CYCLE ALIGNMENT FOR UNDERSTANDING-GUIDED VISUAL EDITING IN UNIFIED MULTIMODAL MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Image editing demands both fine-grained visual understanding and controllable generation. A model must comprehend objects and layouts in the source image, parse editing instructions, and modify specific regions while preserving identity and background. Unified multimodal models (UMMs) inherently possess both capabilities within a single architecture, making them well-suited for editing. However, in conventional UMM training, the understanding and generation branches are optimized independently. Despite sharing the same backbone, the potential for understanding to directly guide generation remains largely unexploited. To exploit this potential, we propose Cycle Alignment (CycleA), a post-training paradigm that closes the loop between understanding and generation. Specifically, CycleA routes the generation branch's output back into the frozen understanding branch, where source-image VQA constraints provide differentiable semantic reconstruction feedback to the generation branch. This feedback operates directly in its native feature space through a lightweight frozen projector, avoiding the decode-encode overhead. To prevent the mapped features from drifting, we further introduce a feature alignment loss that keeps them close to real visual features. Across the evaluated configurations, CycleA improves aggregate image-editing scores over BAGEL after a 2,000-step post-training schedule following 5,000-step projector pretraining, while text-to-image benchmark scores remain comparable.

1 INTRODUCTION

Image editing is a fundamental capability for creative workflows, enabling users to perform object replacement, style transfer, background modification, and attribute adjustment through natural language instructions. This task demands both fine-grained visual understanding and controllable generation. A model must comprehend objects, attributes, and spatial layouts in the source image, accurately parse the semantic intent of editing instructions, and execute local modifications while preserving subject identity, background content, and overall composition. Image editing is therefore not merely a generation task, but one that heavily depends on visual understanding and semantic control.

Unified multimodal models (UMMs) combine visual understanding and generation within a single architecture, making them well-suited for editing. However, the understanding and generation branches are typically trained with separate objectives and receive no cross-branch supervision. The understanding branch learns rich semantic representations, but these are never used to directly guide the generation branch. The generation branch, in turn, receives only low-level visual supervision that cannot enforce semantic correctness. Despite sharing the same backbone, the two branches operate in a one-way manner, and the understanding capability is not leveraged to directly improve generation. This matters most for image editing, where generating the correct edit requires precisely understanding the image and instruction.

Current editing models perform well on many tasks, but still struggle with semantic failures such as attribute misbinding, subject identity drift, and object omission. For instance, when asked to "change the red car to blue", the model may alter the color but also unintentionally modify the car's shape or the surrounding background. These are semantic mistakes rather than pixel-level inaccuracies. The generation branch does not receive differentiable semantic feedback from the

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

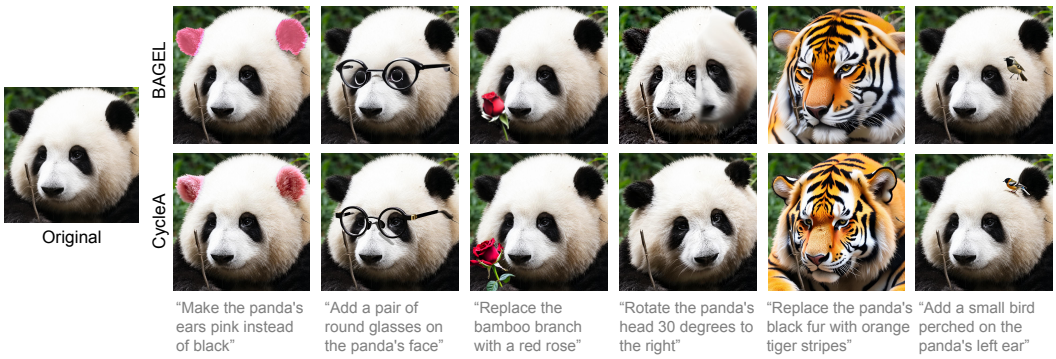

Figure 1: Qualitative comparison of image editing results. In these examples, CycleA more accurately follows several editing instructions than the baseline BAGEL, including attribute modification, object addition, replacement, and pose change.

model's own understanding branch, leaving its semantic knowledge disconnected from generation-side optimization.

To bridge understanding and generation more tightly, several methods have explored leveraging understanding capabilities for generation. Representation alignment methods such as REPA (Yu et al., 2024) and VA-VAE (Yao et al., 2025) regularize generative features with external pretrained visual encoders. Reconstruction-based methods such as ROSS (Wang et al., 2025) and Lumos (Ma et al., 2025a) add lightweight decoders or external DINO features (Caron et al., 2021) to regularize visual representations. RecA (Xie et al., 2025a) conditions the generation branch on the model's own understanding embeddings for image reconstruction. Chain-of-thought reasoning (Wei et al., 2022) and test-time verification (Guo et al., 2025) leverage understanding at inference time. Recent work also backpropagates understanding-derived objectives into visual generators. Dual-Process Image Generation uses an external frozen VLM to score decoded generator outputs, while UNO combines captioning and visual-regression supervision to steer generative representations (Luo et al., 2025; Liu et al., 2026). CycleA takes a distinct latent-space route within the same UMM. An estimated latent is passed through a lightweight frozen projector into the model's understanding pathway, avoiding pixel-space decoding and re-encoding while enabling targeted image QA supervision during post-training.

Based on this idea, we introduce Cycle Alignment (CycleA), a post-training paradigm that closes the loop between understanding and generation. Concretely, CycleA takes the latent predicted by the generation branch and, instead of decoding it to pixels and re-encoding it with the ViT, passes it through a frozen per-patch projector that maps it into the understanding branch's feature space. The frozen understanding branch then answers a VQA query about this mapped feature, and the cross-entropy error of that answer backpropagates into the generation branch as a semantic gradient. A feature alignment loss between the mapped feature and the encoder's feature of the real image keeps the mapping faithful and supplies a short gradient path that bypasses the UMM. Across the evaluated configurations, CycleA improves aggregate image-editing scores over BAGEL after a short post-training schedule, while maintain the generation ability.

Our contributions are summarized as follows.

- We propose Cycle Alignment (CycleA), a differentiable understanding-feedback post-training paradigm for unified multimodal models. CycleA routes generated visual latents back into the frozen understanding branch, enabling the model's own understanding capability to provide semantic guidance for visual generation.
- We introduce a latent-space generation-to-understanding feature path for cross-branch feedback. By mapping generated VAE latents into ViT understanding features, CycleA enables semantic feedback from the frozen understanding branch without decoding latents into pixels or re-encoding images through ViT.

- We develop a VQA-based supervision scheme that combines targeted image QA feedback with a feature-space constraint, providing structured semantic supervision that transfers to instruction-based editing without paired editing examples.

2 Related Work

2.1 Unified Multimodal Models

Unified multimodal models (UMMs) integrate visual understanding and generation within a single framework. Early approaches rely on quantized auto-regressive modeling, transforming images into discrete token sequences processed like language (Team, 2024; Xie et al., 2025b; Wu et al., 2025a; Lin et al., 2024; Wang et al., 2024; Qu et al., 2025). While these methods demonstrate the feasibility of unified modeling, their generative quality is constrained by the discretization bottleneck of VQ-based tokenizers. To overcome this limitation, recent approaches combine multimodal large language models for understanding with diffusion models for generation, yielding substantially improved expressivity. Within this hybrid paradigm, some research arranges an MLLM backbone sequentially with a diffusion decoder, predicting through special query tokens (Zhou et al., 2025; Zhang et al., 2025a; Liao et al., 2025) or intermediate latent representations (Ma et al., 2025b), where understanding and generation operate in separate stages with distinct representations. Some other research emphasizes parallel architectures that process understanding and generation within a unified backbone, including integrated transformers (Xie et al., 2025b; Chen et al., 2025b; Li et al., 2025) as well as Mixture-of-Transformers formulations (Deng et al., 2025; Wu et al., 2025c; Chen et al., 2025a) that allocate distinct pathways to each modality, maintaining separate representations even within a shared backbone. Despite the shared parameters, effective coordination between these representations remains an open challenge.

2.2 Instruction-Based Image Editing

Instruction-based image editing allows users to modify images through natural language descriptions. InstructPix2Pix (Brooks et al., 2023) first introduces a conditional diffusion model that edits images from natural language instructions, generating training pairs by combining GPT-3 with Stable Diffusion. MagicBrush (Zhang et al., 2023a) provides a manually annotated dataset for instruction-guided editing, enabling more reliable evaluation. Subsequent methods improve instruction following through MLLM-guided derivation (Fu et al., 2024) or complex instruction modeling (Huang et al., 2024). DiffEdit (Couairon et al., 2023) automates mask generation for localized editing, and ControlNet (Zhang et al., 2023b) enables spatially controllable editing through conditional inputs. Recent specialized editing models such as Step1X-Edit (Liu et al., 2025), OmniGen (Xiao et al., 2025), AnyEdit (Yu et al., 2025), UltraEdit (Zhao et al., 2024), ICEdit (Zhang et al., 2025b), OmniEdit (Wei et al., 2025), and FLUX.1 Kontext (Labs et al., 2025) achieve strong performance by training on large-scale curated editing pairs with multimodal LLM backbones or flow-matching formulations. Despite these advances, the training objectives of existing editing methods primarily optimize generative or reconstruction objectives, and generally do not use the UMM's own understanding pathway as a differentiable post-training signal.

CycleA instead invokes the model's own understanding branch during post-training. Its generic image QA signal improves semantic representations used by generation, while instruction fidelity is measured in downstream editing evaluation.

2.3 Semantic-Guided Visual Generation

Recent studies have increasingly highlighted the importance of understanding-oriented priors in improving generative modeling. Representation alignment methods such as REPA (Yu et al., 2024) and REPA-E (Leng et al., 2025) regularize diffusion training by aligning intermediate features with pretrained semantic visual representations such as DINOv2 (Oquab et al., 2024). This alignment substantially accelerates convergence and improves generation quality. VA-VAE (Yao et al., 2025) applies similar alignment to VAE latents to improve reconstruction and generation quality. RAE (Zheng et al., 2025) takes a more direct approach by redesigning latent spaces around semantically rich encoder representations, enabling high-quality generation without relying on traditional VAEs. In the context

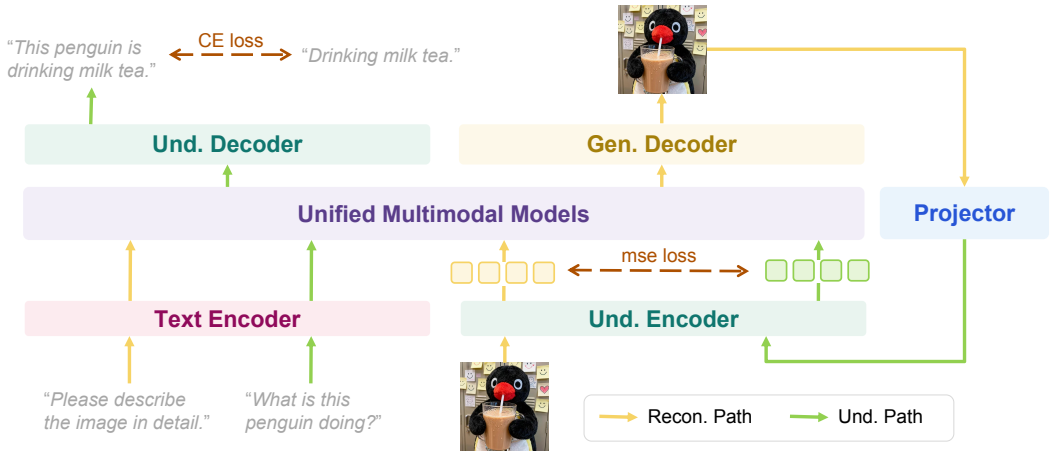

Figure 2: Overview of the Cycle Alignment pipeline. The generation branch produces an estimated clean latent along the yellow path, which is mapped directly into the understanding branch's feature space by a frozen projector. Any image icon is illustrative and does not denote an intervening VAE decode. The understanding branch processes the mapped features through a VQA objective along the green path, producing an answer that is aligned with the ground truth via cross-entropy loss. Additionally, a feature alignment loss is applied between the mapped features and the real visual features to provide a short-gradient-path supervision. Gradients from the VQA loss pass through the frozen understanding branch and the frozen projector to the generation branch, whereas the feature-alignment loss provides a shorter path through the projector directly to the generation branch.

of unified multimodal models, RecA (Xie et al., 2025a) leverages dense features from the model's own visual understanding encoder as prompts and optimizes the model to reconstruct the input image, thereby realigning understanding and generation through feature-level conditioning. UniRL (Mao et al., 2025) applies reinforcement learning to jointly optimize understanding and generation with preference signals. More recent approaches make the connection between understanding and generation explicit. Dual-Process Image Generation backpropagates a VLM-based objective through a separately decoded generator output (Luo et al., 2025). UNO applies captioning and visual-regression objectives to generative representations in a UMM (Liu et al., 2026). SPAR redesigns the tokenizer and employs semantic-pixel self-alignment with adaptive routing to build a shared representation space (Li et al., 2026). CycleA shares the broad goal of understanding-guided generation, but focuses on a latent-space post-training path within an existing UMM. A frozen projector maps predicted VAE latents directly to the same model's frozen understanding feature space, and targeted image QA supervision complements feature alignment without pixel decoding. Unlike Dual-Process, CycleA does not require an external VLM or a decoding and encoding loop. Unlike UNO, it uses targeted QA rather than captioning as its semantic objective. Unlike SPAR, it does not redesign or jointly train the tokenizer.

3 Cycle Alignment

Cycle Alignment constructs a differentiable feedback path by routing an estimated clean latent into the understanding branch, allowing image QA supervision to guide the generation parameters through semantic properties of reconstructed training images. We next introduce the preliminaries, pipeline, and alternating objectives.

3.1 Preliminaries and Motivation

Unified multimodal models (UMMs) integrate visual understanding and generation within a single backbone, typically employing a shared LLM for both tasks. Different UMM families adopt different designs for separating the understanding and generation data flows, but a common characteristic is that the two tasks are trained with separate objectives and receive no direct supervision from each

other. In this work, we take BAGEL (Deng et al., 2025) as the primary instance, which employs a Mixture-of-Transformers architecture that routes understanding and generation tokens through separate expert modules over a shared LLM.

The generation branch is trained with a flow-matching objective. Given a condition c and an image $\mathbf{x}$, the image is encoded into latent $\mathbf{z}$ via a frozen VAE encoder, with a noise level $t \in [0, 1]$ sampled to produce the noised latent $\mathbf{z}_t = (1 - t)\,\mathbf{z} + t\,\boldsymbol{\epsilon}$. The model predicts a velocity field and is optimized with

$$\mathcal{L}_{\text{gen}}(\Theta) = \mathbb{E}_{t,\,\boldsymbol{\epsilon}}\left[\left\|\mathbf{v}_\Theta(\mathbf{z}_t, c, t) - (\boldsymbol{\epsilon} - \mathbf{z})\right\|_2^2\right], \tag{1}$$

where Θ denotes the model parameters, $\mathbb{E}_{t,\,\boldsymbol{\epsilon}}$ denotes the expectation over the sampled noise level t and noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This loss measures reconstruction fidelity but does not explicitly assess object attributes, spatial relations, or whether an instruction-specified modification is applied to the intended region.

The understanding branch is trained with a cross-entropy objective on visual question answering. Given an image x and a question q, the image is encoded into visual features

$$f_{\text{gt}} = E_{\text{und}}(x), \tag{2}$$

in which E_{und} denotes the frozen visual understanding encoder. Given a ground-truth answer $a = (a_1, a_2, \ldots, a_{L_a})$, the model is trained to predict each token, optimized with

$$\mathcal{L}_{\text{und}}(\Theta) = -\sum_{i=1}^{L_a} \log p_\Theta(a_i \mid f_{\text{gt}}, q, a_{<i}), \tag{3}$$

where L_a is the answer length, $a_{<i} = (a_1, \ldots, a_{i-1})$ denotes the preceding ground-truth answer tokens. This objective requires reasoning about visual semantics such as object attributes, spatial relations, and scene composition, thereby providing a structured semantic evaluator for reconstructed images.

In standard UMM training, the total loss is the sum of these two objectives

$$\mathcal{L}(\Theta) = \mathcal{L}_{\text{gen}}(\Theta) + \mathcal{L}_{\text{und}}(\Theta). \tag{4}$$

The two tasks are trained within the same model and interact indirectly through shared parameters. However, the understanding objective never directly acts on the generation output. The understanding branch never receives generated images or latents as input, so gradients from $\mathcal{L}_{\text{und}}$ never directly evaluate or guide the generation results. While this separation effectively prevents interference between tasks operating at different abstraction levels, it also leaves the understanding branch's semantic reasoning capability unused as a source of feedback for the generation process.

This limitation is particularly relevant to image editing, where the model must modify only instruction-specified regions while preserving everything else. The generation loss does not explicitly assess whether the requested region is modified or whether object identity and unrelated content are preserved, whereas the understanding branch is trained to reason about such properties. Importantly, the VQA objective does not directly evaluate an editing instruction during training. It provides generic semantic reconstruction supervision on source images, whose usefulness for instruction-based editing is assessed through downstream evaluation. This motivates us to construct a differentiable path from the generation output to the understanding objective, enabling $\mathcal{L}_{\text{und}}$ gradients to directly guide generation. In the following sections, we present Cycle Alignment, which builds this path through a lightweight projector and a VQA-based supervision scheme.

3.2 Cycle Alignment Pipeline

Our key idea is to route the generation branch's output back to the understanding branch, providing semantic feedback through the understanding objective. Specifically, we map the generation branch's predicted latent into the understanding feature space and optimize answer likelihood for a question associated with the reconstructed training image. The overall pipeline of our method is illustrated in Fig. 2.

Cycle Alignment is a post-training approach evaluated on two UMM architectures, and its training samples are standard image-question-answer triplets without editing instructions or paired source-target images. Given a training sample, the generation branch follows the reconstruction formulation

Model	Add	Adj.	Ext.	Repl.	Rem.	Bkg.	Style	Hybrid	Action	Overall
MagicBrush	2.84	1.58	1.51	1.97	1.58	1.75	2.38	1.62	1.22	1.83
Instruct-Pix2Pix	2.45	1.83	1.44	2.01	1.50	1.44	3.55	1.20	1.46	1.88
AnyEdit	3.18	2.95	1.88	2.47	2.23	2.24	2.85	1.56	2.65	2.45
UltraEdit	3.44	2.81	2.13	2.96	1.45	2.83	3.76	1.91	2.98	2.70
ICEdit	3.58	3.39	1.73	3.15	2.93	3.08	3.84	2.04	3.68	3.05
Step1X-Edit	3.88	3.14	1.76	3.40	2.41	3.16	4.63	2.64	2.52	3.06
UniWorld-V1	3.82	3.64	**2.27**	3.47	3.24	2.99	4.21	**2.96**	2.74	3.26
OmniGen2	3.57	3.06	1.77	3.74	3.20	3.57	**4.81**	2.52	**4.68**	3.44
BAGEL	3.81	3.59	1.58	3.85	3.16	3.39	4.51	2.67	4.25	3.42
BAGEL + RecA	**4.20**	3.86	1.68	4.28	3.75	3.85	4.73	2.94	4.56	3.75
CycleA	4.09	**3.96**	1.95	**4.34**	**4.29**	**3.86**	4.71	2.60	4.50	**3.88**

Table 1: ImgEdit comparison including RecA. Best results are in **bold**. Baselines are cited from (Ye et al., 2025), and the RecA result is cited from (Xie et al., 2025a).

of RecA (Xie et al., 2025a) and predicts the velocity $v_\Theta(z_t, c, t)$. Under the linear interpolation, we estimate the clean latent as

$$\hat{z}_0 = z_t - t\, v_\Theta(z_t, c, t). \tag{5}$$

This estimate is used only to construct the auxiliary feedback path during training. The original multi-step inference procedure remains unchanged. Since a representation gap exists between the VAE latent space and the understanding branch's feature space, we introduce a lightweight per-patch projector g_ω to map the estimated clean latent into the feature space expected by the understanding branch

$$\hat{f} = g_\omega(\hat{z}_0). \tag{6}$$

The projector is pretrained independently and frozen during Cycle Alignment training. The mapped features $\hat{f}$ are fed into the frozen understanding branch together with a question q and its ground-truth answer a, yielding the VQA loss

$$\mathcal{L}_{\text{vqa}} = -\sum_{i=1}^{L_a} \log p_\Theta\Big(a_i \mid \hat{f},\, q,\, a_{<i}\Big). \tag{7}$$

We additionally regularize the mapped features toward the real-image features

$$\mathcal{L}_{\text{feat}} = \Big\|\hat{f} - f_{\text{gt}}\Big\|_2^2. \tag{8}$$

Through this design, Cycle Alignment provides the generation branch with gradients for the semantic properties queried by the image QA objective.

3.3 Training Objectives and Strategy

Cycle Alignment is optimized in two stages. We first establish a stable bridge from generative latents to understanding features, and then train the generation branch with alternating reconstruction and semantic objectives. The following components describe how this optimization preserves generation quality while introducing understanding feedback.

Projector pretraining. We pretrain the projector g_ω before Cycle Alignment to bridge the generative latent space and the understanding feature space. The lightweight per-patch MLP maps each $p \times p$ latent patch to the understanding feature dimension, with real understanding features resized to its target grid. Given an image, we obtain latent patches from the frozen generative encoder and visual features from the frozen understanding encoder. The visual features are resized to the same patch grid, and g_ω is trained with an MSE loss

$$\mathcal{L}_{\text{proj}} = \|g_\omega(z_{\text{patch}}) - f_{\text{gt}}\|_2^2. \tag{9}$$

After pretraining, g_ω is frozen during Cycle Alignment. This prevents the bridge from absorbing VQA gradients and keeps the mapped features in the feature space expected by the frozen understanding branch. Appendix A.2 provides the projector architecture and pretraining hyperparameters.

Model	Semantic Consistency	Perceptual Quality	Overall	Model	Semantic Consistency	Perceptual Quality	Overall
Gemini 2.0	6.73	6.61	6.32	OmniGen	5.96	5.89	5.06
Instruct-Pix2Pix	3.58	5.49	3.68	Step1X-Edit	7.09	6.76	6.70
MagicBrush	4.68	5.66	4.52	BAGEL	7.36	**6.83**	6.52
AnyEdit	3.18	5.82	3.21	CycleA	**8.03**	6.79	**7.08**

Table 2: Compact GEdit-Bench-EN comparison. Best results are in **bold**. Baseline results are cited from (Deng et al., 2025).

Variant	Add	Adj.	Ext.	Repl.	Rem.	Bkg.	Style	Hybrid	Action	Overall
Baseline	3.81	3.59	1.58	3.85	3.16	3.39	4.51	2.67	4.25	3.42
Captioning	3.94	3.77	2.00	4.32	3.48	3.76	4.75	3.01	**4.52**	3.71
Full noise	3.91	3.53	**2.45**	3.68	3.78	3.05	4.71	2.82	4.23	3.59
L_{feat} only	**4.13**	3.56	2.20	4.04	3.55	3.55	**4.80**	**3.03**	4.19	3.71
L_{vqa} only	4.06	3.91	2.17	4.22	3.66	3.80	4.73	2.75	4.44	3.77
CycleA	4.09	**3.96**	1.95	**4.34**	**4.29**	**3.86**	4.71	2.60	4.50	**3.88**

Table 3: Ablation results on ImgEdit. Adj., Ext., Repl., Rem., and Bkg. denote Adjust, Extract, Replace, Remove, and Background. Full noise uses $t_{\max} = 1.0$. Best results are in **bold**.

VQA supervision data. CycleA uses generic image question answer triplets that query source-image objects, attributes, spatial relations, and scene content. Unlike open-ended captions, these questions provide explicit assertions whose answer losses supply targeted gradients to the generation branch.

Training loss. The objectives are applied on disjoint alternating steps rather than summed in a single update.

$$\mathcal{L}_{\text{GEN}} = \lambda_{\text{mse}}\mathcal{L}_{\text{gen}}, \qquad \mathcal{L}_{\text{ALIGN}} = \lambda_{\text{align}}\mathcal{L}_{\text{vqa}} + \lambda_{\text{feat}}\mathcal{L}_{\text{feat}}. \tag{10}$$

Even steps optimize $\mathcal{L}_{\text{GEN}}$, whereas odd steps optimize $\mathcal{L}_{\text{ALIGN}}$. No update jointly sums all three terms.

Alternating training strategy. Even steps optimize standard T2I flow matching, while odd steps optimize Cycle Alignment. Restricting odd-step noise to $[0, t_{\max}]$ keeps $\hat{\mathbf{z}}_0$ close to the true latent and provides faithful pseudo features.

Frozen and trainable parameters. Only generation-side components are trainable. The visual encoder, understanding branch, and g_ω remain frozen but differentiable, allowing gradients to reach the generation branch.

VQA as supervision. VQA directly verifies targeted properties such as color and object count, producing more focused gradients than an open-ended image description.

4 Experiments

4.1 Experimental Setup

We implement CycleA on BAGEL-7B-MoT (Deng et al., 2025) as our primary instance. We additionally apply CycleA to Nexus-Gen (Zhang et al., 2025a) as an initial test on a second architecture, with details in the Appendix. On Nexus-Gen, the reported ImgEdit Overall score increases from 2.68 to 2.74, providing preliminary evidence that the training path can be adapted to a different generation design. The main training runs for 2000 steps with a learning rate of 4×10^{-5}, constant schedule, and 200-step linear warmup. Both λ_{align} and λ_{feat} are set to 1.0, and $t_{\max} = 0.3$.

Optimization uses AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, gradient clipping at 1.0, and a per step batch size of one. Training runs on six A800 80G GPUs with full parameter sharding and CPU offload. The frozen VAE forward pass uses `no_grad`, while the frozen understanding pathway retains gradient flow back to the trainable generation parameters.

Training data. The main training stage uses two data streams. The generation stream uses the same type of paired text and image data as RecA (Xie et al., 2025a). The alignment stream uses image question answer triplets from the dataset of Qin et al. (2025), covering object identity, attributes, spatial relations, and scene-level semantics.

We take the conversation turn as the question and answer pair and draw the reconstruction condition from 30 templates covering fidelity, style, color, geometry, texture, object count, composition, and environment. The generation stream reuses the image caption data of RecA. Independent data loaders sample the two streams in strict one to one alternation, keeping their objectives and conditioning signals separate.

Benchmarks. We evaluate on ImgEdit (Ye et al., 2025) and GEdit-Bench-EN (Liu et al., 2025). ImgEdit covers nine editing categories including Add, Adjust, Extract, Replace, Remove, Background, Style, Hybrid, and Action. GEdit-Bench-EN assesses semantic consistency and perceptual quality. We also report text-to-image generation results on DPGBench to verify that understanding-guided training does not degrade generation capability. For ImgEdit, we report the sample-level Overall score returned by the evaluation pipeline rather than an unweighted mean of the rounded category summaries. Category sizes and rounding can therefore make the displayed category mean differ from Overall.

Evaluation protocol. Following standard protocols, all benchmark results are produced by GPT-4.1 as the judge, a widely adopted automatic evaluator for image editing benchmarks including ImgEdit and GEdit-Bench-EN. We use the default evaluation prompts provided by each benchmark to ensure consistency.

Baselines. We compare against BAGEL (Deng et al., 2025), InstructPix2Pix (Brooks et al., 2023), MagicBrush (Zhang et al., 2023a), AnyEdit (Yu et al., 2025), UltraEdit (Zhao et al., 2024), ICEdit (Zhang et al., 2025b), Step1X-Edit (Liu et al., 2025), UniWorld-V1 (Lin et al., 2025), OmniGen2 (Wu et al., 2025b), OmniGen (Xiao et al., 2025), and Gemini 2.0 (Mallick & Kilpatrick, 2025).

4.2 Editing Performance

We first examine whether CycleA improves instruction following across diverse editing operations. Quantitative results establish the overall and category-level trends, after which qualitative examples illustrate how those gains appear in generated images.

Quantitative results. Table 1 reports ImgEdit results (Ye et al., 2025). CycleA alone achieves 3.88 Overall, improving BAGEL by 0.46 and outperforming the listed standalone editing baselines. Within the BAGEL-based variants, CycleA is strongest on Adjust and provides large gains on Replace, Remove, and Background, highlighting the benefit of semantic feedback for precise object and attribute edits. Table 1 also compares CycleA with RecA. CycleA obtains 3.88 Overall versus 3.75 for RecA, with stronger results on Adjust, Replace, Remove, and Background. RecA performs better on Add, Style, Hybrid, and Action, showing a different category-level profile. On GEdit-Bench-EN, CycleA improves BAGEL by 0.67 in semantic consistency and 0.56 Overall (Tab. 2). In the matched Qwen-VL-Max control in Appendix A.1, BAGEL, GEN only, and CycleA score 3.81, 3.92, and 4.13 Overall. This separates an alignment gain beyond continued training in that setting, but the different judge precludes decomposing the main GPT-4.1 gain.

Qualitative results. We compare CycleA with the baseline BAGEL (Deng et al., 2025) in Fig. 1. Further qualitative comparisons are provided in the Appendix. In the displayed examples, CycleA more accurately follows several editing instructions than the BAGEL baseline. These examples visually illustrate the instruction-following improvements reflected in the quantitative results.

4.3 Ablation Studies

We conduct ablation studies to validate the contribution of each design choice in CycleA, with results summarized in Tab. 3. All ablation variants are trained with the same hyper-parameters as the default configuration unless otherwise specified.

VQA vs. captioning. We compare VQA supervision with captioning supervision, where the frozen understanding pathway is supervised through open-ended caption prediction instead of targeted

Model	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attri.	Overall
Harmon	**0.99**	0.87	0.69	0.86	0.45	0.51	0.73
SD3-Medium	**0.99**	0.94	0.72	**0.89**	0.33	0.60	0.74
FLUX.1-dev	**0.99**	0.85	0.74	0.79	0.21	0.48	0.68
Playground-v3	**0.99**	**0.95**	0.72	0.82	0.50	0.54	0.76
BAGEL	**0.99**	0.93	**0.80**	0.86	**0.51**	0.63	**0.79**
CycleA	**0.99**	0.94	0.76	0.87	**0.51**	**0.68**	**0.79**

Table 4: GenEval results on text-to-image generation. All best values, including ties, are in **bold**. Underlined CycleA values improve over BAGEL.

question answering. As shown in Tab. 3, captioning achieves an overall score of 3.71, falling short of VQA-based CycleA. VQA questions target specific semantic aspects of the image, while captions tend to describe the image broadly without focusing on whether the intended content is correctly generated. Under the evaluated configuration, this result suggests that targeted VQA supervision is more effective than open-ended captioning for steering the generation branch.

Impact of noise level $t_{\max}$. We vary the upper bound of the noise range in the alignment step from $t_{\max} = 0.3$ to $t_{\max} = 1.0$, which covers the full noise schedule. As reported in Tab. 3, increasing $t_{\max}$ to 1.0 drops the overall score from 3.88 to 3.59. The degradation is most severe in Background and Replace, two categories that demand high fidelity to the original image. When $t_{\max}$ is large, the single-step prediction $\hat{z}_0$ deviates further from the clean latent, introducing errors in the pseudo features fed to the understanding branch. This behavior is consistent with noisier pseudo features providing a less faithful input to the frozen understanding pathway and weakening the semantic feedback received by the generation branch. The observed result supports using a restricted noise range so that the estimated clean latent remains close to the training image and yields stable understanding features.

Contribution of VQA and feature losses. We disentangle the roles of L_{vqa} and L_{feat} in Tab. 3. Using only L_{feat} yields 3.71, while only L_{vqa} yields 3.77. Both improve over the baseline, showing that each loss contributes to editing quality. L_{feat} provides short-gradient-path feature supervision, while L_{vqa} supplies gradients for the semantic properties queried in the image QA data. Combining both losses gives 3.88, higher than either single-loss variant, demonstrating that feature-level and semantic supervision are complementary.

4.4 Further Results and Analysis

We next examine text-to-image performance and initial cross-architecture transfer.

T2I generation. CycleA obtains 84.23 versus BAGEL's 84.03 on DPGBench (Hu et al., 2024). On GenEval (Ghosh et al., 2023) in Table 4, it ties the best Overall score of 0.79 while improving BAGEL on Two Object, Colors, and Color Attribution. The results show that editing post-training preserves text-to-image generation and strengthens several compositional attributes.

Discussion. Editing scores improve over BAGEL while text-to-image scores remain comparable. The GEN-only control and loss ablations support gains beyond continued training and contributions from both objectives, although downstream scores do not directly measure the internal mechanism. Nexus-Gen provides preliminary evidence on a second architecture.

5 Conclusion

In this work, we propose Cycle Alignment (CycleA), a post-training paradigm that closes the understanding-generation loop in unified multimodal models. CycleA constrains the generated results in the native feature space of the understanding branch, and steers generation through task-verifiable VQA supervision, enabling semantically accurate image editing. Extensive experiments demonstrate the effectiveness of CycleA on editing tasks, with consistent overall improvements over strong baselines across most categories while preserving the original generation capability.

AI USE STATEMENT

In this work, we used GPT-4.1 and Qwen-VL-Max as automatic evaluators under the corresponding benchmark protocols. Generative AI tools also assisted with translation, implementation of the proposed method, interpretation of evaluation results, and language refinement. We did not use generative AI tools to formulate mathematical claims or proofs, or to design experiments. Other required-disclosure tasks are not applicable. The authors reviewed all AI-assisted work and take responsibility for the final content of this paper.

REPRODUCIBILITY STATEMENT

The method and training objectives are described in Sections 3.2 and 3.3. The training data, optimization configuration, benchmarks, evaluation protocols, and ablation studies are documented in Section 4. Appendix A.1 provides additional training controls, evaluator consistency analyses, text-to-image evaluation, cross-architecture results, and qualitative analyses. Appendix A.2 documents the projector architecture, alternating training procedure, parameter-freezing policy, optimization settings, and hyperparameters.

REFERENCES

Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021.

Jiuhai Chen, Zhiyang Xu, Xichen Pan, Yushi Hu, Can Qin, Tom Goldstein, Lifu Huang, Tianyi Zhou, Saining Xie, Silvio Savarese, et al. Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset. *arXiv preprint arXiv:2505.09568*, 2025a.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025b.

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. In *International Conference on Learning Representations*, 2023.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Tsu-Jui Fu, Wenze Hu, Xianzhi Du, William Wang, Yinfei Yang, and Zhe Gan. Guiding instruction-based image editing via multimodal large language models. In *International Conference on Learning Representations*, 2024.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 2023.

Ziyu Guo, Renrui Zhang, Chengzhuo Tong, Zhizheng Zhao, Rui Huang, Haoquan Zhang, Manyuan Zhang, Jiaming Liu, Shanghang Zhang, Peng Gao, et al. Can we generate images with cot? let's verify and reinforce image generation step by step. *arXiv preprint arXiv:2501.13926*, 2025.

Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment. *arXiv preprint arXiv:2403.05135*, 2024.

Yuzhou Huang, Liangbin Xie, Xintao Wang, Ziyang Yuan, Xiaodong Cun, Yixiao Ge, Jiantao Zhou, Chao Dong, Rui Huang, Ruimao Zhang, et al. Smartedit: Exploring complex instruction-based image editing with multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

Xingjian Leng, Jaskirat Singh, Yunzhong Hou, Zhenchang Xing, Saining Xie, and Liang Zheng. Repa-e: Unlocking vae for end-to-end tuning of latent diffusion transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Han Li, Xinyu Peng, Yaoming Wang, Zelin Peng, Xin Chen, Rongxiang Weng, Jingang Wang, Xunliang Cai, Wenrui Dai, and Hongkai Xiong. Onecat: Decoder-only auto-regressive model for unified understanding and generation. *arXiv preprint arXiv:2509.03498*, 2025.

Hongxiang Li, Hongxu Chen, Chenyang Zhu, Xiaoshuang Huang, Jiayin Cai, Xiaolong Jiang, Yao Hu, and Long Chen. Spar: Semantic-pixel self-alignment and adaptive routing for unified multimodal models. In *European Conference on Computer Vision*, 2026.

Chao Liao, Liyang Liu, Xun Wang, Zhengxiong Luo, Xinyu Zhang, Wenliang Zhao, Jie Wu, Liang Li, Zhi Tian, and Weilin Huang. Mogao: An omni foundation model for interleaved multi-modal generation. *arXiv preprint arXiv:2505.05472*, 2025.

Bin Lin, Zongjian Li, Xinhua Cheng, Yuwei Niu, Yang Ye, Xianyi He, Shenghai Yuan, Wangbo Yu, Shaodong Wang, Yunyang Ge, et al. Uniworld-v1: High-resolution semantic encoders for unified visual understanding and generation. *arXiv preprint arXiv:2506.03147*, 2025.

Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024.

Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, et al. Step1x-edit: A practical framework for general image editing. *arXiv preprint arXiv:2504.17761*, 2025.

Zeyu Liu, Zanlin Ni, Yang Yue, Cheng Da, Huan Yang, Di Zhang, Kun Gai, and Gao Huang. Steering visual generation in unified multimodal models with understanding supervision. *arXiv preprint arXiv:2605.05781*, 2026.

Grace Luo, Jonathan Granskog, Aleksander Holynski, and Trevor Darrell. Dual-process image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17972–17983, 2025.

Shuailei Ma, Kecheng Zheng, Ying Wei, Wei Wu, Fan Lu, Yifei Zhang, Chen-Wei Xie, Biao Gong, Jiapeng Zhu, and Yujun Shen. Learning visual generative priors without text. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025a.

Yiyang Ma, Xingchao Liu, Xiaokang Chen, Wen Liu, Chengyue Wu, Zhiyu Wu, Zizheng Pan, Zhenda Xie, Haowei Zhang, Xingkai Yu, et al. Janusflow: Harmonizing autoregression and rectified flow for unified multimodal understanding and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025b.

Shrestha Basu Mallick and Logan Kilpatrick. Gemini 2.0: Flash, flash-lite and pro. *Google for Developers Blog*, 2025.

Weijia Mao, Zhenheng Yang, and Mike Zheng Shou. Unirl: Self-improving unified multimodal models via supervised and reinforcement learning. *arXiv preprint arXiv:2505.23380*, 2025.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research Journal*, 2024.

Yiming Qin, Bomin Wei, Jiaxin Ge, Konstantinos Kallidromitis, Stephanie Fu, Trevor Darrell, and XuDong Wang. Chain-of-visual-thought: Teaching vlms to see and think better with continuous visual tokens. *arXiv preprint arXiv:2511.19418*, 2025.

Liao Qu, Huichao Zhang, Yiheng Liu, Xu Wang, Yi Jiang, Yiming Gao, Hu Ye, Daniel K Du, Zehuan Yuan, and Xinglong Wu. Tokenflow: Unified image tokenizer for multimodal understanding and generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025.

Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.

Haochen Wang, Anlin Zheng, Yucheng Zhao, Tiancai Wang, Zheng Ge, Xiangyu Zhang, and Zhaoxiang Zhang. Reconstructive visual instruction tuning. In *International Conference on Learning Representations*, 2025.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024.

Cong Wei, Zheyang Xiong, Weiming Ren, Xeron Du, Ge Zhang, and Wenhu Chen. Omniedit: Building image editing generalist models through specialist supervision. In *International Conference on Learning Representations*, 2025.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 2022.

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025a.

Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, et al. Omnigen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025b.

Size Wu, Wenwei Zhang, Lumin Xu, Sheng Jin, Zhonghua Wu, Qingyi Tao, Wentao Liu, Wei Li, and Chen Change Loy. Harmonizing visual representations for unified multimodal understanding and generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025c.

Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Ji Xie, Trevor Darrell, Luke Zettlemoyer, and XuDong Wang. Reconstruction alignment improves unified multimodal models. *arXiv preprint arXiv:2509.07295*, 2025a.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. In *International Conference on Learning Representations*, 2025b.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025.

Yang Ye, Xianyi He, Zongjian Li, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, Li Yuan, et al. Imgedit: A unified image editing dataset and benchmark. *Advances in Neural Information Processing Systems*, 2025.

Qifan Yu, Wei Chow, Zhongqi Yue, Kaihang Pan, Yang Wu, Xiaoyang Wan, Juncheng Li, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. Anyedit: Mastering unified high-quality image editing for any idea. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. *arXiv preprint arXiv:2410.06940*, 2024.

Hong Zhang, Zhongjie Duan, Xingjun Wang, Yuze Zhao, Weiyi Lu, Zhipeng Di, Yixuan Xu, Yingda Chen, and Yu Zhang. Nexus-gen: Unified image understanding, generation, and editing via prefilled autoregression in shared embedding space. *arXiv preprint arXiv:2504.21356*, 2025a.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems*, 2023a.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2023b.

Zechuan Zhang, Ji Xie, Yu Lu, Zongxin Yang, and Yi Yang. Enabling instructional image editing with in-context generation in large scale diffusion transformer. *Advances in Neural Information Processing Systems*, 2025b.

Haozhe Zhao, Xiaojian Ma, Liang Chen, Shuzheng Si, Rujie Wu, Kaikai An, Peiyu Yu, Minjia Zhang, Qing Li, and Baobao Chang. Ultraedit: Instruction-based fine-grained image editing at scale. *Advances in Neural Information Processing Systems*, 2024.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. *arXiv preprint arXiv:2510.11690*, 2025.

Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. In *International Conference on Learning Representations*, 2025.

A Appendix

A.1 Additional Experiments and Analysis

Additional Training Ablations

We add two matched controls to distinguish Cycle Alignment from ordinary post-training and generic auxiliary supervision. GEN only uses the same 2,000 training steps, learning rate of 4×10^{-5}, 200-step warmup, generation data, and trainable parameters as CycleA while optimizing only the generation loss. Shuffled QA uses the same 2,000-step alternating training configuration as CycleA, but reassigns each complete question-answer pair to another image within the same parquet file. We use a fixed per-file derangement, so no image retains its original QA pair while the image order remains unchanged. All generated outputs are retained for evaluation without post hoc filtering. All variants in Table 5 are evaluated once with Qwen-VL-Max under the same ImgEdit protocol. The benchmark field `compose` is reported as Hybrid to match the main paper.

GEN only improves the Overall score from 3.81 to 3.92, showing that continued generation training contributes part of the gain. CycleA further raises it to 4.13 and outperforms GEN only in seven of nine categories. Shuffled QA instead scores 3.67, which is 0.46 below CycleA and also below BAGEL and GEN only. Its largest degradations relative to CycleA occur on Hybrid and Action. Because the training form is unchanged while the image-QA correspondence is removed, this result supports that CycleA benefits from semantically matched QA supervision rather than merely from adding an auxiliary cross-entropy objective.

Model	Add	Adj.	Ext.	Repl.	Rem.	Bkg.	Style	Hybrid	Action	Overall
BAGEL	4.01	3.70	3.03	4.05	3.82	3.79	4.09	3.85	4.53	3.81
GEN only	4.08	3.88	**3.18**	4.09	4.60	3.62	4.19	**4.19**	3.81	3.92
Shuffled QA	4.03	3.93	2.68	3.96	4.43	3.81	4.20	1.09	2.44	3.67
CycleA	**4.26**	**4.17**	3.12	**4.29**	**4.69**	**4.29**	**4.21**	3.96	**4.61**	**4.13**

Table 5: Matched training controls on ImgEdit evaluated with Qwen-VL-Max. GEN only uses only the generation loss, while Shuffled QA preserves CycleA's training configuration but removes image-QA correspondence. All post-trained variants use 2,000 training steps, and no generated outputs are filtered. Adj., Ext., Repl., Rem., and Bkg. denote Adjust, Extract, Replace, Remove, and Background. Best results are in **bold**.

Independent Evaluator Validation

We separately examine whether the advantage of CycleA depends on the automatic evaluator. We repeat the BAGEL and CycleA comparison with Qwen-VL-Max through the DashScope OpenAI compatible API using the model identifier `qwen-vl-max`. The original image, edited image, instruction, category specific prompt, and evaluated samples are identical to those used by the original ImgEdit protocol. All decoding parameters use the DashScope defaults.

Evaluator	Model	Add	Adj.	Ext.	Repl.	Rem.	Bkg.	Style	Hybrid	Action	Overall
GPT-4.1	BAGEL	3.81	3.59	1.58	3.85	3.16	3.39	4.51	**2.67**	4.25	3.42
	CycleA	**4.09**	**3.96**	**1.95**	**4.34**	**4.29**	**3.86**	**4.71**	2.60	**4.50**	**3.88**
	Δ (8/9)	+0.28	+0.37	+0.37	+0.49	+1.13	+0.47	+0.20	−0.07	+0.25	**+0.46**
Qwen-VL-Max	BAGEL	4.01	3.70	3.03	4.05	3.82	3.79	4.09	3.85	4.53	3.81
	CycleA	**4.26**	**4.17**	**3.12**	**4.29**	**4.69**	**4.29**	**4.21**	**3.96**	**4.61**	**4.13**
	Δ (9/9)	+0.25	+0.47	+0.09	+0.24	+0.87	+0.50	+0.12	+0.11	+0.08	**+0.32**

Table 6: Category-level consistency across automatic evaluators on ImgEdit. Δ denotes the CycleA improvement over BAGEL within each evaluator, and the value in parentheses reports the number of improved categories. Adj., Ext., Repl., Rem., and Bkg. denote Adjust, Extract, Replace, Remove, and Background. Best results within each evaluator are in **bold**. Absolute scores are not compared across evaluators.

CycleA improves over BAGEL under both evaluators. GPT-4.1 gives an Overall gain of 0.46 across eight categories, while Qwen-VL-Max gives a gain of 0.32 across all nine categories. Both evaluators identify strong improvements on Remove, Background, and Adjust. The consistent within-evaluator ordering shows that the overall advantage of CycleA is retained when the automatic evaluator is changed.

Qualitative Results Compared with RecA

We compare CycleA with RecA on single-turn instruction editing from ImgEdit. The results are shown in Fig. 3. CycleA produces more faithful edits on instructions requiring precise attribute changes and object removal.

Additional Qualitative Results

Figures 4 and 5 provide additional comparisons with BAGEL. In these selected examples, CycleA more faithfully follows several instructions and exhibits fewer cases of attribute misbinding, object omission, or unintended background modification.

Text-to-Image Generation

To compare text-to-image performance before and after editing post-training, we evaluate CycleA on DPGBench (Hu et al., 2024) and GenEval (Ghosh et al., 2023). Fine-grained GenEval results are reported in the main paper, while the complementary DPGBench comparison is provided here.

Model	DPGBench
SDXL	74.65
SD-3.5-Medium	83.79
FLUX.1-dev	84.00
OmniGen2	83.57
UniWorld-V1	83.86
BAGEL	84.03
CycleA	**84.23**

Table 7: Text-to-image generation results on DPGBench. Best result is in **bold**.

Figure 6 shows some text-to-image generation samples from CycleA. The model produces high-quality images across diverse prompts and can correctly generate unusual attribute compositions that are rare in training data, such as a green teddy bear or purple broccoli, where many models default to the typical color.

Initial Architecture Transfer

As an initial test of architecture transfer, we also apply CycleA to Nexus-Gen (Zhang et al., 2025a), which connects an autoregressive LLM with a diffusion decoder through a shared embedding space. As reported in Tab. 8, the Overall score increases from 2.68 to 2.74 after adapting the cross-space mapper. Because this is a single point estimate on one additional architecture, we treat it as preliminary evidence that the training path can be adapted beyond BAGEL. Nexus-Gen uses a generation design distinct from BAGEL's Mixture-of-Transformers, making this result a useful initial architecture-transfer check.

Model	Overall
Nexus-Gen	2.68
Nexus-Gen + CycleA	2.74

Table 8: Initial architecture-transfer result on ImgEdit.

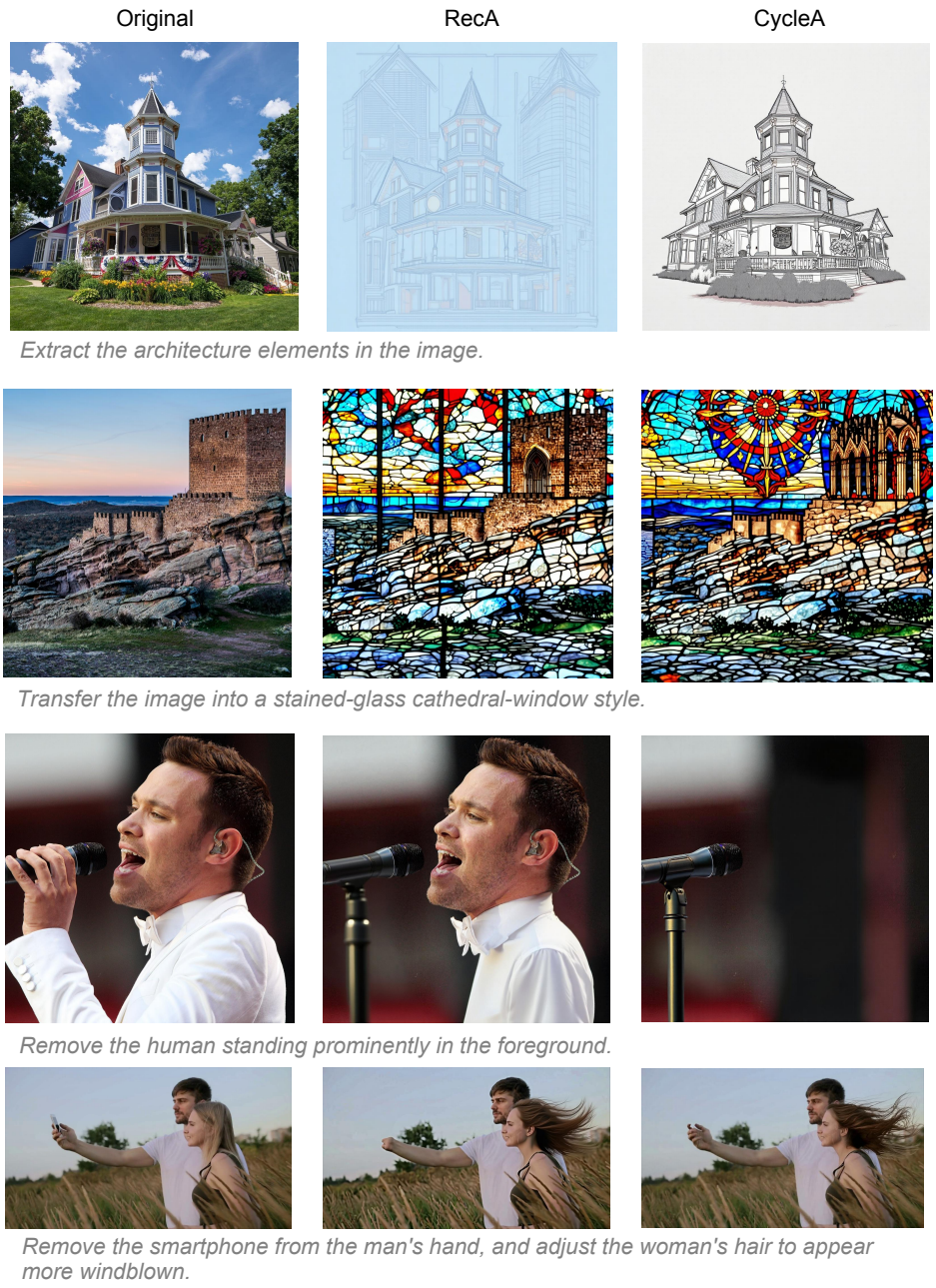

Figure 3: Qualitative comparison of instruction-based image editing between CycleA and RecA on ImgEdit.



Figure 4: Qualitative comparison of image editing results between CycleA and BAGEL. Red arrows highlight locations where the baseline produces editing errors.

Category-wise Analysis

CycleA delivers its largest gains on Add, Adjust, Replace, Remove, and Background, while Extract and Hybrid remain closer to the BAGEL baseline. Figure 7 illustrates representative examples from these more compositionally demanding categories. The category-wise pattern suggests that the semantic feedback is particularly effective for localized object and attribute changes, complementing the overall benchmark improvements.

A.2 Implementation Details

Architecture of the Projector

The projector g_ω bridges the VAE latent space and the understanding branch's visual feature space. It operates on each latent patch independently, mapping it to the feature dimension expected by the frozen understanding encoder's connector.

Dimensions. For BAGEL-7B-MoT, the VAE encoder produces latents with per-pixel dimension $d_z = 16$ and spatial compression factor 8. We group latent pixels into $p \times p$ patches ($p = 2$), yielding a per-patch input dimension of $p^2 \times d_z = 64$. The understanding branch uses a SigLIP-L ViT with feature dimension $d_f = 1152$, so the projector maps $\mathbb{R}^{64} \to \mathbb{R}^{1152}$.

Architecture. We use a 3-block MLP with hidden dimension $d_f = 1152$, GELU activation σ, and LayerNorm. Defining a per-block map $\phi(\mathbf{h}) = \sigma(\mathrm{LN}(\mathbf{h}))$ (with bias folded into each linear layer), the projector is

$$g_\omega(\mathbf{x}) = W_3\, \phi\big(W_2\, \phi(W_1 \mathbf{x})\big), \tag{11}$$

where $W_1 \in \mathbb{R}^{1152\times 64}$, $W_2, W_3 \in \mathbb{R}^{1152\times 1152}$, and LN is LayerNorm. The total parameter count is $\sim$2.7M, negligible relative to the 7B backbone. Weights are Xavier-uniformly initialized. Biases are zero.

Patch grid alignment. The VAE latent grid (at $1/8$ resolution) and the SigLIP patch grid (at $1/14$ resolution, 16×16 for 224-pixel input) generally differ in spatial size. To obtain a regression target at the projector's native grid resolution, we bilinearly resize the *real* SigLIP features *down* to the VAE latent grid before computing the pre-training loss. During Cycle Alignment, the projector output is fed directly into the frozen understanding branch at the VAE latent grid resolution (controlled by the target grid size $g = 14$), with no further resizing.

Why an MLP. The mapping is per-patch and position-independent, motivating a shared MLP without cross-patch interaction and keeping the projector lightweight.

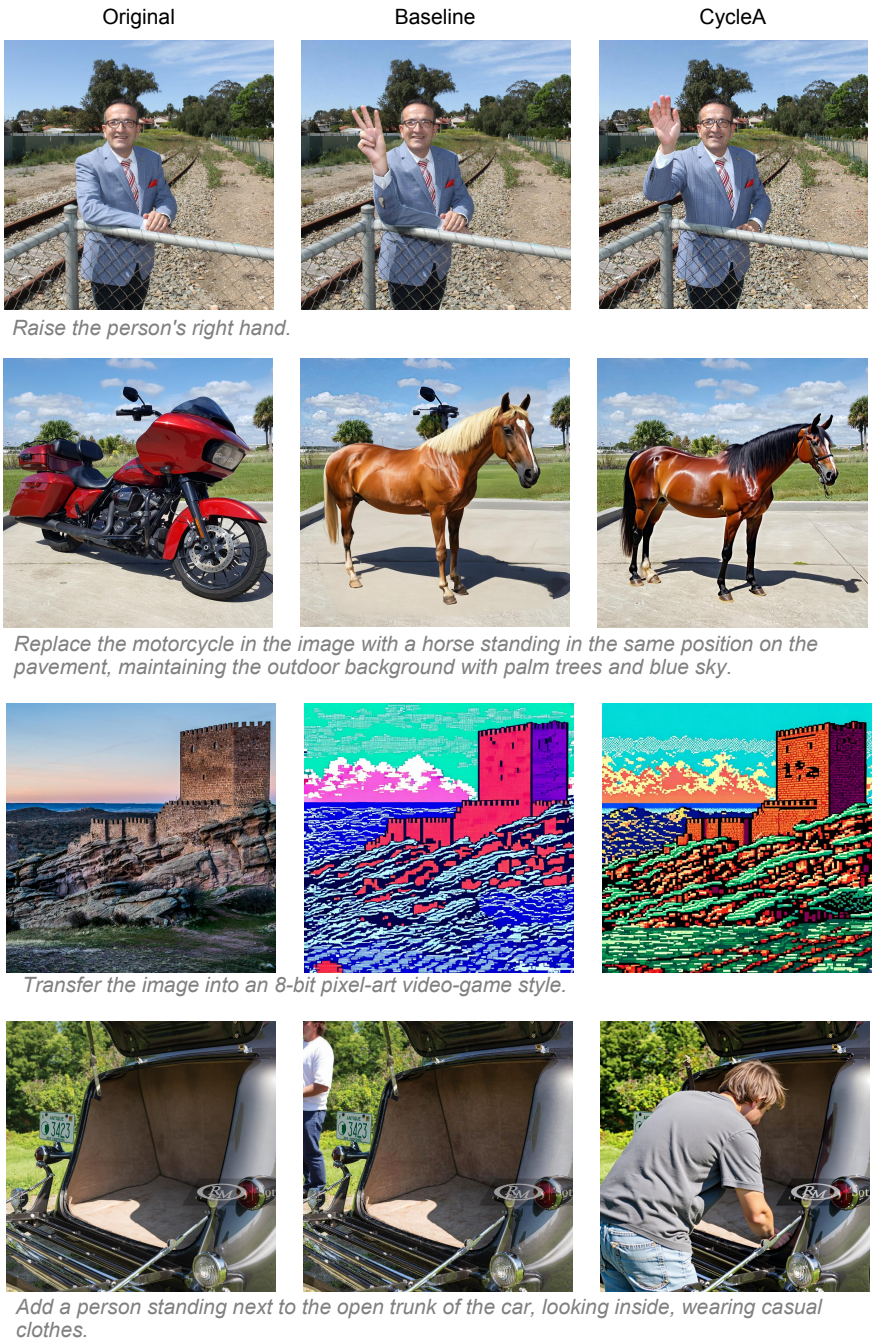

Figure 5: Additional qualitative comparisons between BAGEL and CycleA on image editing.

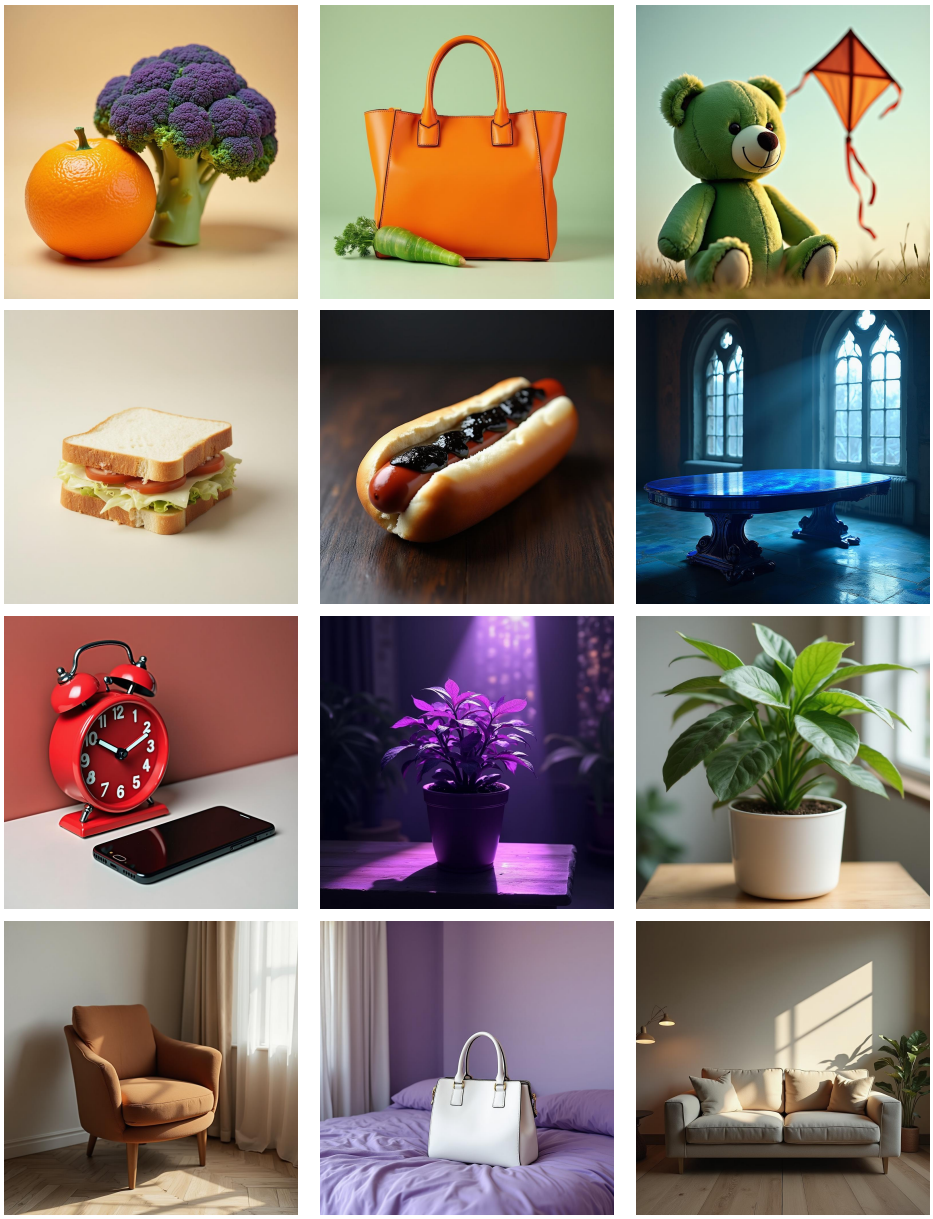

Figure 6: Text-to-image generation results from CycleA.



Figure 7: Representative editing results for the compositionally demanding Extract and Hybrid categories.

Pre-training and freezing. The projector is pre-trained with an MSE loss against real SigLIP features for 5K steps (AdamW, lr 10^{-4}, batch size 8, both VAE and ViT frozen), then frozen for the entire Cycle Alignment training. If the projector were trainable during CycleA, it could absorb VQA gradients and collapse into a mapping that satisfies the loss without remaining in the understanding branch's native feature space. This would shortcut the understanding branch instead of passing gradients through it. Freezing forces the VQA gradient to pass through the frozen understanding branch back to the generation branch, which is the intended feedback path.

CYCLE ALIGNMENT TRAINING DETAILS

Two-way alternating schedule. Cycle Alignment (CycleA) alternates two step kinds in a strict round-robin schedule. Each step performs exactly one forward pass and one backward pass.

- **ALIGN** (odd steps). Cycle CE and ViT-feature MSE are applied to a CoVT-style $(\mathbf{x}, q, a)$ sample with restricted small-t regime $t \sim \mathcal{U}(0, t_{\max})$.
- **GEN** (even steps). Standard flow-matching MSE is applied to a text-to-image sample $(\mathbf{x}, c)$ with full-range logit-normal timestep $t \sim \mathrm{LogitNormal}(0, 1)$ and loss weight λ_{mse}.

This design avoids the inflated-loss and mismatched-gradient-scale problem that arises when a flow-matching MSE and a cycle CE are summed inside the same step. The projector's bilinear alignment is a low-pass operation, so the cycle CE gradient is near-zero on high-frequency latent components and must be complemented by a dedicated GEN step that maintains full-band generation capability.

ALIGN step formulation. Given an image $\mathbf{x}$ with question q and answer a, we (i) encode $\mathbf{x}$ to a clean latent $\mathbf{z}_0$, (ii) sample $t \sim \mathcal{U}(0, t_{\max})$, (iii) form $\mathbf{z}_t = (1-t)\mathbf{z}_0 + t\,\boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and (iv) run the gen branch once to obtain the velocity prediction $\mathbf{v}_\theta(\mathbf{z}_t, c)$ and the single-step clean-latent estimate

$$\hat{\mathbf{z}}_0 \;=\; \mathbf{z}_t - t\,\mathbf{v}_\theta(\mathbf{z}_t, c). \tag{12}$$

The small-t regime keeps $\hat{\mathbf{z}}_0$ close to the true $\mathbf{z}_0$, which is critical for the projector g_ω to produce a faithful ViT-feature proxy. We then map $\hat{\mathbf{z}}_0$ through g_ω to obtain fake ViT features $\tilde{\mathbf{F}}_{\mathrm{vit}} = g_\omega(\mathrm{patchify}(\hat{\mathbf{z}}_0))$ at the target grid size g, and feed them into the frozen understanding branch with (q, a) to compute the token-level VQA cross-entropy loss $\mathcal{L}_{\mathrm{vqa}}$. The ALIGN loss is

$$\mathcal{L}_{\mathrm{ALIGN}} \;=\; \lambda_{\mathrm{align}}\,\mathcal{L}_{\mathrm{vqa}} \;+\; \lambda_{\mathrm{feat}}\,\mathcal{L}_{\mathrm{feat}}, \tag{13}$$

where $\mathcal{L}_{\mathrm{feat}} = \|\tilde{\mathbf{F}}_{\mathrm{vit}} - \mathbf{F}^{\star}_{\mathrm{vit}}\|_2^2$ is the feature-alignment MSE against real ViT features $\mathbf{F}^{\star}_{\mathrm{vit}} = \mathcal{A}(\mathrm{ViT}(\mathbf{x}_{224}), g, g)$, computed under `no_grad` with the image resized to 224×224.

GEN step formulation. GEN uses the standard BAGEL flow-matching objective with full-range logit-normal t. The velocity target is $\mathbf{v}_{\mathrm{tgt}} = \boldsymbol{\epsilon} - \mathbf{z}_0$, and the per-token generation loss is

$$\mathcal{L}_{\mathrm{gen}} \;=\; \big\|\, \mathbf{v}_\theta(\mathbf{z}_t, c) - \mathbf{v}_{\mathrm{tgt}} \,\big\|_2^2 \tag{14}$$

The loss is averaged across tokens and GPUs before scaling. This gives $\mathcal{L}_{\mathrm{GEN}} = \lambda_{\mathrm{mse}}\,\mathcal{L}_{\mathrm{gen}}$. GEN shares the LLM backbone and generation experts with ALIGN, but uses an independent forward path and loss formulation. GEN maintains full-band generation capability, while ALIGN maintains cross-modal alignment.

Option-A freezing. To keep the cycle gradient from contaminating the understanding branch's parameters, we adopt an allowlist-based freezing policy. Only generation-side parameters are trainable, including the four lightweight adapters that bridge the VAE latent space and the LLM hidden space (the latent projection, the velocity regression projection, the timestep embedding, and the latent position embedding), and all generation-expert sub-layers of the MoT decoder (the attention $Q/K/V/O$ projections, the feed-forward MLP, and their respective normalization layers). All understanding-side parameters are frozen, namely the vision encoder, the vision-to-LLM connector, the understanding experts, the shared LLM layers, the language-modeling head, and the projector g_ω.

Therefore, BAGEL's MoT layers insert a gradient detachment on the understanding path by default, to prevent the understanding branch's backward pass from propagating updates into the shared backbone. In CycleA, however, the cycle gradient must flow from $\tilde{\mathbf{F}}_{\mathrm{vit}}$ *through* the understanding experts all the

way back to the generation branch's velocity prediction $\mathbf{v}_\theta$, and this detachment severs the feedback path. Therefore, we explicitly disable the in-layer detachment in CycleA mode so that autograd passes through the frozen understanding-expert parameters. The allowlist still guarantees that these parameters are not updated by the optimizer, while gradients are allowed to flow through them.

Optimization. We use AdamW (β_1=0.9, β_2=0.95) with a constant LR schedule, a 200-step warm-up, peak learning rate 4×10^{-5}, and gradient clipping at 1.0. Cycle Alignment training runs for 2k alternating steps on 6×A800-80G with FSDP full sharding and CPU offload. We disable EMA to avoid the host-RAM cost of a `FullStateDict` gather during checkpoint saves. All forward passes through the frozen VAE use `torch.no_grad`.

Cycle hyper-parameters. The key hyper-parameters of CycleA fall into three groups. They cover schedule settings ($t_{\max}$ and g), loss weights (λ_{align}, λ_{feat}, λ_{mse}), and optimizer configuration. The complete list is given in Tab. 9.

Hyper-parameter	Value
Cycle timestep upper bound $t_{\max}$	0.3
Projector target grid size g	14
λ_{align} (VQA CE)	1.0
λ_{feat} (feature-alignment MSE)	1.0
λ_{mse} (GEN MSE)	1.0
Warm-up steps	200
Peak LR	4×10^{-5}
Batch size (per step)	1

Table 9: Cycle Alignment hyper-parameters.