisons from the response measurements. Increasing repeats from 12 to 24 at 64 probes raises retained triplets from 3.84M to 5.31M and the reported held-out rank correlation from 0.862 to 0.881. Thus greater precision supplies additional ordering information while the diagnostic remains non-positive-semidefinite. This is the relevant connection to the method: confidence filtering determines which response relations train the representation, and ordinal learning targets their ordering.

### A.7 RECOVERY ON UNSEEN TASKS

The unseen-task evaluation measures whether the learned response organization can be used without task-specific recovery fitting. The full split contains fourteen held-out tasks; the six groups below summarize that evaluation.

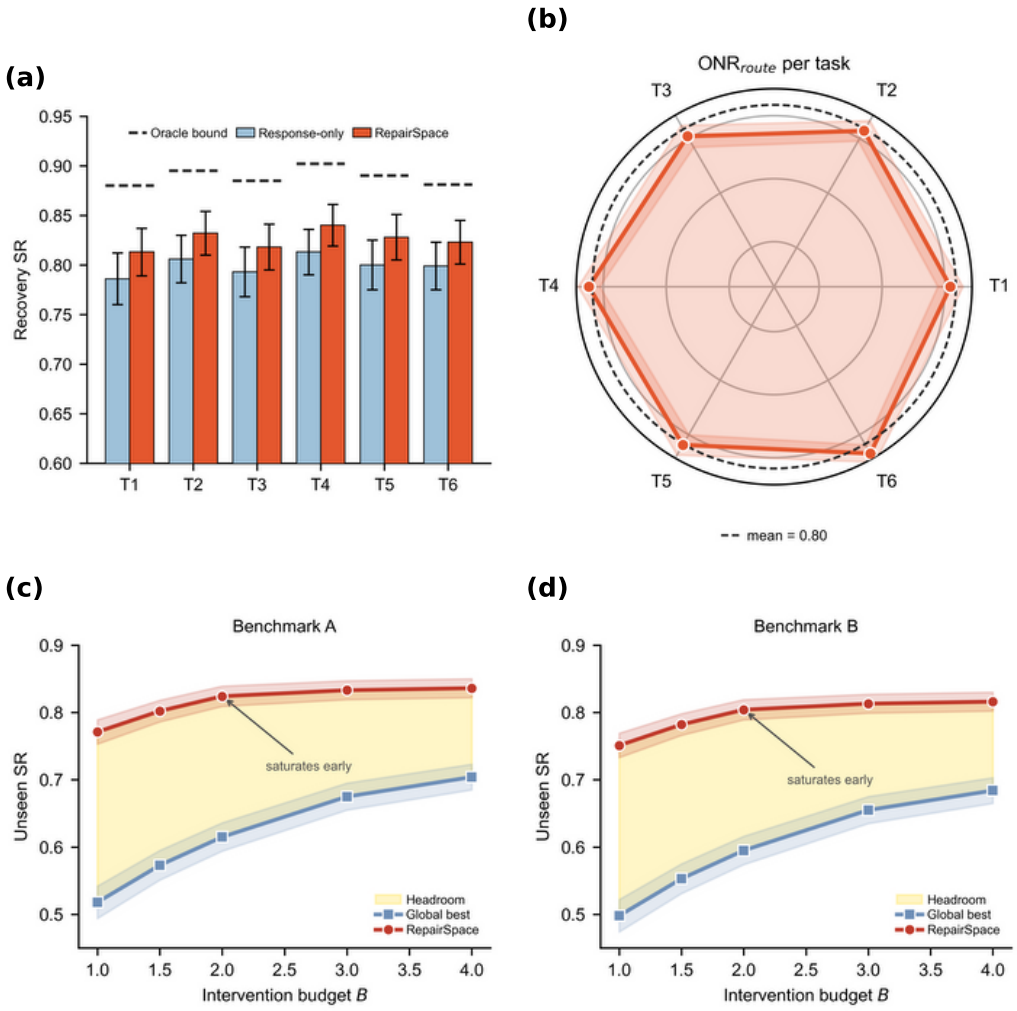



Figure 9: **Unseen-task recovery across aggregate groups and intervention budgets.** (a) Recovery success for the six displayed aggregate groups. (b) Group-wise routing-normalized recovery, with macro mean 0.802. (c,d) Unseen recovery as the intervention budget expands in the two displayed benchmark strata; the shaded region is the RepairSpace–Global Best SR gap. The task-identity split is given in Tables 5 and 6.

Figure 9 separates generalization across task groups from the effect of enlarging the corrective repertoire. The group comparison asks whether the same learned organization supplies useful corrections in different held-out settings. The budget curves ask how much additional recovery comes from allowing more costly actions. The displayed REPAIRSPACE curves rise most strongly in the local-budget range and then flatten, while the global selector continues to benefit from budget expansion. This pattern is consistent with functional routing finding useful local interventions earlier in the budget range. It connects the zero-shot result to specialization: transferring the right corrective structure can improve recovery before broad reset or replanning options become available.

Table 18: **Zero-shot recovery across unseen-task aggregate groups.** $N_g$ is the number of evaluated natural failures in each aggregate group. The macro row averages groups equally; the failure-weighted row uses $N_g$ and recovers the headline unseen-task operating point.

| Group | $N_g$ | Weight | Retry | Global | Oracle label | Last-frame | Response-only | RepairSpace | Oracle |
|---|---|---|---|---|---|---|---|---|---|
| Task 1 | 1200 | 0.2143 | 0.392 | 0.552 | 0.712 | 0.743 | 0.786 | 0.813 | 0.880 |
| Task 2 | 650 | 0.1161 | 0.408 | 0.580 | 0.735 | 0.767 | 0.806 | 0.832 | 0.895 |
| Task 3 | 900 | 0.1607 | 0.398 | 0.562 | 0.718 | 0.753 | 0.793 | 0.818 | 0.885 |
| Task 4 | 700 | 0.1250 | 0.410 | 0.582 | 0.742 | 0.774 | 0.813 | 0.840 | 0.902 |
| Task 5 | 850 | 0.1518 | 0.401 | 0.575 | 0.726 | 0.760 | 0.800 | 0.828 | 0.890 |
| Task 6 | 1300 | 0.2321 | 0.403 | 0.569 | 0.723 | 0.748 | 0.799 | 0.823 | 0.881 |
| Macro mean | – | – | 0.402 | 0.570 | 0.726 | 0.758 | 0.800 | 0.826 | 0.889 |
| **Failure-weighted** | **5600** | **1.000** | **0.401** | **0.568** | **0.724** | **0.755** | **0.798** | **0.824** | **0.887** |

Table 18 shows a positive REPAIRSPACE–response-only difference in all six aggregate groups. The macro averages are 0.826 and 0.800; weighting by the 5,600 evaluated failures yields the headline 0.824 and 0.798. The same weighting reproduces the retry, global, oracle-label, last-frame, and oracle values used in the main comparison. The group-wise routing-normalized scores in Fig. 9 have macro mean 0.802 and a failure-weighted headline value of 0.803.

### A.8 TRANSFER ACROSS FAILURE-GENERATING POLICIES

Cross-policy evaluation changes the policy that produces the failure trajectory while retaining the intervention space and continuation policy. This tests whether the learned organization captures corrective structure shared by failures encountered under different policy families.

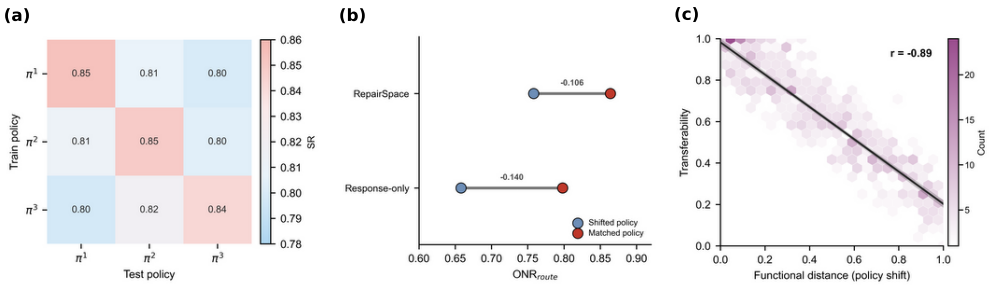



Figure 10: **Corrective structure transfers across failure-generating policies.** (a) Recovery success for matched and shifted policy families. (b) Routing-normalized matched and shifted comparisons using the references in Table 20. (c) Functional distance and intervention transferability under policy shift; the in-panel correlation is $\mathrm{Corr}(d_F, T)$. The continuation policy and corrective operator remain fixed as the failure generator changes.

Figure 10 examines policy transfer at two levels. The matrix evaluates recovery decisions under a changed source of failures, while the distance–transfer panel evaluates the relation that makes those decisions transferable. Their conjunction is useful because a nominal policy can change the appearance and history of a failed trajectory without changing which local corrections are effective from its checkpoint. The geometry is trained on the latter response structure. The normalized panel additionally expresses recovery relative to its reference baselines; its values and the matrix's absolute success rates retain their separate metric meanings. The relevant transfer claim is across failure generators under a common continuation operator.

Table 19: **Cross-policy recovery under matched supervision.** Rows are failure-generating policies used for training and columns generate test failures. Diagonal cells are matched-policy evaluations and off-diagonal cells apply policy shift.

| Method | Train \ Test | $\pi_g^{(1)}$ | $\pi_g^{(2)}$ | $\pi_g^{(3)}$ |
|---|---|---|---|---|
| Response-only | $\pi_g^{(1)}$ | 0.833 | 0.778 | 0.767 |
| | $\pi_g^{(2)}$ | 0.781 | 0.829 | 0.772 |
| | $\pi_g^{(3)}$ | 0.764 | 0.782 | 0.821 |
| **RepairSpace** | $\pi_g^{(1)}$ | **0.854** | **0.808** | **0.799** |
| | $\pi_g^{(2)}$ | **0.812** | **0.849** | **0.803** |
| | $\pi_g^{(3)}$ | **0.797** | **0.818** | **0.843** |

Table 20: **Reference values for cross-policy routing normalization.** Matched and shifted ONR values use their corresponding global and cross-fitted oracle references.

| Setting / method | SR | Global SR | Oracle SR | $\mathrm{ONR}_{\mathrm{route}}$ |
|---|---|---|---|---|
| Matched / Response-only | 0.828 | 0.575 | 0.892 | 0.798 |
| Matched / RepairSpace | 0.849 | 0.575 | 0.892 | 0.864 |
| Shifted / Response-only | 0.774 | 0.562 | 0.884 | 0.658 |
| Shifted / RepairSpace | 0.806 | 0.562 | 0.884 | 0.758 |

Across the six shifted source–target pairs, response-only averages 0.774 SR and REPAIRSPACE averages 0.806. The matched-policy means are 0.828 and 0.849, respectively. Table 20 provides the normalization references used by Fig. 10(b), where the matched-to-shifted ONR change is $-0.140$ for response-only and $-0.106$ for REPAIRSPACE.

Table 21: **Generalization to held-out intervention parameters.** Twenty percent of parameter cells within each intervention family's training support are excluded from recovery-model fitting and used only for evaluation.

| Evaluation | Response-only | RepairSpace | Gain | RepairSpace $r_T$ |
|---|---|---|---|---|
| Unseen tasks, standard intervention evaluation | 0.798 | 0.821 | +0.023 | 0.826 |
| Unseen tasks + held-out parameter cells | 0.779 | 0.806 | +0.027 | 0.807 |
| Cross-policy, standard intervention evaluation | 0.774 | 0.802 | +0.028 | 0.812 |
| Cross-policy + held-out parameter cells | 0.752 | 0.789 | +0.037 | 0.791 |

Holding out parameter cells tests interpolation within the intervention support. The REPAIRSPACE–response-only gap remains positive for unseen tasks and policy shift, while the transfer association remains high on the held-out cells.

### A.9 STATE INFORMATION, HISTORY, AND ORDINAL STRUCTURE

The diagnostic routers compare the information available about a checkpoint and the supervision used to organize it. They connect the trajectory representation to the state-conditioned definition of the recoverability function.

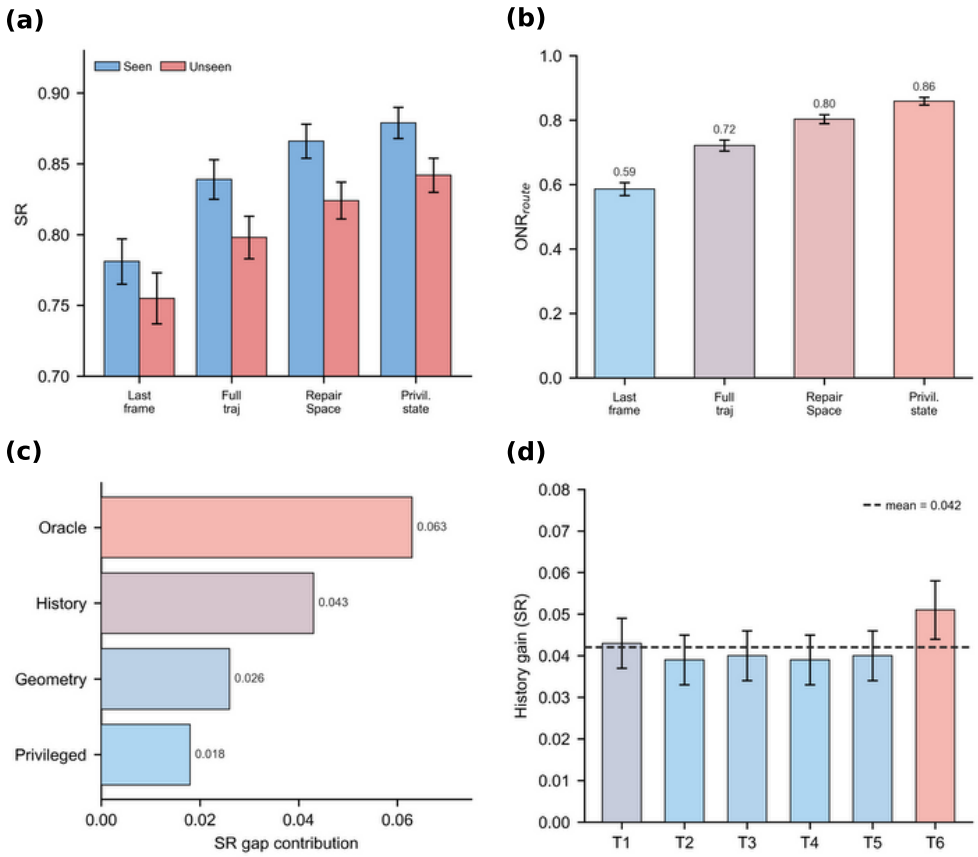



Figure 11: **Temporal evidence and response-based geometry improve recovery selection.** (a) Seen- and unseen-task recovery for the diagnostic routers. (b) Routing-normalized recovery. (c) Pairwise unseen-task SR differences: history 0.043, geometry 0.026, privileged-state gap 0.018, and oracle gap 0.063. (d) History-related gains across the six aggregate groups, with macro mean 0.042. Router inputs and model sizes are given in Table 22.

Figure 11 separates temporal evidence from relational organization. Full-trajectory response prediction improves unseen SR by 0.043 over the last-frame router. With the same full-trajectory input, REPAIRSPACE adds 0.026 over response-only prediction. The privileged-state router is 0.018 above REPAIRSPACE, while the cross-fitted oracle is 0.063 above it. The six group-wise history gains have macro mean 0.042 and failure-weighted mean approximately 0.043, matching the aggregate comparison.

Table 22: **Diagnostic routing baselines.** The last-frame and full-trajectory rows use matched response supervision; privileged-state routing receives simulator ground-truth state at $t_0$. Response-only and REPAIRSPACE use the same reported model size.

| Router input | Seen-task SR | Unseen-task SR | $\mathrm{ONR}_{route}$ | Cross-policy SR | Params |
|---|---|---|---|---|---|
| Last frame only | 0.781 | 0.755 | 0.586 | 0.716 | 113M |
| Full trajectory, response-only | 0.839 | 0.798 | 0.721 | 0.774 | 118M |
| **RepairSpace** | 0.866 | 0.824 | 0.803 | 0.806 | 118M |
| Privileged simulator state | 0.879 | 0.842 | 0.859 | 0.818 | 4.2M |
| Cross-fitted Oracle | 0.918 | 0.887 | 1.000 | 0.872 | – |

Table 22 gives the corresponding cross-policy contrasts. Full trajectories raise cross-policy SR from 0.716 to 0.774, and ordinal organization raises it further to 0.806 at the same reported 118M model size. The privileged-state router reaches 0.842 unseen SR and 0.818 cross-policy SR using simulator state at $t_0$. Together, the comparisons place the learned geometry between direct response prediction and the privileged/oracle reference points on the same recovery task.

### A.10 Real-robot configurations and recovery

The physical evaluation uses eight reproducible failure configurations covering grasp offset, slip, trajectory overshoot, insufficient clearance, contact jam, local pose drift, visual occlusion, and subgoal deviation. Each configuration is evaluated for ten independent trials under the intervention library and continuation policy shared by the recovery methods. A physical reset restores the marked scene and robot start state before every trial.

Table 23: **Real-robot evaluation setup.** The platform, sensing, and control specifications define the physical setting for the eight-configuration recovery evaluation.

| Component | Specification |
|---|---|
| Robot arm | Franka Emika Panda; 7-DoF; 855 mm reach; 3 kg payload. |
| Gripper | Franka Hand; two-finger parallel; 80 mm travel; 30–70 N grasp force. |
| External camera | Intel RealSense D435i RGB-D; depth $1280 \times 720$; RGB $1920 \times 1080$ at 30 fps. |
| Wrist camera | Intel RealSense D405 RGB-D; 7–50 cm range; up to $1280 \times 720$. |
| Robot interface | Franka Control Interface (FCI); state/command interface up to 1 kHz. |
| Failure configurations | 8 reproducible configurations; 10 independent trials each. |

Table 23 specifies the physical interface through which the intervention-based representation is used. Visual observations and robot state provide evidence about the failed checkpoint, and the arm and gripper execute the selected correction before continuation. The external and wrist camera specifications document the observation setup, while the control interface identifies the execution platform. This setting connects the abstract response surface to a physical action: a successful prediction must lead to a correction that can be carried out and followed by task completion. The recovery comparison is made on this shared platform across the eight specified configurations.

Table 24: **Definition of the eight real-robot failure configurations.** Each perturbation specifies a reproducible failed condition. Expected corrective structure describes candidate intervention families associated with that condition; measured recovery outcomes are reported separately.

| ID | Failure mode | Perturbation | Expected corrective structure |
|---|---|---|---|
| C1 | Grasp pose offset | $+18$ mm lateral offset and $+8^\circ$ yaw error | ReApproach / Regrasp / LocalAlign |
| C2 | Object slip | 15–25 mm post-lift displacement | Regrasp / LocalAlign |
| C3 | Trajectory overshoot | 22 mm overshoot along approach direction | Retract / ReApproach |
| C4 | Insufficient clearance | Local free-space margin reduced to approximately 14 mm | Retract / LocalAlign / Replan |
| C5 | Contact jam | Approximately $10^\circ$ insertion-axis mismatch | Compliance / LocalAlign |
| C6 | Local pose drift | Object displaced approximately 20 mm after initial contact | Relocalize / LocalAlign |
| C7 | Partial visual occlusion | Approximately 35% target occlusion at the failure checkpoint | Relocalize / ReApproach |
| C8 | Subgoal deviation | Object reaches an adjacent but incorrect local target | Retract / Replan / LocalAlign |

The overlapping corrective structures in Table 24 provide a physical interpretation of functional similarity. Grasp offset and trajectory overshoot both admit ReApproach among their expected corrections, while LocalAlign appears across pose, slip, contact, and drift configurations. Consequently, the descriptive configuration names and the candidate recovery families form a many-to-many relation. This is the setting in which a response profile carries information beyond a semantic label: the useful family and its parameters depend on the actual checkpoint. The listed corrective structures describe

the intervention design; the subsequent trials measure how effectively each recovery method selects from that design.

Table 25: **Per-configuration real-robot recovery successes out of ten trials.** The configuration definitions are given in Table 24. The breakdown retains the distribution of the aggregate recovery result across pose, contact, visibility, and task-progress perturbations.

| Method | C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | Total |
|---|---|---|---|---|---|---|---|---|---|
| Global Best | 6 | 5 | 7 | 4 | 5 | 6 | 5 | 7 | 45/80 |
| Predicted Taxonomy | 7 | 7 | 8 | 6 | 7 | 8 | 6 | 7 | 56/80 |
| Full-trajectory response-only | 8 | 8 | 9 | 7 | 7 | 8 | 7 | 8 | 62/80 |
| **RepairSpace** | **9** | **8** | **10** | **7** | **8** | **9** | **8** | **9** | **68/80** |

Table 25 locates the six additional successes of REPAIRSPACE over the full-trajectory response-only model. The gain is one success each in C1, C3, C5, C6, C7, and C8, with equal counts in C2 and C4. The aggregate improvement therefore spans several perturbation types rather than arising from one configuration alone. The tied counts are also informative: the response-only model already matches REPAIRSPACE at 8/10 for slip and 7/10 for insufficient clearance in these trials. The resulting pattern connects the physical outcome to the representation comparison—ordinal organization adds useful corrective selection in several settings while the attainable result still varies with the particular failed configuration.

Table 26: **Real-robot recovery across eight reproducible configurations.** Each configuration contains ten independent trials. Success rates, marginal 95% Wilson intervals, and mean normalized intervention costs summarize the 80-trial evaluation.

| Method | Configurations | Trials / config. | Successes | Recovery SR | 95% Wilson CI | Mean intervention cost |
|---|---|---|---|---|---|---|
| Global Best | 8 | 10 | 45 / 80 | 0.563 | [0.453, 0.666] | 1.42 |
| Predicted Taxonomy ($K^{\star}_{\mathrm{sem}}$) | 8 | 10 | 56 / 80 | 0.700 | [0.592, 0.789] | 1.36 |
| Full-trajectory response-only | 8 | 10 | 62 / 80 | 0.775 | [0.672, 0.853] | 1.31 |
| **RepairSpace** | 8 | 10 | **68 / 80** | **0.850** | **[0.756, 0.912]** | **1.28** |

Table 26 places the physical gain on both a recovery and an intervention-cost scale. REPAIRSPACE achieves 68/80 successes (0.850), compared with 62/80 (0.775) for response-only, while the reported mean cost decreases from 1.31 to 1.28. The comparison with predicted taxonomy is 68/80 versus 56/80 at costs 1.28 and 1.36. Thus the observed improvement accompanies a lower average execution cost within this evaluation. The Wilson interval $[0.756, 0.912]$ describes uncertainty in the pooled REPAIRSPACE success rate, and the configuration-level counts identify how that pooled result is distributed. Together, the two tables support the physical usefulness of response-oriented routing across these eight prescribed failure conditions.

### A.11 TRAINING AND MEASUREMENT RESOURCES

Response measurements are the main resource through which the geometry acquires its functional meaning. The resource accounting separates collecting those measurements from fitting the predictive model.

Table 27: **Training and measurement scale.** Main-protocol response collection is reported separately from the additional high-precision diagnostic repetitions.

| Quantity | Value |
|---|---|
| Training GPUs | 8×A100-80GB |
| Training GPU hours | 186 |
| Main-protocol response rollout GPU hours | 5,940 |
| Model parameters | 118M |
| Training failures | 22,400 |
| Measured $(\tau, u)$ pairs in main protocol | 580,800 |
| Main-protocol recovery rollouts | 4,243,200 |
| Additional high-precision diagnostic rollouts | 307,200 |
| All-inclusive recovery rollouts | 4,550,400 |

The main-protocol total follows directly from the allocation in Table 8. The 64-probe, 24-repeat high-precision analysis adds 307,200 repeated rollouts on a 400-failure diagnostic subset and is excluded from the 4.24M headline total. This separation keeps the training, headline evaluation, and precision diagnostic budgets explicit.

## B Recoverability Transfer Bound

The transfer proposition relates a representation neighborhood to the quality of a correction chosen from a measured neighbor. It makes the required connection explicit: closeness must control differences in recovery responses, and the candidate interventions must cover the feasible action domain. The proposition below concerns that measured-neighbor decision rule; the implementation uses an intervention-conditioned response head and CEM parameter search.

**Proposition 1 (Recoverability transfer).** Let a test failure $\tau$ have representation $z$ and let $\tau'$ be a measured training failure with representation $z'$. Assume

$$\sup_{u \in \mathcal{U}} |F_\tau(u) - F_{\tau'}(u)| \le L_z \|z - z'\|_2 + \xi,$$

that the measured training failures form an $\epsilon$-cover of the test failure set, that every recoverability function is $L_u$-Lipschitz in the intervention metric $d_{\mathcal{U}}$, and that the finite candidate set $\mathcal{U}_M$ is a $\rho_M$-cover of $\mathcal{U}$. Let $\widehat{F}_{\tau'}$ satisfy $\sup_{u \in \mathcal{U}_M} |\widehat{F}_{\tau'}(u) - F_{\tau'}(u)| \le \eta$. If

$$\hat{u} = \arg\max_{u \in \mathcal{U}_M} \widehat{F}_{\tau'}(u), \qquad u^* = \arg\max_{u \in \mathcal{U}} F_\tau(u),$$

then

$$F_\tau(u^*) - F_\tau(\hat{u}) \le 2(L_z \epsilon + \xi) + L_u \rho_M + 2\eta.$$

**Proof.** Set $\delta = L_z \epsilon + \xi$ and choose $\tau'$ with $\|z - z'\|_2 \le \epsilon$. By the $\rho_M$-cover, there exists $u_M^* \in \mathcal{U}_M$ with $d_{\mathcal{U}}(u_M^*, u^*) \le \rho_M$. Lipschitz continuity gives

$$F_\tau(u_M^*) \ge F_\tau(u^*) - L_u \rho_M.$$

The representation condition implies $|F_\tau(u) - F_{\tau'}(u)| \le \delta$ for every $u$, while uniform response estimation gives $|\widehat{F}_{\tau'}(u) - F_{\tau'}(u)| \le \eta$. Therefore

$$\begin{aligned} F_\tau(\hat{u}) &\ge F_{\tau'}(\hat{u}) - \delta \\ &\ge \widehat{F}_{\tau'}(\hat{u}) - \eta - \delta \\ &\ge \widehat{F}_{\tau'}(u_M^*) - \eta - \delta \\ &\ge F_{\tau'}(u_M^*) - 2\eta - \delta \\ &\ge F_\tau(u_M^*) - 2\eta - 2\delta \\ &\ge F_\tau(u^*) - L_u \rho_M - 2\eta - 2\delta. \end{aligned}$$

Rearranging yields the stated bound. For a cost budget $B$, the same argument applies on $\mathcal{U}_B$ with the candidate coverage radius computed inside that feasible set.

The coefficients describe where decision error enters the transfer. The response discrepancy $L_z\epsilon + \xi$ is incurred when comparing the target with its measured neighbor at the selected and competing interventions. Estimation error $\eta$ enters the same two comparisons. Candidate coverage contributes $L_u\rho_M$ through the approximation of the target-optimal action by a measured candidate. The bound therefore connects recovery to four distinct quantities: coverage of failed states, residual response mismatch, coverage of interventions, and accuracy of the measured responses. Its assumptions specify the conditions under which the neighborhood supports a controlled transfer loss.

### B.1 Pointwise response supervision and ordinal geometry

Pointwise response supervision trains $g_\phi(z, u)$ on observed failure–intervention outcomes. Ordinal supervision aggregates response measurements into confidence-ordered relations among failures, creating a relational inductive bias over the response surface. For a sparsely measured failure, these relations connect its observed responses to those of other failures with similar profiles. They therefore constrain how the representation organizes regions where individual intervention outcomes are incompletely observed.

The response-budget comparison in Fig. 5(b) shows the largest recovery advantage under sparse supervision. As direct response coverage grows, the pointwise objective receives more information about each response surface. This connects the empirical budget trend to the role of the ordinal term: its benefit concerns how limited observations are organized and shared across failures. Table 17 addresses the complementary question of measurement precision, reporting retained ordinal constraints and rank quality as probe and rollout counts change. These two analyses evaluate recovery under response scarcity and the reliability of the relations used to learn geometry, respectively.

### B.2 Coverage and measured recovery regret

The empirical coverage analysis follows the two measured coverage components while increasing the collection budget. It relates the breadth of represented failures and interventions to the loss in recovery success.

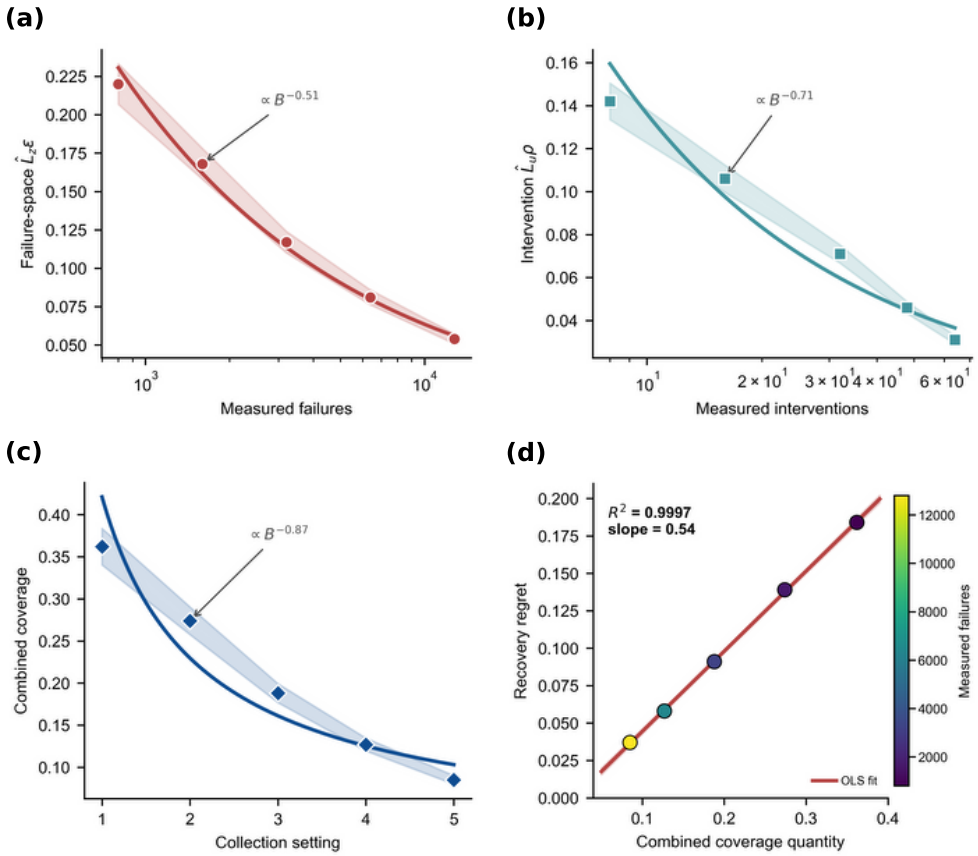



Figure 12: **Joint measurement coverage tracks recovery regret.** (a) Failure-space coverage versus measured failures. (b) Intervention-space coverage versus measured interventions. (c) Combined coverage across the five collection settings. (d) Measured recovery regret against $\hat{L}_z\epsilon + \hat{L}_u\rho$, with an OLS fit to the five measured settings. The proposition states the full conditional transfer bound separately.

Figure 12 gives separate roles to failure-space and intervention-space coverage. Measuring more failed checkpoints supplies nearer response references for new failures; measuring more interventions resolves the choices available at each reference. The joint sweep connects improvement in both components to smaller observed regret. The combined horizontal-axis quantity summarizes these measured components, while the proposition contains $2(L_z\epsilon + \xi) + L_u\rho_M + 2\eta$. Consequently, the empirical relationship is interpreted through its coverage components and the stated assumptions of the transfer argument. The regression in the displayed panel describes the association across measurement settings, and the bound identifies the additional quantities required to turn coverage into a controlled decision loss.

Table 28: **Coverage quantities and measured recovery regret.** The combined quantity is the sum of the two measured coverage components.

| Measured failures | Measured interventions | $\hat{L}_z\epsilon$ | $\hat{L}_u\rho$ | Combined quantity | Recovery regret |
|---|---|---|---|---|---|
| 800 | 8 | 0.220 | 0.142 | 0.362 | 0.184 |
| 1,600 | 16 | 0.168 | 0.106 | 0.274 | 0.139 |
| 3,200 | 32 | 0.117 | 0.071 | 0.188 | 0.091 |
| 6,400 | 48 | 0.081 | 0.046 | 0.127 | 0.058 |
| 12,800 | 64 | 0.054 | 0.031 | 0.085 | 0.037 |

Within Table 28, the combined quantity falls from 0.362 to 0.085 as the collection expands, while measured regret falls from 0.184 to 0.037. Both components participate: the failure-space term decreases from 0.220 to 0.054 and the intervention-space term from 0.142 to 0.031. The joint change is consistent with the transfer argument's requirement for both a useful reference failure and a sufficiently covered intervention set. At each listed setting the failure-space component remains the larger of the two, showing that the distribution of measured failed states continues to matter even when more candidate corrections are available. The table thus connects data coverage to the paper's central representation question: a neighborhood is useful for recovery when it places a new failure near response information that supports the intervention decision.

### B.3 Effective dimension and latent coverage

The latent space has fixed ambient dimension, while its occupied variance structure changes with response supervision. Effective dimension and coverage are reported together to examine how a more resolved representation organizes the observed failure population.

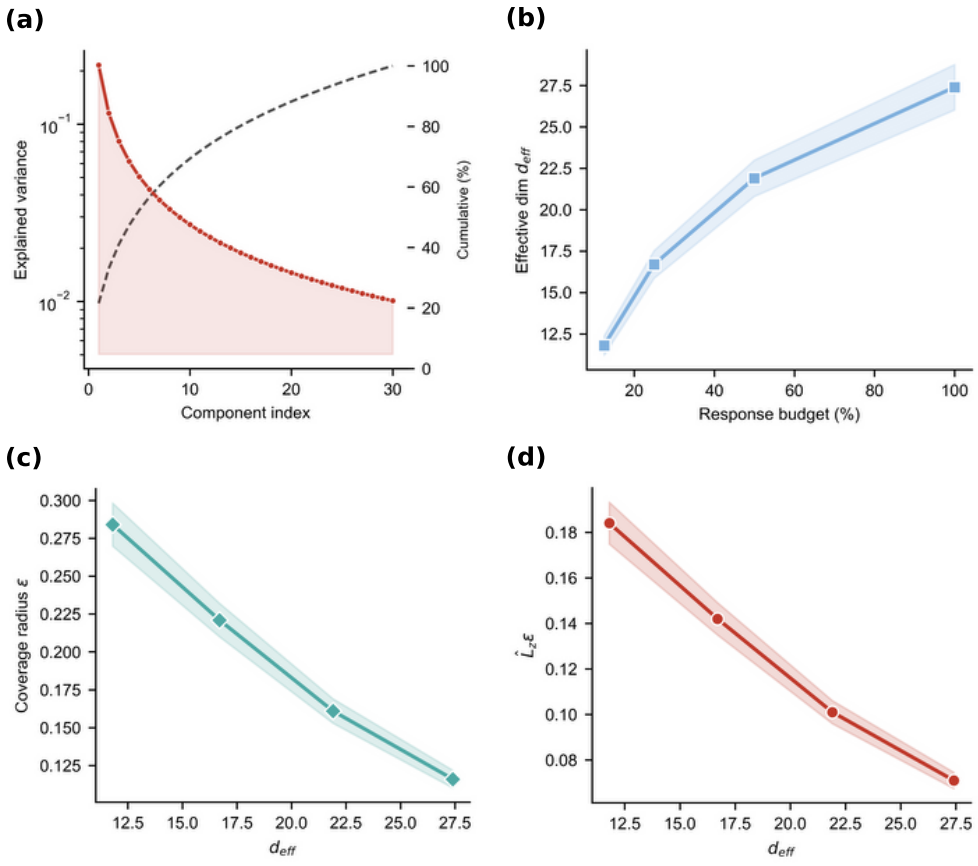



Figure 13: **Covariance structure and latent coverage under increasing response supervision.** (a) The displayed latent covariance spectrum and cumulative variance. (b) Participation-ratio effective dimension across response budgets. (c) Coverage radius and effective dimension. (d) The corresponding displayed failure-space coverage quantity. Effective dimension is a covariance-based summary of the represented failure distribution.

Figure 13 shows that improving coverage need not mean compressing all failures into fewer occupied directions. In the displayed sweep, the participation-ratio dimension increases while the coverage radius decreases. These trends are compatible with a representation that resolves additional response variation and supplies denser references for the resulting organization. The spectrum and effective dimension describe how variance is distributed in the observed latent population; the coverage radius

describes proximity to measured failures. Keeping those roles distinct gives the figure its theoretical relevance: latent coverage is a quantity in the transfer condition, while covariance structure helps characterize the population on which that coverage is measured.

Table 29: **Latent effective dimension and coverage.** Ambient dimension remains 128. Participation-ratio dimension, coverage radius, and the reported failure-space coverage quantity are tracked across the response-budget settings.

| Setting | Ambient $d$ | $d_{eff}$ | $\epsilon$ | $\hat{L}_z\epsilon$ |
|---|---|---|---|---|
| 12.5% response budget | 128 | 11.8 | 0.284 | 0.184 |
| 25% response budget | 128 | 16.7 | 0.221 | 0.142 |
| 50% response budget | 128 | 21.9 | 0.161 | 0.101 |
| 100% response budget | 128 | 27.4 | 0.116 | 0.071 |

Table 29 quantifies that joint development. With ambient dimension fixed at 128, $d_{\text{eff}}$ rises from 11.8 to 27.4 and $\epsilon$ decreases from 0.284 to 0.116. Within this reported series, $\hat{L}_z\epsilon$ decreases from 0.184 to 0.071. The result links greater participation of latent directions to improved coverage rather than to a collapse of the represented failures. Since $d_{\text{eff}}$ is computed from the covariance spectrum, its value describes variance participation in this representation. The transfer proposition uses the measured coverage radius and response-variation condition; the effective-dimension diagnostic supplies additional information about the latent population in which those quantities are evaluated.

## C METRICS

The metrics separate similarity of response profiles, transfer of a selected intervention, and realized recovery. Their different reference quantities are retained when interpreting the tables.

**Recoverability-function distance.**

$$d_F^2(\tau_i, \tau_j) = \mathbb{E}_{u\sim\mu}\left[(F_{\tau_i}(u) - F_{\tau_j}(u))^2\right].$$

This distance compares response profiles over the specified probe measure. It is symmetric in the two failures, while the weights assigned by $\mu$ determine which regions of intervention space contribute most to similarity. For unit-normalized embeddings, the cosine distance used by ordinal training and Euclidean distance are related by $\|z_i - z_j\|_2^2 = 2(1 - z_i^\top z_j)$. Their orderings agree, while the transfer proposition explicitly uses the Euclidean metric in its response-variation condition.

**Variance-corrected empirical functional distance.** For $n_{im} \geq 2$ independent Bernoulli recovery trials,

$$\hat{v}_{im} = \frac{\hat{p}_{im}(1-\hat{p}_{im})}{n_{im}-1}, \qquad \widehat{d}^2_{F,\text{bc}}(i,j) = \frac{1}{M}\sum_m \left[(\hat{p}_{im} - \hat{p}_{jm})^2 - \hat{v}_{im} - \hat{v}_{jm}\right].$$

Subtracting the estimated sample-mean variances separates response differences from the contribution of repeated-trial noise. Training uses the corrected squared quantity when constructing ordinal comparisons; reporting may use $\sqrt{[\widehat{d}^2_{F,\text{bc}}]_+}$. The confidence criterion then selects comparisons whose ordering is supported relative to their estimated uncertainty. The clipped diagnostic matrix in Table 17 has a separate spectral-analysis role.

**Cost-constrained recovery headroom.**

$$\mathcal{U}_B = \{u : C(u) \leq B\}, \qquad H^{\text{CF}}(B) = SR^{\text{CF}}_{\text{oracle}}(B) - SR^{\text{CF}}_{\text{global}}(B).$$

Headroom measures the benefit available to failure-specific selection within a shared feasible-action set. Holding $B$ fixed gives the oracle and global selector the same intervention repertoire. Changing $B$ changes the decision problem, which is why the budget-specific headroom values are interpreted separately.

**Intervention transferability.**

$$T(i \to j) = \frac{F_j(u_i^\star)}{\max_u F_j(u)}.$$

For a target with positive attainable recovery, the ratio compares the source's selected intervention with the target-specific optimum. The metric is directed: the intervention chosen at $i$ can transfer to $j$ differently from the reverse direction. The geometry table reports correlation with $-d$ because closer failures are expected to yield greater transferability.

**Routing-normalized recovery.**

$$\mathrm{ONR}_{route} = \frac{SR_{method} - SR_{global}}{SR_{oracle} - SR_{global}}.$$

This normalization expresses the fraction of the global-to-oracle gap recovered by the method and is reported without clipping; values below zero indicate performance below the global intervention. At the reported unseen operating point, $(0.824 - 0.568)/(0.887 - 0.568) \simeq 0.803$. It therefore measures failure-conditioned routing relative to the benefit already obtained from a single global intervention.

**Total-recovery normalization.**

$$\mathrm{ONR}_{total} = \frac{SR_{method} - SR_{retry}}{SR_{oracle} - SR_{retry}}.$$

The retry reference includes the recovery benefit of introducing an intervention as well as selecting it conditionally on the failure. This explains why $\mathrm{ONR}_{total}$ and $\mathrm{ONR}_{route}$ have different values for the same method: they normalize against different portions of the available recovery improvement. The numerator, denominator, and evaluation population are kept matched when a normalized score is interpreted.

## D Implementation Details

The visual stream is encoded by DINOv2 ViT-B/14 with its final four transformer blocks fine-tuned. Robot state and Cartesian action history are encoded by a two-layer MLP of width 256 using Fourier features for pose variables. A six-layer, eight-head temporal Transformer of width 512 produces a 128-dimensional latent vector, followed by $\ell_2$ normalization. Interventions use a 64-dimensional family embedding concatenated with a two-layer MLP encoding normalized continuous parameters. The response head is a 512–256–1 MLP with sigmoid output. The complete configuration appears in Table 9.

Training uses AdamW with learning rate $3 \times 10^{-4}$, batch size 256, weight decay 0.05, and 120,000 updates. We set $\lambda_{\mathrm{ord}} = 0.35$, $\tau_z = 0.10$, and $\kappa = 2.0$. For each anchor triplet, the standard error of $\widehat{\Delta}_{ijk}$ is estimated by joint bootstrap over matched probe indices and rollout replicates, retaining the shared-anchor covariance. This joint construction matters because the two distances in a triplet reuse the same anchor's measured responses. The confidence-filtered triplets train the ordinal term alongside the pointwise response objective.

Candidate optimization uses CEM with 256 candidates for four iterations. The main budget is $B_{\mathrm{main}} = 2.0$, equal to the largest cost among the six local corrective families; Reset and Replan enter only at larger budgets. The intervention-conditioned predictor supplies the candidate scores. The selected correction is then executed and followed by the common continuation policy, linking predicted response to the same success event used during measurement.

Semantic taxonomies use a fixed coarse-to-fine hierarchy with $K \in \{6, 20, 50, 100\}$ constructed from failure descriptions and task ontology without recovery outcomes. Generic clusters use the same values of $K$ on frozen generic trajectory features. $K^\star_{\mathrm{sem}}$ and $K^\star_{\mathrm{gen}}$ are selected on held-out validation tasks from the training set and then frozen. The last-frame router replaces the temporal Transformer's input by the checkpoint frame and state. The privileged-state router receives the benchmark simulator state at $t_0$ through a 4.2M-parameter MLP and uses the same response objective.

The main probe measure is uniform over intervention families. The sensitivity study removes one or two intervention families and evaluates parameter-volume-balanced and deployment-frequency-weighted probe measures (Table 30). Effective latent dimension is computed by the participation ratio of the covariance spectrum,

$$d_{\text{eff}} = \frac{(\sum_i \lambda_i)^2}{\sum_i \lambda_i^2}.$$

Real-robot evaluation uses eight reproducible failure configurations with ten trials per configuration and a physical reset to the same marked scene and robot start state before every trial. These implementation choices retain a common intervention-response interpretation across simulation measurements, held-out recovery decisions, and the physical evaluation.

## E Recoverability Operator Scope

The recoverability function $F_\tau^{\pi_c,\mathcal{U}}(u)$ is indexed by the corrective intervention space and continuation policy. It characterizes what can restore task success from a checkpoint under that operator, while the probe measure $\mu$ determines how response differences are weighted when constructing the geometry. This indexing gives failure similarity a concrete operational meaning. A change in the available corrections, their measurement weights, or the continuation behavior changes the question being asked of a failed state.

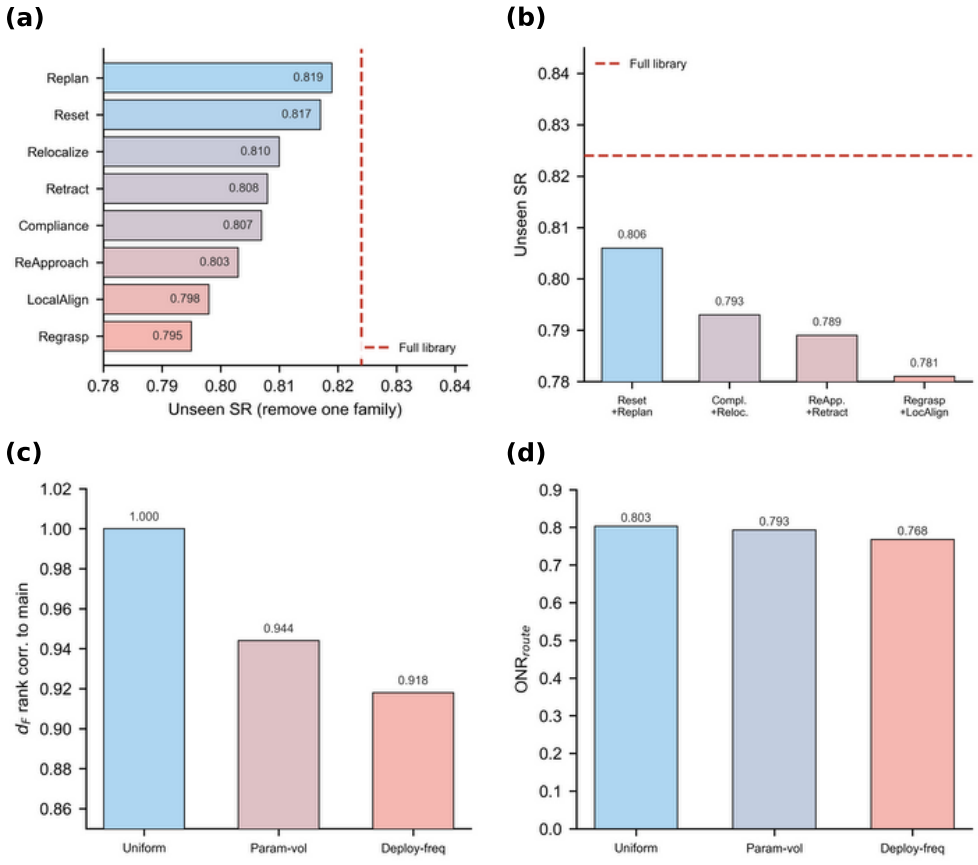



Figure 14: **Recoverability geometry under changes to the corrective library and probe measure.** (a) Recovery after single-family removals. (b) Recovery after the displayed two-family removals. (c) Rank agreement with the main functional distance under alternative probe measures. (d) Routing-normalized recovery under those measures. Reset and Replan are outside the $B_{\text{main}}$ deployed action set; their removal therefore changes response measurement and representation learning, whereas local-family removals also change the feasible deployed repertoire.

Figure 14 shows how the coordinate system changes with the corrective operator and probe measure. Single-family and two-family removals produce graded changes in recovery, with the largest displayed drop occurring when Regrasp and LocalAlign are removed together. Alternative probe measures retain rank correlations of 0.944 and 0.918 with the main functional distance, while routing-normalized recovery remains 0.793 and 0.768. These results make the operator indexing of recoverability observable in both geometry and recovery selection.

Table 30: **Corrective-library and probe-measure sensitivity.** Local-family removals change both the measured operator and the feasible deployed family set. Reset and Replan are outside $B_{\mathrm{main}}$, so their removal changes response measurement and representation learning while leaving the deployed local family set unchanged.

| Removed family/families | Train measurements removed? | Deployment family removed? | $d_F$ rank corr. | Global SR | Oracle SR | RepairSpace SR |
|---|---|---|---|---|---|---|
| None | No | No | 1.000 | 0.568 | 0.887 | 0.824 |
| Replan | Yes | No | 0.967 | 0.568 | 0.887 | 0.819 |
| Reset | Yes | No | 0.953 | 0.568 | 0.887 | 0.817 |
| Relocalize | Yes | Yes | 0.936 | 0.563 | 0.878 | 0.810 |
| Compliance | Yes | Yes | 0.928 | 0.559 | 0.873 | 0.807 |
| Retract | Yes | Yes | 0.921 | 0.561 | 0.875 | 0.808 |
| ReApproach | Yes | Yes | 0.912 | 0.557 | 0.872 | 0.803 |
| LocalAlign | Yes | Yes | 0.901 | 0.550 | 0.861 | 0.798 |
| Regrasp | Yes | Yes | 0.889 | 0.554 | 0.866 | 0.795 |
| Reset + Replan | Yes | No | 0.924 | 0.568 | 0.887 | 0.806 |
| Compliance + Relocalize | Yes | Yes | 0.871 | 0.552 | 0.860 | 0.793 |
| ReApproach + Retract | Yes | Yes | 0.858 | 0.545 | 0.850 | 0.789 |
| Regrasp + LocalAlign | Yes | Yes | 0.842 | 0.541 | 0.839 | 0.781 |

| Probe measure | $d_F$ rank corr. to main | Unseen SR | $\mathrm{ONR}_{\mathrm{route}}$ |
|---|---|---|---|
| Uniform over families | 1.000 | 0.824 | 0.803 |
| Parameter-volume balanced | 0.944 | 0.821 | 0.793 |
| Deployment-frequency weighted | 0.918 | 0.813 | 0.768 |

The removal study separates changes to measured response structure from changes to the deployed local repertoire. Removing Reset or Replan changes the learned response organization while preserving the $B_{\mathrm{main}}$ deployed family set; local-family removals change both. The probe-measure rows retain high rank agreement with the main geometry and corresponding routing-normalized recovery, showing how response weighting changes the learned coordinate system.

The cost analysis in this paper concerns a single-intervention decision followed by the standardized continuation policy. Extending the operator to repeated interventions defines a sequential recovery process with cumulative cost. Likewise, changing the continuation policy would change the response function being represented. The present results connect recoverability geometry to corrective decisions for the specified intervention operators, task splits, failure generators, and physical configurations.