Holistic EvoLution via Intrinsic eXchange for Unified Multimodal Models

Shenghao Dong* **Yuhang Yu** **Bo Li✉** **Jinwei Chen** **Hao Zhang**
vivo BlueImage Lab, vivo Mobile Communication Co., Ltd., China
dshao2020@gmail.com, libra@vivo.com

Abstract

Unified Multimodal Models (UMMs) integrate visual generation and understanding within a shared parameter space, yet existing post-training typically relies on external supervision and additional data, and rarely couples these two capabilities for iterative improvement. We propose Holistic EvoLution via Intrinsic eXchange (HELIX) for Unified Multimodal Models, a two-stage cyclic post-training framework that forms a double-helix coupling between generation and understanding: generation produces controllable visual evidence while understanding provides semantic judgments. A target-driven Data Bridge connects the two stages and supplies reliable training signals for both generation and understanding updates. In Stage 1, the understanding branch provides a self-evaluated likelihood gain reward to optimize the generative policy for stronger semantic faithfulness. In Stage 2, the generation branch supplies reconstruction-based visual evidence, and we refine the understanding branch with Reconstruction and Prompt-Alignment rewards. Extensive experiments on Bagel, trained at 512px, demonstrate consistent gains on compositional and knowledge-intensive benchmarks, outperforming the baseline by +5.6 on T2I-CompBench, +7.1 on GenEval, +1.54 on DPG-Bench and +0.02 on WISE when evaluated at 1024px.

1 Introduction

Recent text-to-image (T2I) models have achieved impressive fidelity and diversity [1–3], yet they still struggle to follow complex instructions that require deep semantic understanding, including external world knowledge, precise spatial relations, fine-grained attribute binding, and compositional reasoning [4–7]. These failures highlight a fundamental limitation of standard T2I architectures: generation quality does not necessarily imply strong semantic comprehension.

To address this gap, recent research has shifted toward Unified Multimodal Models (UMMs) that integrate understanding and generation within a single parameter space [8–11]. Although UMMs have rapidly advanced through large-scale pre-training [12–14], post-training for UMMs remains comparatively underexplored. Existing post-training methods often rely on external datasets or auxiliary models, and are commonly framed as one-shot optimization that performs a single-pass refinement of UMMs. Some focus on generation, while dual-task approaches also update both capabilities [15–20]. This motivates a cycle-based post-training strategy that leverages the intrinsic dual capabilities of UMMs.

In parallel, reinforcement-learning-based alignment methods such as GRPO have been applied to T2I generation, but they usually require external preference data or additional reward models [21, 22]. As shown in Figure 1, such auxiliary reward models can be brittle under compositional

*Intern at vivo Mobile Communication Co., Ltd.
✉Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

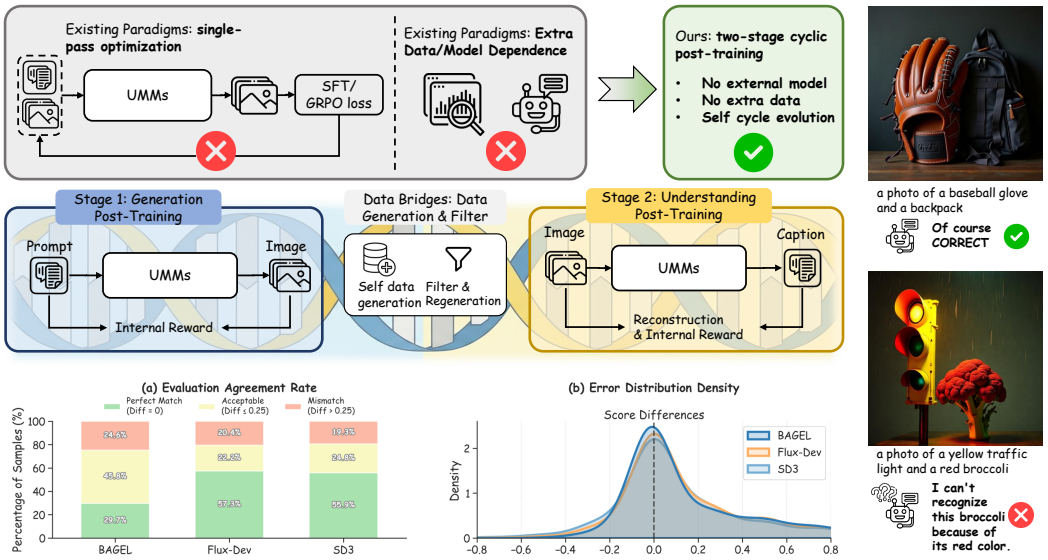


Figure 1: **HELIX overview and intrinsic-score validation.** HELIX (**H**olistic **E**voLution via **I**ntrinsic e**X**change) realizes a double-helix coupling between generation and understanding in Unified Multimodal Models. Stage 1 improves generation strand using intrinsic rewards from the understanding strand; Stage 2 updates the understanding strand using reconstruction-based and intrinsic signals from the improved generation strands. A Data Bridge synthesizes stable prompts and images for training. Bottom: consistency between GenEval reward and UMM-internal reward on images generated by the UMM, FLUX, and SD3.

generalization, struggling with fine-grained attribute binding and counter-intuitive yet valid concepts such as a red broccoli, which can lead to misleading optimization signals and reward hacking.

These limitations point to a cyclic post-training paradigm that avoids external reward models and explicitly couples the two capabilities within UMMs. We view this coupling as a double helix: generation produces controllable visual evidence, while understanding provides semantic judgments, and model-internal signals link them into an intrinsic exchange loop. Concretely, we treat the generation and understanding branches as two strands, and use a Data Bridge to deliver stable cross-stage training data and model-intrinsic reward signals, thereby realizing a two-stage interleaved cyclic training framework. Compared to one-shot post-training, the cycle enables iterative improvement by enhancing the quality of intrinsic rewards and providing stable training data, without introducing additional supervision. We compare images generated by the UMM, FLUX, and SD3 in Figure 1 [23, 24]. The UMM-internal evaluation largely agrees with GenEval scores, while a certain gap remains.

Building on this perspective, we propose HELIX, Holistic EvoLution via Intrinsic eXchange for Unified Multimodal Models, an RL-based two-stage cyclic post-training framework. At the start of each stage, a target-driven Data Bridge synthesizes prompts from predefined semantic targets. During training, we optimize the policy with intrinsic rewards. Together, these mechanisms connect the generation and understanding strands. In Stage 1, the understanding branch provides an intrinsic self-evaluated likelihood gain reward to optimize the generative policy for semantic faithfulness. In Stage 2, we strengthen the understanding policy using reconstruction-based reward from the improved generation strands, encouraging descriptions and judgments that are grounded in actionable visual evidence. Alternating these stages yields a self-reinforcing loop that strengthens intrinsic evaluation and provides stable training signals across cycles, without external paired training data or auxiliary reward models.

We evaluate HELIX on challenging text-to-image benchmarks that test compositional binding, dense prompt following, and knowledge-to-vision grounding, including GenEval, T2I-CompBench, DPG-Bench, and WISE. Our experiments are trained at 512^2 resolution and evaluated at both 512^2 and 1024^2 resolution. HELIX consistently improves over the baseline across all benchmarks, with particularly strong gains on GenEval and T2I-CompBench.

In summary, we make the following contributions: (1) We introduce HELIX, an RL-based two-stage cyclic post-training framework for unified multimodal models. (2) We design intrinsic rewards and a data bridge to connect the generation and understanding strands, where intrinsic signals are exchanged across strands to enable stable self-evolution without external preference data or auxiliary reward models. (3) We demonstrate consistent improvements on challenging text-to-image benchmarks covering compositional binding, dense prompt following, and knowledge-to-vision grounding, with gains transferring from the training resolution to the native resolution.

2 Related Work

2.1 Unified Multimodal Models

Unified multimodal models (UMMs) aim to unify multimodal understanding and generation within a single end-to-end backbone. By sharing parameters under a unified interface, UMMs promote cross-modal synergy and reduce system-level complexity. Recent work spans several architectural paradigms. Autoregressive designs discretize images into visual tokens and learn next-token prediction for both understanding and synthesis[11, 25–27]. Show-o introduces a discrete-diffusion scheduling mechanism to mitigate error accumulation in visual token generation [28]. Hybrid architectures couple an autoregressive Transformer backbone with a diffusion or flow-matching generation head, enabling higher-fidelity continuous image synthesis while retaining the strengths of autoregressive understanding[10]. Other systems keep a strong pretrained multimodal model fixed for reasoning and route intermediate representations through learnable queries to an external image generator [9, 12]. To improve scalability and efficiency, sparse and modular Mixture-of-Transformers designs have also been introduced, with Bagel as a representative instantiation [8]. This work adopts Bagel as the base UMM and investigates post-training optimization to further strengthen unified multimodal performance.

2.2 Post-training Strategies for UMMs

Post-training for UMMs generally falls into three categories. First, inference-time strategies augment generation with explicit reasoning steps [29–31], which enhances reliability but incurs high computational costs without improving intrinsic model parameters. Second, reinforcement learning approaches (e.g., FlowGRPO) and reconstruction-based methods (e.g., RecA) align models using preference data or cross-modal objectives [17, 21]. However, they typically rely on text or image data or external reward models. Third, recent self-improvement frameworks like SRUM [16], UniRL [15], and UniGame [18] explore intrinsic scoring or adversarial loops to refine capabilities. Pan et al. [20] study token-level intrinsic generation rewards; SUDER [19] uses dual rewards with joint and alternating updates. HELIX combines prior correction, adaptive targets, cascade filtering, and truth-guided regeneration across cycles.

3 Method

3.1 Overview: HELIX as a Double-Helix Cycle

HELIX (**H**olistic **E**vo**L**ution via **I**ntrinsic e**X**change) casts a unified multimodal model as two intertwined strands: a generation strand and an understanding strand. The two strands are coupled by intrinsic rewards and a Data Bridge that enable bidirectional signal exchange in a cyclic post-training loop. Each cycle performs an alternating optimization of two stages. In Stage 1, the generation strand is optimized utilizing intrinsic rewards produced by the understanding strand. In Stage 2, the understanding strand is updated via reconstruction and intrinsic rewards, thereby providing a more stable rewards for Stage 1 of the subsequent cycle.

3.2 Preliminary

3.2.1 Unified Multimodal Model

A unified multimodal model (UMM) with parameters θ induces two conditional policies: a generation policy $\pi_\theta(y \mid c)$ and a understanding policy $\pi_\theta(x \mid y)$. Here c denotes conditions, y an image,

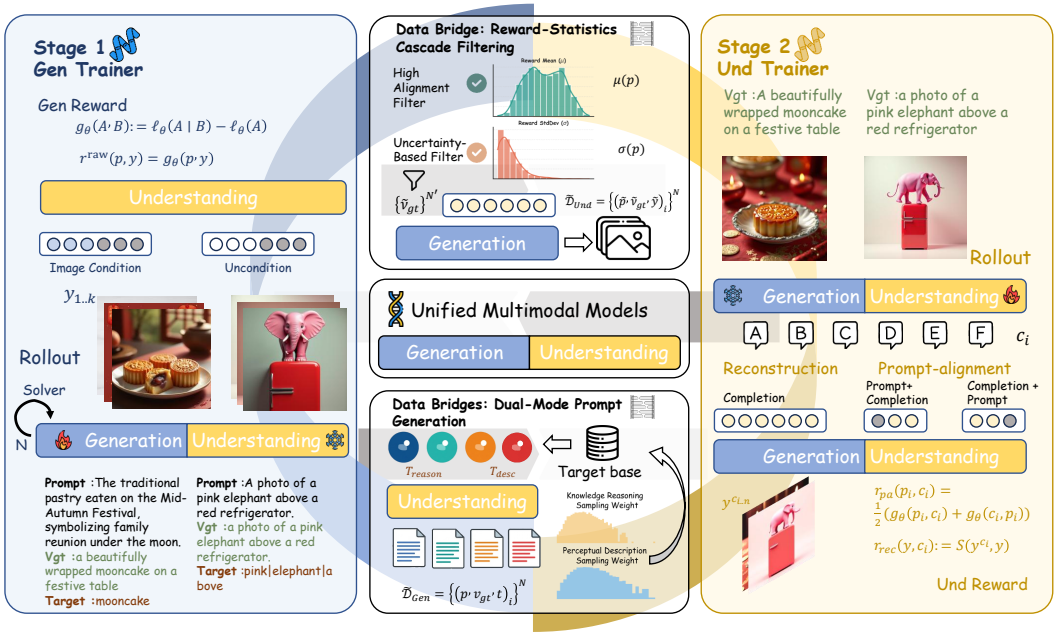


Figure 2: **HELIX framework overview.** We cast generation and understanding as two intertwined strands of a double helix, connected by a target-driven Data Bridge. The cycle starts from a text-only Target Base. In **Stage 1**, the generative policy is optimized with the intrinsic self-evaluated likelihood gain reward g_θ produced by the understanding strand. The resulting trajectories are distilled by a cascade filter to retain reliable and informative targets/prompts, and the Data Bridge regenerates target-consistent samples for the next stage. In **Stage 2**, the understanding policy is refined on this bridged data with two signals: a Reconstruction reward r_{rec} for visual faithfulness and a Prompt-Alignment reward r^{pa} for semantic consistency, closing the helix loop for continual self-evolution.

and x a text output. We refer to them as the generation and understanding strands, and optimize the two strands in alternating stages.

ODE-based Image Generation. Images are sampled from the generation policy conditioned on a control signal $c \in \mathcal{C}$. An ODE-based sampler is adopted: initialize $z(0) \sim \mathcal{N}(0, I)$ and integrate

$$\frac{dz(t)}{dt} = u_\theta(z(t), t, c), \quad t \in [0, 1], \tag{1}$$

where c is instantiated as textual control, visual control, or their combination.

Visual Understanding. Visual understanding is cast as conditional language modeling. For a text sequence $x = (x_1, \ldots, x_{|x|})$, the conditional distribution factorizes as

$$\pi_\theta(x \mid y) = \prod_{i=1}^{|x|} \pi_\theta(x_i \mid x_{<i}, y). \tag{2}$$

3.2.2 Group Relative Policy Optimization

We adopt Group Relative Policy Optimization (GRPO) for Stage 2. Given a context c and a behavior policy $\pi_{\theta_{\text{old}}}$, GRPO samples a group of G responses $\{o_i\}_{i=1}^G$ and assigns each a scalar reward r_i. It then applies group-wise advantage standardization and a clipped KL-regularized objective:

$$\hat{A}_i = \frac{r_i - \text{mean}(\{r_j\}_{j=1}^G)}{\text{std}(\{r_j\}_{j=1}^G) + \epsilon}; \quad \mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}\Big[\min\big(\rho(\theta)\hat{A}, \text{clip}_\varepsilon(\rho(\theta))\hat{A}\big)\Big] - \beta_{\text{KL}}\mathbb{E}[\text{KL}(\pi_\theta \| \pi_{\text{ref}})]\,. \tag{3}$$

Here $\rho(\theta)$ is the per-token importance ratio between π_θ and $\pi_{\theta_{\text{old}}}$, and $\text{clip}_\varepsilon(\cdot)$ clips to $[1-\varepsilon, 1+\varepsilon]$. In Stage 2, $c = (y, s_m)$ is an image-conditioned instruction with input image y and system prompt s_m; rewards are computed per caption and standardized within each group.

3.2.3 Diffusion Negative-aware Finetuning

For Stage 1, we employ Diffusion Negative-aware Finetuning (DiffusionNFT) to perform reward-weighted policy optimization on the forward diffusion noising process [32]. Given a raw reward $r^{\text{raw}}(x_0, c)$, we map it to an optimality weight $r(x_0, c) \in [0, 1]$ by per-condition normalization and clipping, and optimize

$$\mathcal{L}(\theta) = \mathbb{E}\big[r\,\mathcal{L}^{+}(x_t, t, c) + (1-r)\,\mathcal{L}^{-}(x_t, t, c)\big]\,, \quad \mathcal{L}^{\pm}(x_t, t, c) = \big\|v_\theta^{\pm}(x_t, t, c) - v\big\|_2^2. \tag{4}$$

The expectation is over c, $x_0 \sim \pi_{\text{old}}(\cdot \mid c)$, diffusion time t, and noise used to construct x_t. Here v denotes the target velocity field and π_{old} is a fixed reference policy. The implicit positive and negative policies are $v_\theta^{+} := (1-\eta)v^{\text{old}} + \eta v_\theta$ and $v_\theta^{-} := (1+\eta)v^{\text{old}} - \eta v_\theta$, with mixing coefficient η. In Stage 1, c is the generation prompt p, and $r(x_0, c)$ is obtained from our intrinsic reward in Section 3.4.

3.3 Data Bridge

To provide high-quality and stable data for self-evolving cycles, we design a Data Bridge. The Data Bridge couples the two HELIX strands across stages. It synthesizes target-driven prompts for Stage 1 using Algorithm 1 and converts Stage 1 reward statistics into target-consistent training data for Stage 2.

Dual-Mode Prompt Generation. We organize target-driven synthesis with a unified triplet $\mathcal{T} = (p, v_{gt}, t)$, where t is a target concept, p is the prompt used to condition Stage 1 rollouts, and v_{gt} is a visual-truth description that explicitly contains t for ideal scene. We synthesize (p, v_{gt}) with two complementary modes mixed by ratio λ. The Perceptual Descriptive mode aligns p with v_{gt} by injecting explicit visual constraints like cardinality, attributes, and spatial relations. The Knowledge Reasoning mode decouples abstract semantics from concrete entities by using implicit causal or metaphorical cues as p, while keeping v_{gt} as an entity-grounded description containing t.

Adaptive Target Reweighting. Instead of uniform sampling, we adaptively reweight targets t based on Stage 1 rewards. After each cycle, we group prompts by their associated target and summarize the Stage 1 reward standard deviation $\sigma(p)$ within each group. Targets with larger dispersion are treated as learnable yet unstable, and receive more synthesis budget in the next cycle. We smooth $u_c(t)$ with an EMA memory, initialize unseen targets with a high-quantile value q, normalize it, and convert it into sampling weights via a temperatured softmax:

$$w_c(t) = \frac{\exp(\tilde{u}_c(t)/T)}{\sum_{t'} \exp(\tilde{u}_c(t')/T)}. \tag{5}$$

It allocates more budget to high-variance targets and forms a self-paced curriculum across cycles.

Reward-Statistics Cascade Filtering. Instead of directly reusing noisy rollout data, we transfer data across stages by selecting prompts using Stage-1 reward statistics. Specifically, (i) we apply a cascade filter that first keeps high-quality prompts by the reward mean $\mu(p)$ and then prioritizes challenging ones by the reward standard deviation $\sigma(p)$; (ii) using the selected prompt set $\mathcal{P}^*$, we construct $\tilde{\mathcal{D}}_{Und}$ by regenerating images conditioned on the model-synthesized visual truth v_{gt} for target-consistent Stage 2 evidence. Appendix L traces this data flow, and Algorithm 2 gives the filtering and regeneration procedure.

3.4 Stage 1: Generation Post-training

Stage 1 post-trains the generation branch using DiffusionNFT on prompts sampled from $\tilde{\mathcal{D}}_{Gen}$. We formulate each training instance as a unified triplet $\mathcal{T} = (p, v_{gt}, t)$. Specifically, t is a target entity sampled from predefined target sets. v_{gt} represents the visual truth, defined as a detailed description that explicitly depicts a scene containing the target t. Finally, p serves as the generation prompt for image generation. In this stage, the generation policy is strictly conditioned on the prompt p.

Sampling and policy optimization. Given a prompt p, we synthesize K rollout images by integrating the conditional ODE in Equation (1),

$$\hat{y}_k \sim \pi_\theta(\cdot \mid p), \qquad k = 1, \ldots, K. \tag{6}$$

Each rollout yields a scalar raw reward $r^{\text{raw}}(p, \hat{y}_k)$. We aggregate these rewards into prompt-level statistics, including the mean reward $\mu(p)$ and reward standard deviation $\sigma(p)$, which are used for per-prompt reward normalization in Stage 1 and as the selection signal for cascade filtering in Algorithm 2. Stage 1 optimization follows the DiffusionNFT objective in Section 3.2, where we substitute the collected rollouts and their rewards into Equation (4).

Self-evaluated likelihood gain Reward. Crucially, Stage 1 operates without any external reward model. Instead, the reward is computed intrinsically by leveraging the model's own understanding capability to calculate the likelihood $\pi_\theta(X \mid \hat{y})$ of a reference text X given the generated image $\hat{y}$.

Formally, let $X = (x_1, \ldots, x_{|X|})$ denote a generic reference text sequence. We define the token-normalized conditional log-likelihood as:

$$\ell_\theta(X \mid \hat{y}) := \frac{1}{|X|} \sum_{i=1}^{|X|} \log \pi_\theta(x_i \mid x_{<i}, \hat{y}). \tag{7}$$

To mitigate text-frequency and prompt-prior biases, we derive a prior-corrected likelihood gain by subtracting an unconditional baseline:

$$g_\theta(\hat{y}, X) := \ell_\theta(X \mid \hat{y}) - \ell_\theta(X), \tag{8}$$

where $\ell_\theta(X)$ represents the unconditional log-likelihood computed without visual inputs. Intuitively, $g_\theta(\hat{y}, X)$ quantifies the information gain contributed specifically by the generated image $\hat{y}$ in explaining the text X.

Target View and Expansion View. Each triplet $\mathcal{T} = (p, v_{\text{gt}}, t)$ facilitates a dual-perspective evaluation of the generated image $\hat{y}$. The Target View, denoted as $X_{\text{tgt}} := v_{\text{gt}}$, explicitly encapsulates the target concept t, thereby enforcing strict semantic faithfulness to the core subject.

Complementing this, the Expansion View employs an auxiliary description $X_{\text{exp}} := v_{\text{exp}}$, which is generated by the UMM to imagine a comprehensive scene specification solely from the text prompt p, independent of the generated image. This expansion serves a dual purpose: it enriches the supervision with fine-grained visual cues to promote detailed alignment, and effectively functions as an implicit chain-of-thought distillation mechanism.

We apply the likelihood-gain function to both views and form the Stage 1 raw reward as

$$g_{\text{tgt}}(\hat{y}) := g_\theta(\hat{y}, X_{\text{tgt}}), \quad g_{\text{exp}}(\hat{y}) := g_\theta(\hat{y}, X_{\text{exp}}), \quad r^{\text{raw}}(p, \hat{y}) = \lambda_{\text{tgt}} g_{\text{tgt}}(\hat{y}) + \lambda_{\text{exp}} g_{\text{exp}}(\hat{y}). \tag{9}$$

Here λ_{tgt} and λ_{exp} balance the two components.

3.5 Stage 2: Understanding Post-training

In Stage 2, the visual-understanding branch is post-trained on the regenerated dataset $\tilde{\mathcal{D}}_{\text{Und}}$ by updating the conditional text policy $\pi_\theta(x \mid y)$. Each training instance retains (p, y, v_{gt}, t) from Section 3.3, where y is regenerated under the visual-truth constraint associated with the target t, and p is the corresponding generation prompt from Stage 1. Stage 2 employs a dual-reward framework: a caption-to-image reconstruction reward that scores a sampled description by its ability to regenerate the input image through the UMM's generative branch, and a prompt-alignment reward that encourages the understanding policy to recover the original generation prompt associated with the image.

Sampling and policy optimization. Given an input image y_{gt}, Stage 2 samples a group of G candidate captions from the current understanding policy under a mode-dependent system instruction s_m:

$$x^{(g)} \sim \pi_\theta(\cdot \mid y_{gt},\, s_m), \qquad g = 1, \ldots, G, \tag{10}$$

where s_m matches the caption style of each data mode. We use a constraint-focused instruction for Perceptual Descriptive samples and a reasoning-oriented instruction for Knowledge Reasoning samples. Stage 2 is optimized with GRPO as described in Section 3.2.

Caption-to-Image Reconstruction Reward. To enforce visual grounding in Stage 2, we reward captions that are not only well-formed but also sufficient to reconstruct the input image. For each sampled caption $x^{(g)}$, we use it as the control condition for the UMM generation strand to obtain

a reconstructed image $\tilde{y}^{(g)}$. We then define the reconstruction reward by a hybrid visual similarity function $S(\cdot,\cdot)$ between $\tilde{y}^{(g)}$ and the original image y. Maximizing this reward encourages the understanding strand to generate captions that are faithful to the visual content.

$$\tilde{y}^{(g)} \sim \pi_{\theta_{\text{old}}}(\cdot \mid x^{(g)}),\ r_{\text{rec}}\Big(y, x^{(g)}\Big) := S\Big(\tilde{y}^{(g)},\ y\Big)\,. \tag{11}$$

Prompt-Alignment Reward. To align the sampled caption $x^{(g)}$ with the ground-truth prompt p, we introduce a text-side semantic signal based on the likelihood gain formulation in Equation (8). This objective serves a dual role: it enforces the recovery of fine-grained attributes for perceptual samples, while distilling implicit reasoning cues for reasoning samples. The intrinsic reward is same as Stage 1, measuring the semantic similarity between the sampled caption and the rollout prompt, and encouraging captions that remain tightly grounded in the target concept. To mitigate reward hacking effects, Stage 2 uses a bidirectional form by averaging gains in both directions:

$$r_{\text{pa}}\Big(x^{(g)}, p\Big) := \tfrac{1}{2}\Big(g_\theta(p, x^{(g)}) + g_\theta(x^{(g)}, p)\Big). \tag{12}$$

The Stage 2 raw reward combines the caption-to-image reconstruction score and the prompt-alignment score:

$$r^{\text{raw}}\Big(y, p, x^{(g)}\Big) := \lambda_{\text{pa}}\, r_{\text{pa}}\Big(x^{(g)}, p\Big) + \lambda_{\text{rec}}\, r_{\text{rec}}\Big(y, x^{(g)}\Big)\,, \tag{13}$$

where λ_{pa} and λ_{rec} balance the two reward components.

4 Experiments

4.1 Experimental Setup

Training Data. Our annotation-free framework uses word-level semantic targets without paired training annotations. We construct two complementary target pools: (i) a perceptual descriptive set using COCO category names [33] for Perceptual Descriptive mode, and (ii) a Knowledge Reasoning mode set using class vocabularies from WordNet [34], ImageNet-1K [35], and QuickDraw [36].

Model Architecture. We employ Bagel [8] as the baseline, initialized with official pre-trained weights. All experiments are conducted strictly as a post-training phase. Notably, while we constrain the training resolution to 512×512 for efficiency, we explicitly conduct evaluations at the model's native 1024×1024 resolution to assess its generalization capabilities.

Evaluation Benchmarks. We adopt default settings for all tasks, with a primary emphasis on text-to-image generation. To this end, we employ T2I-CompBench [5] and GenEval [7] for compositional binding, alongside DPG-Bench [37] and WISE [4] for advanced prompt following and reasoning. Complementarily, we evaluate visual understanding capabilities using MME [38], MMBench [39], MMVet [40], MMMU [41], and MMVP [42].

4.2 Main Results

Generation Performance on Text-to-Image Benchmarks. We evaluate text-to-image generation on compositional prompt following and knowledge reasoning using T2I-CompBench, GenEval, DPG-Bench, and WISE. Our gains are achieved in an annotation-free post-training setting, where supervision is synthesized internally from target concepts without external annotated corpora. As shown in Table 1, HELIX consistently improves over the Bagel base model and prior post-training baselines. On T2I-CompBench, the overall score increases from 82.2 to 87.8, a 6.8% relative improvement, with clear gains on spatial relationships and text rendering, indicating stronger layout control and better text fidelity under multi-constraint prompts. On GenEval, we reach 85.9, with notable improvements on counting and position, suggesting more reliable handling of cardinality and spatial constraints. We also improve dense prompt following on DPG-Bench, where the DPG score increases by 1.54 points, reflecting better consistency under long and structured prompts. For knowledge reasoning, we observe consistent gains on WISE across resolutions, improving WiScore from 0.52 to 0.53 at 512^2 and from 0.52 to 0.54 at 1024^2. Overall, these results indicate that HELIX strengthens both perceptual compositionality and knowledge-driven generation, and the gains transfer reliably from the training resolution to the native resolution.

Table 1: **Full Quantitative Evaluation.** Comparison with state-of-the-art models. Baselines include Specialized T2I Models (SDXL [43], SD-3 [23], FLUX.1 [24]) and Unified Multimodal Models (Janus-Pro [11], Show-O2 [44], OmniGen2 [45], Harmon [9]). We specifically compare with Bagel-based optimization methods: **RecA** [17] and **SRUM** [16], which are both trained and evaluated at **1024px** resolution. *Note:* Our method is trained exclusively at **512px**, yet we report results evaluated at both 512px (in-distribution) and 1024px (zero-shot resolution extrapolation). **Bold** indicates the best result, and underlined indicates the second best.

Model	T2I-CompBench									GenEval							DPG	WISE
	3D Sp.	Color	Cplx.	Non-Sp.	Num.	Shape	Spat.	Text.	**Avg.**	Sgl.	Two	Cnt.	Col.	Pos.	Attr.	**Avg.**	Score	WiScore
Specialized T2I Models																		
SDXL	72.3	77.8	75.0	85.3	57.1	72.2	77.1	78.4	74.4	98.0	74.0	39.0	85.0	15.0	23.0	55.7	74.65	0.43
SD-3-Medium	77.8	91.6	84.7	86.1	72.8	83.7	88.2	89.0	84.3	99.0	94.0	72.0	89.0	33.0	60.0	74.0	84.08	0.45
FLUX.1-schnell	79.4	84.5	82.0	85.6	72.8	82.2	85.5	86.4	82.3	**100.0**	87.0	76.0	84.0	22.0	49.0	70.0	-	0.40
FLUX.1-dev	76.4	90.6	83.5	87.5	75.3	80.2	84.2	87.1	83.1	98.0	81.0	74.0	79.0	22.0	45.0	66.5	84.00	0.50
Unified Multimodal Models																		
Janus-Pro	76.2	84.3	80.3	80.5	56.4	65.1	79.7	69.7	74.0	99.0	89.0	59.0	**90.0**	**79.0**	66.0	80.3	84.19	0.35
Show-O2	**88.6**	87.7	87.9	85.9	69.7	74.0	86.6	82.2	82.8	97.2	80.3	61.9	78.2	27.3	52.3	66.2	82.21	0.40
OmniGen2	82.2	92.2	86.9	88.5	72.0	84.0	**90.1**	**90.9**	85.8	**100.0**	**95.0**	64.0	88.0	55.0	76.0	80.0	83.57	-
Harmon	-	-	-	-	-	-	-	-	-	99.4	87.3	68.7	86.4	44.9	51.1	72.9	80.93	0.41
Evaluation of Bagel-Based Optimization																		
1024-base (Default 1024px) and Ours (Test@1024) – compared against all models																		
Bagel (Base@1024)	78.0	89.3	83.3	85.0	70.4	81.9	81.5	87.9	82.2	99.1	93.0	79.9	86.0	51.3	63.4	78.8	84.03	0.52
+ SRUM	83.1	92.9	**88.7**	88.5	78.5	84.2	86.9	89.6	86.6	98.0	94.0	83.0	**90.0**	64.0	**83.0**	85.3	-	-
+ RecA	-	-	-	-	-	-	-	-	-	99.3	93.9	80.3	87.6	60.8	72.6	82.4	85.29	0.52
Ours (Test@1024)	**83.4**	**93.6**	88.5	**88.6**	**81.3**	**86.7**	89.9	90.6	**87.8**	99.4	94.4	**89.4**	**90.7**	**67.1**	74.3	**85.9**	**85.57**	**0.54**
Δ vs Base	+5.4	+4.3	+5.2	+3.6	+10.9	+4.8	+8.4	+2.7	+5.6	+0.3	+1.4	+9.5	+4.7	+15.8	+10.9	+7.1	+1.54	+0.02
512-base (Default 512px) vs. Ours (Test@512)																		
Bagel (Base@512)	75.9	91.3	85.9	87.0	78.7	82.6	86.6	87.3	84.4	99.1	90.9	73.1	86.2	45.8	57.3	75.4	84.31	0.52
Ours (Test@512)	**81.9**	**93.4**	**87.8**	**87.9**	**80.5**	**86.2**	**89.4**	**90.9**	**87.3**	**99.7**	**97.3**	**82.4**	**91.6**	**66.4**	**71.6**	**84.8**	**85.73**	**0.53**
Δ vs Base	+6.0	+2.1	+1.9	+0.9	+1.8	+3.6	+2.8	+3.6	+2.9	+0.6	+6.4	+9.3	+5.4	+20.6	+14.3	+9.4	+1.42	+0.01

Table 2: **Visual Understanding Performance.** HELIX exhibits gains and regressions across understanding tasks. MME-P and MME-C denote the perception and cognition splits of the MME benchmark. † indicates CoT inference.

Method	MMBench	MMMU	MME-P	MME-C	MM-Vet	MMVP
Bagel (Base)	85.0	53.0 / 55.3†	**1687**	**701**	**67.2**	69.3
SRUM	84.8	55.2†	1673	677	67.0	**70.0**
Ours	**86.5**	53.7	1681	688	65.9	**70.0**

Table 3: **Generation-side ablations.** Impact of components on generation performance. Bold and underline denote best and second-best results.

Configuration	T2I-Comp	GenEval	DPG	WISE
Base Model	84.4	75.4	84.31	0.5659
Stage 1 Only (One-shot)	**87.43**	83.98	84.91	0.5582
Full Cycle (Ours)	87.25	**84.81**	85.73	0.6180
w/o Adaptive Target Reweighting	86.44	80.43	85.09	**0.6209**
w/o Prompt-Align. Reward r^{pa}	**87.43**	82.67	**85.80**	0.5950
w/o Recon. Reward r_{rec}	85.79	78.29	83.58	0.5813
w/o Intrinsic Reward	76.38	30.90	67.18	0.2287

Impact on Visual Understanding. Beyond improving generation, HELIX largely preserves and in several cases strengthens the model's visual understanding. We observe a clear gain on MM-Bench (86.5), suggesting improved general multimodal competence under standard evaluation. On MMMU, the score increases to 53.7 under standard inference, indicating that the understanding strand benefits on knowledge- and reasoning-heavy questions without relying on test-time scaling. We also improve MMVP to 70.0. MME-P, MME-C, and MM-Vet decline relative to Base. We characterize the outcome as understanding preservation with selective grounding gains. Direct caption evaluations assess Stage 2 in Appendix I.

4.3 Ablation Study and Training Dynamics

We conduct ablation studies to validate the contribution of each component in HELIX, with results in Table 3. For WISE, we use Qwen3-VL 30B A3B [46] as the judge, while the main results are evaluated with GPT-4o.

Effectiveness of the two-stage cycle. One-shot Stage 1 post-training yields a large lift over the base model, improving T2I-Comp from 84.4 to 87.43 and GenEval from 75.4 to 83.98. However, it optimizes generation with a fixed understanding strand, limiting the intrinsic evaluator. With the full cycle, the model maintains strong T2I-Comp at 87.25 while improving semantics-sensitive metrics, raising GenEval to 84.81, DPG to 85.73, and WISE from 0.5659 to 0.6180. Table 4 further shows that the full cycle preserves and modestly improves visual understanding, increasing MMBench from 85.00 to 86.50 and MMVP from 69.33 to 70.00 while improving MMMU from 53.0 to 53.7. The dynamics in Figure 3 show a sustained increase in the S1 evaluation score under cyclic training,

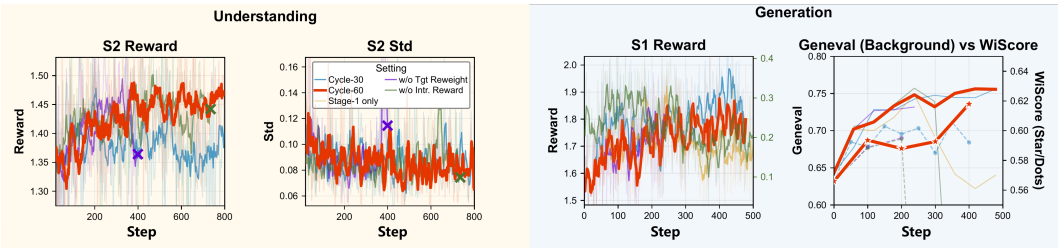


Figure 3: **Training dynamics.** Reward trajectories and evaluation trends across HELIX training. The panels show Stage 2 reward mean/std, Stage 1 intrinsic reward, and GenEval/WiScore scores.

whereas Stage 1 Only plateaus earlier and later regresses. The multi-cycle control in Section 4.4 tests continued generation-only training.

Adaptive target reweighting improves stability and prevents mid-training degradation. Table 3 shows that removing Adaptive Target Reweighting leads to clear drops on compositional benchmarks, with GenEval decreasing from 84.81 to 80.43 and T2I-Comp from 87.25 to 86.44. The understanding-side results show a similar pattern: removing reweighting reduces MMVP from 70.00 to 69.00 and MM-Vet from 65.9 to 65.0, suggesting weaker visual grounding and discriminative stability. This is consistent with Figure 3: without reweighting, the S1 evaluation score tends to plateau and shows a mid-to-late decline, while the S2 reward becomes less stable. We attribute this to the self-evolving setting, where uniform sampling over-emphasizes already-stable targets, produces less informative reward variation, and causes the intrinsic exchange loop to drift. In contrast, reweighting keeps training focused on unstable concepts, sustaining informative rewards and yielding improvements across cycles.

Table 4: **Understanding-side ablations.** Visual-understanding results.

Config.	MMB	MMMU	MMVP	Vet	MME-P	MME-C
Base	85.00	53.00	69.33	**67.2**	**1687.0**	701.0
w/o Tgt Rew.	85.40	53.60	69.00	65.0	<u>1685.7</u>	698.2
w/o PA r^{pa}	85.40	53.30	<u>69.67</u>	64.8	1684.6	<u>701.8</u>
w/o Rec. r_{rec}	<u>85.57</u>	53.60	68.67	64.8	1679.3	**702.5**
w/o Intr.	85.05	53.40	69.33	65.2	1681.8	681.1
Full	**86.50**	**53.70**	**70.00**	<u>65.9</u>	1681.0	688.0

Stage 2 rewards: prompt alignment vs. reconstruction grounding. Ablating the Prompt-Alignment reward mainly affects semantics-heavy generation, reducing GenEval from 84.81 to 82.67 and WISE from 0.6180 to 0.5950, while leaving T2I-Comp comparable. On understanding benchmarks, this variant keeps MMVP relatively high at 69.67 but lowers MMMU and MM-Vet, indicating that prompt alignment mainly sharpens semantic preference and knowledge-related judgments. In contrast, removing the Reconstruction reward leads to broader generation degradation, with GenEval dropping to 78.29 and DPG to 83.58. It also weakens MMVP to 68.67, suggesting reduced fine-grained visual discrimination despite preserving some general understanding scores. These results indicate that reconstruction-based training provides the key visual grounding signal for the understanding strand, while prompt alignment complements it by sharpening semantic preferences.

Intrinsic rewards stabilize self-evolution. Figure 3 shows that full-cycle training yields steadily increasing rewards and improved stability, while Stage 1 Only plateaus earlier and exhibits a mid-to-late reward drop, consistent with weaker semantic generalization in Table 3. Replacing likelihood-gain rewards with prompt-based explicit scoring further destabilizes training and can cause collapse, with discrete-score behavior further examined in Appendix J. The understanding-side ablation also shows that removing intrinsic reward fails to match the full model on MMBench and MMVP, reinforcing that dense image-conditioned feedback is needed not only for generation but also for stable cross-strand improvement. Each default cycle uses 30 Stage 1 epochs and 60 generation optimizer updates; epochs and updates are distinct units.

4.4 Multi-cycle Generation-Budget Control

We train an eight-cycle Stage-1-only control with understanding frozen. Both paths use 240 Stage-1 epochs, 69,120 candidates, 480 generation updates, and global rollout batch 48. Full HELIX outperforms the control across the four external evaluations in Table 5, supporting the contribution of cyclic generation–understanding exchange. Training costs are reported in Appendix G.

Table 5: Eight-cycle control at the same generation budget. WISE-V denotes WISE_Verified no-CoT; Table 3 uses a separate WISE evaluation protocol.

Method	T2I-Comp	GenEval	DPG	WISE-V
Stage-1-only	78.64	77.35	85.46	0.334
Full HELIX	87.25	84.81	85.73	0.384
Difference	+8.61	+7.46	+0.27	+0.050

Table 6: Data Bridge decomposition under matched ablation settings. GenEval percentages; the first bridge intervention affects cycle 2, so cycle 1 is identical.

Bridge variant	Cycle 1	Cycle 2	Cycle 3
Full bridge	69.05	71.77	73.32
Direct rollout reuse	69.05	71.95	72.43
Mean-only filter	69.05	70.31	69.78

4.5 Data Bridge Decomposition

Data Bridge decomposition. We compare the full bridge with direct Stage-1 rollout reuse and mean-only filtering over three cycles. All variants use four GPUs, 30 Stage-1 epochs per cycle, global batch 24, 12,960 generated candidates, and LoRA rank/alpha 32/64, with Stage 2 and adaptive target sampling held fixed. Table 6 shows that, by cycle 3, the full bridge exceeds rollout reuse by 0.89 GenEval points and mean-only filtering by 3.54 points. These comparisons support truth-guided regeneration and the additional standard-deviation gate. Detailed protocols appear in Appendix H.

A blinded automatic caption comparison on 400 fixed inputs gives Cycle 8 a 57.5% preference over the initial checkpoint; full caption evaluations and independent reward audits appear in Appendices I and J.

5 Conclusion

We present HELIX, a cyclic post-training framework for UMMs that couples image generation and image-conditioned understanding into an intrinsic exchange loop without external reward models or paired training annotations. HELIX introduces a Data Bridge that synthesizes target-driven prompts and transfers Stage 1 reward statistics into target-consistent data for Stage 2 via cascade filtering and truth-guided regeneration. Across text-to-image benchmarks, HELIX improves compositional generation on binding- and structure-sensitive benchmarks and strengthens knowledge-based generation, with understanding preservation and selective grounding gains.

Acknowledgments and Disclosure of Funding

This work received no specific funding. The authors declare no competing interests.

References

[1] Tianhong Li, Yonglong Tian, He Li, et al. Autoregressive Image Generation without Vector Quantization. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 56424–56445. Curran Associates, Inc., 2024. doi: 10.52202/079017-1797.

[2] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, et al. Zero-shot text-to-image generation. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8821–8831. PMLR, 18–24 Jul 2021. URL `https://proceedings.mlr.press/v139/ramesh21a.html`.

[3] Robin Rombach, Andreas Blattmann, Dominik Lorenz, et al. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, June 2022.

[4] Yuwei Niu, Munan Ning, Mengren Zheng, et al. Wise: A world knowledge-informed semantic evaluation for text-to-image generation, 2025.

[5] Kaiyi Huang, Kaiyue Sun, Enze Xie, et al. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, pages 78723–78747, 2023.

[6] Kaiyi Huang, Chengqi Duan, Kaiyue Sun, et al. T2i-compbench++: An enhanced and comprehensive benchmark for compositional text-to-image generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(5):3563–3579, 2025. doi: 10.1109/TPAMI.2025.3531907.

[7] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 52132–52152. Curran Associates, Inc., 2023.

[8] Chaorui Deng, Deyao Zhu, Kunchang Li, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

[9] Size Wu, Wenwei Zhang, Lumin Xu, et al. Harmonizing visual representations for unified multimodal understanding and generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 17739–17750, October 2025.

[10] Yiyang Ma, Xingchao Liu, Xiaokang Chen, et al. Janusflow: Harmonizing autoregression and rectified flow for unified multimodal understanding and generation, 2024.

[11] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, et al. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

[12] Jiuhai Chen, Zhiyang Xu, Xichen Pan, et al. Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset. *arXiv preprint arXiv:2505.09568*, 2025.

[13] Teng Li, Quanfeng Lu, Lirui Zhao, et al. Unifork: Exploring modality alignment for unified multimodal understanding and generation. *arXiv preprint arXiv:2506.17202*, 2025.

[14] Yan Yang, Haochen Tian, Yang Shi, et al. A survey of unified multimodal understanding and generation: Advances and challenges. *Authorea Preprints*, 2025.

[15] Weijia Mao, Zhenheng Yang, and Mike Zheng Shou. Unirl: Self-improving unified multimodal models via supervised and reinforcement learning, 2025.

[16] Weiyang Jin, Yuwei Niu, Jiaqi Liao, et al. Srum: Fine-grained self-rewarding for unified multimodal models, 2025.

[17] Ji Xie, Trevor Darrell, Luke Zettlemoyer, et al. Reconstruction alignment improves unified multimodal models. *arXiv preprint arXiv:2509.07295*, 2025.

[18] Zhaolong Su, Wang Lu, Hao Chen, et al. Unigame: Turning a unified multimodal model into its own adversary, 2025.

[19] Jixiang Hong, Yiran Zhang, Guanzhong Wang, Yi Liu, Ji-Rong Wen, and Rui Yan. Suder: Self-improving unified large multimodal models for understanding and generation with dual self-rewards, 2025. URL `https://arxiv.org/abs/2506.07963`.

[20] Jiadong Pan, Liang Li, Yuxin Peng, Yu-Ming Tang, Shuohuan Wang, Yu Sun, Hua Wu, Qingming Huang, and Haifeng Wang. Learning to generate via understanding: Understanding-driven intrinsic rewarding for unified multimodal models, 2026. URL `https://arxiv.org/abs/2603.06043`.

[21] Jie Liu, Gongye Liu, Jiajun Liang, et al. Flow-grpo: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470*, 2025.

[22] Jing Wang, Jiajun Liang, Jie Liu, et al. Grpo-guard: Mitigating implicit over-optimization in flow matching via regulated clipping, 2025. URL `https://arxiv.org/abs/2510.22319`.

[23] Patrick Esser, Sumith Kulal, Andreas Blattmann, et al. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv:2403.03206*, 2024. doi: 10.48550/arXiv.2403.03206. URL https://arxiv.org/abs/2403.03206.

[24] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

[25] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024. doi: 10.48550/arXiv.2405.09818. URL https://github.com/facebookresearch/chameleon.

[26] Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024.

[27] Chengyue Wu, Xiaokang Chen, Zhiyu Wu, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. *arXiv preprint arXiv:2410.13848*, 2024.

[28] Jinheng Xie, Weijia Mao, Zechen Bai, et al. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

[29] Ziyu Guo, Renrui Zhang, Chengzhuo Tong, et al. Can we generate images with cot? let's verify and reinforce image generation step by step. *arXiv preprint arXiv:2501.13926*, 2025.

[30] Rongyao Fang, Chengqi Duan, Kun Wang, et al. Got: Unleashing reasoning capability of multimodal large language model for visual generation and editing. *arXiv preprint arXiv:2503.10639*, 2025.

[31] Dongzhi Jiang, Renrui Zhang, Haodong Li, et al. Draco: Draft as cot for text-to-image preview and rare concept generation. *arXiv preprint arXiv:2512.05112*, 2025.

[32] Kaiwen Zheng, Huayu Chen, Haotian Ye, et al. Diffusionnft: Online diffusion reinforcement with forward process. *arXiv preprint arXiv:2509.16117*, 2025.

[33] Tsung Yi Lin, Michael Maire, Serge Belongie, et al. Microsoft coco: Common objects in context. *Springer International Publishing*, 2014.

[34] George A. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38 (11):39–41, 1995. doi: 10.1145/219717.219748. URL https://doi.org/10.1145/219717.219748.

[35] Jia Deng, Wei Dong, Richard Socher, et al. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20–25 June 2009, Miami, Florida, USA*, pages 248–255. IEEE Computer Society, 2009. doi: 10.1109/CVPR.2009.5206848. URL https://doi.org/10.1109/CVPR.2009.5206848.

[36] David Ha and Douglas Eck. A neural representation of sketch drawings. *arXiv preprint arXiv:1704.03477*, 2017. doi: 10.48550/arXiv.1704.03477. URL https://doi.org/10.48550/arXiv.1704.03477.

[37] Xiwei Hu, Rui Wang, Yixiao Fang, et al. ELLA: Equip Diffusion Models with LLM for Enhanced Semantic Alignment. *arXiv preprint arXiv:2403.05135*, 2024. doi: 10.48550/arXiv.2403.05135. URL https://arxiv.org/abs/2403.05135.

[38] Chaoyou Fu, Peixian Chen, Yunhang Shen, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. doi: 10.48550/arXiv.2306.13394. URL https://arxiv.org/abs/2306.13394.

[39] Yuan Liu, Haodong Duan, Yuanhan Zhang, et al. Mmbench: Is your multi-modal model an all-around player? In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part VI*, volume 15064 of *Lecture Notes in Computer Science*, pages 216–233. Springer, 2024. doi: 10.1007/978-3-031-72658-3_13. URL https://doi.org/10.1007/978-3-031-72658-3_13.

[40] Weihao Yu, Zhengyuan Yang, Linjie Li, et al. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023. doi: 10.48550/arXiv.2308.02490. URL https://arxiv.org/abs/2308.02490.

[41] Xiang Yue, Yuansheng Ni, Kai Zhang, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9556–9567, June 2024.

[42] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, et al. Eyes wide shut? exploring the visual shortcomings of multimodal llms, 2024.

[43] Dustin Podell, Zion English, Kyle Lacey, et al. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=di52zR8xgf.

[44] Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *arXiv preprint arXiv:2506.15564*, 2025. URL https://arxiv.org/abs/2506.15564.

[45] Chenyuan Wu, Pengfei Zheng, Ruiran Yan, et al. Omnigen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025.

[46] Shuai Bai, Yuxuan Cai, Ruizhe Chen, et al. Qwen3-VL technical report. *arXiv preprint*, November 2025.

[47] Hongyuan Dong, Jiawen Li, Bohong Wu, Jiacong Wang, Yuan Zhang, and Haoyuan Guo. Benchmarking and improving detail image caption, 2024. URL https://arxiv.org/abs/2405.19092.

[48] Roopal Garg, Andrea Burns, Burcu Karagol Ayan, Yonatan Bitton, Ceslee Montgomery, Yasumasa Onoe, Andrew Bunner, Ranjay Krishna, Jason Baldridge, and Radu Soricut. Imageinwords: Unlocking hyper-detailed image descriptions, 2024. URL https://arxiv.org/abs/2405.02793.

A Detail Settings

Backbone and training protocol. The main experiments post-train BAGEL-7B-MoT (14B total parameters). We adopt a two-stage cyclic training protocol (HELIX) with $C = 8$ cycles. Stage 1 post-trains the generation strand using DiffusionNFT-style reward-weighted optimization, and Stage 2 post-trains the understanding strand using GRPO with group-wise standardized advantages. We employ LoRA with two separate adapters: a Stage 1 adapter for the generation strand (rank r=32) and a Stage 2 adapter for the understanding strand (rank r=32). Training is performed with bf16 mixed precision.

Stage 1 (Generation post-training). Stage 1 is trained for 30 epochs per cycle using the `dpm2` solver. For rollout sampling, we generate K=24 candidates per prompt using 10 inference steps with a guidance scale of 4.5 for training acceleration. For evaluation, we increase the sampling steps to 50. Optimization is performed with a per-device batch size of 6. The intrinsic reward is computed as a weighted sum of the target view and expansion view likelihood gains defined in Equation (9), with coefficients set to λ_{tgt}=0.5 and λ_{exp}=0.5. We utilize an intrinsic reward based on self-evaluated likelihood gain (Geneval) and apply KL regularization with β=0.0001. The implicit-policy update uses a strength of β_{nft}=1.0 with decay type 1, and EMA is enabled for the generation parameters.

Stage 2 (Understanding post-training). Stage 2 updates the image-conditioned text policy with GRPO. We use a learning rate of 5×10^{-5} and train for 1 epoch per cycle (all filter samples from stage 1). The rollout configuration uses a group size of G=16 (i.e., 16 sampled captions per image) and a per-device training batch size of 2 with gradient accumulation steps set to 2. For reward computation, we use one reconstruction per sampled caption and set the weighting coefficients to λ_{pa}=0.1 and λ_{rec}=2.0. The similarity metric utilizes DINOv3 features (or alternatively the UMM's internal ViT) for semantic alignment and the UMM's native VAE features for structural consistency. We set the KL penalty coefficient to β=0.01 and use the DAPO-style loss with ϵ_{high}=0.3. For reward computation, each sampled caption is evaluated with one reconstruction, and the maximum completion length is 1536 tokens.

Data Bridge. At the start of each cycle, we synthesize a batch of samples using a dynamic mixing ratio of λ=0.5 to balance Perceptual Descriptive and Knowledge Reasoning prompts. Within each cycle, prompts are filtered using reward statistics from Stage 1 and then regenerated under the visual-truth constraint to produce target-consistent training pairs for Stage 2. The unseen-target quantile is q=0.7. Cascade filtering retains the top 80% by mean reward, then the top 80% of that subset by standard deviation.

Resolution and compute. Training uses a resolution of 512, while evaluation is reported at both 512 and 1024 resolutions. Experiments are conducted on 8 NVIDIA H20 GPUs. With the configuration of 30 generation epochs and 1 understanding epoch per cycle, the total training time is approximately 50 hours (400 GPU-hours).

B Broader Impacts

This work improves text-to-image controllability in unified multimodal models via annotation-free cyclic post-training, reducing reliance on human annotations and external reward models. More reliable compositional generation can benefit assistive creation, education, and rapid prototyping.

Stronger generation also raises misuse risks, such as producing deceptive imagery or misinformation. Responsible deployment should include safety filtering and provenance mechanisms, and intrinsic evaluators may still inherit biases or blind spots, so monitoring and safety-oriented extensions remain important.

C Algorithms for the Data Bridge

This appendix provides the pseudocode of the Data Bridge used in HELIX. The main paper describes the design motivations and training signals in text; here we summarize the procedural steps for target-driven dual-mode prompt synthesis and reward-statistics cascade filtering with truth-guided regeneration.

Algorithm 1 Dual-Mode Prompt Generation

Input: target pool $\mathcal{S}$, text policy π_θ, mode ratio λ, initial weights w_0, previous-cycle record $\mathcal{R}^{(c-1)}$, EMA memory, unseen-target quantile q, temperature T
$w_c \leftarrow \textsc{Reweight}(\mathcal{R}^{(c-1)}, \mathrm{EMA}, q, T)$ using Equation (5); use w_0 if the record is empty
Initialize $\tilde{\mathcal{D}}_{\mathrm{Gen}} \leftarrow \emptyset$
for each target $t \sim w_c(\mathcal{S})$ in the synthesis budget **do**
  Sample a synthesis mode with ratio λ
  if Perceptual Descriptive **then**
    Select a Simple template type from {`plain`, `color`, `count`, `pos`, `cpos`}
    $(p, v_{\mathrm{gt}}) \leftarrow \pi_\theta(\textsc{PerceptualTemplate}, t)$
  else
    $(p, v_{\mathrm{gt}}) \leftarrow \pi_\theta(\textsc{ReasoningTemplate}, t)$
    Validate that v_{gt} contains the target phrase t; otherwise discard
  end if
  $\tilde{\mathcal{D}}_{\mathrm{Gen}} \leftarrow \tilde{\mathcal{D}}_{\mathrm{Gen}} \cup \{(p, v_{\mathrm{gt}}, t)\}$
end for
Output: $\tilde{\mathcal{D}}_{\mathrm{Gen}}$

Algorithm 2 Reward-Statistics Cascade Filtering & Regeneration

Input: $\tilde{\mathcal{D}}_{\mathrm{Gen}}$, Stage 1 reward record $\mathcal{R}^{(c)}$, updated generation policy π_θ, retention percentages $a\%, b\%$
Phase I: Statistical Filtering
for each $(p, v_{\mathrm{gt}}, t) \in \tilde{\mathcal{D}}_{\mathrm{Gen}}$ **do**
  $\{r_k\}_{k=1}^{K} \leftarrow \textsc{ReadRewards}(\mathcal{R}^{(c)}, p)$
  $\mu(p) \leftarrow K^{-1} \sum_k r_k$; $\sigma(p) \leftarrow [K^{-1} \sum_k (r_k - \mu(p))^2]^{1/2}$
end for
$\mathcal{P}_{\mathrm{HQ}} \leftarrow \textsc{TopByMean}(\tilde{\mathcal{D}}_{\mathrm{Gen}}, a\%)$
$\mathcal{P}^* \leftarrow \textsc{TopByStd}(\mathcal{P}_{\mathrm{HQ}}, b\%)$
Phase II: Truth-Guided Regeneration
$\tilde{\mathcal{D}}_{\mathrm{Und}} \leftarrow \emptyset$
for each $(p, v_{\mathrm{gt}}, t) \in \mathcal{P}^*$ **do**
  $y \sim \pi_\theta(\cdot \mid v_{\mathrm{gt}})$
  $\tilde{\mathcal{D}}_{\mathrm{Und}} \leftarrow \tilde{\mathcal{D}}_{\mathrm{Und}} \cup \{(p, y, v_{\mathrm{gt}}, t)\}$
end for
Output: $\tilde{\mathcal{D}}_{\mathrm{Und}}$

C.1 Dual-Mode Prompt Generation with Adaptive Target Reweighting

Algorithm 1 summarizes how the Data Bridge constructs the Stage 1 prompt set $\tilde{\mathcal{D}}_{\mathrm{Gen}}$. The previous-cycle record $\mathcal{R}^{(c-1)}$ contains prompt tuples and their candidate rewards; it is empty before the first cycle. Reweighting uses the stored within-prompt standard deviations, aggregated by target, and retains the initial weights when no previous-cycle record is available.

C.2 Reward-Statistics Cascade Filtering and Truth-Guided Regeneration

Algorithm 2 consumes the current cycle's recorded Stage 1 rewards $\mathcal{R}^{(c)}$. Statistical filtering reuses these records; image generation in the bridge occurs during the regeneration phase. We retain the original prompt and target metadata alongside each regenerated image to preserve the Stage 2 supervision interface.

D Additional Analysis and Experiments

This appendix provides additional evidence for the cyclic intrinsic-exchange mechanism, including understanding-side ablations, cross-backbone transfer, target-vocabulary dependence, editing-

oriented evaluation, and stability analysis. These results complement the main ablation study in Section 4.3.

D.1 Capability Requirements of the Initial UMM

HELIX is designed to improve a unified multimodal model that already exposes non-trivial generation and understanding capabilities, rather than to bootstrap an arbitrary weak model from scratch. In particular, the initial UMM should be able to (i) generate recognizable visual evidence from text prompts, (ii) condition language predictions on generated images, and (iii) produce likelihood-gain signals with non-degenerate variation across samples. If the initial understanding strand is severely biased or fails to distinguish target-consistent from target-inconsistent images, the intrinsic reward can inherit this bias and may provide weaker guidance.

The Data Bridge mitigates this failure mode but does not eliminate the requirement. The target view anchors the reward to explicit semantic concepts, the expansion view increases coverage over fine-grained attributes, cascade filtering removes low-quality or uninformative rollouts, and Stage 2 reconstruction updates refresh the evaluator using regenerated target-consistent visual evidence. The cross-backbone results on JanusFlow-1.3B in Appendix D.4 support that HELIX is not specific to Bagel, while the smaller magnitude of gains is consistent with the framework depending on the base UMM's initial competence.

D.2 Target Vocabulary Dependence and Annotation-free Post-training

The Data Bridge uses semantic target vocabularies as concept anchors for prompt synthesis. These targets provide coverage over objects, attributes, and relational concepts, but they are not paired image–text annotations or preference labels. Thus, the training setting is better described as annotation-free post-training: supervision is synthesized internally from target concepts and model-intrinsic feedback, without external annotated image–text data or auxiliary reward models.

The choice of target vocabulary affects coverage and curriculum quality. A substantially smaller or randomly sampled target pool would still instantiate the same HELIX procedure, but it would concentrate training on fewer concepts and could plateau earlier on compositional or knowledge-intensive benchmarks due to missing semantic coverage. Adaptive target reweighting can allocate more updates to unstable targets within the available pool, but it cannot recover concepts that are absent from the pool. This is why we use broad disjoint target sets and report both generation and understanding-side ablations to verify that the cyclic updates improve semantic grounding rather than overfitting to a narrow target list.

D.3 Full Understanding-side Ablations

Table 7 reports understanding-side ablations under both a matched low-resolution (224px) setting and the native-resolution setting. Across both settings, HELIX improves MMBench and MMVP; MMMU is stable at 224px and improves from 53.0 to 53.7 at native resolution. Other understanding dimensions show mixed changes.

Notably, the low-resolution setting serves as a stricter test, where fine-grained visual evidence is significantly compressed. HELIX still achieves the best performance on MMBench (81.36) and MMVP (64.00), suggesting that the improvements are robust to degraded visual inputs and are not merely a byproduct of higher-resolution details.

From the ablations, removing Adaptive Target Reweighting slightly reduces MMVP and destabilizes performance across both resolutions, consistent with its role in maintaining informative training signals. Removing the Reconstruction reward leads to noticeable drops in MMVP (especially at native resolution: $70.00 \rightarrow 68.67$), confirming its importance for fine-grained visual grounding. In contrast, removing the Prompt-Alignment reward has a smaller effect on MMVP but slightly impacts MMMU and MM-Vet, suggesting that it primarily refines semantic preference and knowledge-based judgment.

Finally, removing intrinsic rewards degrades MM-Vet and MME-C more substantially, indicating weaker multimodal reasoning consistency. Overall, these results support understanding preservation with selective grounding gains, with regressions retained in the table.

Table 7: **Full understanding-side ablations on Bagel.** Each entry reports 224px / native-resolution performance. Best results are in **bold**, and second-best are underlined.

Variant	MMBench	MMMU	MMVP	MM-Vet	MME-P	MME-C
Base Model	81.18 / 85.00	51.30 / 53.00	62.00 / 69.33	45.4 / **67.2**	1575.7 / **1687.0**	642.9 / 701.0
w/o Adaptive Target Reweighting	81.27 / 85.40	50.30 / 53.60	**64.00** / 69.00	46.9 / 65.0	1568.1 / 1685.7	649.6 / 698.2
w/o Prompt-Align. Reward r^{pa}	81.09 / 85.40	51.20 / 53.30	62.67 / 69.67	45.9 / 64.8	**1576.6** / 1684.6	640.0 / **701.8**
w/o Recon. Reward r_{rec}	81.09 / **85.57**	50.80 / 53.60	63.33 / 68.67	**48.9** / 64.8	1575.2 / 1679.3	647.5 / **702.5**
w/o Intrinsic Reward	81.27 / 85.05	50.90 / 53.40	63.33 / 69.33	47.5 / 65.2	**1580.5** / 1681.8	645.0 / 681.1
Full Cycle (Ours)	**81.36 / 86.50**	51.30 / **53.70**	**64.00 / 70.00**	48.2 / 65.9	1577.5 / 1681.0	622.9 / 688.0

Table 8: GenEval across backbones. S/T denote single/two objects, Cnt counting, Col colors, Pos position, and Attr color attribute. All entries are percentages.

Backbone	Model	S	T	Cnt	Col	Pos	Attr	Overall
JanusFlow-1.3B	Base	89.69	45.20	30.94	73.67	31.50	25.25	49.38
	HELIX	94.69	52.27	29.38	72.61	38.00	27.25	52.37
BLIP3-o-4B	Base	90.94	68.18	15.00	67.82	49.50	34.75	54.36
	HELIX	91.25	69.44	20.00	71.81	48.75	34.00	55.88
BLIP3-o-8B	Base	90.62	69.44	14.69	76.86	47.75	37.50	56.14
	HELIX	90.62	71.46	16.88	76.33	49.75	37.00	57.01

Table 9: **Cross-backbone understanding results on JanusFlow-1.3B.**

Model	MMBench	MMMU	MMVP	MM-Vet	MME-P	MME-C
JanusFlow Baseline	64.69	30.30	38.67	38.4	1297.0	243.9
JanusFlow + HELIX	**64.95**	**31.10**	**39.33**	38.2	**1300.8**	**248.6**
Δ	+0.26	+0.80	+0.66	-0.2	+3.8	+4.7

D.4 Cross-backbone Transfer

We evaluate HELIX beyond BAGEL on JanusFlow-1.3B and BLIP3-o-4B/8B. Table 8 presents their GenEval results together, including all six categories. Overall improves by 2.99 points on JanusFlow-1.3B, 1.51 on BLIP3-o-4B, and 0.86 on BLIP3-o-8B, supporting transfer across model implementations and scales. We evaluate the 4B checkpoint after Stage 1 of cycle 5 and the 8B checkpoint reported in the rebuttal.

For BLIP3-o we use our HELIX adaptation with BF16 LoRA rank 64 and alpha 128, 512×512 images, and 24 candidates per prompt. Stage 1 uses 30 epochs per cycle and learning rate 10^{-4}. Stage 2 uses up to 80 regenerated bridge instances, one epoch, learning rate 5×10^{-5}, caption group size 4 per rank, and gradient accumulation 2. Feature/decoder sampling uses 30/50 steps and CFG 3.0/3.0. GenEval uses 553 prompts with four samples each (2,212 images), evaluation seed 34396, and the same sampling parameters for Base and HELIX.

The BLIP3-o transfer implementation uses all eligible bridge prompts up to the 80-instance cap, with truth-guided regeneration. The 4B exact Overall scores are 0.543647 and 0.558755, a gain of 0.015107 before rounding. The 8B run uses the same core configuration with a smaller Stage-1 per-rank micro-batch for memory.

Table 9 reports the accompanying JanusFlow understanding results.

D.5 Resolution and Aspect-ratio Stability

HELIX is trained at 512×512 resolution for rollout efficiency in Stage 1. To test whether the learned improvements transfer beyond this fixed square setting, we evaluate generation under randomly sampled inference aspect ratios $\{4{:}3, 3{:}4, 9{:}16, 16{:}9, 1{:}1\}$. HELIX improves random-aspect GenEval Overall from 0.74 to 0.83. The largest gains occur on spatial and compositional categories, including position (0.44 to 0.63), color attribute (0.53 to 0.70), counting (0.70 to 0.81), and two-

Table 10: **Random-aspect-ratio GenEval evaluation.** During inference, aspect ratios are randomly sampled from $\{4{:}3, 3{:}4, 9{:}16, 16{:}9, 1{:}1\}$.

Method	Overall	Two-object	Position	Color Attr.	Counting
Base Model	0.74	0.91	0.44	0.53	0.70
HELIX	**0.83**	**0.97**	**0.63**	**0.70**	**0.81**
Δ	+0.09	+0.06	+0.19	+0.17	+0.11

Table 11: **ImgEdit results.** Category-level editing scores under matched internal evaluation.

Method	Bg.	Style	Extract	Remove	Replace	Adjust	Add	Compose	Action	Overall
Base 512	2.06	4.12	1.74	3.13	**3.86**	2.68	3.29	2.19	**4.06**	2.97
HELIX 512	**2.56**	**4.31**	**1.90**	**3.21**	3.80	**2.88**	**3.80**	**2.35**	4.05	**3.19**
Base 1024	2.80	3.87	1.65	2.76	3.14	3.32	3.49	1.58	3.21	2.95
HELIX 1024	**2.89**	**4.37**	**1.74**	**2.86**	**3.38**	**3.48**	**3.76**	**1.62**	**3.40**	**3.15**

object composition (0.91 to 0.97). This suggests that the Stage 1 optimization does not merely overfit to 1:1 inference.

We also evaluate variable-resolution editing using GEdit, where image sizes and aspect ratios vary across examples. HELIX improves SC/PQ/Overall from 7.070/6.314/6.145 to 7.272/6.448/6.337. Together with the random-aspect GenEval results, this indicates that the gains transfer beyond fixed-resolution square generation to more diverse resolution and aspect-ratio settings.

D.6 Editing-oriented Evaluation

We further evaluate whether HELIX improves editing-oriented controllability beyond one-shot text-to-image generation. Editing is a useful stress test because the model must preserve irrelevant visual content while applying localized or semantic changes specified by the instruction. We evaluate on ImgEdit and GEdit under matched internal settings, and therefore focus on relative improvements over reproduced baselines rather than direct comparison to externally reported numbers.

ImgEdit. Table 11 reports category-level ImgEdit results. At 512^2, HELIX improves the overall score from 2.97 to 3.19, with gains on background editing, style transfer, extraction, adjustment, addition, and composition. The largest improvements appear in categories that require instruction-conditioned visual modification rather than global regeneration, such as background editing (+0.50), addition (+0.51), adjustment (+0.20), and composition (+0.16). At 1024^2, HELIX also improves the overall score from 2.95 to 3.15, with consistent gains on background editing, style transfer, adjustment, addition, and action editing. These results suggest that cyclic intrinsic post-training improves editing controllability and that the gains are not restricted to the training resolution.

GEdit. We also evaluate HELIX on GEdit, which uses variable input resolutions and aspect ratios. HELIX improves the average SC/PQ/Overall scores from 7.070/6.314/6.145 to 7.272/6.448/6.337. Table 12 reports category-level Overall scores and absolute gains. HELIX improves 7 out of 11 categories, with especially clear gains on text change (+0.785), tone transfer (+0.759), material alteration (+0.372), subject removal (+0.584), motion change (+0.067), style change (+0.027), and color alteration (+0.010). The strongest improvements occur in edits requiring semantic reinterpretation or fine-grained attribute manipulation, suggesting that HELIX improves not only generation fidelity but also instruction-conditioned editability. The few regressions, such as subject addition and background change, indicate that preserving or inserting large visual entities remains challenging and merits further study.

D.7 Conditional Surrogate Stability

We analyze cyclic generation–understanding updates through a shared surrogate potential. Let g, u denote generation and understanding parameters, and define

$$\Phi(g, u) = J_{\mathrm{gen}}(g, u) + \lambda J_{\mathrm{und}}(g, u) - \tau C(g, u). \tag{14}$$

Table 12: **GEdit results.** Category-level variable-resolution editing results. We report Overall scores and absolute gains.

Category	Base	HELIX	Δ
Background change	**6.741**	6.590	-0.151
Color alter	6.771	**6.781**	+0.010
Material alter	6.104	**6.476**	+0.372
Motion change	4.155	**4.222**	+0.067
PS human	**4.764**	4.615	-0.149
Style change	5.778	**5.805**	+0.027
Subject add	**7.947**	7.763	-0.184
Subject remove	6.688	**7.272**	+0.584
Subject replace	**6.744**	6.740	-0.004
Text change	6.677	**7.462**	+0.785
Tone transfer	5.227	**5.986**	+0.759
Average	6.145	**6.337**	+0.192

We model one cycle as approximate block ascent on this analytical potential:

$$g_{k+1} = g_k + \alpha_k(\nabla_g \Phi(g_k, u_k) + \xi_k^g + b_k^g), \tag{15}$$

$$u_{k+1} = u_k + \beta_k(\nabla_u \Phi(g_{k+1}, u_k) + \xi_k^u + b_k^u). \tag{16}$$

Assume block-smoothness constants L_g, L_u, an upper bound on Φ, finite initial expected potential, and bias bounds $\|b_k^g\|, \|b_k^u\| \le \delta_k$. Let $\mathcal{F}_k$ denote information before the g update and $\mathcal{F}_{k+1/2}$ additionally include that update. Each bias is measurable before its respective block update; each noise has conditional mean zero and conditional second moment at most σ_g^2 or σ_u^2 relative to that block's information. Assume deterministic stepsizes and bias bounds, with $\sum_k \alpha_k \delta_k^2$ and $\sum_k \beta_k \delta_k^2$ finite; each stepsize sequence has divergent sum and square-summable squares, and eventually $\alpha_k \le 1/(3L_g)$, $\beta_k \le 1/(3L_u)$.

Proposition D.1 (Weighted surrogate stationarity)**.** *Under these assumptions, writing* $G_k = \mathbb{E}\|\nabla_g \Phi(g_k, u_k)\|^2$ *and* $U_k = \mathbb{E}\|\nabla_u \Phi(g_{k+1}, u_k)\|^2$*, we have*

$$\sum_k (\alpha_k G_k + \beta_k U_k) < \infty.$$

Consequently $\liminf_k G_k = \liminf_k U_k = 0$*. If additionally* $\min(\alpha_k, \beta_k) \ge c(\alpha_k + \beta_k)$ *for some* $c > 0$ *eventually, then* $\liminf_k (G_k + U_k) = 0$*.*

Proof. Block smoothness, conditional unbiasedness and Young's inequality give

$$\mathbb{E}[\Phi(g_{k+1}, u_k) - \Phi(g_k, u_k)] \ge \tfrac{1}{4}\alpha_k G_k - r_k^g,$$

where $r_k^g = \alpha_k \delta_k^2 + \frac{3}{2} L_g \alpha_k^2 (\sigma_g^2 + \delta_k^2)$. The analogous u inequality has $\frac{1}{4}\beta_k U_k - r_k^u$, with $r_k^u = \beta_k \delta_k^2 + \frac{3}{2} L_u \beta_k^2 (\sigma_u^2 + \delta_k^2)$. These residuals are summable: stepsizes are bounded eventually, so the weighted squared-bias assumptions also control the squared-step bias terms. Add the two inequalities and telescope; the upper potential bound implies the finite weighted gradient sum. Divergence of each stepsize sum gives the separate liminf conclusions. Under the additional comparability condition, the weighted sum bounds $c \sum_k (\alpha_k + \beta_k)(G_k + U_k)$, which yields the joint liminf conclusion. □

The two block measures are evaluated at their respective update points within a cycle. The proposition establishes surrogate stationarity under the stated smoothness, noise, bias, and stepsize assumptions.

D.8 Methodological Novelty Relative to Related Post-training Methods

HELIX should be understood as a closed-loop co-evolution framework rather than as a simple aggregation of existing post-training components. Self-rewarding methods such as SRUM primarily use the model's own understanding capability to improve generation, but the evaluator itself is not explicitly refreshed as a second strand of the loop. Reconstruction-alignment methods such as RecA show the value of reconstruction grounding, but do not formulate post-training as an iterative process

in which generation and understanding repeatedly provide training signals to each other. UniRL and UniGame are closely related self-improvement paradigms, but their objectives focus on self-generated SFT/RL data or adversarial consistency rather than a stabilized intrinsic feedback loop between a generation policy and an understanding policy.

The Data Bridge is central to making this loop trainable. It synthesizes target-driven prompts, filters unstable samples through reward statistics, and regenerates cleaner training pairs before they are used by the understanding strand. The generation-budget control and independent reward audits in Appendices G and J examine the contribution of cyclic evaluator updates.

E Data Bridge Self-Synthesized Data Examples

This section provides concrete examples of the self-synthesized prompt set used by the Data Bridge to train Stage 1.

Table 13: **Examples of generated samples.** The Samples from both Perceptual Descriptive and Knowledge Reasoning modes.

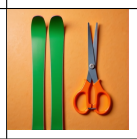
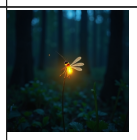
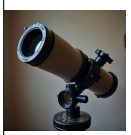
Mode	**Prompt** p	**Visual truth** v_{gt}	`Target` t	**Generated Image**
simple	a photo of a red spoon left of a white bird	a photo of a red spoon left of a white bird	`cpos|red|spoon|left of`	
simple	a photo of a pink tv below a green bird	a photo of a pink tv below a green bird	`cpos|pink|tv|below`	
simple	a photo of six carrots	a photo of six carrots	`count|six|carrot`	
simple	a photo of a green skis left of a orange scissors	a photo of a green skis left of a orange scissors	`cpos|green|skis|left of`	
simple	a photo of a bottle above a pizza	a photo of a bottle above a pizza	`pos|bottle|above`	
reason	The portable audio device used for listening to music in silent environments, using earphones to partially block external sounds.	a pair of earphones with audio cables	`earphone`	
reason	The small insect that uses bioluminescence to attract mates and predators alike under the cover of night.	a glowing firefly in a dark nighttime forest	`firefly`	
reason	The team sport involving kicking a spherical ball through goalposts, with rules and strategies based on the physics of motion and ball control.	a football game in progress with players in action	`football`	
reason	The optical device allowing stargazers to view distant celestial bodies by focusing light through a series of lenses.	a telescope with eyepiece and objective lens	`telescope`	
reason	The flower known for its beauty and resilience, often growing in water and symbolizing purity in many cultures.	a lotus flower blooming in a body of water	`lotus flower`	

Table 14: Generation budgets and absolute training costs, with GPU types specified. NR means not reported or unavailable.

Strategy	Cycles	S1 epochs	Candidates	Gen. updates	Cost
One-shot Stage 1	1	30	8,640	60	NR
Stage-1-only control	8	240	69,120	480	163.94 H200 GPU-h
Full HELIX	8	240	69,120	480	$\sim$400 H20 GPU-h
SRUM	–	–	–	–	NR

F Limitations

HELIX improves generation with selective changes in understanding; its applicability and evaluation limitations are discussed below. First, the method assumes an initial UMM with non-trivial generation and understanding capabilities; if the base understanding strand is severely biased, intrinsic rewards may inherit these biases and provide weaker guidance.

G Additional Controls and Computational Cost

Bridge decomposition after three cycles yields GenEval 73.32 for the full bridge, 72.43 for rollout reuse, and 69.78 for mean-only selection, as detailed in Appendix H. Direct caption evaluation finds broadly stable CAPTURE scores and selective grounding gains; a blinded Qwen72B comparison of initial and Cycle 8 captions gives 57.5% overall preference, as reported in Appendix I. The intrinsic reward correlates positively with independent Qwen72B judgments, with $\rho = 0.4627$, as shown in Appendix J. BLIP3-o-4B improves GenEval from 54.36 to 55.88, and BLIP3-o-8B improves from 56.14 to 57.01; the complete results appear in Appendix D.4.

The eight-cycle Stage-1-only control starts from BAGEL base and retains the target-driven prompt generator, adaptive planner, and intrinsic generation reward. It matches Full HELIX's generation budget while keeping the understanding branch frozen and disabling Stage 2. The comparison tests cyclic generation–understanding exchange against continued generation-only training.

P0 GenEval is 77.3508 and DPG is 85.4639. WISE_Verified no-CoT uses the same 1,000-entry manifest for P0 and Full, 512$\times$512 images, 50 sampling steps, CFG 4.5, the same seed mapping, and the same Qwen3.5 judge. Its scores are 0.334 and 0.384. The main WISE results use GPT-4o, and the WISE ablation uses Qwen3-VL. Full GenEval and DPG scores are 84.81 and 85.73.

The T2I-CompBench results reported in our rebuttal are 78.64 for the eight-cycle Stage-1-only control and 87.25 for Full HELIX, a difference of +8.61 points, as included in Table 5.

The control uses four H200 GPUs for 40.984 hours. Full HELIX uses eight H20 GPUs for approximately 50 hours, with 30 Stage-1 epochs and one Stage-2 epoch per cycle. The generation budgets match, and Full HELIX additionally trains the understanding branch in Stage 2. SRUM's absolute training duration was unavailable in the records used for this comparison.

H Data Bridge Decomposition

We compare the full bridge with direct reuse of Stage-1 rollout images and mean-only selection. The three runs use four GPUs, three cycles of 30 generation epochs, per-device/global batch 6/24, 24 candidates per prompt, 144 candidates per epoch (12,960 total), 512px resolution, and LoRA rank/alpha 32/64. Stage 2 and adaptive target sampling are enabled throughout. The variants alter cross-stage sample selection or regeneration.

The complete three-cycle trajectory is reported in the main-text Table 6.

Evaluation occurs after Stage 1 and before the current cycle's bridge and Stage 2. Cycle 1 is therefore identical; the first intervention affects subsequent cycles. By cycle 3, the full bridge exceeds direct reuse by 0.89 points and mean-only filtering by 3.54 points, supporting truth-guided regeneration and the standard-deviation gate. The adaptive-reweighting ablation gives Full/without-reweighting scores of 87.25/86.44 on T2I-CompBench, 84.81/80.43 on GenEval, and 85.73/85.09 on DPG.

Table 15: CAPTURE description evaluation (0–100 scale). Each entry is Base/Full Cycle 7; Object P is object precision.

Dataset	Overall	Object F1	Attribute F1	Relation F1	Object P
DetailCaps	61.0556/60.9209	65.2703/65.3839	59.8013/59.4847	53.6546/53.3543	71.7688/72.5837
IIW-400	58.4079/58.4260	62.6718/62.4592	57.0548/57.1838	51.1315/51.4483	–

Table 16: Automatic Qwen72B pairwise preferences on 400 fixed inputs, in percentages.

Dimension	C8 win	Tie	Initial win	Preference
Comprehensiveness	48.5	27.0	24.5	62.0
Specificity	45.0	26.0	29.0	58.0
Overall	43.5	28.0	28.5	57.5

Table 17: External alignment of likelihood-gain rewards.

Metric	Base	Full (C8-era)
Qwen expected-score Spearman ρ	0.4532	0.4627
Detector soft-score Spearman ρ	0.2673	0.2689
Detector ROC-AUC	0.6843	0.6992

I Direct Caption and Grounding Evaluation

We compare BAGEL Base and Full Cycle 7 with the same image inputs, the prompt `Describe this image in detail.`, greedy decoding, and a 512-token cap. DetailCaps-4870 [47] provides three detailed reference descriptions per image from GPT-4o, GPT-4V, and Gemini-1.5-Pro; IIW-400 is a fixed ImageInWords subset [48]. We use CAPTURE [47] to measure object, attribute, and relation correspondence with these references, as reported in Table 15.

HELIX improves object precision and F1 on DetailCaps and attribute and relation F1 on IIW-400, while CAPTURE Overall remains close to Base on both datasets. The results support understanding preservation with selective grounding gains.

In a complementary training-process audit, 400 fixed visual inputs are captioned by an initial checkpoint and Cycle 8 under the same prompt and decoding settings. Qwen2.5-VL-72B-Instruct judges anonymous A/B captions in randomized order. Preference is the Cycle 8 win rate plus half the tie rate.

J Intrinsic Reward Reliability

We audit 300 GenEval prompt–image pairs, with 100 each for counting, position, and color attribute, spanning 280 unique prompts. Generated images are fixed while the Base or Full understanding evaluator scores them. The Full evaluator is the understanding policy after Cycle 7 Stage 2, used in the Cycle 8 generation stage. Independent references are the original BF16 Qwen2.5-VL-72B-Instruct and the GenEval detector; neither participates in HELIX optimization. The Qwen 1–5 expected score uses first-token probabilities, and the detector supplies a continuous soft score and a binary success label for ROC-AUC.

The intrinsic reward is positively correlated with both independent evaluators, and Full maintains higher point estimates than Base across all three metrics. In the readout ablation, PMI gain achieves $\rho = 0.4606$ versus $\rho = 0.1003$ for uncorrected conditional log-likelihood, a paired difference of $+0.3603$. This comparison supports text-only prior correction for extracting image-conditioned semantic evidence.

We also ask BAGEL to generate a scalar in $[0, 1]$ by greedy decoding and score the output text. Table 18 shows that Full improves agreement with both Qwen72B and the GenEval detector under this direct-scoring readout. Direct scores form five discrete levels, while likelihood gain supplies continuous feedback for candidate ranking in group-relative optimization.

Table 18: Directly generated scalar-score audit on the same 300 pairs.

Metric	Base	Full (C8-era)
Distinct output scores	4	5
Qwen expected-score Spearman ρ	0.3710	0.4519
Qwen hard-score Spearman ρ	0.5734	0.6862
Detector soft-score Spearman ρ	0.3047	0.3771
Detector ROC-AUC	0.5975	0.6458

K Scope, Reproducibility, and Broader Impacts

HELIX applies to UMMs that can generate recognizable visual evidence, make image-conditioned language predictions, and distinguish candidate images through likelihood gain. These initial capabilities determine the quality of its self-training signals. The reported audits use independent automatic judges, and the principal training comparisons use single runs.

The reported annotation-free setup uses predefined concept names from external word lists to control semantic coverage. Training prompts, visual specifications, images, and rewards are synthesized by the model. A model-generated target vocabulary is an alternative instantiation. Regenerated evidence inherits the model's generation errors; cascade filtering and truth-guided regeneration improve the reliability of cross-stage transfer.

The method supports controllable image generation and multimodal assistance. Potential risks include misleading synthetic imagery, biased outputs, and inaccurate visual descriptions. Responsible release calls for provenance, upstream licenses, intended-use guidance, and safeguards appropriate to generative models.

Language models support prompt synthesis, image-conditioned text prediction, and reward computation. Independent Qwen VLMs provide the automatic evaluations described above.

L Data Bridge: Expanded Data Flow

The Data Bridge converts a word-level semantic target pool $\mathcal{S}$ into Stage 1 prompts and then into Stage 2 image–text training instances. At cycle c, it performs five operations: target sampling, prompt synthesis, reward profiling, cascade filtering, and truth-guided regeneration. The two synthesis modes share a tuple (p, v_{gt}, t): t is the target concept, p conditions Stage 1 generation, and v_{gt} explicitly describes the intended scene containing t.

Target sampling with adaptive reweighting. We aggregate the previous cycle's within-prompt reward standard deviations by target to obtain a dispersion score $u_{c-1}(t)$. We update an EMA memory with these scores, assign unseen targets a high-quantile value controlled by q, and normalize the resulting scores to obtain $\tilde{u}_c(t)$. A softmax with temperature T gives the current-cycle sampling weights:

$$w_c(t) = \frac{\exp(\tilde{u}_c(t)/T)}{\sum_{t'} \exp(\tilde{u}_c(t')/T)}. \tag{17}$$

We sample $t \sim w_c(\mathcal{S})$, allocating more synthesis budget to targets whose candidates have varied rewards. The first cycle uses the initial target weights; later cycles incorporate the preceding cycle's statistics.

Dual-mode prompt synthesis. Given t, the UMM synthesizes (p, v_{gt}) using Perceptual Descriptive or Knowledge Reasoning mode, mixed by ratio λ. Perceptual prompts state visual constraints such as cardinality, attributes, and spatial relations explicitly. Knowledge-reasoning prompts express the target through indirect causal or metaphorical cues, while v_{gt} names the target in a concrete scene. Following Algorithm 1, the resulting tuples form $\tilde{\mathcal{D}}_{\mathrm{Gen}}$. Here v_{gt} is a model-synthesized visual specification.

Reward profiling during Stage 1. For each prompt p, Stage 1 generates K candidates $\hat{y}_k$ and records their intrinsic rewards $r_k = r^{\mathrm{raw}}(p, \hat{y}_k)$, defined in Equation (9). We compute the mean $\mu(p) = K^{-1} \sum_k r_k$ and standard deviation $\sigma(p) = [K^{-1} \sum_k (r_k - \mu(p))^2]^{1/2}$. These recorded

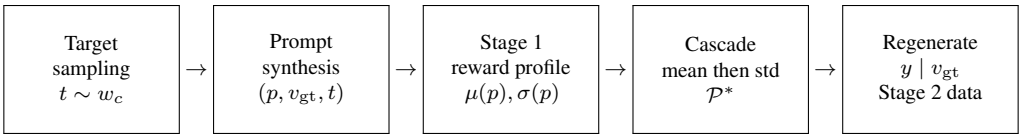


Figure 4: Data Bridge flow. Stage 1 generates candidates from p; the bridge reuses their reward statistics, then regenerates Stage 2 images from explicit v_{gt}.

statistics provide the selection signal for the current cycle's bridge and the dispersion scores for the next cycle's target sampling.

Reward-statistics cascade filtering. After Stage 1, we retain the top $a\%$ of tuples by $\mu(p)$ as $\mathcal{P}_{\mathrm{HQ}}$, then retain the top $b\%$ within this subset by $\sigma(p)$ as $\mathcal{P}^*$. The mean gate selects prompts whose candidates receive higher intrinsic quality scores. The standard-deviation gate then prioritizes prompts with differentiated candidate rewards, exposing distinctions made by the current evaluator across images. Both gates use the rewards already collected during Stage 1.

Truth-guided regeneration. For each retained tuple, the updated generation branch samples a new image $y \sim \pi_\theta(\cdot \mid v_{\mathrm{gt}})$. Conditioning on the explicit visual specification makes the intended target available directly during regeneration, including when p contains only indirect reasoning cues. Following Algorithm 2, we retain $(p, y, v_{\mathrm{gt}}, t)$ in $\tilde{\mathcal{D}}_{\mathrm{Und}}$: y is the Stage 2 visual input, p supplies the prompt-alignment reference, and v_{gt}, t preserve the scene specification and target identity. Stage 2 updates the understanding policy on these instances, and the updated policy supplies intrinsic feedback in the next cycle.

M Data Bridge Prompt Templates

The templates below specify the two prompt-synthesis modes. The output field `metadata` provides the visual specification v_{gt}, and `target` denotes the semantic target t. Training and evaluation settings are given in Appendix A.

M.1 Perceptual Descriptive Mode

```
You are a data generator for vision-language training.
Task: Describe a photo containing specifically these OBJECTS (Nouns):
[objs_str].

Requirement 1 (Object Integrity): Treat listed words strictly as NOUNS
(physical objects).
  - Do NOT turn them into adjectives.
  - Do NOT add unlisted objects.

Return ONLY a valid JSON object inside a code block.
Format:
"‘json
{
   "prompt": "string (start with ’a photo of’)",
   "metadata": "string (same as prompt)"
}
"‘

Global Constraints:
  - Output: all lowercase, ASCII only.
  - One sentence only.
```

```
>> Appended Constraints (depending on ‘simple_type‘):

[If type is plain/color/count]:
Template Type: SINGLE-OBJECT.
Requirement 2: Include EXACTLY ONE object from the list, exactly once.

(plain) → Requirement 3 (Plain): Use NO color, NO relation, NO counting.
Example: "a photo of a cat"

(color) → Requirement 3 (Color): The USER message will specify a color for
the object. You MUST place that color immediately before the object. Example:
"a photo of a blue cat"

(count) → Requirement 3 (Counting): The USER message will specify a number
word. You MUST use that number word and pluralize the object correctly.
Example: "a photo of five cats"

[If type is pos/cpos]:
Template Type: TWO-OBJECT-RELATION.
Requirement 2: You MUST include EXACTLY TWO objects from the list, each
exactly once.
Requirement 3: Use EXACTLY ONE relation from: ["left of", "right of",
"above", "below"]
Requirement 4: Structure constraints:
  - Start with ONE main object.
  - Introduce the second object ONLY via the relation.
  - BAD: "a and b are left of each other"
  - GOOD: "a is left of b"
Requirement 5: DO NOT use vague terms like ’next to’, ’near’, ’beside’.

(pos) → Requirement 6 (Pos): The USER message may specify relation. Do NOT
add colors. Example: "a photo of a cat left of a sofa"

(cpos) → Requirement 6 (Colored Pos): The USER message will specify colors
for BOTH objects. You MUST place each color immediately before its object.
Example: "a photo of a blue cat left of a red sofa"
```

Figure 5: **System Prompt Template for Perceptual Descriptive Mode.** This figure illustrates the complete prompt construction logic, including the base instruction and all conditional constraints.

M.2 Knowledge Reasoning Mode

```
You are an expert in World Knowledge Reasoning.  Your goal is to test a model's
ability to infer concepts from context.
Target Concept:  "{target}"
Domain:  {domain} | Style:  {style}

Task:  Create a KNOWLEDGE-BASED INFERENCE PROMPT and its VISUAL TRUTH.

Output a valid JSON object with exactly THREE fields:
1.  "prompt":  A reasoning sentence (10-30 words) that implies the target using
WORLD KNOWLEDGE.
  - DO NOT simply describe what the object looks like (e.g., "It is round and
red").
  - MUST use scientific principles, cultural associations, historical context,
or causal relationships.
2.  "metadata":  The precise VISUAL description of the answer (3-15 words).
  - MUST include the target phrase "{target}".
  - MUST describe the specific visual state implied by the prompt.
3.  "target":  The original concept "{target}".

=== Examples (Notice:  Prompts are about KNOWLEDGE, not just APPEARANCE) ===

Example 1 (Scientific Inference):
Target:  "apple tree"
Output:  {
  "prompt":  "The botanical specimen undergoing chlorophyll breakdown as the
season shifts to cooler temperatures.",
  "metadata":  "an apple tree with yellow autumn leaves",
  "target":  "apple tree"
}

Example 2 (Cultural/Historical Inference):
Target:  "pumpkin"
Output:  {
  "prompt":  "The iconic vegetable carved into glowing lanterns to ward off
spirits during All Hallows' Eve.",
  "metadata":  "a carved halloween pumpkin with a candle inside",
  "target":  "pumpkin"
}

Example 3 (Functional/Causal Inference):
Target:  "compass"
Output:  {
  "prompt":  "The navigational instrument that relies on the planet's
magnetosphere to determine cardinal directions.",
  "metadata":  "a magnetic compass pointing north",
  "target":  "compass"
}
```

Figure 6: **System Prompt for Knowledge Reasoning Mode.** The strict instruction requires the model to generate logic-based prompts rather than visual descriptions, verified by a visual ground truth.

N More Qualitative Results on Text-to-Image Generation

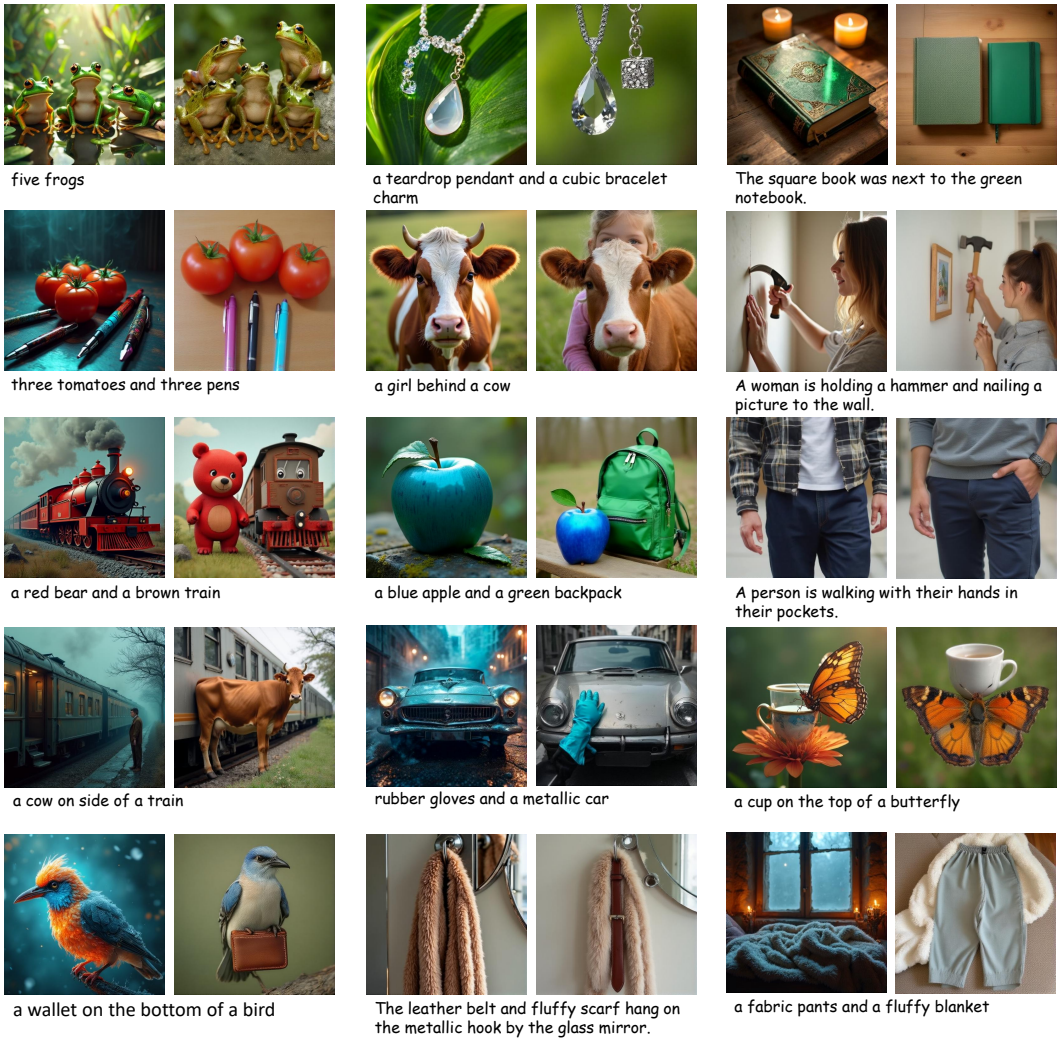


Figure 7: **Qualitative results on T2I-CompBench (512px).** For each prompt, the left image is generated by the Bagel baseline, and the right image is generated by our post-trained model.



Figure 8: **Qualitative results on DPG-Bench (1024px).** The top row shows images generated by the Bagel baseline, while the bottom row shows images generated by our post-trained model for the same prompt.

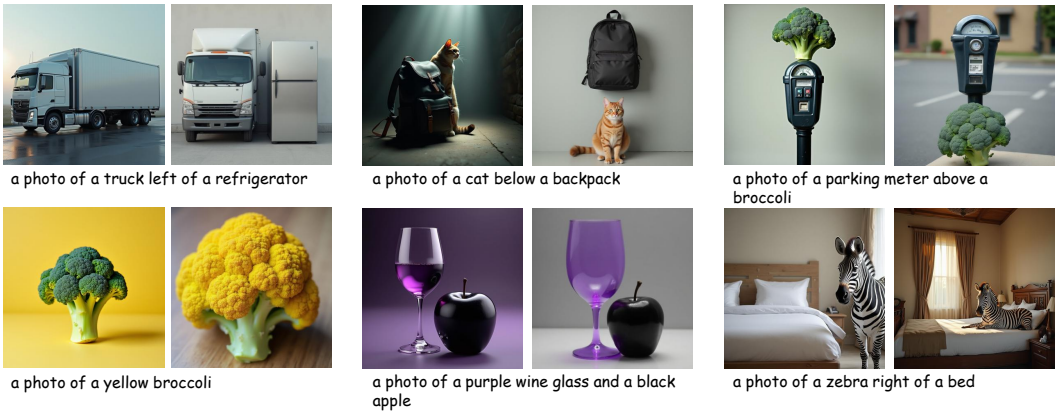


Figure 9: **Qualitative results on GenEval (1024px).** For each prompt, the Bagel baseline is shown on the left and our post-trained model on the right, illustrating improved compositional instruction following (e.g., spatial relations, attribute binding, and uncommon concepts).

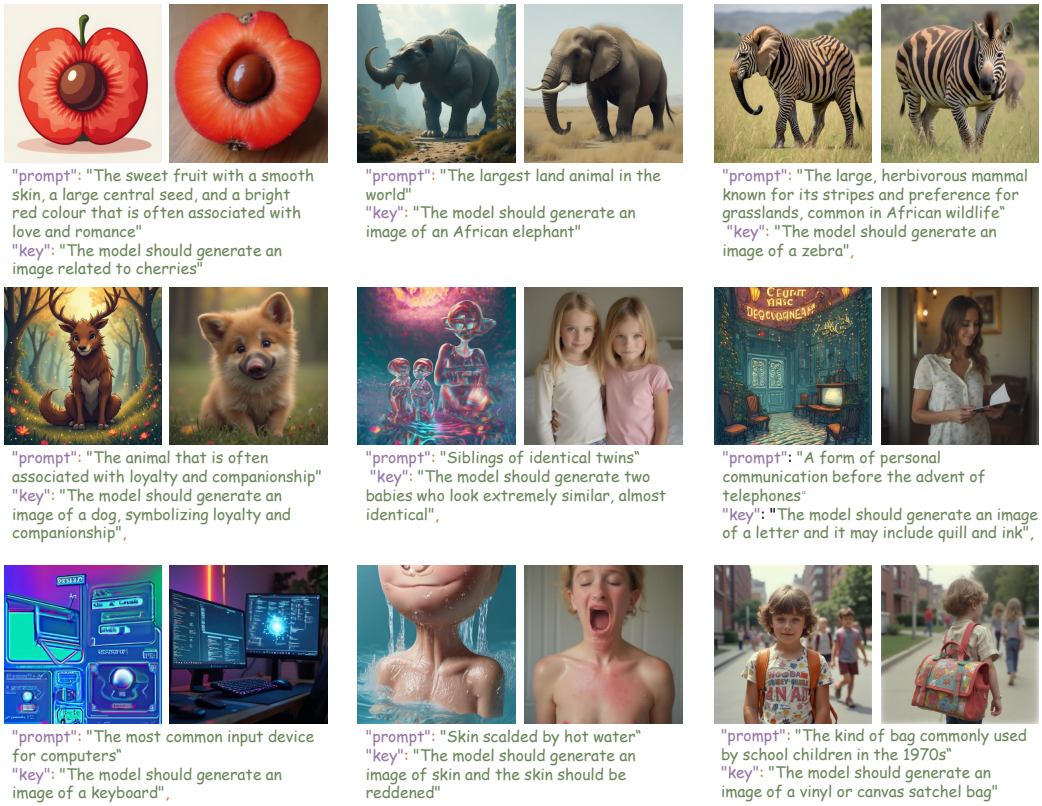


Figure 10: **Qualitative results on WISE (1024px).** The Bagel baseline (left) and our post-trained model (right) are compared on knowledge- and concept-driven prompts, showing improved world-knowledge grounding and more faithful visual realization of the intended semantics.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the proposed HELIX framework, the two-stage cyclic training design, the intrinsic reward/Data Bridge components, and the empirical scope on text-to-image benchmarks; the reported claims are matched by experiments and ablations in the main paper and appendix.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The appendix includes a dedicated Limitations section discussing the base-model capability requirement and the risk that biased or weak understanding strands may provide less reliable intrinsic rewards.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Appendix D.7 provides a surrogate theoretical analysis of HELIX dynamics. It states the assumptions used in the analysis, including smoothness, bounded variance, bounded bias, and sufficiently small step sizes, and presents Theorem D.1 on asymptotic stationarity of the surrogate dynamics with a proof sketch. We explicitly clarify that this is not an end-to-end convergence proof of the full implemented HELIX algorithm, but a formal stability explanation for the cyclic updates.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The method, objectives, Data Bridge algorithms, training protocol, hyperparameters, resolution settings, and benchmark evaluation setup are described in the main paper and Appendix A.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The code is not publicly released during the anonymous review period. We plan to release the training and evaluation code, configuration files, and prompt-generation scripts after acceptance. To support reproducibility during review, the paper provides the algorithmic details, hyperparameters, training protocol, and evaluation protocols in the main paper and appendix.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental setup specifies the backbone, training stages, optimizer-relevant hyperparameters, rollout settings, filtering ratios, reward coefficients, benchmark suites, and evaluation resolutions in the main text and Appendix A.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Due to the computational cost of cyclic post-training for unified multimodal models, the paper does not report confidence intervals or multiple-seed error bars. Instead, it reports consistent trends across multiple benchmarks, ablations, training dynamics, resolutions, and a cross-backbone transfer experiment.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix A reports the hardware and compute budget: experiments use 8 NVIDIA H20 GPUs, with approximately 50 hours of training, corresponding to about 400 GPU-hours.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research uses standard ML benchmarks, existing model backbones, and self-synthesized text-targeted training signals; no human-subject data collection or prohibited data use is involved, and the work is intended for responsible model improvement and evaluation.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss broader impacts in the appendix, including potential benefits for controllable visual generation and multimodal assistance, as well as risks such as misleading synthetic content, biased outputs, and misuse in deceptive media.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release a new high-risk model, dataset, or scraped data asset as part of the anonymous submission. If models or code are released later, appropriate usage guidelines and safeguards should be provided.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the original sources for the existing models, benchmarks, and vocabularies used in this work. Public assets are used according to their respective licenses or terms of use; proprietary APIs, when used for evaluation, are accessed only through their intended interfaces.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not introduce or release a new dataset, benchmark, or standalone asset. The Data Bridge produces self-synthesized training instances internally, and examples are documented in the appendix.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The research does not involve crowdsourcing experiments or human-subject studies.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The research does not involve crowdsourcing or human-subject studies, so IRB or equivalent approval is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLM-based tools were used solely for writing, editing, or formatting purposes and did not affect the core methodology, scientific rigor, or originality of the research.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.