BEYOND FEATURE FIDELITY: RECOVERY AND READOUT ADAPTATION IN MULTIMODAL LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Feature recovery targets the geometry of a clean visual representation; multimodal prediction depends on how a readout uses that representation. We examine this distinction in controlled systems connecting a shared visual encoder to frozen language-side models through learned readouts. Across four downstream models and nine benchmarks, recovery has mixed effects under a clean-trained readout. Readout adaptation improves the reported utility in most evaluated model–benchmark combinations, yet adapting directly to raw features is often competitive with or better than adapting after explicit recovery. In a GQA control, additional readout training on recovered clean features raises the score from 58.82 to 59.35, while adaptation using recovered clean and corrupted features reaches 59.67. This limits attribution of the entire adaptation gain to distribution matching. Separate fidelity analyses characterize task-specific and task-agnostic recovery settings. Together, the results motivate evaluating the representation–readout pair through distinct measurements of feature fidelity, fixed-readout utility, and adapted-readout utility.

1 INTRODUCTION

Many contemporary multimodal large language models (MLLMs) couple a visual encoder with a language model via a learned vision–language interface (Liu et al., 2023; Li et al., 2023). Under visual distribution shifts such as blur, noise, compression, and exposure variation, the distribution of encoder features can change (Chhipa et al., 2023; Tang et al., 2026). Recovering these features toward their clean counterparts is a natural robustness strategy, but its usefulness also depends on whether the downstream interface can consume the recovered representation.

Feature-recovery methods are typically optimized and evaluated with representation-space criteria such as reconstruction error, cosine similarity, and structural consistency (Yang et al., 2024; Parikh et al., 2026). These metrics measure *representation fidelity*, rather than whether the recovered information remains usable by a fixed multimodal interface. Thus, a recovered representation can move closer to its clean counterpart without improving downstream utility.

We study this interaction in a controlled DINOv3–LLaVA-1.5-7B system (Siméoni et al., 2025; Liu et al., 2024), comparing raw and recovered features under clean-trained and adapted readouts. On GQA (Hudson & Manning, 2019), inserting recovery changes Avg-All-C from 58.99 to 58.82, while adapting the readout to recovered inputs raises the score to 59.67. Direct adaptation to raw features reaches 59.74. These observations motivate distinguishing three effects: recovery under a fixed readout, adaptation to a fixed representation, and the additional value of recovery when both readouts are adapted.

Limited utility after recovery can arise from changes in task-relevant information, readout sensitivity to the transformed features, or both. We investigate these possibilities with a controlled 2×2 design that varies the representation supplied to the readout and whether the readout is adapted. Holding the recovered features fixed while updating the readout measures the utility effect of that update. A clean-only continuation control further tests how much of the observed gain can also arise without including corrupted inputs in adaptation. These comparisons establish readout dependence, while not uniquely separating information loss from interface mismatch.

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.

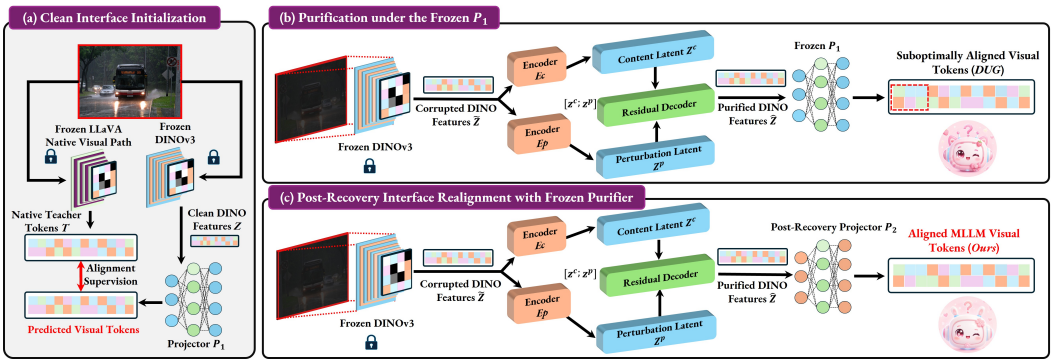


Figure 1: Controlled evaluation of representation recovery and readout adaptation in multimodal systems.

We evaluate these controlled effects across multiple MLLM families and downstream tasks. Recovery under the clean-trained readout has heterogeneous and often negative utility effects, whereas readout adaptation improves utility in most evaluated model–benchmark combinations. Moreover, adapting directly to raw corrupted features is frequently competitive with adapting after explicit recovery, indicating that recovery and readout adaptation contribute distinct effects. We further examine these observations when recovery is trained on an unlabeled, task-agnostic subset of Open Images (Kuznetsova et al., 2020) and transferred without downstream question–answer supervision, as well as under different adaptation-data scales.

Our contributions are threefold:

- We introduce a controlled representation–readout evaluation that distinguishes the utility effect of recovery under a fixed readout, the effect of readout adaptation, and the marginal benefit of recovery when both readouts are adapted.
- Across four downstream models and nine benchmarks, we show that recovery-only utility effects are heterogeneous, readout adaptation improves recovered-input utility in most evaluated settings, and explicit recovery provides no consistent advantage over direct adaptation to raw corrupted features.
- We complement these utility comparisons with separate representation-fidelity analyses, a clean-only continuation control, adaptation-data scaling, task-agnostic recovery, and statistical and corruption-level diagnostics.

2 Related Work

Robust multimodal understanding under visual corruption. Multimodal models remain sensitive to common visual corruptions. Q-Instruct improves low-level perception through instruction-based visual-quality supervision (Wu et al., 2024), while Robust-U1 reconstructs corrupted visual content and jointly reasons over corrupted and recovered observations (Tang et al., 2026). These approaches improve robustness through enhanced visual supervision or pixel-space recovery.

Visual representation recovery and denoising. Beyond pixel-space recovery, representation-space methods recover or stabilize visual features. Denoising Vision Transformers removes positional artifacts from ViT features using cross-view consistency (Yang et al., 2024), while latent denoising recovers clean teacher features with auxiliary reconstruction and structural objectives (Parikh et al., 2026). Such methods primarily evaluate recovery through feature reconstruction, alignment, and structural consistency, making representation fidelity a central criterion.

Vision–language interfaces and representation alignment. Modern multimodal architectures expose visual representations through different interfaces, including continuous visual prefixes, gated cross-attention, Q-Formers, and learned projectors (Tsimpoukelli et al., 2021; Alayrac et al., 2022; Li et al., 2023; Liu et al., 2023). JoyAI-Image further couples a multimodal language model

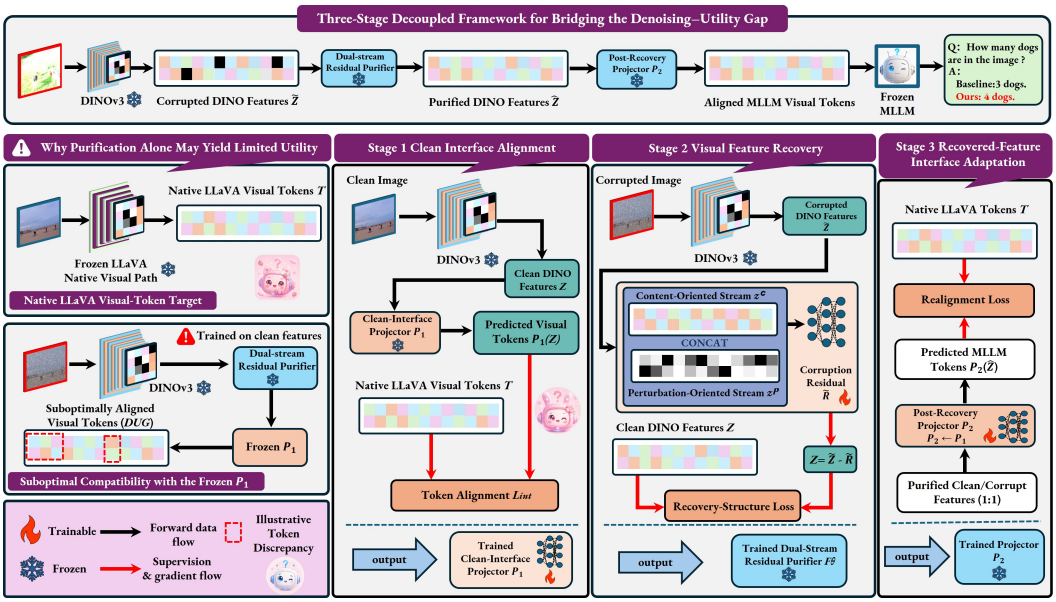


Figure 2: Overview of the controlled framework. The left panel motivates why improved feature fidelity may not improve downstream utility under a fixed interface. The three stages align a clean interface, recover corrupted DINOv3 features, and adapt the downstream interface to recovered features.

and a diffusion transformer through a shared interface for understanding and generation (Song et al., 2026). DeCo traces semantic flow through different projector designs and shows that the interface affects which visual semantics reach the language model (Yao et al., 2024). These studies establish that interface design shapes the usability of visual representations.

Representation fidelity versus downstream utility. Prior work establishes separate roles for feature recovery and interface design, but it remains unclear whether a higher-fidelity recovered representation is more useful to a fixed clean-trained readout. We study this question by separating representation recovery from interface adaptation.

3 Methodology

We study how representation recovery and readout adaptation affect multimodal prediction. Our controlled system consists of a shared visual encoder, a model-specific readout, and a frozen language-side model. We intervene on the representation supplied to the readout and on the readout parameters, while measuring the resulting feature fidelity and downstream utility. Figure 2 summarizes this design.

3.1 Problem Formulation and DUG Definition

We consider multimodal robustness under visual distribution shifts. Given a clean image x, a corruption process $\mathcal{C}_{c,s}(\cdot)$ generates a corrupted observation:

$$\tilde{x} = \mathcal{C}_{c,s}(x), \tag{1}$$

where c denotes the corruption type and s denotes the corruption severity.

A frozen visual encoder E maps clean and corrupted observations into a shared representation space:

$$z = E(x), \qquad \tilde{z} = E(\tilde{x}), \tag{2}$$

where $z, \tilde{z} \in \mathbb{R}^{N \times d}$.

In our controlled experiments, we use DINOv3 as the shared visual encoder. Keeping the encoder fixed allows us to isolate the influence of representation changes from variations caused by different visual backbones.

A representation recovery module R_θ aims to restore corrupted features toward their clean counterparts:

$$\hat{z} = R_\theta(\tilde{z}). \tag{3}$$

We distinguish representation fidelity from downstream utility.

Representation fidelity measures how close the recovered representation is to the clean reference:

$$\mathcal{F}(R_\theta) = \phi(\hat{z}, z), \tag{4}$$

We use a higher-is-better convention for $\mathcal{F}$: for example, ϕ can be cosine similarity or negative reconstruction error. Tables report MSE in its usual lower-is-better form and mark metric directions explicitly.

However, high representation fidelity does not necessarily imply high multimodal utility. For an MLLM m, let $P^{(m)}$ denote the vision–language interface and $L^{(m)}$ denote the frozen language-side computation. The downstream utility is defined as:

$$\mathcal{U}^{(m)}(R, P) = S\left(L^{(m)}(P(R(\tilde{z}))), a\right), \tag{5}$$

where $S(\cdot)$ is the benchmark-specific evaluation metric.

The question is whether an increase in representation fidelity is accompanied by an increase in utility under a fixed readout:

$$\Delta\mathcal{F} > 0 \Rightarrow \Delta\mathcal{U} > 0. \tag{6}$$

We use *Denoising–Utility Gap (DUG)* to denote an improvement in representation fidelity without a corresponding improvement in downstream utility under a fixed readout. Establishing this pattern requires fidelity and utility measurements for the same recovery and evaluation setting. DUG describes a discrepancy between these outcomes, rather than a numerical difference between their metrics.

3.2 Controlled Representation Recovery

To study the relationship between representation recovery and downstream utility, we train recovery modules using paired clean and corrupted visual representations without downstream multimodal supervision.

The recovery module maps a corrupted representation to a recovered representation,

$$\hat{z} = R_\theta(\tilde{z}). \tag{7}$$

A common restoration component of the recovery objective is

$$\mathcal{L}_{\text{rest}} = \lambda_1 \|\hat{z} - z\|_2^2 + \lambda_2 \big(1 - \cos(\hat{z}, z)\big). \tag{8}$$

Different recovery configurations may augment this restoration component with additional structure-preserving regularization. In particular, the task-agnostic Open Images recovery configuration uses the auxiliary terms reported in Appendix 9.

Recovery modules are trained with representation-level supervision and no downstream question–answer labels. For each recovery setting, the trained module is held fixed during readout adaptation and downstream evaluation. Representation fidelity is evaluated separately for each setting.

3.3 Readout Adaptation

We test readout dependence by adapting the readout while holding the visual encoder and recovery module fixed. This intervention measures how updating the readout changes the utility of a given representation.

For each evaluated MLLM m, we construct a model-specific replacement readout $P^{(m)}$ that maps the shared DINOv3 representation space into the native visual-token space expected by the frozen downstream model:

$$P^{(m)} : \mathcal{Z} \to \mathcal{T}^{(m)}. \tag{9}$$

We first train a clean-trained interface using clean visual representations:

$$P_f^{(m)} = \arg\min_P \ell(P(z), t^{(m)}), \tag{10}$$

Here, $t^{(m)}$ denotes the native visual-token target supplied by the original MLLM visual pathway. In the main GQA experiments, this target is extracted from the corresponding clean image.

The clean-trained readout serves as the fixed reference condition. Each adapted readout is initialized from the corresponding clean-trained readout and optimized on its adaptation distribution:

$$P_a^{(m)} = \arg\min_P \mathbb{E}_{(z', t^{(m)}) \sim \mathcal{D}_{\text{adapt}}} \left[\ell\big(P(z'), t^{(m)}\big) \right]. \tag{11}$$

In the GQA experiments, the adaptation inputs for each image pair are

$$\begin{aligned} \text{Raw-Adapted:} \quad & z' \in \{E(x), E(\tilde{x})\}, \\ \text{Recovered-Adapted:} \quad & z' \in \{R_\theta(E(x)), R_\theta(E(\tilde{x}))\}, \\ \text{Clean-only Adapted:} \quad & z' = R_\theta(E(x)). \end{aligned}$$

All three continuation conditions use clean-image native visual tokens as targets and start from the same clean-trained readout. The clean-only condition changes the adaptation input distribution by excluding corrupted inputs.

During interface adaptation, all upstream visual encoders, recovery modules, and language models remain frozen. Thus, adaptation only changes how the downstream model consumes a given representation distribution.

3.4 Controlled Representation–Readout Analysis

We evaluate a 2×2 design that varies the representation supplied to the readout and whether the readout is adapted. The representation is either the encoder output directly (*Raw*) or the output of the recovery module (*Recovered*); the readout is either held at its clean-trained parameters (*Frozen*) or adapted to the corresponding input distribution (*Adapted*). This yields Raw-Frozen, Recovered-Frozen, Raw-Adapted, and Recovered-Adapted. The following contrasts distinguish the utility effects of recovery, readout adaptation, and their combination.

Recovery effect. Comparing Recovered-Frozen and Raw-Frozen measures the downstream utility effect of inserting the recovery intervention while keeping the readout fixed:

$$\Delta_{\text{rec}} = \mathcal{U}(\text{Recovered-Frozen}) - \mathcal{U}(\text{Raw-Frozen}). \tag{12}$$

A non-positive point estimate of Δ_{rec} indicates no observed utility gain under the fixed readout. Interpreting this result as DUG additionally requires improved fidelity in the same recovery and evaluation setting. Statistical uncertainty is assessed separately in Appendix C.

Interface adaptation effect. Comparing Recovered-Adapted and Recovered-Frozen holds the recovered representation fixed while changing only the downstream readout:

$$\Delta_{\text{adapt}} = \mathcal{U}(\text{Recovered-Adapted}) - \mathcal{U}(\text{Recovered-Frozen}). \tag{13}$$

Because the recovered representation is unchanged, this comparison measures the utility effect of changing the readout. It does not, by itself, distinguish distribution matching from benefits due to additional readout optimization; we evaluate this confound with a matched continuation control in Section 4.3.

Marginal recovery effect. Finally, comparing Recovered-Adapted and Raw-Adapted evaluates whether explicit representation recovery provides additional benefit once both representations are paired with adapted interfaces:

$$\Delta_{\text{marg}} = \mathcal{U}(\text{Recovered-Adapted}) - \mathcal{U}(\text{Raw-Adapted}). \tag{14}$$

These contrasts distinguish changes to the representation from changes to the readout. They quantify readout-dependent utility without uniquely attributing the observed effects to information loss or distribution mismatch.

4 Experiments

Our experiments examine three questions: how recovery affects utility under a fixed readout; how adapting the readout changes the utility of raw and recovered features; and whether explicit recovery remains beneficial after adaptation. We evaluate these comparisons across four model architectures and nine benchmarks, then examine a clean-only continuation control, adaptation-data scale, and the recovery-training distribution. Separate representation-level analyses characterize fidelity.

4.1 Experimental Setup

We evaluate representative multimodal large language models, including LLaVA-1.5, Qwen2.5-VL, Qwen3.5, and LLaVA-OneVision-2, which cover different vision–language coupling mechanisms.

A DINOv3 visual encoder defines the shared visual representation space. The recovery module is trained to map corrupted representations toward their clean counterparts without downstream task supervision. During evaluation, the recovery module remains frozen and is applied before the vision–language interface.

We consider common visual degradations including noise, blur, compression, and exposure variation. Downstream utility is evaluated on GQA, TextVQA, TallyQA, POPE, VQAv2, VizWiz (Gurari et al., 2018), MMStar (Chen et al., 2024), BLINK (Fu et al., 2024), and NaturalBench (Li et al., 2024).

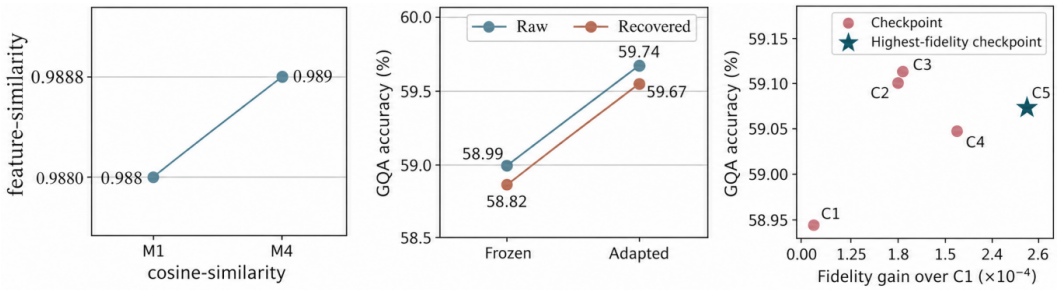
We use the four representation–readout conditions defined in Section 3.4 and summarized in Table 5.

For controlled cross-model evaluation, each architecture is connected to the shared DINOv3 representation through a separately trained diagnostic replacement readout. The readout maps DINOv3 features into the native visual-token space expected by the frozen downstream multimodal model; the native visual backbone, native multimodal components, and language-side model remain frozen. Consequently, cross-model results are interpreted through within-model condition differences rather than absolute native benchmark performance across MLLMs.

4.2 Main Results and Generalization

We first compare the utility effects of recovery and readout adaptation across models and benchmarks.

Table 1 reports the four controlled conditions across four models and nine benchmarks. Recovery under a frozen readout does not yield a consistent utility gain; its effect varies across tasks and architectures.



(a) Feature fidelity (b) Downstream utility (c) Fidelity–utility decoupling

Figure 3: Representation recovery and readout adaptation affect downstream utility through different interventions. The four conditions compare raw and recovered features under frozen and adapted readouts. Recovery has heterogeneous utility effects under a fixed readout, while adaptation can change the utility of the same representation.

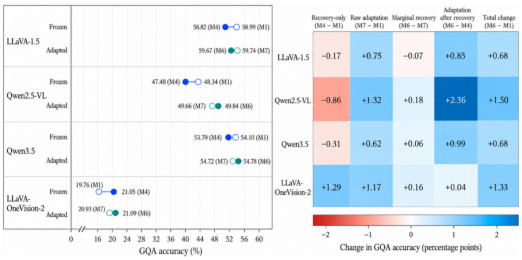
Changing the readout produces a substantially more consistent effect. Holding the recovered input fixed, Recovered-Adapted outperforms Recovered-Frozen in 33 of 36 model–benchmark combinations. Because the representation is unchanged within each comparison, this result establishes substantial readout dependence of downstream utility.

Explicit recovery, however, provides no uniform advantage once the readout is adapted. Recovered-Adapted exceeds Raw-Adapted in only 10 of 36 combinations; direct adaptation to raw corrupted features is competitive with or better than recovery followed by adaptation in the remaining cases. These results motivate evaluating the representation–readout pair rather than judging a recovery intervention from representation fidelity alone. The counts summarize point estimates, not statistically significant effects.

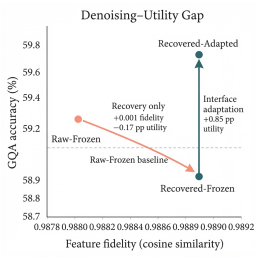
Table 1: Downstream utility across four models and nine benchmarks. Scores follow Appendix A.5 and characterize controlled systems using a shared DINOv3 encoder and model-specific replacement readouts. The best and second-best results within each setting are highlighted.

Model	Condition	GQA	TextVQA	TallyQA	POPE	VQAv2	VizWiz	MMStar	BLINK	NaturalBench
LLaVA-1.5	Raw-Frozen	58.99	9.96	54.00	82.42	65.45	17.83	27.53	36.12	46.86
	Recovered-Frozen	58.82	9.57	49.01	80.91	64.62	17.29	27.38	**36.14**	45.69
	Raw-Adapted	**59.74**	**10.11**	**55.29**	**82.47**	**66.71**	**18.38**	**28.16**	35.77	**47.42**
	Recovered-Adapted	59.67	9.97	54.74	82.18	66.53	18.28	27.71	35.12	46.92
Qwen2.5-VL	Raw-Frozen	48.34	7.62	53.97	75.24	60.83	17.23	34.22	38.52	48.45
	Recovered-Frozen	47.48	6.82	48.62	74.73	57.50	13.72	33.84	37.82	46.59
	Raw-Adapted	49.66	**8.70**	55.38	75.87	**63.40**	**20.06**	**35.53**	38.79	**49.61**
	Recovered-Adapted	**49.84**	8.17	**55.82**	**76.17**	62.97	18.43	35.24	**39.05**	49.28
Qwen3.5	Raw-Frozen	54.10	10.87	60.29	83.18	68.63	20.24	39.49	41.47	62.79
	Recovered-Frozen	53.79	9.98	58.29	82.87	68.05	17.88	36.91	41.14	61.47
	Raw-Adapted	54.72	11.17	**60.84**	83.14	**69.59**	**21.46**	**40.44**	**42.22**	**63.30**
	Recovered-Adapted	**54.78**	**11.23**	60.76	**83.42**	69.37	20.75	39.64	41.07	63.18
OneVision-2	Raw-Frozen	19.76	10.11	32.77	77.16	23.36	11.30	34.44	24.88	46.77
	Recovered-Frozen	21.05	10.19	33.99	77.01	24.57	10.74	33.42	25.58	44.50
	Raw-Adapted	20.93	**10.44**	35.64	**77.84**	26.05	**12.32**	**35.80**	**26.55**	**47.71**
	Recovered-Adapted	**21.09**	10.38	**36.67**	77.73	**26.19**	11.54	35.78	25.43	47.16

Across multimodal architectures. We further examine whether the observed behavior is specific to a single MLLM. Figure 4a decomposes the recovery-only and interface-adaptation effects across four architectures. On GQA, the recovery-only effect is negative for LLaVA-1.5, Qwen2.5-VL, and Qwen3.5, but positive for OneVision-2, indicating that the utility effect of representation recovery varies across the evaluated architectures. In contrast, adapting the readout to the recovered representation produces a positive GQA change for all four evaluated architectures, although the magnitude differs substantially across models.



(a) Cross-model utility effects of recovery and readout adaptation.



(b) Controlled representation–readout comparisons.

Figure 4: Controlled representation–readout comparisons. (a) Recovery-only and recovered-input adaptation effects across downstream models. (b) Utility changes between the evaluated representation–readout conditions. These utility comparisons do not by themselves establish an improvement in representation fidelity.

Table 2: Matched-optimization control on GQA. Clean-only adaptation excludes recovered corrupted inputs; scores are percentages.

Condition	**Adaptation input**	**GQA ↑**	**Δ vs. Frozen**
Recovered-Frozen	Frozen	58.82	0.00
Clean-only Adapted	Recovered clean	59.35	+0.53
Recovered-Adapted	Recovered clean + corrupted	**59.67**	**+0.85**

Together, the task-level and cross-model results show that the utility effect of recovery varies across tasks and architectures and depends on the readout paired with the representation.

4.3 Readout Dependence and Matched-Optimization Control

Fidelity varies across recovery settings. The task-specific module does not improve encoder-space fidelity on the GQA diagnostic subset (Appendix F), whereas the Open Images-trained module improves all four fidelity metrics on ImageNet-1K (Appendix B.3). These measurements characterize different recovery and evaluation settings.

To distinguish distribution matching from additional readout optimization, Table 2 adds a clean-only continuation control to the canonical LLaVA-1.5 GQA setting.

With the same initialization and optimization budget, clean-only adaptation raises GQA from 58.82 to 59.35; including recovered corrupted inputs reaches 59.67. Thus, the full 0.85-point gain cannot be attributed solely to matching the corrupted-input distribution.

4.4 Robustness Across Recovery Settings

We next examine the scope of the readout-adaptation results along two dimensions: the amount of adaptation data and the distribution used to train the recovery module. These comparisons test whether the observed utility effects persist with smaller adaptation sets and with task-agnostic recovery.

Interface adaptation with limited data. We vary the fraction of adaptation data while fixing the visual representation, downstream model, and optimization budget. The 0% condition uses the frozen readout; nonzero fractions use progressively larger subsets of the same adaptation distribution. The 100% scores in this scaling study differ from the adapted scores in Table 1, so nonzero fractions are compared within the scaling study.

On both GQA and POPE, the 1% setting underperforms the corresponding frozen baselines. Larger subsets improve utility relative to this setting, although performance is not monotonic in data scale. For example, raw-input GQA reaches 59.65 at 10%, compared with 59.44 at 100%.

The amount of data needed to exceed the frozen baseline also depends on the representation and task. At 5% on POPE, the raw-input score of 81.88 remains below Raw-Frozen (82.42), while the recovered-input score of 81.94 exceeds Recovered-Frozen (80.91). Thus, smaller adaptation sets can improve utility, but do not uniformly match or exceed the frozen baselines.

Table 3: Effect of interface-adaptation data scale on downstream utility. The 0% condition uses the original frozen interface, while nonzero ratios adapt the interface using increasing fractions of distribution-matched data. All values are percentages. Within each representation row, the best result is highlighted in darker blue and bolded, while the second-best result is highlighted in lighter blue.

(a) GQA

	Adaptation Data					
Representation	0%	1%	5%	10%	25%	100%
Raw	58.99	55.85	59.32	**59.65**	59.55	59.44
Recovered	58.82	56.87	59.04	59.24	59.23	**59.65**

(b) POPE

	Adaptation Data					
Representation	0%	1%	5%	10%	25%	100%
Raw	82.42	77.17	81.88	82.10	82.16	**82.52**
Recovered	80.91	78.47	81.94	81.90	82.15	**82.51**

Task-specific versus task-agnostic recovery. We evaluate a recovery module trained on an Open Images subset (Kuznetsova et al., 2020) without downstream question–answer supervision. On held-out ImageNet-1K representations, it improves all four fidelity metrics (Appendix B.3).

Under the frozen readout, downstream utility changes are mixed: GQA and NaturalBench improve slightly, while the other five benchmarks decrease relative to Raw-Frozen. Readout adaptation increases all seven benchmark scores (Table 4), including GQA ($59.09 \rightarrow 59.84$), VQAv2 ($65.29 \rightarrow 66.63$), and MMStar ($27.24 \rightarrow 28.78$). The utility gains from readout adaptation therefore extend to recovery trained on task-agnostic data.

The ImageNet fidelity evaluation and downstream utility evaluations use different datasets. They characterize the same recovery module but do not constitute matched fidelity–utility measurements.

Table 4: Generalization across recovery training distributions. The recovery module is trained either on task-specific data or on a task-agnostic Open Images subset. We report downstream utility for recovered representations before and after adapting the vision–language interface. All values are percentages. Within each recovery-training setting, the better of the two interface conditions is highlighted in light blue and bolded.

Recovery Training	Condition	GQA	POPE	VQAv2	VizWiz	MMStar	BLINK	NaturalBench
Task-specific	Recovered-Frozen	58.82	80.91	64.62	17.29	27.38	**36.14**	45.69
	Recovered-Adapted	**59.67**	**82.18**	**66.53**	**18.28**	**27.71**	35.12	**46.92**
OpenImages Generic	Recovered-Frozen	59.09	82.37	65.29	17.79	27.24	35.82	47.03
	Recovered-Adapted	**59.84**	**82.51**	**66.63**	**18.30**	**28.78**	**36.17**	**47.36**

The purifier-design, corruption-specific, multi-seed, bootstrap, recovery–utility sweep, and additional interface-compatibility analyses are reported in the appendix.

5 Conclusion

We study representation recovery and readout adaptation as separate interventions in shared-encoder multimodal systems. Recovery under a clean-trained readout has heterogeneous utility effects, whereas readout adaptation improves utility in most evaluated settings. Direct adaptation to raw features is frequently competitive with or better than adaptation after explicit recovery. The clean-only continuation control shows that part of the adaptation gain is also obtained without corrupted training inputs. Separate fidelity analyses characterize the task-specific and task-agnostic recovery settings. These results motivate evaluating representations together with their downstream readouts and reporting fidelity, fixed-readout utility, and adapted-readout utility separately.

REFERENCES

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, et al. Flamingo: A visual language model for few-shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL https://arxiv.org/abs/2204.14198.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Qiao Yu, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models? In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 27056–27087, 2024. doi: 10.52202/079017-0850.

Prakash Chandra Chhipa, Johan Rodahl Holmgren, Kanjar De, Rajkumar Saini, and Marcus Liwicki. Can self-supervised representation learning methods withstand distribution shifts and corruptions? In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 4469–4478, 2023. URL https://arxiv.org/abs/2308.02525.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: Multimodal large language models can see but not perceive. In *Computer Vision – ECCV 2024*, pp. 148–166, 2024. doi: 10.1007/978-3-031-73337-6_9.

Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. VizWiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3608–3617, 2018. doi: 10.1109/CVPR.2018.00380.

Drew A. Hudson and Christopher D. Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6700–6709, 2019.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, 2020. doi: 10.1007/s11263-020-01316-z.

Baiqi Li, Zhiqiu Lin, Wenxuan Peng, Jean de Dieu Nyandwi, Daniel Jiang, Zixian Ma, Simran Khanuja, Ranjay Krishna, Graham Neubig, and Deva Ramanan. NaturalBench: Evaluating vision-language models on natural adversarial samples. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 17044–17068, 2024. doi: 10.52202/079017-0542.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023. URL https://arxiv.org/abs/2301.12597. arXiv:2301.12597.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://arxiv.org/abs/2304.08485.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. URL https://arxiv.org/abs/2310.03744.

Dhruv Parikh, Jacob Fein-Ashley, Rajgopal Kannan, and Viktor Prasanna. Latent denoising improves visual alignment in large multimodal models, 2026. URL https://arxiv.org/abs/2604.21343. arXiv:2604.21343.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, et al. DINOv3, 2025. URL https://arxiv.org/abs/2508.10104. arXiv:2508.10104.

Lin Song, Wenbo Li, Guoqing Ma, Wei Tang, Bo Wang, Yuan Zhang, Yijun Yang, Yicheng Xiao, Jianhui Liu, Yanbing Zhang, Guohui Zhang, Wenhu Zhang, Hang Xu, Nan Jiang, Xin Han, Haoze Sun, Maoquan Zhang, Haoyang Huang, and Nan Duan. JoyAI-Image: Awaking spatial intelligence in unified multimodal understanding and generation, 2026. URL https://arxiv.org/abs/2605.04128. arXiv:2605.04128.

Jiaqi Tang, Jianmin Chen, Youyang Zhai, Wei Wei, Runtao Liu, Mengjie Zhao, Xiangyu Wu, Qingfa Xiao, and Qifeng Chen. Robust-U1: Can MLLMs self-recover corrupted visual content for robust understanding?, 2026. URL https://arxiv.org/abs/2606.08063. arXiv:2606.08063.

Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, S. M. Ali Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 200–212, 2021.

Haoning Wu, Zicheng Zhang, Erli Zhang, Chaofeng Chen, Liang Liao, Annan Wang, Kaixin Xu, Chunyi Li, Jingwen Hou, Guangtao Zhai, Geng Xue, Wenxiu Sun, Qiong Yan, and Weisi Lin. Q-Instruct: Improving low-level visual abilities for multi-modality foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25490–25500, 2024.

Jiawei Yang, Katie Z. Luo, Jiefeng Li, Congyue Deng, Leonidas J. Guibas, Dilip Krishnan, Kilian Q. Weinberger, Yonglong Tian, and Yue Wang. Denoising vision transformers, 2024. URL https://arxiv.org/abs/2401.02957. arXiv:2401.02957.

Linli Yao, Lei Li, Shuhuai Ren, Lean Wang, Yuanxin Liu, Xu Sun, and Lu Hou. DeCo: Decoupling token compression from semantic abstraction in multimodal large language models, 2024. URL https://arxiv.org/abs/2405.20985. arXiv:2405.20985.

A Experimental and Reproducibility Details

We provide additional details of the controlled representation–interface evaluation protocol, model-specific readouts, optimization settings, benchmark evaluation procedures, and data construction used throughout the paper. Unless otherwise specified, downstream results use seed 123. Additional multi-seed and statistical analyses are reported in Appendix C.

A.1 Controlled Representation–Interface Conditions

Our experiments independently vary two factors: whether the visual representation is raw or recovered, and whether the vision–language interface remains frozen or is adapted to the corresponding representation distribution. This produces the four semantic conditions summarized in Table 5.

Table 5: Controlled representation–interface evaluation design. Representation recovery and interface adaptation are varied independently.

Representation	Frozen Interface	Adapted Interface
Raw	Raw-Frozen	Raw-Adapted
Recovered	Recovered-Frozen	Recovered-Adapted

A.2 Shared Representation and Model-Specific Readouts

All evaluated architectures consume a shared DINOv3 representation containing 576 patch tokens with feature dimension 1024. Special and register tokens are removed before the representation is passed to the model-specific vision–language readout. Because the native visual token spaces differ across architectures, each MLLM uses a separately trained readout that maps the shared DINOv3 representation into the visual-token space expected by its frozen language-side computation.

Table 6: Model-specific vision–language readouts used in the cross-model analysis. All models consume the same 576×1024 DINOv3 representation. Dynamic token counts denote image-dependent native visual-token sequences.

Model	Native Target Stage	Target Tokens	Target Dim.	Readout	Trainable Params.	Token Alignment
LLaVA-1.5	Post-native projector	576	4096	$1024 \to 4096 \to 4096$	20.98M	Direct 576-token alignment
Qwen2.5-VL	Native visual output	Dynamic	3584	$1024 \to 4096 \to 3584$	18.88M	Deterministic linear resampling
Qwen3.5	Native visual output	Dynamic	4096	$1024 \to 4096 \to 4096$	20.98M	Deterministic linear resampling
OneVision-2	Native visual output	Dynamic	4096	$1024 \to 4096 \to 4096$	20.98M	Deterministic linear resampling

For LLaVA-1.5, the target consists of 576 native visual tokens with dimension 4096, so the DINOv3 token sequence already matches the native token count. For Qwen2.5-VL, Qwen3.5, and OneVision-2, the native number of visual tokens is image dependent. We therefore deterministically resample the DINOv3 sequence to the native token count before applying the model-specific readout.

Each readout is a two-layer MLP with GELU activation. During interface training and adaptation, only the readout parameters are optimized; the DINOv3 encoder, recovery module, native multimodal model, and language model remain frozen. Raw-Adapted and Recovered-Adapted use the same optimization protocol within each architecture. Consequently, the cross-model experiments are interpreted through controlled within-model differences rather than absolute scores across architectures.

A.3 Interface Training and Optimization

The clean-trained interface is optimized to reproduce the native visual tokens of each MLLM from clean DINOv3 representations. The training objective combines mean-squared error with token-wise cosine distance over valid visual tokens. The same optimizer and optimization budget are used for all four evaluated architectures.

Table 7: Common optimization settings for clean-trained and adapted vision–language readouts.

Setting	Value
Optimizer	AdamW
Learning rate	1×10^{-4}
Weight decay	0.01
Adam β	(0.9, 0.95)
Gradient clipping	1.0
Scheduler	None
Batch size	64
Gradient accumulation	1
Maximum optimization steps	750
Minimum steps before stopping	375
Validation subset size	1024
Precision	bfloat16
Default seed	123
Checkpoint criterion	Minimum validation MSE + cosine loss

The clean-trained interfaces for all four architectures are trained on 72,140 clean representation–native-token pairs. Adapted readouts are initialized from the corresponding clean-trained interface and optimized only on the target representation distribution.

A.4 Task-Agnostic Recovery Implementation

For the task-agnostic Open Images experiment, we use a residual dual-stream variational recovery module. The content branch uses a hidden dimension of 1024 and a latent dimension of 128, while the nuisance branch uses a hidden dimension of 512 and a latent dimension of 64. The decoder hidden dimension is 1024, and the reconstructed residual is added to the input representation. The implementation contains approximately 5.52M trainable parameters.

Table 8: Architecture and optimization settings of the task-agnostic Open Images recovery module.

Setting	Value
Input / output dimension	1024
Content hidden / latent	1024 / 128
Nuisance hidden / latent	512 / 64
Decoder hidden dimension	1024
Activation	GELU
Layer normalization	Yes
Dropout	0
Approx. parameters	5.52M
Optimizer	AdamW
Learning rate	3×10^{-4}
Weight decay	0.01
Adam β	$(0.9, 0.95)$
Gradient clipping	1.0
Batch size	32
Scheduler	Cosine
Warmup steps	200
Training steps	12,000
Minimum early-stop step	8,000
Training seed	123

The task-agnostic recovery objective combines direct restoration losses with structure-preserving regularization. Table 9 lists the loss weights used to train the Open Images recovery module.

Table 9: Loss weights used for the task-agnostic Open Images recovery module.

Loss Term	Weight
Restoration MSE	1.0
Restoration cosine	1.0
Restoration Smooth-L_1	0.5
Global cosine	0.1
Relational consistency	0.2
KL regularization	0.005
Total-correlation regularization	0.02
Cycle consistency	0.3
Minimum clean-identity term	0.01
Minimum residual-L_2 term	0.02

The canonical task-specific experiments use a fixed task-specific recovery checkpoint throughout downstream evaluation. In all downstream conditions, the recovery module is frozen; interface adaptation therefore never changes the recovered representation itself.

A.5 Benchmark Evaluation Protocol

We evaluate all representation–interface conditions using the same benchmark-specific scorer within each task. Model generations are decoded only from the continuation after the prompt. Empty generations are treated as incorrect. Unless the benchmark requires a dedicated protocol, answers are lowercased, punctuation and articles are removed, and whitespace is canonicalized before scoring.

Avg-All-C is the macro-average of the available Clean, Medium, and Heavy condition scores. Each condition score is computed over its evaluated rows. When all three conditions are present,

$$S_{\text{Avg-All-C}} = \frac{S_{\text{Clean}} + S_{\text{Medium}} + S_{\text{Heavy}}}{3}.$$

Thus, Avg-All-C includes clean inputs and is distinct from the corrupted-only Avg-C score. Evaluated-row counts include condition expansion and should not be interpreted as counts of unique images.

Table 10: Benchmark-specific scoring protocols. TextVQA and VizWiz use the soft score $\min(1, \text{matches}/3)$, whereas VQAv2 uses normalized exact-match accuracy. These study-specific scoring rules differ from the corresponding official VQA metrics. Counts refer to evaluated rows.

Benchmark	Split	Evaluated rows	Metric	Answer Processing
GQA	val	30,000	Normalized exact match	Lowercase, punctuation/article removal
TextVQA	val	15,000	VQA-style soft accuracy	Standard normalized short-answer matching
TallyQA	test	30,000	Exact count accuracy	Normalization + number words 0–20
POPE	val2014	27,000	Canonical yes/no accuracy	Official first-sentence yes/no extraction
VQAv2	screening-seed123	15,000	Implemented normalized exact match	Normalized answer matching
VizWiz	val	12,957	Implemented soft accuracy	Standard normalized short-answer matching
MMStar	val	4,500	Normalized exact match	Lowercase, punctuation/article removal
BLINK	val	5,703	Normalized exact match	Lowercase, punctuation/article removal
NaturalBench	train	22,800	Normalized exact match	Lowercase, punctuation/article removal

POPE answer parsing. POPE predictions are evaluated using the same parser across all models. Empty generations are marked incorrect before applying the yes/no parser to nonempty responses. For nonempty responses, only the first sentence is considered and commas are removed. Responses containing the tokens "No", "no", or "not" are classified as negative; all other nonempty responses are classified as affirmative. The same procedure is applied to LLaVA-1.5, Qwen2.5-VL, Qwen3.5, and OneVision-2, with no architecture-specific parser.

A.6 Interface-Adaptation Scaling Protocol

The adaptation-scaling experiment varies the number of unique distribution-matched samples available to the interface while holding the configured optimization budget fixed. The nonzero subsets are nested and are constructed with seed 123. Raw and recovered conditions use the same example identities at each ratio and are initialized from the same clean-trained interface.

The 0% condition reported in the main paper corresponds to the original frozen interface and involves no adaptation training. We therefore omit it from the adaptation-training protocol in Table 11 and report only the nonzero adaptation subsets.

Table 11: Data subsets used for interface adaptation. All reported conditions use the same 750-step optimization budget.

Adaptation Ratio	Unique Samples	Nested	Optimization Steps
1%	76	Yes	750
5%	384	Yes	750
10%	769	Yes	750
25%	1,922	Yes	750
100%	7,691	Yes	750

All adaptation settings use AdamW with learning rate 10^{-4} and batch size 64. Because the optimization budget is fixed across ratios, the experiment varies the diversity of available adaptation examples rather than allocating more optimization steps to larger subsets.

A.7 Corruption Generation Protocol

We evaluate five common corruption families at three severity levels. The corruption instances are deterministic and shared across evaluated models. Gaussian-noise randomness is seeded from a hash of the protocol identifier, sample identifier, corruption family, and severity.

Training excludes the held-out heavy corruption settings used for final test evaluation. Downstream labels are not used to train either the recovery module or the representation–interface readout.

Table 12: Parameters used to construct the three corruption severity levels. For exposure corruptions, the value denotes the multiplicative brightness factor.

Corruption	Severity 1	Severity 2	Severity 3
Gaussian Blur (radius)	1.0	2.0	4.0
Gaussian Noise (σ)	8	16	32
JPEG (quality)	70	45	25
Overexposure (factor)	1.35	1.75	2.25
Underexposure (factor)	0.75	0.55	0.35

A.8 Task-Agnostic Open Images Recovery Data

We train the task-agnostic recovery module on paired clean and corrupted DINOv3 features from 72,140 images in Open Images. Images are assigned using a deterministic SHA256-sorted balanced round-robin policy. Training uses paired features without downstream question–answer annotations.

Clean and corrupted feature pairs are generated using the same corruption protocol as the task-specific setting, and the clean DINOv3 representation is used as the recovery target. The recovery model is trained for 12,000 steps with seed 123. No GQA, POPE, VQAv2, or other downstream QA annotations are used during this training.

Table 13: Task-agnostic Open Images recovery-data protocol.

Setting	Value
Images	72,140
Supervision	Clean/corrupted feature pairs
QA supervision	None
Training seed	123
Corruption seed	20260920
Recovery steps	12,000
Downstream annotations used	No
ImageNet-1K validation images	50,000

We evaluate this module on 50,000 ImageNet-1K validation images (Deng et al., 2009) to measure representation fidelity on a separate dataset (Appendix B.3).

B Recovery Design and Training Controls

We examine how recovery architecture and training objective affect representation fidelity and downstream utility, and assess additional readout optimization using a clean-only continuation control.

B.1 Recovery Architecture and Objective Ablation

Table 14 compares alternative recovery designs using both representation-level fidelity metrics and downstream utility. The full recovery objective obtains the strongest reconstruction metrics, but does not yield the highest downstream GQA score. This provides an additional example that improvements in representation fidelity do not translate monotonically into downstream utility.

Table 14: Sensitivity to recovery architecture and training objective. We report both representation fidelity and downstream utility. Lower MSE and normalized MSE are better, whereas higher cosine similarity, alignment, and downstream scores are better.

Variant	MSE ↓	Norm-MSE ↓	Cosine ↑	Alignment ↑	GQA ↑	POPE ↑
Raw	0.001885	0.023075	0.988466	0.996742	58.99	82.42
Linear	0.001975	0.024196	0.988140	0.996402	59.09	82.36
Shallow	0.001944	0.023839	0.988423	0.996650	**59.17**	82.30
Rec-only	0.001906	0.023372	0.988560	**0.996849**	59.14	**82.41**
Align-only	0.002035	0.024946	0.987695	0.995914	59.09	82.30
Full	**0.001864**	**0.022826**	**0.988735**	0.996833	59.09	82.37

B.2 Matched-Optimization Control

The clean-only control in Table 2 continues readout training on recovered clean features. Comparing this condition with Recovered-Adapted assesses the additional utility associated with including corrupted inputs during adaptation.

B.3 Task-Agnostic Representation Validation

We additionally evaluate the Open Images-trained recovery module on held-out ImageNet-1K representations. This test measures whether the generic recovery module improves representation fidelity independently of downstream multimodal supervision.

Table 15: Representation-level validation of the task-agnostic Open Images recovery module on ImageNet-1K. Lower error and higher similarity are better.

Representation	MSE ↓	Norm-MSE ↓	Cosine ↑	Alignment ↑
Raw	0.003091	0.032864	0.982917	0.993639
Recovered	**0.002993**	**0.031825**	**0.983556**	**0.994151**

Recovery reduces both reconstruction-error metrics and increases both similarity metrics on ImageNet-1K, extending its fidelity improvements to a dataset distinct from the Open Images training set.

C Statistical Reliability

We evaluate the stability of the controlled comparisons across random seeds and quantify uncertainty using paired bootstrap confidence intervals.

C.1 Multi-Seed Evaluation

Table 16: GQA performance across three random seeds. Mean and standard deviation are computed over the three runs. All values are percentages.

Seed	Raw-Frozen	Recovered-Frozen	Raw-Adapted	Recovered-Adapted
42	59.03	58.67	59.63	59.67
123	58.99	58.82	59.74	59.67
3407	59.18	58.93	59.45	59.44
Mean	59.07	58.81	59.61	59.59
Std.	0.10	0.13	0.15	0.13

Across all three seeds, Recovered-Frozen remains below Raw-Frozen, whereas interface adaptation increases the utility of the recovered representation. The qualitative controlled comparison is therefore stable across the evaluated random seeds.

C.2 Bootstrap Confidence Intervals

We additionally estimate uncertainty for the four controlled effects on the primary LLaVA-1.5 downstream benchmarks. The analysis uses a paired bootstrap over the intersection of evaluation-example identifiers in the two conditions being compared. We use 5,000 bootstrap resamples and report 95% percentile confidence intervals corresponding to the 2.5th and 97.5th percentiles of the bootstrap distribution.

These intervals condition on the fitted checkpoints and describe evaluation-sample uncertainty under the stated example-level resampling scheme. They do not measure variability across training seeds. Because questions and corruption instances may share an image, example-level resampling may underrepresent within-image dependence. The reported intervals are therefore interpreted with this limitation.

We denote the recovery effect by Δ_{rec}, the interface-adaptation effect by Δ_{adapt}, the raw-adaptation effect by Δ_{raw}, and the marginal recovery effect after adaptation by Δ_{marg}.

$$\Delta_{\mathrm{rec}} = \mathcal{U}(\text{Recovered-Frozen}) - \mathcal{U}(\text{Raw-Frozen}), \tag{15}$$

$$\Delta_{\mathrm{adapt}} = \mathcal{U}(\text{Recovered-Adapted}) - \mathcal{U}(\text{Recovered-Frozen}), \tag{16}$$

$$\Delta_{\mathrm{raw}} = \mathcal{U}(\text{Raw-Adapted}) - \mathcal{U}(\text{Raw-Frozen}), \tag{17}$$

$$\Delta_{\mathrm{marg}} = \mathcal{U}(\text{Recovered-Adapted}) - \mathcal{U}(\text{Raw-Adapted}). \tag{18}$$

Table 17: Paired-bootstrap confidence intervals for the controlled utility effects on LLaVA-1.5. Each entry reports the percentage-point difference followed by its 95% percentile confidence interval.95% percentile confidence interval. Differences are computed before rounding and may differ by 0.01 percentage points from subtracting the displayed scores.

Benchmark	Δ_{rec}	Δ_{adapt}	Δ_{raw}	Δ_{marg}
GQA	-0.17 [-0.44, 0.11]	$+0.85$ [0.58, 1.14]	$+0.75$ [0.53, 0.96]	-0.07 [-0.26, 0.12]
TextVQA	-0.39 [-0.60, -0.18]	$+0.40$ [0.18, 0.62]	$+0.16$ [-0.02, 0.34]	-0.14 [-0.29, 0.01]
TallyQA	-4.99 [-5.41, -4.59]	$+5.72$ [5.29, 6.16]	$+1.28$ [0.98, 1.57]	-0.55 [-0.79, -0.31]
POPE	-1.51 [-1.77, -1.23]	$+1.26$ [1.00, 1.52]	$+0.04$ [-0.14, 0.24]	-0.29 [-0.46, -0.12]
VQAv2	-0.83 [-1.23, -0.43]	$+1.91$ [1.47, 2.31]	$+1.26$ [0.97, 1.57]	-0.18 [-0.43, 0.07]

For the benchmarks listed here, recovery under a frozen readout has negative point estimates, whereas adaptation on recovered inputs has positive point estimates. These comparisons characterize readout-dependent utility changes without uniquely identifying their mechanism. Small differences between adapted raw and recovered conditions are not interpreted as evidence of equivalence.

D Corruption-Level Robustness

We further disaggregate GQA performance by corruption family and severity. This analysis tests whether the observed behavior is driven by a small subset of visual degradations.

The effect of recovery alone varies across corruption families and severity levels rather than producing a uniform utility gain. Adapted-interface conditions are generally competitive with or stronger than their corresponding frozen conditions, further supporting the distinction between representation recovery and representation usability.

Table 18: Corruption-specific GQA performance across corruption families and severity levels. All values are percentages.

Corruption	Severity	Raw-Frozen	Recovered-Frozen	Raw-Adapted	Recovered-Adapted
Gaussian Blur	1	59.31	60.20	60.20	59.80
	2	59.76	59.76	60.96	60.46
	3	55.28	54.64	56.12	56.47
Gaussian Noise	1	59.28	59.67	60.06	60.93
	2	60.86	60.46	60.66	61.16
	3	58.32	57.76	59.12	58.52
JPEG	1	57.57	56.87	58.57	58.67
	2	58.59	57.80	59.68	60.08
	3	58.06	58.45	58.25	58.55
Overexposure	1	56.83	57.25	57.35	57.35
	2	58.64	59.26	59.47	58.74
	3	60.52	60.62	62.07	62.37
Underexposure	1	59.59	59.98	59.59	59.98
	2	61.26	61.26	61.76	62.06
	3	60.12	59.52	60.58	60.27

E Additional Fidelity–Utility Diagnostics

We provide additional analyses of how representation metrics and downstream utility vary across recovery checkpoints.

E.1 Recovery Checkpoint Sweep

Table 19 evaluates five recovery checkpoints. The fidelity metrics improve overall from C1 to C5, although MSE and normalized MSE increase slightly from C2 to C3. GQA reaches its highest score at C3 (59.16), while POPE reaches its largest score at C2 (82.40). C5 obtains 59.09 and 82.37, respectively. Thus, the checkpoint with the best final fidelity metrics does not maximize either downstream score in this sweep.

Table 19: Recovery–utility sweep across five checkpoints. Representation fidelity improves overall from C1 to C5, while the displayed GQA and POPE scores are non-monotonic across the checkpoint sequence. C1–C5 identify the displayed checkpoints.

Checkpoint	MSE ↓	Norm-MSE ↓	Cosine ↑	Alignment ↑	GQA ↑	POPE ↑
C1	0.001894	0.023185	0.988469	0.996452	58.93	82.27
C2	0.001886	0.023095	0.988594	0.996724	59.14	82.40
C3	0.001887	0.023099	0.988596	0.996759	**59.16**	82.30
C4	0.001871	0.022913	0.988668	0.996777	59.05	82.34
C5	**0.001864**	**0.022826**	**0.988735**	**0.996833**	59.09	82.37

E.2 Fidelity–Utility Correlation

We summarize the association between representation metrics and downstream utility over the same five recovery checkpoints. Because the sweep contains only five checkpoints, these values are descriptive diagnostics rather than formal statistical evidence.

The associations vary by metric and task. The reported GQA Pearson correlation with global alignment is 0.79, and the POPE Spearman correlation with MSE is -0.70. Because lower MSE indicates higher fidelity, the latter is directionally consistent with better fidelity accompanying higher utility. These five-checkpoint diagnostics do not support a general claim of no association. Their narrower observation is that downstream utility is non-monotonic across the displayed checkpoint sequence and is not maximized by C5.

Table 20: Descriptive correlation between representation-recovery metrics and downstream utility over five recovery checkpoints.

Benchmark	Recovery Metric	Pearson	Spearman
GQA	MSE	-0.25	-0.10
	Normalized MSE	-0.24	-0.10
	Cosine	0.52	0.20
	Global Alignment	0.79	0.20
POPE	MSE	-0.52	-0.70
	Normalized MSE	-0.51	-0.70
	Cosine	0.65	0.40
	Global Alignment	0.67	0.40

F Interface Compatibility Diagnostics

We complement downstream evaluation with direct measurements of how each representation distribution interacts with the corresponding vision–language interface. We report statistics both before and after the model-specific interface transformation.

Table 21: Representation–interface compatibility diagnostics on the GQA interface sample. "Pre" denotes DINOv3 features measured against the corresponding clean DINOv3 reference, while "Post" denotes measurements after the model-specific readout transformation. Lower MSE and normalized MSE and higher cosine similarity are better.

Condition	Stage	Cosine ↑	MSE ↓	Norm-MSE ↓	Pooled Cosine ↑	Norm Ratio
Raw-Frozen	Pre	0.988435	0.001886	0.023091	0.996749	1.001307
	Post	0.998598	0.001082	0.005069	0.999791	1.008561
Recovered-Frozen	Pre	0.894237	0.022155	0.272020	0.826113	1.134325
	Post	0.976949	0.011225	0.056457	0.981581	1.002465
Raw-Adapted	Pre	0.988435	0.001886	0.023091	0.996749	1.001307
	Post	0.992660	0.003378	0.017589	0.998397	1.021070
Recovered-Adapted	Pre	0.894237	0.022155	0.272020	0.826113	1.134325
	Post	0.992517	0.003464	0.017791	0.998598	0.997084

On this GQA diagnostic subset, task-specific recovery lowers token-wise cosine similarity to the clean DINOv3 reference from 0.988435 to 0.894237 and increases MSE from 0.001886 to 0.022155, indicating reduced encoder-space fidelity. Readout adaptation changes the post-interface statistics while leaving the input features unchanged. These measurements characterize changes before and after the readout; they do not isolate the cause of the downstream utility changes.

G Additional Benchmark Analyses

G.1 Detailed POPE Metrics

POPE accuracy alone may obscure changes in answer distribution or class imbalance. We therefore report precision, recall, F1, and yes-answer ratio in Table 22. All conditions are evaluated with the same canonical yes/no parser described in Appendix A.5.

The yes-answer ratio ranges from 41.11% to 42.92% across the evaluated conditions. For task-specific recovered inputs, readout adaptation increases precision from 87.60% to 87.88% and recall from 72.02% to 74.65%.

Table 22: Detailed POPE evaluation. All metric values except sample count are percentages.

Source	Condition	n	Acc.	Prec.	Recall	F1	Yes Ratio
Canonical	Clean-Frozen	27000	82.42	87.81	75.30	81.07	42.87
Canonical	Raw-Frozen	27000	82.42	87.81	75.30	81.07	42.87
Canonical	Recovered-Frozen	27000	80.91	87.60	72.02	79.05	41.11
Canonical	Raw-Adapted	27000	82.47	88.19	74.98	81.05	42.51
Canonical	Recovered-Adapted	27000	82.18	87.88	74.65	80.73	42.47
OpenImages	Recovered-Frozen	27000	82.37	87.71	75.29	81.03	42.92
OpenImages	Recovered-Adapted	27000	82.51	87.93	75.36	81.16	42.85

G.2 TallyQA-Complex

On TallyQA-Complex, recovery under the frozen readout lowers accuracy from 40.17 to 38.48. Adapting the readout to recovered features raises accuracy to 40.77, exceeding Raw-Frozen while remaining below Raw-Adapted (41.24). This subset therefore also shows a benefit from readout adaptation without an additional advantage from explicit recovery.

Table 23: Performance on the TallyQA-Complex subset. All values are percentages.

Condition	TallyQA-Complex ↑
Raw-Frozen	40.17
Recovered-Frozen	38.48
Raw-Adapted	41.24
Recovered-Adapted	40.77

The complex counting subset exhibits the same qualitative asymmetry as the main TallyQA evaluation: recovery alone reduces downstream utility, while interface adaptation recovers a substantial portion of the lost performance.

G.3 Relation to Main-Text Robustness Experiments

The main paper reports the complete task-agnostic Open Images transfer experiment and the interface-adaptation data-scaling study. We do not duplicate their utility tables here. The additional controls in this appendix instead test robustness to recovery architecture, optimization, representation fidelity, corruption family, random seed, and downstream task, while Sections A.6 and A.8 provide the corresponding reproducibility details.