GENERATION BEFORE RECONSTRUCTION FOR STABLE END-TO-END LATENT GENERATIVE MODELING

Anonymous authors
Paper under double-blind review

ABSTRACT

Latent generative models typically train a VAE for reconstruction and then fit a generative model on the frozen latent space. Since reconstruction-optimized latents are not necessarily generation-friendly, jointly training both models is an appealing alternative. However, direct end-to-end training remains challenging, as it is prone to latent collapse and faces a generation–reconstruction conflict. We revisit this problem and identify two key insights. First, the entropy term in the Kullback-Leibler divergence (KL) objective is essential for preventing collapse. Second, reconstruction and generation exhibit asymmetric learning dynamics: reconstruction is fast and strongly supervised, whereas generation is slower and harder to optimize. Based on these observations, we introduce an entropy-preserving end-to-end objective for stable joint training and propose GenFirst, a simple generation-before-reconstruction strategy that first shapes a generation-friendly latent space and then refines reconstruction. GenFirst generalizes across exact-likelihood continuous autoregressive priors and SiT-based flow-matching priors. With our end-to-end objective and GenFirst, SiT achieves a gFID of 0.97 with CFG and 1.45 without CFG on ImageNet-256, while MMDiT reaches a GenEval score of 0.90 on text-to-image generation. We further extend the framework to shared visual latents for generation, representation learning, and reconstruction, as well as continuous text latents for unified text–image generation, demonstrating its applicability across generative priors, objectives and modalities.

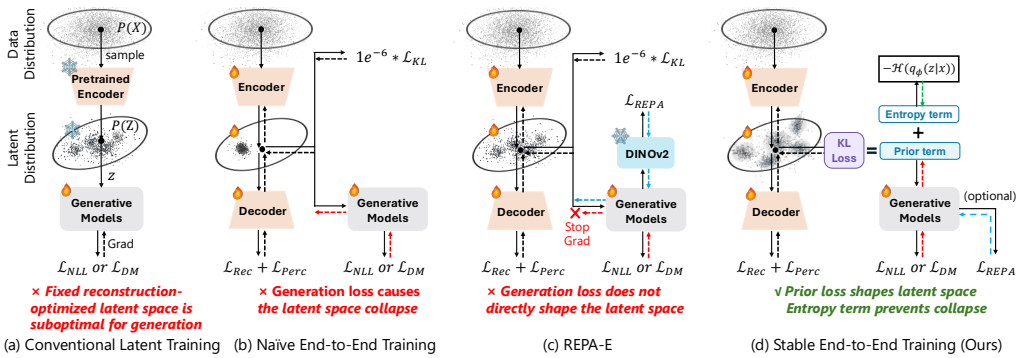


Figure 1: **Comparison of latent generative modeling paradigms.** (a) Conventional two-stage methods train generative models on a fixed, reconstruction-optimized latent space that may be suboptimal for generation. (b) Naïve end-to-end training allows the generative loss to update the encoder but causes latent collapse. (c) REPA-E avoids collapse by stopping generative gradients and reshaping the latent space through a representation-alignment loss, preventing the generative objective from directly shaping the latent space. (d) Our method enables direct end-to-end training: the prior loss shapes a generation-friendly latent space, while the entropy term prevents collapse.

1 INTRODUCTION

Latent generative models (Rombach et al., 2022; Peebles & Xie, 2023) have become a dominant paradigm for high-resolution image generation. Instead of modeling images directly in pixel space, they first use a variational autoencoder (VAE) Kingma & Welling (2013) or visual tokenizer to com-

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.

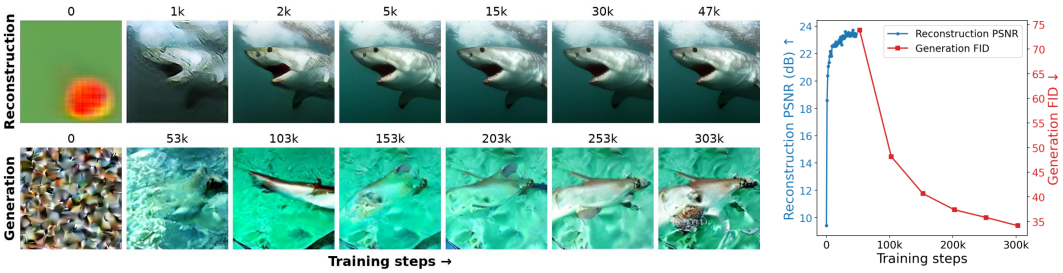


Figure 2: **Visualization of the asymmetric learning dynamics between reconstruction and generation.** Reconstruction reaches good image fidelity within much fewer training steps, while generation requires a longer optimization process to learn a sampleable latent distribution.

press raw pixels into a low-dimensional latent space and then train a generative model on this latent distribution. Such a two-stage pipeline greatly reduces modeling complexity and has enabled huge progress in image generation. However, recent works (Yao et al., 2025; Leng et al., 2025) show that a VAE latent space optimized for reconstruction is not necessarily optimal for generation. Reconstruction tends to preserve instance-level details, making the latent distribution overly dispersed and harder for generative models to fit and sample from.

A natural solution is to jointly train the VAE and generative model, allowing the generative objective to directly shape the latent space. REPA-E Leng et al. (2025) jointly trains diffusion models with a VAE, but observes severe latent collapse when gradients from the generative model are propagated to the VAE. It therefore stops the diffusion gradient and instead uses representation alignment for indirect supervision. SimFlow Zhao et al. (2025) avoid the latent collapse by fixing VAE variance during end-to-end training. Together, these designs highlight the promise of end-to-end latent generative modeling, while leaving its feasibility and optimization mechanism unclear.

In this paper, we therefore ask the first fundamental question: **Is direct end-to-end training of VAEs and generative models feasible?** REPA-E Leng et al. (2025) observed that naively propagating the generative loss through the VAE causes latent collapse. To answer this question, we revisit end-to-end training from the perspective of the VAE objective and identify a previously overlooked prior–entropy imbalance (§3.1). LDM Rombach et al. (2022) established a practical VAE training recipe widely adopted by modern latent generative models, where a small KL weight weakly regularizes the latent space. The KL regularizer in the VAE objective consists of a prior-fitting term and a negative posterior-entropy term. The former encourages latent samples to match the Gaussian prior, whereas the latter preserves posterior uncertainty and prevents degeneracy. In naive end-to-end training, the added generative objective introduces a strong prior-fitting force, while posterior entropy remains suppressed by the inherited small KL weight. This imbalance drives the posterior variance toward zero and the latent representations toward a nearly constant value, ultimately causing latent collapse (§5.2). Our analysis turns latent collapse into a correctable imbalance in the objective: stable end-to-end training requires explicitly preserving posterior entropy and balancing it against prior fitting.

After resolving the collapse issue, end-to-end training still faces a generation–reconstruction conflict Yao et al. (2025): reconstruction favors preserving instance-specific details, whereas generation favors a smoother and more modelable latent distribution, potentially discarding information needed for faithful reconstruction. This results in a generation–reconstruction trade-off (§3.2).

Therefore, we ask a second fundamental question: **How can we mitigate the generation–reconstruction conflict to learn a latent space suited for generative modeling?** As shown in Figure 2, our experiments reveal asymmetric learning dynamics between reconstruction and generation (§3.3). Reconstruction improves rapidly under direct supervision, whereas generation requires longer optimization to learn a sampleable latent distribution. Strong reconstruction pressure early in training can therefore anchor the encoder to a reconstruction-oriented latent space that is difficult for the generative model to learn. Conversely, maintaining excessive generative pressure can limit the recovery of visual details needed for high-quality decoding. These observations suggest that the key is not only how two objectives are weighted but also when each objective shapes the latent space.

This observation motivates **GenFirst**, a generation-before-reconstruction strategy for our end-to-end training framework. Rather than first learning a reconstruction-oriented latent space, GenFirst allows the generative objective to shape the latent distribution early, under weak reconstruction

pressure and explicit entropy preservation. Once the latent distribution can be effectively modeled by the generative prior, reconstruction is progressively strengthened to recover visual details. This ordering aligns with the asymmetric learning dynamics of the two objectives: generation needs to shape the latent geometry early, while reconstruction, being faster, can be strengthened later.

In summary, our contributions are threefold: (1) we identify a prior–entropy imbalance as the key cause of latent collapse in direct end-to-end training and show that preserving posterior entropy is essential; (2) we reveal the asymmetric learning dynamics of generation and reconstruction and propose GenFirst to first shape a generation-friendly latent space and then refines reconstruction; (3) we validate GenFirst with both exact-likelihood continuous autoregressive priors Zheng et al. (2025c) and flow-matching priors Ma et al. (2024), achieving a gFID of 0.97 on ImageNet and a GenEval score of 0.90 on text-to-image generation. Beyond image generation, we extend the framework to shared visual latents for generation, representation learning, and reconstruction, as well as continuous unified text–image modeling, demonstrating its generality across priors and modalities.

2 Related Work

Detailed discussions of related works are provided in Appendix A.

Latent Generative Models. Latent generative models (Rombach et al., 2022; Bao et al., 2023; Zheng et al., 2023; Peebles & Xie, 2023; Ma et al., 2024; Sun et al., 2024; Li et al., 2024; Tian et al., 2024; Gu et al., 2025) use image tokenizers, such as VAEs Rombach et al. (2022), to compress images into compact latent spaces, reducing the complexity of generative modeling. However, recent studies (Yu et al., 2024; Yao et al., 2025; Leng et al., 2025) show that reconstruction-oriented latent spaces are not necessarily optimal for generation. Recent works improve latent representations through representation alignment (Yu et al., 2024; Yao et al., 2025; Leng et al., 2025), but do not directly study the stable joint optimization of the latent encoder and the generative model.

End-to-End Generative Modeling. Prior work explores joint optimization of VAEs and generative models through different formulations. LSGM Vahdat et al. (2021) jointly trains VAEs with score-based generative models, while REPA-E Leng et al. (2025) enables end-to-end tuning by stopping diffusion gradients and introducing representation alignment. Several works (Tschannen et al., 2024; Zheng et al., 2025c; Zhao et al., 2025; Chen et al., 2026a) jointly optimize alternative latent representations, such as normalizing flows, with generative priors. Unified Latents Heek et al. (2026) jointly learns bitrate-controlled latents before retraining a larger generation-oriented prior. Another line of work (Zhai et al., 2024; Wang et al., 2025a; Li & He, 2025; Yu et al., 2026; Ma et al., 2026a) removes VAEs and directly models pixels. However, these approaches do not provide a systematic understanding of direct end-to-end VAE and generative model training, especially regarding latent collapse, and the generation–reconstruction conflict. We revisit this problem from the VAE objective and propose GenFirst for stable end-to-end latent generative modeling.

3 Revisiting End-to-End Latent Generative Modeling

A standard latent generative model is trained in two stages. In the first stage, a VAE is trained to encode images into a continuous latent space. Given an image x, the encoder $q_\phi(z|x)$ predicts a mean μ and a variance σ^2, which define a Gaussian posterior $\mathcal{N}(\mu, \mathrm{diag}(\sigma^2))$. A latent vector z is then sampled from this posterior and decoded by D_ψ to reconstruct the image $\hat{x} = D_\psi(z)$. The VAE is optimized with a reconstruction loss, typically an ℓ_1 loss between x and $\hat{x}$, together with perceptual losses computed on $\hat{x}$, including LPIPS and GAN losses. In practice, a weak KL regularization term, usually with a very small weight such as 10^{-6} Rombach et al. (2022), is also applied to μ and σ^2 to mildly constrain the posterior distribution. The VAE training objective is

$$\mathcal{L}_{\mathrm{VAE}} = \lambda_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec}} + \lambda_{\mathrm{LPIPS}}\mathcal{L}_{\mathrm{LPIPS}} + \lambda_{\mathrm{GAN}}\mathcal{L}_{\mathrm{GAN}} + \lambda_{\mathrm{KL}}D_{\mathrm{KL}}(q_\phi(z|x)\|p_0(z)). \tag{1}$$

In the second stage, the pretrained VAE is frozen and a generative prior is trained on latents sampled from the encoder posterior. For an exact-likelihood generative prior $p_\theta(z)$, the prior loss is $\mathcal{L}_{\mathrm{prior}} = \mathbb{E}_{x,\, z\sim q_\phi(z|x)}\left[-\log p_\theta(z)\right]$. A natural end-to-end alternative is to jointly optimize the VAE and prior:

$$\mathcal{L}_{\mathrm{naive}} = \mathcal{L}_{\mathrm{VAE}} + \lambda_{\mathrm{prior}}\mathcal{L}_{\mathrm{prior}}. \tag{2}$$

While this lets the prior directly shape the latent space, it also changes the forces acting on the encoder posterior. Without proper balancing, the posterior can degenerate and cause latent collapse.

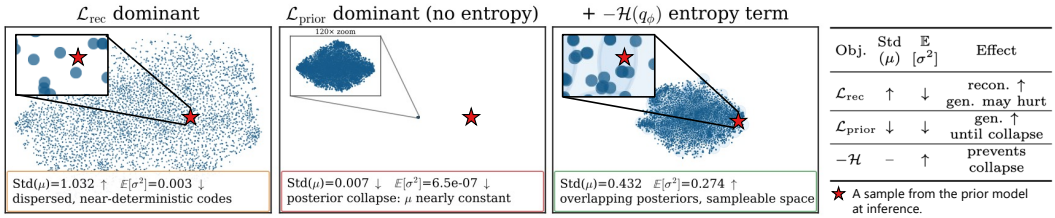


Figure 3: Qualitative effects of each objective on the latent posterior. Here $z = \mu_\phi(x) + \sigma_\phi(x)\epsilon$.

3.1 Prior-Entropy Imbalance Causes Latent Collapse

To understand why naive end-to-end training collapses, we revisit the KL term. For the standard Gaussian prior $p_0(z) = \mathcal{N}(0, I)$, the KL divergence can be decomposed as

$$D_{\text{KL}}(q_\phi(z|x)\|p_0(z)) = \mathbb{E}_{q_\phi}[\log q_\phi(z|x) - \log p_0(z)] = \underbrace{\mathbb{E}_{q_\phi}[-\log p_0(z)]}_{\text{prior fitting}} - \underbrace{\mathcal{H}(q_\phi(z|x))}_{\text{entropy}}. \quad (3)$$

The prior-fitting term pulls latent samples toward high-probability regions of the prior $p_0(z)$. The entropy term counteracts this effect by keeping the posterior from degenerating into point masses. As described above, naive end-to-end training treats the VAE KL loss as an engineering regularizer and keeps the same tiny weight following LDM Rombach et al. (2022), typically 10^{-6}. This overlooks the entropy-preserving role of the KL term when a learned prior loss is introduced. Once the generative objective is added, it introduces additional prior-fitting pressure and breaks the balance between prior fitting and posterior entropy: $\lambda_{\text{prior}} + \lambda_{\text{KL}} \gg \lambda_{\text{KL}}$. This prior–entropy imbalance drives the posterior variance toward zero and pushes the posterior means to collapse to nearly the same code, resulting in a collapsed latent space. Based on this analysis, we explicitly separate prior fitting and posterior entropy in Equation 2, and use $\mathcal{L}_{\text{prior}}$ to denote the generative objective:

$$\mathcal{L}_{\text{E2E}} = \lambda_{\text{rec}}\mathcal{L}_{\text{rec}} + \lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}} + \lambda_{\text{GAN}}\mathcal{L}_{\text{GAN}} + \lambda_{\text{prior}}\mathcal{L}_{\text{prior}} - \lambda_{\text{ent}}\mathcal{H}(q_\phi(z|x)). \quad (4)$$

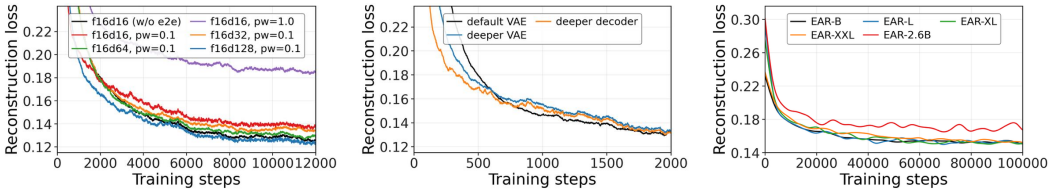
In this view, the entropy term is not a minor regularizer, but rather provides a counteracting force to learned prior fitting and prevents latent collapse.

To further illustrate how different losses affect the latent space, we visualize two statistics: $\text{Std}(\mu)$ and $\mathbb{E}[\sigma^2]$. $\text{Std}(\mu)$ measures how widely the posterior means are distributed across images, while $\mathbb{E}[\sigma^2]$ measures the average posterior uncertainty. As shown in Figure 3, different objectives push the latent posterior in different directions. Reconstruction increases $\text{Std}(\mu)$ but reduces uncertainty, improving fidelity while creating latent gaps. Without sufficient entropy, prior fitting reduces both statistics and collapses the posterior. Adding entropy increases posterior overlap and prevents collapse and improves sampleability. More discussion about latent collapse is provided in Appendix C.

3.2 Generation–Reconstruction Trade-off

After preventing latent collapse with entropy, end-to-end training still faces a generation–reconstruction conflict. We vary only λ_{prior} while keeping reconstruction-related losses unchanged. As shown in Figure 5(a), stronger prior fitting improves generation but degrades reconstruction: gFID improves from 120.60 to 8.06, while rFID worsens from 0.53 to 4.20. This indicates that generation and reconstruction prefer different operating points and cannot generally be optimized simultaneously with a fixed prior weight. We analyze this trade-off using the exact-likelihood AR prior from FARMER Zheng et al. (2025c), which allows direct control of the generative objective.

We further investigate whether increasing latent dimension, VAE size, or model capacity can alleviate this conflict (Figure 4); detailed settings are provided in Appendix B.1. Increasing the latent dimension relaxes the information bottleneck and allows stronger prior fitting, but only shifts the trade-off: even increasing the dimension from 16 to 128 only supports $\lambda_{\text{prior}} \approx 0.1$ before reconstruction degrades. Scaling the VAE cannot recover the lost reconstruction details from a more generation-oriented latent space and may even hurt reconstruction. Scaling the prior improves generation but introduces stronger latent shaping pressure, further degrading reconstruction. These results suggest that capacity scaling alone cannot resolve the generation–reconstruction conflict.



(a) Latent dim (b) VAE size (c) Prior size

Figure 4: Larger latent dimension, VAE and model size for balancing generation and reconstruction.

3.3 Loss Balancing Is Not Enough

The above results show that the latent dimension, VAE size, or model capacity do not eliminate the generation–reconstruction conflict. We next examine whether better loss balancing can alleviate it. As shown in Figure 5(a), the prior loss weight λ_{prior} strongly affects this trade-off. We therefore test several common weighting schemes with simple tuning, including constant weighting, cosine decay, and PI-style adaptive control Shao et al. (2020). However, Table 2 shows that none achieves a satisfactory balance, highlighting the difficulty of dynamically balancing the two objectives.

To understand this failure, we revisit the asymmetric learning dynamics between reconstruction and generation. As shown in Figure 2, reconstruction is a fast and strongly supervised mapping problem: given latent samples, the decoder receives direct supervision from the target images and can quickly recover visual details. In contrast, generation is a slower distribution-modeling problem: the prior must learn the global latent distribution before it can produce realistic samples. Reconstruction can reach good image fidelity within far fewer training steps, whereas generation requires substantially longer training to produce recognizable samples. Motivated by this asymmetry, we propose GenFirst, treating the conflict as an optimization-order rather than a loss-balancing problem. GenFirst first shapes a generation-friendly latent space and then strengthens reconstruction to recover details.

4 Method

4.1 Stable End-to-End Latent Generative Training

Given an image x, the encoder predicts the posterior parameters $\mu_\phi(x)$ and $\sigma_\phi(x)^2$, and samples $z = \mu_\phi(x) + \sigma_\phi(x) \odot \epsilon_z,\ \epsilon_z \sim \mathcal{N}(0, I)$. The decoder reconstructs the image from z. We optimize the end-to-end objective (Equation 4), where the form of $\mathcal{L}_{\text{prior}}$ depends on the generative model.

Continuous autoregressive prior. We instantiate the exact-likelihood prior with the continuous autoregressive model from FARMER Zheng et al. (2025c) and refer to the resulting end-to-end model as **EAR**. Given $z \in \mathbb{R}^{h\times w\times c}$, we flatten it into $N = hw$ continuous tokens and optimize

$$\mathcal{L}_{\text{prior}}^{\text{AR}} = \mathbb{E}_{x,\, z\sim q_\phi(z|x)} \left[-\log \prod_{i=1}^{N} p_\theta(z_i|z_{<i}, c) \right],\ p_\theta(z_i|z_{<i}, c) = \sum_{k=1}^{K} \pi_{i,k} \mathcal{N}\left(z_i; m_{i,k}, \text{diag}(s_{i,k}^2)\right). \tag{5}$$

Following (Tschannen et al., 2023; 2024), for each token z_i, a causal Transformer predicts a K-component diagonal Gaussian mixture based on prefix tokens $\{z_1, z_2, ..., z_{i-1}\}$ and condition c. Here, $\pi_{i,k}$, $m_{i,k}$, and $s_{i,k}^2$ denote the mixture weight, mean, and diagonal variance of component k, respectively. During end-to-end training, EAR not only fits the latent distribution, but also back-propagates the gradient to reshape the VAE latent space into one that is easier to model.

SiT prior. For SiT, we train a velocity field instead of an explicit likelihood. Given $\epsilon_t \sim \mathcal{N}(0, I)$ and $t \sim \mathcal{U}(0, 1)$, we construct $z_t = (1-t)\epsilon_t + tz,\ v = z - \epsilon_t$, and optimize

$$\mathcal{L}_{\text{prior}}^{\text{SiT}} = \mathbb{E}_{x,\, z\sim q_\phi(z|x),\, \epsilon_t,\, t} \left[\|v_\theta(z_t, t) - (z - \epsilon_t)\|_2^2 \right]. \tag{6}$$

Since SiT perturbs z with standard Gaussian noise, the latent value scale must remain bounded; otherwise, an excessively large z would make the injected noise relatively too small, allowing the model to exploit a shortcut to reduce the denoising loss. We therefore keep the standard VAE KL regularization with a small weight 10^{-6} for the SiT prior loss: $\mathcal{L}_{\text{E2E}}^{\text{SiT}} = \mathcal{L}_{\text{E2E}} + 10^{-6} D_{\text{KL}}\left(q_\phi(z|x) \,\|\, \mathcal{N}(0, I)\right)$. This small KL term is used only for latent-scale regularization and is distinct from the explicit posterior-entropy term in Equation 4, which provides the primary anti-collapse regularization.

4.2 GENERATION-BEFORE-RECONSTRUCTION TRAINING

Motivated by the asymmetric learning dynamics discussed in section 3.3, we propose **GenFirst**, a two-stage training schedule based on the same end-to-end objective in Equation 4. Throughout training, we keep the reconstruction, LPIPS, and GAN weights unchanged from standard VAE training, and tie the entropy weight to the prior weight, i.e., $\lambda_{\text{ent}} = \lambda_{\text{prior}}$. GenFirst uses a larger prior weight in the generation-first stage and a smaller one in the reconstruction-refinement stage:

$$\lambda_{\text{prior}}^{(1)} > \lambda_{\text{prior}}^{(2)}, \qquad \lambda_{\text{ent}}^{(s)} = \lambda_{\text{prior}}^{(s)}, \quad s \in \{1, 2\}. \tag{7}$$

In the generation-first stage, the stronger generative objective plays the main role in shaping a generation-friendly latent space, while reconstruction preserves image information and posterior entropy prevents latent collapse. In the reconstruction-refinement stage, we reduce the prior and entropy weights so that reconstruction receives greater relative influence and recovers visual details without substantially disrupting the generation-friendly latent space formed in the first stage. In practice, we use a longer generation-first stage for latent-space formation and a shorter reconstruction-refinement stage for recovering visual details. Unlike adaptive weighting, GenFirst does not merely adjust loss magnitudes; it explicitly separates latent-space formation from reconstruction refinement according to the asymmetric generation–reconstruction learning dynamics.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

We evaluate class-conditional generation on ImageNet Deng et al. (2009) at 256×256 and 512×512, and text-to-image generation on GenEval Ghosh et al. (2023) and DPG-Bench Hu et al. (2024). We instantiate our framework in two settings: (1) EAR, an exact-likelihood continuous AR from Zheng et al. (2025c) for ImageNet generation; (2) EiT, our diffusion-based instantiation using SiT Ma et al. (2024) for ImageNet generation and MMDiT Esser et al. (2024) for text-to-image generation. EAR jointly trains the VAE and AR prior from scratch in a single end-to-end stage. Following REPA-E Leng et al. (2025), EiT first jointly trains the VAE and diffusion prior, followed by a prior-only stage with the VAE frozen. For 512×512 generation, we reuse the VAE learned at 256×256 and train only the corresponding generative prior. Full training details are provided in Appendix D.1.

5.2 ANALYSIS

For ease of analysis, we use the exact-likelihood EAR as the default setting, and report gFID Heusel et al. (2017) and IS Salimans et al. (2016) without guidance. More analyses are in Appendix B.

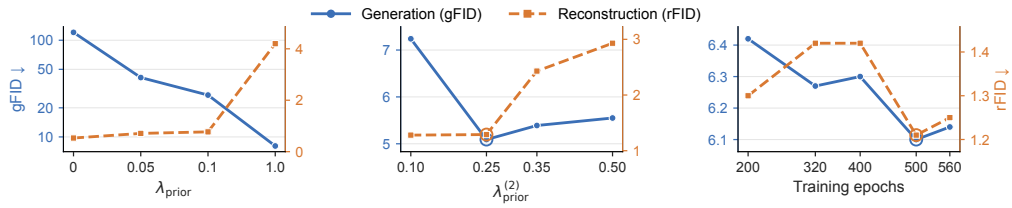
Does end-to-end training learn a generation-friendly latent space? We compare four protocols in Table 1: *Frozen VAE* is the standard two-stage baseline; *Naive E2E* propagates the generative loss to the VAE without explicit entropy preservation; *Stable E2E* uses our entropy-preserving objective and GenFirst strategy; and *Frozen E2E-trained VAE* freezes the VAE learned by our Stable E2E objective with GenFirst and trains a new prior. Naive E2E collapses for both EAR and SiT, whereas Stable E2E remains stable. The E2E-trained VAE substantially improves a newly trained prior, reducing gFID by 30.66 for EAR and 4.33 for SiT. This demonstrates that end-to-end training learns a more generation-friendly latent space. EAR and SiT differ in joint optimization: EAR directly models the latent density with exact likelihood, while SiT lacks an explicit likelihood over latents, making joint optimization less effective than training a new prior on frozen E2E-trained VAE.

Table 1: **Stable E2E learns generation-friendly latents.** EAR is trained until gFID converges, whereas SiT is evaluated after 80 epochs. *Collapsed* indicates NaN training loss.

Setting	EAR				SiT (80 epochs)			
	gFID↓	IS↑	rFID↓	PSNR↑	gFID↓	IS↑	rFID↓	PSNR↑
Frozen VAE	36.33	46.55	**0.53**	**26.18**	7.90	122.60	0.74	**25.67**
Naive E2E	294.90	1.48	0.70	25.11	Collapsed			
Stable E2E	**5.10**	**145.22**	1.26	21.93	92.38	13.94	**0.57**	25.13
Frozen E2E-trained VAE	5.67	137.39	1.26	21.93	**3.57**	**166.49**	**0.57**	25.13

Table 2: **Comparison of loss-balancing strategies.** *Collapsed* indicates NaN training loss.

Strategy	gFID (W/ CFG)↓	IS (W/ CFG)↑	rFID↓	PSNR↑
GenFirst	**2.10**	246.70	1.26	21.93
Constant	3.04	**251.2**	3.27	19.52
Cosine decay	5.45	136.11	0.66	23.87
PI adaptive	Collapsed			
Frozen VAE	6.97	147.04	**0.53**	**26.18**



(a) Fixed prior weight (b) Reconstruction-refinement-stage prior weight (c) Generation-first-stage duration

Figure 5: **Generation–reconstruction trade-off and GenFirst schedule ablations.**

Table 3: Comparison with REPA-E.

Method	gFID↓	IS↑	Prec.↑	Rec.↑
REPA	7.90	122.60	0.70	**0.65**
REPA-E (scratch)	4.34	154.30	0.75	0.63
REPA-E (SD-VAE init.)	4.07	161.80	0.76	0.62
REPA-E (VA-VAE init.)	3.46	159.80	0.77	0.63
EiT (scratch)	3.78	161.42	**0.78**	0.61
EiT (SD-VAE init.)	3.57	166.49	0.77	0.62
EiT (VA-VAE init.)	**2.79**	**175.40**	**0.78**	0.63

Table 4: Reconstruction comparison on ImageNet-256.

VAE	rFID↓	PSNR↑	SSIM↑	LPIPS↓
FLUX.1-dev VAE	0.18	31.59	0.90	0.05
E2E-trained VAE (Flux.1-dev VAE init.)	0.23	29.43	0.85	0.07
SD-VAE	0.74	25.67	0.72	0.13
E2E-trained VAE (SD-VAE init.)	0.57	25.13	0.71	0.14
VA-VAE	0.28	26.32	0.76	0.11
E2E-trained VAE (VA-VAE init.)	0.36	25.82	0.74	0.12

How does prior weighting affect the generation–reconstruction trade-off? Figure 5(a) shows that increasing the prior loss weight improves generation but hurts reconstruction. As λ_{prior} increases from 0 to 1.0, gFID improves from 120.60 to 8.06, while rFID worsens from 0.53 to 4.20. This reveals that stronger latent shaping improves generation but sacrifices reconstruction fidelity.

Can simple loss balancing strategies resolve the conflict? We compare GenFirst with three representative weighting baselines in the EAR setting: constant weighting, cosine decay, and PI-based adaptive control Shao et al. (2020); detailed settings are provided in Appendix D.2. As shown in Table 2, different weighting strategies reach different trade-off points: constant weighting improves generation at the cost of reconstruction, cosine decay favors reconstruction, and PI-controlled run becomes numerically unstable in our setting. GenFirst achieves the best generation quality with a gFID of 2.10 and maintains an rFID of 1.26. These results suggest that it is difficult to solve the conflict by simply tuning or scheduling the prior weight; the optimization order is critical.

Effect of GenFirst schedule. We ablate the reconstruction-refinement-stage prior weight and generation-first-stage duration. As shown in Figure 5(b), reducing $\lambda^{(2)}_{\text{prior}}$ from 0.5 to 0.25 improves generation and reconstruction, while further reducing it to 0.1 improves reconstruction but degrades generation. We set $\lambda^{(2)}_{\text{prior}}$ as 0.25. Figure 5(c) shows that varying the generation-first-stage duration from 200 to 560 epochs yields stable performance, with 500 epochs achieving the best trade-off.

Comparison with REPA-E. We compare EiT, which jointly optimizes SiT Ma et al. (2024) and the VAE using our end-to-end objective with GenFirst, with REPA Yu et al. (2024) and REPA-E Leng et al. (2025) in Table 3 under the same 80-epoch prior-only training budget. Compared with REPA-E, EiT consistently achieves lower gFID across VAE initializations: 3.78 vs. 4.34 from scratch, 3.57 vs. 4.07 with SD-VAE initialization, and 2.79 vs. 3.46 with VA-VAE initialization. These results show that our end-to-end formulation with GenFirst further improves generation beyond REPA-E.

Effect on VAE reconstruction. As shown in Table 4, E2E-trained VAEs retain competitive reconstruction quality with moderate degradation across different VAE architectures. For example, VA-VAE changes from 0.28 to 0.36 rFID and from 26.32 to 25.82 PSNR. End-to-end training sacrifices only a small amount of reconstruction fidelity to learn more generation-friendly latents.

5.3 Main Results

Class-conditional generation. As shown in Tables 5 and 6, GenFirst consistently improves both autoregressive and diffusion-based generation. EAR achieves a gFID of 2.10 with CFG using only 312M parameters. EiT reaches a gFID of **0.97** with CFG and **1.45** without CFG on ImageNet 256×256. At 512×512, EAR and EiT achieve gFID scores of 1.87 and 1.23, respectively, showing that the learned latent space transfers effectively to higher-resolution generation.

Table 5: **Class-conditional generation on ImageNet 256×256.** *: with low-temperature sampling.

Method	Epochs	#Params	W/o CFG				W/ CFG			
			gFID↓	IS↑	Prec.↑	Rec.↑	gFID↓	IS↑	Prec.↑	Rec.↑
Latent Diffusion										
DiT (Peebles & Xie, 2023)	1400	675M	9.62	121.5	0.67	0.67	2.27	278.2	**0.83**	0.57
MaskDiT (Zheng et al., 2023)	1600	675M	5.69	177.9	0.74	0.60	2.28	276.6	0.80	0.61
SiT (Ma et al., 2024)	1400	675M	8.61	131.7	0.68	0.67	2.06	270.3	0.82	0.59
MDTv2 (Gao et al., 2023)	1080	675M	–	–	–	–	1.58	314.7	0.79	0.65
REPA (Yu et al., 2024)	800	675M	5.78	158.3	0.70	0.68	1.29	306.3	0.79	0.64
VA-VAE (Yao et al., 2025)	800	675M	2.17	205.6	0.77	0.65	1.35	295.3	0.79	0.65
DDT (Wang et al., 2025b)	400	675M	6.27	154.7	0.68	**0.69**	1.26	310.6	0.79	0.65
RAE (Zheng et al., 2025a)	800	839M	1.51	242.9	0.79	0.63	1.13	262.6	0.78	**0.67**
End-to-End Latent Generative Models										
SimFlow (Zhao et al., 2025)	160	1.4B	13.72	105.2	0.67	0.62	2.15	276.8	0.83	0.57
REPA-E (Leng et al., 2025)	800	675M	1.69	219.3	0.77	0.67	1.12	302.9	0.79	0.66
EAR (Ours, w/o GenFirst)	500	312M	5.17	147.5	0.70	0.59	3.04	251.2	0.73	0.55
EAR (Ours)	640	312M	3.83*	164.0*	0.65	0.64	2.10	246.7	0.72	0.59
EiT (Ours)	80	675M	2.79	175.4	0.78	0.63	1.38	271.3	0.79	0.63
EiT (Ours)	480	675M	1.60	216.8	0.78	0.65	0.99	270.7	0.79	**0.67**
EiT (Ours)	800	675M	**1.45**	226.7	0.79	0.65	**0.97**	283.4	0.79	**0.67**

Method	#Params	W/ CFG	
		gFID↓	IS↑
Latent Autoregressive			
GIVT (Tschannen et al., 2023)	1.67B	2.59	–
LlamaGen (Sun et al., 2024)	1.4B	2.34	253.9
VAR (Tian et al., 2024)	2.0B	1.73	**350.2**
MAR (Li et al., 2024)	943M	1.55	303.7
xAR (Ren et al., 2025)	1.1B	**1.24**	301.6
Pixel Space			
ADM (Dhariwal & Nichol, 2021)	554M	3.94	215.8
PixelFlow (Chen et al., 2025b)	677M	1.98	282.1
PixNerd (Wang et al., 2025a)	700M	2.15	297.0
SiD2 (Hoogeboom et al., 2024)	–	**1.38**	–
TARFlow (Zhai et al., 2024)	1.4B	4.69	–
JetFormer (Tschannen et al., 2024)	2.8B	6.64	–
FARMER (Zheng et al., 2025c)	1.9B	3.60	269.2
JiT (Li & He, 2025)	2B	1.82	292.6
PixelDiT (Yu et al., 2026)	797M	1.61	292.7
DeCo (Ma et al., 2026a)	682M	1.69	**304.0**
DiP (Chen et al., 2026b)	631M	1.79	281.9
PixelGen (Ma et al., 2026b)	676M	1.83	293.6

Table 6: **Class-conditional generation on ImageNet 512×512.**

Method	#Params	gFID↓	IS↑
DiT (Peebles & Xie, 2023)	674M	3.04	240.8
SiT (Ma et al., 2024)	674M	2.62	252.2
REPA (Yu et al., 2024)	675M	2.08	274.6
MAR (Li et al., 2024)	481M	1.73	279.9
Unified Latents Heek et al. (2026)	–	1.31	–
DDT (Wang et al., 2025b)	675M	1.28	**305.1**
EDM2 (Karras et al., 2024b)	1.5B	1.25	–
RAE (Zheng et al., 2025a)	839M	**1.13**	259.6
EAR (Ours)	940M	1.87	280.0
EiT (Ours)	675M	1.23	286.1

Table 7: **Text-to-image generation.** * denotes using Qwen3-1.7B instead of CLIP as text encoder.

Method	#Params	GenEval↑	DPG↑
Emu3-Gen Wang et al. (2024)	8B	0.54	80.60
FLUX.1-dev black-forest labs (2024)	4.8B+12B	0.66	–
SD3-Medium Esser et al. (2024)	5.5B+2B	0.74	84.08
MetaQuery-XL Pan et al. (2025)	7B	0.80	82.05
Janus-Pro-7B Chen et al. (2025c)	7B	0.80	84.19
BLIP3-o-8B Chen et al. (2025a)	8B	0.84	81.60
FLUX.2-dev Labs (2025)	24B+32B	0.87	87.57
Qwen-Image Wu et al. (2025a)	7B+20B	0.87	**88.32**
EiT (Ours)	0.1B+1.3B	**0.90**	82.60
EiT (Ours)*	1.7B+1.3B	0.89	84.65

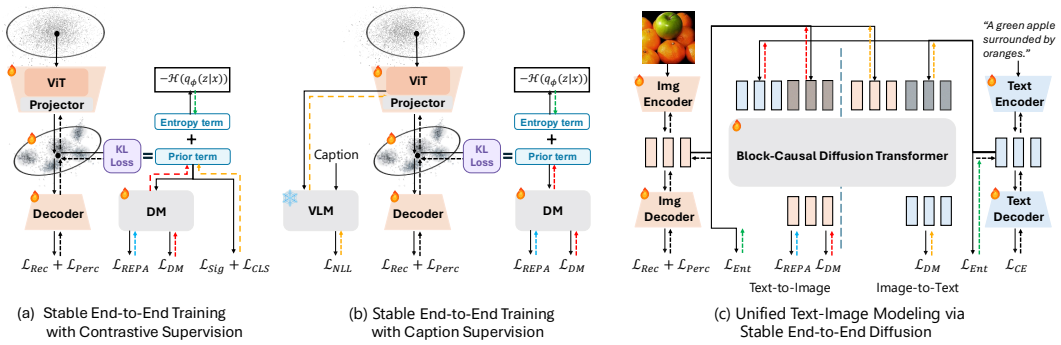
Text-to-image generation. With a 1.3B MMDiT, EiT achieves **0.90** GenEval score Ghosh et al. (2023), outperforming several larger models, and reaches 82.60 DPG-Bench score Hu et al. (2024), further improved to 84.65 with Qwen3-1.7B Yang et al. (2025) as the text encoder.

Training efficiency. Figure 6(a) shows that E2E-trained latents substantially accelerate generative modeling. On ImageNet, GenFirst reaches the final SiT Ma et al. (2024) and REPA Yu et al. (2024) performance with at least 70× and 26× fewer training steps, respectively, and matches REPA-E Leng et al. (2025) with approximately 2× less training. The same trend holds for text-to-image generation, where both EiT variants surpass the 200K-step baseline by 80K steps.



(a) Training convergence (b) Qualitative results

Figure 6: (a) **End-to-end training using GenFirst accelerates convergence across generation tasks.** (b) **Top:** ImageNet samples from EiT and EAR. **Bottom:** text-to-image comparisons with the FLUX.1-dev VAE (left) and our E2E-trained VAE (right) under the same training setting.



(a) Stable End-to-End Training with Contrastive Supervision

(b) Stable End-to-End Training with Caption Supervision

(c) Unified Text-Image Modeling via Stable End-to-End Diffusion

Figure 7: **Further extensions of stable end-to-end latent learning.** (a–b) Shared visual latent learning with representation supervision. (c) Unified continuous text–image latent modeling.

Table 8: **Shared latent space for generation and representation learning.**

Setting	rFID↓	PSNR↑	GenEval↑	DPG↑	Top-1 Acc.↑
Qwen3-VL ViT	–	–	–	–	79.64
E2E	0.42	27.79	0.89	83.70	77.71
E2E + SigLIP	0.39	27.64	**0.90**	**83.76**	79.25
E2E + VLM-NLL	0.46	27.19	0.89	83.49	**81.69**

Table 9: **Unified text–image modeling with continuous latents via stable end-to-end diffusion.**

Setting	Text-to-image		Image-to-text		
	GenEval↑	DPG↑	CIDEr↑	BLEU-4↑	SPICE↑
Frozen text VAE + frozen image VAE	0.80	63.96	20.03	5.36	8.03
E2E-trained text VAE + frozen image VAE	0.82	68.24	30.19	7.46	11.12
Jointly E2E-trained text VAE + image VAE	**0.87**	**72.68**	**33.28**	**7.76**	**11.81**

Qualitative results. Figure 6(b) shows that EAR and EiT produce coherent ImageNet samples with ine-grained details, while replacing FLUX.1-dev VAE black-forest labs (2024) with our E2E-trained VAE improves text-to-image perceptual quality, structural plausibility, and prompt alignment under the same MMDiT and training data. More qualitative results are provided in Appendix E.

5.4 Further Extensions

Shared latent space for generation and representation learning. We replace the FLUX.1-dev encoder with a Qwen3-VL ViT and introduce either latent-level SigLIP or VLM-NLL supervision during end-to-end training (Figure 7(a–b)). As shown in Table 8, end-to-end training with VLM-NLL supervision improves ImageNet linear-probe accuracy from 77.71 to 81.69, exceeding the pretrained Qwen3-VL ViT (79.64), while retaining strong generation quality (0.89 GenEval). This shows that end-to-end latent learning can jointly support generation, representation learning, and reconstruction. Implementation details and additional metrics are provided in Appendix D.3.

Unified text–image modeling. We further extend the framework to continuous text latents and jointly model text-to-image and image-to-text generation with a block-causal MMDiT (Figure 7(c)). As shown in Table 9, jointly training text and image VAEs improves GenEval from 0.80 to 0.87, DPG-Bench from 63.96 to 72.68, and image-to-text CIDEr from 20.03 to 33.28, demonstrating that stable end-to-end latent modeling extends beyond image latents to multimodal continuous representations. Full training details, additional metrics, and qualitative results are provided in Appendix D.4.

6 Conclusion

We revisit end-to-end training of VAEs and generative models and make two key observations. First, we find that latent collapse in naive end-to-end training is mainly caused by a prior-entropy imbalance. Second, we identify asymmetric learning dynamics underlying the generation–reconstruction conflict. Based on these observations, we introduce an entropy-preserving end-to-end objective that enables stable joint training and further propose GenFirst, a generation-before-reconstruction strategy that mitigates the generation–reconstruction conflict. Experiments show that our method learns generation-friendly latent spaces and improves the generation–reconstruction trade-off. We hope this work provides a simple and practical recipe for stable end-to-end latent generative modeling.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript editing, organization, formatting, coding, and language refinement. All AI-assisted suggestions were reviewed and verified by the authors. The authors take full responsibility for the final content of this work, including text or artifacts produced with AI assistance.

ETHICS STATEMENT

This work studies end-to-end latent generative modeling for images and text. Like other generative models, the generated images of our models may inherit biases from the publicly released third-party datasets used for training. Any downstream deployment should assess these risks.

REPRODUCIBILITY STATEMENT

The model formulations and training procedures are described in the main paper. Detailed architecture, optimization, and inference settings are provided in Appendix D.1; additional analyses and ablation settings are provided in Appendices B and D.2; and implementation details for the extensions are provided in Appendices D.3 and D.4.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22669–22679, 2023.

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

black-forest labs. Flux.1-dev. `https://huggingface.co/black-forest-labs/FLUX.1-dev`, 2024.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019.

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.

Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-\sigma: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. *arXiv preprint arXiv:2403.04692*, 2024.

Shoufa Chen, Chongjian Ge, Shilong Zhang, Peize Sun, and Ping Luo. Pixelflow: Pixel-space generative models with flow. *arXiv preprint arXiv:2504.07963*, 2025b.

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*, 2025c.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, and Ramakrishna Vedantam. `https://github.com/salaniz/pycocoevalcap`, 2020.

Yang Chen, Xiaowei Xu, Shuai Wang, Xinwen Zhang, Qiushi Guo, Tiezheng Ge, and Limin Wang. Mimflow: Integrating masked image modeling with normalizing flows for end-to-end image generation. *arXiv preprint arXiv:2606.26016*, 2026a.

Zhennan Chen, Junwei Zhu, Xu Chen, Jiangning Zhang, Xiaobin Hu, Hanzhen Zhao, Chengjie Wang, Jian Yang, and Ying Tai. Dip: Taming diffusion models in pixel space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36136–36146, 2026b.

Zhihong Chen, Xuehai Bai, Yang Shi, Chaoyou Fu, Huanyu Zhang, Haotian Wang, Xiaoyan Sun, Zhang Zhang, Liang Wang, Yuanxing Zhang, et al. Opengpt-4o-image: A comprehensive dataset for advanced image generation and editing. *arXiv preprint arXiv:2509.24900*, 2025d.

Wenda Chu, Bingliang Zhang, Jiaqi Han, Yizhuo Li, Linjie Yang, Yisong Yue, and Qiushan Guo. End-to-end autoregressive image generation with 1d semantic tokenizer. *arXiv preprint arXiv:2605.00503*, 2026.

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.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Shanghua Gao, Pan Zhou, Ming-Ming Cheng, and Shuicheng Yan. Mdtv2: Masked diffusion transformer is a strong image synthesizer. *arXiv preprint arXiv:2303.14389*, 2023.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36: 52132–52152, 2023.

Jiatao Gu, Tianrong Chen, David Berthelot, Huangjie Zheng, Yuyang Wang, Ruixiang Zhang, Laurent Dinh, Miguel Angel Bautista, Josh Susskind, and Shuangfei Zhai. Starflow: Scaling latent normalizing flows for high-resolution image synthesis. *arXiv preprint arXiv:2506.06276*, 2025.

Hongcan Guo, Qinyu Zhao, Yian Zhao, Shen Nie, Rui Zhu, Qiushan Guo, Feng Wang, Tao Yang, Hengshuang Zhao, Guoqiang Wei, et al. Continuous latent diffusion language model. *arXiv preprint arXiv:2605.06548*, 2026.

Jonathan Heek, Emiel Hoogeboom, Thomas Mensink, and Tim Salimans. Unified latents (ul): How to train your latents. *arXiv preprint arXiv:2602.17270*, 2026.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems*, 30, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Emiel Hoogeboom, Thomas Mensink, Jonathan Heek, Kay Lamerigts, Ruiqi Gao, and Tim Salimans. Simpler diffusion (sid2): 1.5 fid on imagenet512 with pixel-space diffusion. *arXiv preprint arXiv:2410.19324*, 2024.

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.

Zilong Huang, Qinghao Ye, Bingyi Kang, Jiashi Feng, and Haoqi Fan. Classification done right for vision-language pre-training. *Advances in Neural Information Processing Systems*, 37:96483–96504, 2024.

Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. *Advances in Neural Information Processing Systems*, 37:52996–53021, 2024a.

Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. Analyzing and improving the training dynamics of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24174–24184, 2024b.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Alexander Kolesnikov, André Susano Pinto, and Michael Tschannen. Jet: A modern transformer-based normalizing flow. *arXiv preprint arXiv:2412.15129*, 2024.

Black Forest Labs. FLUX.2: Frontier Visual Intelligence. `https://bfl.ai/blog/flux-2`, 2025.

Xingjian Leng, Jaskirat Singh, Yunzhong Hou, Zhenchang Xing, Saining Xie, and Liang Zheng. Repa-e: Unlocking vae for end-to-end tuning with latent diffusion transformers. *arXiv preprint arXiv:2504.10483*, 2025.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. *arXiv preprint arXiv:2511.13720*, 2025.

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37: 56424–56445, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. *arXiv preprint arXiv:2401.08740*, 2024.

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 Computer Vision and Pattern Recognition Conference*, pp. 7739–7751, 2025.

Zehong Ma, Longhui Wei, Shuai Wang, Shiliang Zhang, and Qi Tian. Deco: Frequency-decoupled pixel diffusion for end-to-end image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 43600–43610, 2026a.

Zehong Ma, Ruihan Xu, and Shiliang Zhang. Pixelgen: Pixel diffusion beats latent diffusion with perceptual loss. *arXiv e-prints*, pp. arXiv–2602, 2026b.

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. *arXiv preprint arXiv:2304.07193*, 2023.

Xichen Pan, Satya Narayan Shukla, Aashu Singh, Zhuokai Zhao, Shlok Kumar Mishra, Jialiang Wang, Zhiyang Xu, Jiuhai Chen, Kunpeng Li, Felix Juefei-Xu, et al. Transfer between modalities with metaqueries. *arXiv preprint arXiv:2504.06256*, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations*, volume 2024, pp. 1862–1874, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

Sucheng Ren, Qihang Yu, Ju He, Xiaohui Shen, Alan Yuille, and Liang-Chieh Chen. Beyond next-token: Next-x prediction for autoregressive visual generation. *arXiv preprint arXiv:2502.20388*, 2025.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in Neural Information Processing Systems*, 29, 2016.

Axel Sauer, Katja Schwarz, and Andreas Geiger. Stylegan-xl: Scaling stylegan to large diverse datasets. In *ACM SIGGRAPH 2022 conference proceedings*, pp. 1–10, 2022.

Huajie Shao, Shuochao Yao, Dachun Sun, Aston Zhang, Shengzhong Liu, Dongxin Liu, Jun Wang, and Tarek Abdelzaher. Controlvae: Controllable variational autoencoder. In *International conference on machine learning*, pp. 8655–8664. PMLR, 2020.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

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

Z-Image Team. Z-image: An efficient image generation foundation model with single-stream diffusion transformer. *arXiv preprint arXiv:2511.22699*, 2025.

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

Michael Tschannen, Cian Eastwood, and Fabian Mentzer. GIVT: Generative infinite-vocabulary Transformers. *arXiv:2312.02116*, 2023.

Michael Tschannen, André Susano Pinto, and Alexander Kolesnikov. Jetformer: An autoregressive generative model of raw images and text. *arXiv preprint arXiv:2411.19722*, 2024.

Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Advances in neural information processing systems*, 34:11287–11302, 2021.

Shuai Wang, Ziteng Gao, Chenhui Zhu, Weilin Huang, and Limin Wang. Pixnerd: Pixel neural field diffusion. *arXiv preprint arXiv:2507.23268*, 2025a.

Shuai Wang, Zhi Tian, Weilin Huang, and Limin Wang. Decoupled diffusion transformer. *arXiv preprint arXiv:2504.05741*, 2025b.

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.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025a.

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*, pp. 12966–12977, 2025b.

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. *arXiv preprint arXiv:2408.12528*, 2024.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *arXiv preprint arXiv:2506.15564*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

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*, pp. 15703–15712, 2025.

Lijun Yu, José Lezama, Nitesh B Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Vighnesh Birodkar, Agrim Gupta, Xiuye Gu, et al. Language model beats diffusion–tokenizer is key to visual generation. *arXiv preprint arXiv:2310.05737*, 2023.

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.

Yongsheng Yu, Wei Xiong, Weili Nie, Yichen Sheng, Shiqiu Liu, and Jiebo Luo. Pixeldit: Pixel diffusion transformers for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14273–14282, 2026.

Shuangfei Zhai, Ruixiang Zhang, Preetum Nakkiran, David Berthelot, Jiatao Gu, Huangjie Zheng, Tianrong Chen, Miguel Angel Bautista, Navdeep Jaitly, and Josh Susskind. Normalizing flows are capable generative models. *arXiv preprint arXiv:2412.06329*, 2024.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11941–11952. IEEE, 2023.

Bingliang Zhang, Wenda Chu, Yizhuo Li, Linjie Yang, Yisong Yue, Katherine Bouman, Yang Song, and Qiushan Guo. Speediff: Scalable pixel-anchored end-to-end latent diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35893–35903, 2026.

Qinyu Zhao, Guangting Zheng, Tao Yang, Rui Zhu, Xingjian Leng, Stephen Gould, and Liang Zheng. Simflow: Simplified and end-to-end training of latent normalizing flows. *arXiv preprint arXiv:2512.04084*, 2025.

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

Guangting Zheng, Yehao Li, Yingwei Pan, Jiajun Deng, Ting Yao, Yanyong Zhang, and Tao Mei. Hierarchical masked autoregressive models with low-resolution token pivots. *arXiv preprint arXiv:2505.20288*, 2025b.

Guangting Zheng, Qinyu Zhao, Tao Yang, Fei Xiao, Zhijie Lin, Jie Wu, Jiajun Deng, Yanyong Zhang, and Rui Zhu. Farmer: Flow autoregressive transformer over pixels. *arXiv preprint arXiv:2510.23588*, 2025c.

Hongkai Zheng, Weili Nie, Arash Vahdat, and Anima Anandkumar. Fast training of diffusion models with masked transformers. *arXiv preprint arXiv:2306.09305*, 2023.

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*, volume 2025, pp. 6446–6469, 2025.

A More Detailed Related Work

Latent Generative Models. Latent generative models (Rombach et al., 2022; Bao et al., 2023; Zheng et al., 2023; Peebles & Xie, 2023; Ma et al., 2024; Sun et al., 2024; Li et al., 2024; Tian et al., 2024; Zheng et al., 2025b; Gu et al., 2025) use image tokenizers, such as VAEs (Rombach et al., 2022; Yao et al., 2025), to compress images into a lower-dimensional latent space, which simplifies generative modeling while leaving high-frequency details to the tokenizer. However, recent studies (Yu et al., 2024; Yao et al., 2025; Leng et al., 2025) show that a VAE latent space optimized for reconstruction is not necessarily optimal for generation.

Representation Learning for Generation. Recent works explore how to obtain more generation-friendly representations. REPA Yu et al. (2024) improves diffusion models by aligning their intermediate hidden states with features from pretrained visual encoders such as DINOv2 Oquab et al. (2023). VA-VAE Yao et al. (2025) applies representation alignment to VAE training, while REPA-E Leng et al. (2025) further propagates the REPA loss through the diffusion model to align the VAE latent space in an end-to-end manner. Another line of work (Zhai et al., 2024; Tschannen et al., 2024; Zheng et al., 2025c; Li & He, 2025; Yu et al., 2026; Ma et al., 2026a; Chen et al., 2026b) revisits pixel representation and removes the VAE entirely. However, pixel-space generative models still face the challenge of modeling high-dimensional data and often remain less efficient than latent generative models.

End-to-End Generative Modeling. Several works (Chu et al., 2026; Zhang et al., 2026) study joint training of autoencoders and generative models. LSGM Vahdat et al. (2021) jointly trains a VAE with score-based generative models using a complex design. REPA-E Leng et al. (2025) jointly trains a VAE and a diffusion model by stopping the diffusion gradient to the VAE and introducing a representation alignment loss. JetFormer Tschannen et al. (2024) and FARMER Zheng et al. (2025c) jointly train lossless normalizing flows (Kolesnikov et al., 2024; Zhai et al., 2024) and autoregressive models Sun et al. (2024) to map the data distribution to a latent distribution modeled by AR priors. SimFlow and MIMFlow (Zhao et al., 2025; Chen et al., 2026a) jointly train a VAE and normalizing flows by fixing the VAE variance. Unified Latents Heek et al. (2026) first jointly learns bitrate-controlled latents with an encoder, diffusion prior, and decoder, then freezes the encoder and retrains a larger generation-oriented latent diffusion prior for efficient, high-quality synthesis. However, existing works still lack a systematic understanding of direct end-to-end training between VAEs and generative models, especially the cause of latent collapse, the conflict between generation and reconstruction in shaping the latent space, and a simple general training strategy. In this paper, we revisit end-to-end latent generative modeling from the VAE objective, identify a prior-entropy imbalance as the key cause of collapse, and propose a generation-before-reconstruction strategy.

B More Analysis

B.1 Effect of Latent, VAE, and Prior Capacity

Increasing the latent dimension. We increase the latent dimension to relax the information bottleneck Yao et al. (2025). As shown in Figure 4(a), larger latent dimensions tolerate stronger prior weights, but the improvement is limited: even increasing the dimension from 16 to 128 only supports $\lambda_{\text{prior}} \approx 0.1$ before reconstruction deteriorates. Moreover, a larger latent dimension makes the AR prior harder to model. Thus, increasing latent dimension shifts the trade-off but does not remove the generation–reconstruction conflict.

Scaling the VAE. We next test whether a stronger VAE can recover details from a more generation-friendly latent space. Compared with the default VAE ($\text{block} = 2$), we increase both encoder and

Table 10: **Scaling behavior of EAR.** The VAE is first jointly trained with an EAR-L prior under GenFirst and then frozen while varying the EAR model size and GMM capacity.

(a) EAR model scale.

Model	#Params	gFID↓	IS↑
B	174M	13.85	79.00
L	477M	5.67	137.39
XL	775M	3.91	163.71
XXL	1.4B	3.19	181.99

(b) GMM components.

GMM K	gFID↓	IS↑
32	6.08	132.70
64	5.67	137.39
128	5.64	137.24
1024	5.01	144.63
4096	5.00	144.99

Table 11: **Effect of an optional normalizing flow.** The VAE is frozen when training each AR prior.

Setting	gFID↓	IS↑
Frozen VAE + AR	110.10	14.79
Frozen VAE + NF + AR	36.33	46.55
Stable E2E VAE + AR	5.27	149.28
Stable E2E VAE + NF + AR	**5.12**	**151.17**

decoder depth ($\text{block} = 4$), or increase only the decoder depth. Figure 4(b) shows that scaling the VAE does not alleviate the conflict and can even hurt reconstruction, suggesting that VAE capacity alone is insufficient.

Scaling the generative prior. Finally, we scale the AR model to better fit complex latent distributions. Figure 4(c) shows that a larger prior improves generation but further degrades reconstruction, indicating stronger shaping pressure on the latent space. Thus, scaling the prior can amplify the dominance of generation rather than resolve the conflict.

B.2 Additional EAR Analysis

Scalability of the EAR prior. We first jointly train the VAE with an EAR-L prior using GenFirst, then freeze the VAE and train new EAR priors of different sizes from scratch on its latents using a lightweight 6-layer normalizing-flow adapter (Zhai et al., 2024; Tschannen et al., 2023). As shown in Table 10(a), scaling the prior from 174M to 1.4B parameters reduces gFID from 13.85 to 3.19 and increases IS from 79.00 to 181.99.

Effect of GMM capacity. Using the same frozen VAE, we fix EAR-L and vary the number of GMM components K. Table 10(b) shows that increasing K from 32 to 1024 improves gFID from 6.08 to 5.01 and IS from 132.70 to 144.63, with marginal gains beyond $K = 1024$.

Optional normalizing flow. Following FARMER Zheng et al. (2025c), we test whether an lightweight 6-layer normalizing flow (NF) is still needed for continuous AR modeling. Notably, for the frozen-VAE EAR baseline in Table 1, we follow FARMER and jointly train the AR prior with an NF adapter. This is because the pretrained MAR VAE exhibits a substantial distributional mismatch with the continuous AR prior. The NF adapter helps bridge this mismatch and provides a stronger and fairer frozen-VAE baseline. Table 11 shows that NF substantially improves the frozen-VAE baseline, but provides only a marginal gain after end-to-end VAE training. This suggests that end-to-end training already produces latents well suited to continuous AR modeling.

B.3 Additional EiT analysis

Fine-grained evaluation of text-to-image generation. We further report the detailed GenEval and DPG-Bench results of our two text-to-image model variants in Table 12. Version A performs the end-to-end phase on ImageNet, whereas Version B performs the end-to-end phase on the text-to-image training data. The detailed training configurations of both variants are provided in Appendix D.1. Version A performs slightly better overall, while the two variants remain close, indicating that end-to-end VAE training is effective on both ImageNet and text-to-image training data.

Ablation on GenFirst prior weights for SiT. We first use $(\lambda^{(1)}_{\text{prior}}, \lambda^{(2)}_{\text{prior}}) = (1, 0.1)$, which already achieves reasonable generation and reconstruction performance. Reducing the prior weights

Table 12: **Fine-grained evaluation of text-to-image generation.** We report category-wise results on GenEval Ghosh et al. (2023) and DPG-Bench Hu et al. (2024) for two model variants.

GenEval

Method	Single Obj.	Two Obj.	Counting	Color	Position	Color Attr.	Overall
EiT (Version A)	**1.00**	**0.94**	**0.83**	0.94	0.87	**0.82**	**0.90**
EiT (Version B)	0.99	0.93	0.83	**0.95**	**0.88**	0.78	0.89

DPG-Bench

Method	Global	Entity	Attribute	Relation	Other	Overall
EiT (Version A)	**88.30**	86.91	**88.75**	**89.64**	85.66	**82.60**
EiT (Version B)	84.26	**88.10**	87.16	88.87	**89.23**	82.06

Table 13: **Ablation on GenFirst prior weights for SiT.** We vary the prior weights in the generation-first and reconstruction-refinement stages.

$\lambda_{\text{prior}}^{(1)}$	$\lambda_{\text{prior}}^{(2)}$	rFID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	IS↑
1.00	0.10	0.61	24.57	0.70	0.15	4.44	154.25
0.10	0.01	0.57	25.13	0.71	0.14	**3.57**	**166.49**
0.01	0.01	**0.48**	**25.17**	**0.72**	**0.14**	3.84	161.40

to $(0.1, 0.01)$ further improves both generation and reconstruction, with rFID decreasing from 0.61 to 0.57 and FID decreasing from 4.44 to 3.57. This suggests that $(1, 0.1)$ still applies excessive prior pressure and does not reach a favorable generation–reconstruction trade-off. When the weights are further reduced to $(0.01, 0.01)$, reconstruction continues to improve, but generation degrades. We therefore use $(0.1, 0.01)$ as the default setting and leave finer tuning of the two-stage prior weights to future work due to computational constraints.

Effect of end-to-end training and REPA. We further disentangle the contributions of end-to-end training and REPA Yu et al. (2024) in Table 14. Using SD-VAE Rombach et al. (2022) with vanilla SiT-XL/2 Ma et al. (2024) yields an FID of 17.20, while applying REPA during the prior-only phase improves it to 7.90. Replacing the frozen SD-VAE with our end-to-end trained VAE already improves the vanilla SiT-XL/2 result to an FID of 8.09, even without REPA in either training phase. Following REPA-E Leng et al. (2025), introducing REPA during the end-to-end phase further improves the FID to 6.83. Applying REPA again during the subsequent prior-only phase further reduces the FID to 3.57. These results show that end-to-end VAE training and REPA provide complementary rather than conflicting benefits: end-to-end training reshapes the latent space to make it easier for the generative model to learn, while REPA further improves generation through representation alignment.

B.4 Full Benchmark Results

ImageNet 512×512**.** Table 15 reports the full comparison on ImageNet 512×512, including training epochs, model size, precision, and recall. EAR achieves a gFID of 1.87 with 940M parameters. EiT reaches a gFID of 1.63 after only 80 epochs and further improves to 1.23 after 400 epochs, outperforming prior latent diffusion models including DiT, SiT, REPA, DDT, and EDM2. Notably, the VAE used at 512×512 is directly transferred from end-to-end training at 256×256, showing that the learned generation-friendly latent space generalizes to higher-resolution generation.

Text-to-image generation. Table 16 provides the full GenEval category-wise results and DPG-Bench comparison. With only a 1.3B MMDiT, EiT achieves an overall GenEval score of 0.90, outperforming substantially larger models such as FLUX.2-dev, Qwen-Image, and BAGEL. The gains are particularly strong on spatial position (0.87) and attribute binding (0.82), suggesting that the learned latent space improves compositional generation rather than only low-level image fidelity. EiT obtains 82.60 on DPG-Bench with the original CLIP text encoder. Replacing CLIP with Qwen3-1.7B increases DPG-Bench to 84.65 and further improves color, position, and attribute scores, showing that stronger text representations provide complementary gains to end-to-end latent learning.

Table 14: **Effect of end-to-end training and REPA.** We disentangle the contributions of end-to-end latent learning and representation alignment Yu et al. (2024) to SiT Ma et al. (2024) generation.

End-to-end phase			Prior-only phase (80 epochs)		FID↓
VAE	Model	REPA	Model	REPA	
SD-VAE Rombach et al. (2022)	-	-	SiT-XL/2 Ma et al. (2024)	✗	17.20
SD-VAE Rombach et al. (2022)	-	-	SiT-XL/2 Ma et al. (2024)	✓	7.90
E2E-trained VAE (SD-VAE init.)	SiT-XL/2 Ma et al. (2024)	✗	SiT-XL/2 Ma et al. (2024)	✗	8.09
E2E-trained VAE (SD-VAE init.)	SiT-XL/2 Ma et al. (2024)	✓	SiT-XL/2 Ma et al. (2024)	✗	6.83
E2E-trained VAE (SD-VAE init.)	SiT-XL/2 Ma et al. (2024)	✓	SiT-XL/2 Ma et al. (2024)	✓	**3.57**

Table 15: **Full class-conditional comparison on ImageNet** 512×512**.** We report training epochs, parameter count, FID, Inception Score, precision, and recall.

Method	Epochs	#Params	gFID↓	IS↑	Prec.↑	Rec.↑
BigGAN-deep (Brock et al., 2019)	-	158M	8.43	177.9	0.88	0.29
StyleGAN-XL (Sauer et al., 2022)	-	-	2.41	267.8	0.77	0.52
VAR (Tian et al., 2024)	350	2.3B	2.63	303.2	-	-
MAGVIT-v2 (Yu et al., 2023)	1080	307M	1.91	324.3	-	-
MAR (Li et al., 2024)	800	481M	1.73	279.9	-	-
XAR (Ren et al., 2025)	800	608M	1.70	281.5	-	-
ADM (Dhariwal & Nichol, 2021)	871+210	731M	3.85	221.7	0.84	0.53
SiD2 (Hoogeboom et al., 2024)	-	-	1.50	-	-	-
PixNerd Wang et al. (2025a)	340	700M	2.84	245.6	0.80	0.59
JiT Li & He (2025)	600	2B	1.78	306.8	-	-
PixelDiT Yu et al. (2026)	850	797M	1.81	278.6	0.78	0.67
DeCo Ma et al. (2026a)	340	682M	2.22	290.0	0.80	0.60
DiT (Peebles & Xie, 2023)	600	674M	3.04	240.8	0.84	0.54
SiT (Ma et al., 2024)	600	674M	2.62	252.2	0.84	0.57
REPA (Yu et al., 2024)	200	675M	2.08	274.6	0.83	0.58
Unified Latents Heek et al. (2026)	-	-	1.31	-	-	-
DDT (Wang et al., 2025b)	356	675M	1.28	305.1	0.80	0.63
EDM2 (Karras et al., 2024b)	-	1.5B	1.25	-	-	-
RAE (Zheng et al., 2025a)	400	839M	1.13	259.6	0.80	0.63
STARFlow (Gu et al., 2025)	-	1.4B	3.00	-	-	-
SimFlow + REPA-E	160	1.4B	2.74	304.9	0.81	0.57
EAR (Ours)	320	940M	1.87	280.0	0.82	0.58
EiT (Ours)	80	675M	1.63	278.9	0.81	0.61
EiT (Ours)	400	675M	1.23	286.1	0.80	0.63

C A SHORTCUT SOLUTION IN NAIVE END-TO-END TRAINING

When the learned prior objective is strong while posterior entropy is weak, we observe that the encoder can reduce both the posterior variance and the diversity of posterior means. In the extreme case,

$$\sigma_\phi^2(x) \to 0, \qquad \mu_\phi(x) \approx c,$$

where c is nearly constant across different images. Since the posterior noise becomes very small, the decoder may still exploit small residual differences in $\mu_\phi(x)$ for reconstruction, even though the overall latent distribution is already highly concentrated.

For a diffusion or flow-matching prior, such a concentrated latent distribution also makes the denoising objective artificially easy. Consider the linear interpolation used by SiT,

$$z_t = (1-t)\epsilon_t + tz, \qquad v = z - \epsilon_t.$$

If the latent collapses to $z \approx c$, then

$$z_t \approx (1-t)\epsilon_t + tc, \qquad v \approx \frac{c - z_t}{1-t}.$$

Therefore, predicting the target velocity requires little knowledge of the image distribution: the model can approximately recover the target through a simple affine transformation of the noisy input around the nearly constant code c. The denoising loss can thus become small even though the generative model has not learned a meaningful latent distribution.

Table 16: **Full text-to-image comparison on GenEval and DPG-Bench.** † denotes prompt rewriting and * denotes replacing CLIP Radford et al. (2021) with Qwen3-1.7B Yang et al. (2025).

Method	#Params	GenEval							DPG-Bench
		Single-Obj	Two-Obj	Count	Color	Position	Attr	Overall	
PixArt Chen et al. (2024)	4.3B+0.6B	0.98	0.50	0.44	0.80	0.08	0.07	0.48	71.11
DALL-E 2 Ramesh et al. (2022)	-	0.94	0.66	0.49	0.77	0.10	0.19	0.52	-
Show-o Xie et al. (2024)	1.3B	0.95	0.52	0.49	0.82	0.11	0.28	0.53	-
Emu3-Gen Wang et al. (2024)	8B	0.98	0.71	0.34	0.81	0.17	0.21	0.54	80.60
SDXL Podell et al. (2024)	0.81B+2.6B	0.98	0.74	0.39	0.85	0.15	0.23	0.55	74.65
Janus Wu et al. (2025b)	1.3B	0.97	0.68	0.30	0.84	0.46	0.42	0.61	79.68
JanusFlow Ma et al. (2025)	1.3B	0.97	0.59	0.45	0.83	0.53	0.42	0.63	80.09
FLUX.1-dev black-forest labs (2024)	4.8B+12B	0.98	0.81	0.74	0.79	0.22	0.45	0.66	-
DALL-E 3 Betker et al. (2023)	-	0.96	0.87	0.47	0.83	0.43	0.45	0.67	83.50
SD3-Medium Esser et al. (2024)	5.5B+2B	0.99	0.94	0.72	0.89	0.33	0.60	0.74	84.08
Show-o2-7B Xie et al. (2025)	7B	1.00	0.87	0.58	0.92	0.52	0.62	0.76†	86.14
MetaQuery-XL Pan et al. (2025)	7B	-	-	-	-	-	-	0.80†	82.05
Janus-Pro-7B Chen et al. (2025c)	7B	0.99	0.89	0.59	0.90	0.79	0.66	0.80	84.19
Z-Image-Turbo Team (2025)	4B+6B	1.00	0.95	0.77	0.89	0.65	0.68	0.82	84.86
BLIP3-o-8B Chen et al. (2025a)	8B	-	-	-	-	-	-	0.84	81.60
FLUX.2-dev Labs (2025)	24B+32B	1.00	**0.99**	0.79	0.93	0.73	0.78	0.87	87.57
Qwen-Image Wu et al. (2025a)	7B+20B	0.99	0.92	**0.89**	0.88	0.76	0.77	0.87	**88.32**
BAGEL-7B Deng et al. (2025)	7B	0.98	0.95	0.84	0.95	0.78	0.77	0.88†	-
EiT (Ours)	0.1B+1.3B	**1.00**	0.94	0.83	0.94	0.87	0.82	**0.90**	82.60
EiT (Ours)*	1.7B+1.3B	0.99	0.94	0.73	**0.96**	**0.89**	**0.84**	0.89	84.65

This explains why preserving posterior entropy is important during end-to-end training. Increasing the entropy prevents $\sigma_\phi^2(x)$ from vanishing. Under non-trivial posterior noise, if all posterior means were also nearly identical, the latent samples of different images would become indistinguishable to the decoder. The reconstruction objective therefore encourages $\mu_\phi(x)$ to remain image-dependent and distributed across the latent space. Together, entropy preservation and reconstruction discourage both variance and mean collapse. Motivated by this observation, we tie the entropy weight to the prior weight in our default end-to-end objective, $\lambda_{\text{ent}} = \lambda_{\text{prior}}$. This keeps posterior-entropy preservation coupled to the strength of prior-fitting pressure and avoids the degenerate shortcut observed in naive end-to-end training.

D Implementation Details

D.1 Training and Inference Details

We provide the complete architecture, training, and inference configurations for EAR, EiT, and text-to-image EiT in Table 17,Table 18, and Table 19. Below, we summarize the training pipeline and the relationship between different resolutions and model variants.

EAR on ImageNet. The detailed EAR configuration is provided in Table 17. For ImageNet 256×256, we jointly train the MAR VAE Li et al. (2024) and the continuous AR prior Zheng et al. (2025c) from scratch using the two-stage GenFirst schedule. The resulting VAE is then reused for the resolution 512×512 experiment: we keep the VAE frozen and train a larger EAR prior on the higher-resolution latent sequences. Therefore, the resolution 512×512 experiment evaluates whether the latent space learned by GenFirst at resolution 256×256 transfers to higher-resolution generative modeling without further adapting the tokenizer. We follow FARMER Zheng et al. (2025c) for the continuous-AR parameterization and resampling-based classifier-free guidance.

EiT on ImageNet. The ImageNet EiT configuration is summarized in Table 18. At ImageNet 256×256, training consists of an end-to-end phase followed by a prior-only phase. During the end-to-end phase, the VAE and SiT are jointly optimized using the two-stage GenFirst schedule. We then freeze the resulting VAE and train a new SiT on its latent space. For resolution 512×512, we directly reuse and freeze the VA-VAE obtained from the resolution 256×256 end-to-end stage, and train only a new SiT at the higher resolution. We observe that the latent space changes rapidly during end-to-end training, which can leave the decoder insufficiently adapted to the updated latents. Therefore, for the ImageNet-256 EiT results in Table 5 only, we additionally fine-tune the VAE decoder for 10K steps after the end-to-end stage. This improves the 80-epoch unguided gFID from 3.00 to 2.79. This

Table 17: **Detailed training configuration of EAR on ImageNet.** EAR-256 jointly trains the VAE and AR prior with GenFirst. For EAR-512, we reuse and freeze the VAE obtained from EAR-256 and train only the AR prior at the higher resolution.

Configuration	EAR-256	EAR-512
Architecture		
VAE architecture	MAR VAE Li et al. (2024)	EAR-256 VAE (frozen)
Latent configuration	$f16d16$	$f16d16$
AR backbone	EAR-312M	EAR-940M
Transformer depth	24	36
Hidden dimension	1024	1280
Attention heads	16	20
GMM components K	64	1024
Training		
Trainable modules	VAE + AR prior	AR prior only
Training resolution	256×256	512×512
Generation-first stage	500 epochs	–
$\lambda_{\text{prior}}/\lambda_{\text{ent}}$	$1.0/1.0$	–
Reconstruction-refinement stage	140 epochs	–
$\lambda_{\text{prior}}/\lambda_{\text{ent}}$	$0.25/0.25$	–
Prior-only training	–	320 epochs
Global batch size	512	512
Optimizer	AdamW Loshchilov & Hutter (2017), $\beta_1, \beta_2 = 0.9, 0.999$	AdamW, $\beta_1, \beta_2 = 0.9, 0.999$
Learning rate	1×10^{-4}	1×10^{-4}
LR schedule	10-epoch linear warmup, then constant	10-epoch linear warmup, then constant
EMA rate	0.9999	0.9999
Class token drop (for CFG)	0.1	0.1
Inference		
Eval batch size	50	50
Guidance	resampling-based CFG Zheng et al. (2025c)	resampling-based CFG
Weight temperature (Propose stage)	0.9	0.9
Variance temperature (Propose stage)	0.9	0.9
Sample number from p_c/p_u	4/4	4/4
Weight temperature (Weigh stage)	-0.55	-0.5
Variance temperature (Weigh stage)	1.1	1.0
Resampling temperature	0.9	1.0
CFG scale	1.5	2.0

decoder fine-tuning is not used in the other EiT experiments. For the corresponding 80-epoch guided result, we use CFG scale 2.0 with a guidance interval of $[0.2, 0.85]$. Unless otherwise specified, the optimization and REPA settings follow REPA-E Leng et al. (2025); complete architecture, training, and inference configurations are provided in Table 18.

Text-to-image EiT. We consider two variants for obtaining an end-to-end trained FLUX.1-dev VAE black-forest labs (2024), as detailed in Table 19. In *Version A*, the VAE is jointly trained with SiT-XL/2 on ImageNet using the same GenFirst setting as class-conditional EiT. In *Version B*, the VAE is jointly trained with a smaller MMDiT (MMDiT-L/2) directly on the text-to-image training data using GenFirst. These variants compare a VAE shaped directly by text-conditioned generation with one shaped by class-conditional ImageNet generation. After the end-to-end phase, we freeze the resulting VAE and train the same MMDiT-XL/2 prior for both variants. We first train at 256×256 for 200K steps, continue at 512×512 for 80K steps, and finally fine-tune on the OpenAI-4o dataset Chen et al. (2025d) at 512×512 for 40K steps. Complete architecture, optimization, and inference settings are provided in Table 19.

D.2 Details of Loss-Balancing Baselines

We provide additional implementation details for the loss-balancing baselines compared in Table 2. All variants use the same EAR architecture, dataset, and reconstruction-related loss weights; they differ only in how the influence of the generative objective is controlled during training.

Constant weighting. We use a fixed prior weight $\lambda_{\text{prior}} = 1$ throughout 500 training epochs, with $\lambda_{\text{ent}} = \lambda_{\text{prior}}$. We observe that training largely converges around 260 epochs, with little

Table 18: **Detailed training configuration of EiT on ImageNet.** EiT-256 first jointly trains the VAE and SiT with the two-stage GenFirst schedule, after which the VAE is frozen and we train a new SiT prior. EiT-512 directly reuses and freezes the EiT-256 VAE and trains only SiT.

Configuration	**EiT-256**	**EiT-512**
Architecture		
VAE	SD-VAE / VA-VAE	EiT-256 VAE (frozen, VA-VAE version)
Latent configuration	$f8d4/f16d32$	$f16d32$
Generative backbone	SiT-XL/2 Ma et al. (2024) / SiT-XL/1 Ma et al. (2024)	SiT-XL/1 Ma et al. (2024)
Transformer depth	28	28
Hidden dimension	1152	1152
Attention heads	16	16
End-to-end phase		
Trainable modules	VAE + SiT	–
Generation-first stage	80 epochs	–
$\lambda_{\text{prior}}/\lambda_{\text{ent}}$	$0.1/0.1$	–
Reconstruction-refinement stage	40 epochs	–
$\lambda_{\text{prior}}/\lambda_{\text{ent}}$	$0.01/0.01$	–
Prior-only phase		
VAE	frozen	frozen
Trainable modules	SiT only	SiT only
Training resolution	256×256	512×512
Training duration	800 epochs	400 epochs
Shared optimization		
Global batch size	256	256
Optimizer	AdamW Loshchilov & Hutter (2017), $\beta_1, \beta_2 = 0.9, 0.999$	AdamW, $\beta_1, \beta_2 = 0.9, 0.999$
Learning rate	1×10^{-4}	1×10^{-4}
LR schedule	constant	constant
EMA rate	0.9999	0.9999
Class token drop (for CFG)	0.1	0.1
REPA		
λ for SiT	0.5	0.5
λ for VAE	1.5	1.5
Alignment depth	8	8
Encoder	DINOv2-B Oquab et al. (2023)	DINOv2-B
Inference		
Guidance	CFG Ho & Salimans (2022)+Autoguidance Karras et al. (2024a)	CFG
Sampler	SDE	SDE
Sampling steps	250	250
Weak model	20 epochs	–
Autoguidance scale	1.3	–
CFG scale	1.5	2.5
Guidance interval	[0.2, 0.7]	[0.2, 0.8]

additional improvement from longer training. Moreover, as shown in Figure 5(a), smaller constant prior weights improve reconstruction but yield worse gFID (Figure 5(a)). We therefore use $\lambda_{\text{prior}} = 1$ as the constant-weight baseline with the best generation performance among the tested values.

Cosine decay. We train for 320 epochs and smoothly decay the prior weight from 0.5 to 0.1 with a cosine schedule. The entropy weight follows the same schedule, i.e., $\lambda_{\text{ent}} = \lambda_{\text{prior}}$. We performed a simple hyperparameter sweep but did not obtain better results than the constant-weight setting.

Convergence of weighting baselines. Both the constant-weight and cosine-decay baselines reach an empirical plateau in gFID within their training budgets. Extending training beyond this plateau leads to fluctuations in gFID without sustained improvement.

PI-based adaptive control. We additionally test a feedback-based strategy inspired by ControlVAE Shao et al. (2020). Instead of prescribing a fixed schedule, the controller adjusts how strongly the generative objective updates the VAE encoder through a w. Thus, $w = 1.0$ allows the full generative gradient to propagate to the encoder, while smaller w weakens this update. The controller uses the per-step reconstruction error as feedback with target $\ell^* = 0.055$ following Con-

Table 19: **Detailed training configuration of EiT for text-to-image generation.** We consider two settings for the end-to-end latent-shaping phase: joint training with SiT on ImageNet or with a smaller MMDiT on the T2I training mixture. The resulting VAE is then frozen and used to train the target MMDiT.

Configuration	EiT (Version A)	EiT (Version B)
Architecture		
VAE	FLUX.1-dev VAE black-forest labs (2024)	FLUX.1-dev VAE black-forest labs (2024)
Latent configuration	$f8d16$	$f8d16$
Text Encoder	–	clip-vit-large-patch14 Radford et al. (2021)
Generative backbone (E2E phase)	SiT-XL/2 Ma et al. (2024)	MMDiT-L/2 Esser et al. (2024)
Backbone depth (E2E phase)	28	24
Hidden dimension (E2E phase)	1152	1024
Attention heads (E2E phase)	16	16
Training data (E2E phase)	ImageNet	T2I mixture
Training resolution (E2E phase)	256×256	256×256
Text Encoder	clip-vit-large-patch14 Radford et al. (2021)	
Generative backbone (Frozen VAE phase)	MMDiT-XL/2	
Backbone depth (Frozen VAE phase)	28	
Hidden dimension (Frozen VAE phase)	1152	
Attention heads (Frozen VAE phase)	16	
Training data (Frozen VAE phase)	T2I mixture	
Training resolution (Frozen VAE phase)	256×256 (stage1), 512×512 (stage2, stage3)	
End-to-end phase		
Trainable modules	VAE + SiT	VAE + MMDiT
Global batch size	256	256
Generation-first stage	80 epochs	400K steps
$\lambda_{\text{prior}}/\lambda_{\text{ent}}$	$0.1/0.1$	$0.1/0.1$
Reconstruction-refinement stage	40 epochs	200K steps
$\lambda_{\text{prior}}/\lambda_{\text{ent}}$	$0.01/0.01$	$0.01/0.01$
Frozen-VAE MMDiT training		
VAE	Frozen E2E-trained VAE	
T2I mixture, 256^2	200K steps, global batch size 1536	
T2I mixture, 512^2	80K steps, global batch size 512	
4o data fine-tuning, 512^2	40K steps, global batch size 512	
Inference		
Sampler	Euler	
Sampling steps	50	
CFG scale	7.0	
CFG interval	[0.35, 1.0]	

trolVAE Shao et al. (2020). We initialize $w = 1.0$ and constrain it to $[0.004096, 1.0]$. In our experiments, the reconstruction error is substantially above the target during early training, causing w to rapidly decrease toward its lower bound. It subsequently oscillates between different values, leading to abrupt changes in the gradient received by the latent encoder and numerical instability. We also performed simple tuning of the controller hyperparameters, but did not obtain stable training under the tested configurations.

D.3 Toward a Shared Latent Space for Generation and Representation Learning

As discussed in section 5.4, we further examine whether the end-to-end framework can learn a visual representation that simultaneously supports reconstruction, generation, and representation learning. In this section, we describe the autoencoder architecture, training schedule, and discrimination objectives used in the three settings reported in Table 8.

Qwen3-VL-based visual tokenizer. We replace the convolutional encoder of the FLUX.1-dev VAE black-forest labs (2024) with the vision tower of Qwen3-VL-2B-Instruct Bai et al. (2025). At an input resolution of 256×256, the patch size of 16 produces a 16×16 grid of visual tokens. Each token is represented by a 1,024-dimensional feature before the spatial merger. We construct a Gaussian projector on top of these features. We first apply a LayerNorm initialized from the normal-

ization layer of the pretrained Qwen3-VL merger. Two independent linear projections then map each 1,024-dimensional visual token to 64 dimensions, predicting the posterior mean and log-variance, respectively:

$$\boldsymbol{\mu}_i = W_\mu \,\mathrm{LN}(\mathbf{h}_i) + b_\mu, \qquad \log \boldsymbol{\sigma}_i^2 = W_\sigma \,\mathrm{LN}(\mathbf{h}_i) + b_\sigma,$$

where $\mathbf{h}_i \in \mathbb{R}^{1024}$ and $\boldsymbol{\mu}_i, \log \boldsymbol{\sigma}_i^2 \in \mathbb{R}^{64}$. The 64 outputs associated with each ViT token are reshaped into a 2×2 block with 16 channels. The resulting latent therefore has shape $16 \times 32 \times 32$, matching both the dimensionality and spatial rate of the FLUX $f8d16$ latent. This allows us to retain the generative architecture and training setting used in our text-to-image experiments. The decoder is randomly initialized and consists of a 12-layer ViT with width 1,024 and 16 attention heads. Each decoder token predicts a 16×16 RGB patch. We additionally employ two lightweight 3×3 convolutional layers after assembling the image to reduce visible artifacts at patch boundaries.

Decoder warm-up. Directly optimizing the randomly initialized decoder together with the pretrained vision tower exposes the encoder to large and uninformative reconstruction gradients at the beginning of training. In preliminary experiments, this rapidly degraded the semantic structure inherited from Qwen3-VL. We therefore first freeze the Qwen3-VL vision tower and train only the Gaussian bottleneck and decoder for 100K steps. In this warm-up stage, we apply only a weak KL regularizer with weight 10^{-6}; the adversarial loss is enabled after 10K steps.

End-to-end training. After decoder warm-up, we unfreeze the vision tower and jointly optimize the encoder, projector, decoder, and an MMDiT-L/2 generative prior. Apart from the discrimination objective, this stage follows the text-to-image end-to-end training recipe used in our main experiments, including the reconstruction losses, flow-matching objective, entropy regularization, and REPA alignment to DINOv2 features.

Due to the available computational budget, we shorten the original two-stage schedule by a factor of two. We first train for 200K steps with the prior weight set to 0.1. We then train for another 100K steps with the prior weight set to 0.01. For the two discrimination variants, the discrimination-loss weight is the same as the prior weight. The global batch size is 256. We compare the following three settings.

Stable E2E without discrimination supervision. The first setting uses only reconstruction, generative, entropy, and REPA objectives. It serves as the generation–reconstruction baseline and does not receive an explicit discrimination objective.

Latent-level SigLIP supervision. In the second setting, discrimination supervision is applied directly to the same normalized latent consumed by MMDiT, which is a 16×16 sequence of 64-dimensional tokens. A linear stem maps each token to width 1,024, after which a learnable query performs attention pooling. The pooled feature is projected to the 768-dimensional CLIP image–text embedding space. Given normalized image and text embeddings $\mathbf{v}_i$ and $\mathbf{t}_j$, we use the sigmoid pairwise objective Zhai et al. (2023)

$$\mathcal{L}_{\mathrm{Sig}} = -\frac{1}{B} \sum_{i=1}^{B} \sum_{j=1}^{B} \log \sigma \left[y_{ij} \left(\tau \, \mathbf{v}_i^\top \mathbf{t}_j + \beta \right) \right], \tag{8}$$

where $y_{ij} = 1$ for a matched image–caption pair and -1 otherwise. Text features and negatives are gathered across workers, so the loss uses the global batch.

We complement this objective with classification-based caption supervision Huang et al. (2024). The pooled image feature predicts a bag of CLIP caption tokens. Caption tokens form a multi-hot target, which is reweighted using online inverse document frequency and normalized into a probability distribution. The corresponding soft cross-entropy is

$$\mathcal{L}_{\mathrm{CLS}} = -\sum_k \widetilde{q}_k \log p_k, \qquad \widetilde{q}_k \propto q_k \log \frac{N}{f_k}. \tag{9}$$

This weighting suppresses frequent syntactic tokens and concentrates the gradient on visually informative content words. The complete discrimination objective is

$$\mathcal{L}_{\mathrm{disc}}^{\mathrm{SigLIP}} = \mathcal{L}_{\mathrm{Sig}} + \mathcal{L}_{\mathrm{CLS}}. \tag{10}$$

This objective directly shapes the latent representation used by the generative model.

Trunk-level VLM-NLL supervision. In the third setting, we preserve the interface between the Qwen3-VL vision tower and its pretrained language model. Rather than applying supervision after the 16-channel bottleneck latent, we branch from the ViT trunk and feed its native merged visual tokens to the frozen Qwen3-VL-2B language model. We also retain the original DeepStack features injected into the early language-model layers. For an image x and its caption $\mathbf{c} = (c_1, \ldots, c_T)$, the discrimination loss is the caption-token negative log-likelihood

$$\mathcal{L}_{\text{disc}}^{\text{NLL}} = -\frac{1}{T}\sum_{t=1}^{T} \log p_{\text{Qwen}}\left(c_t \mid c_{<t},\, f_{\text{ViT}}(x)\right). \tag{11}$$

The Qwen3-VL language model, language-model head, spatial merger, and DeepStack mergers are all frozen. Consequently, the NLL can only be reduced by preserving visual features that remain readable by the original Qwen3-VL language model. Unlike SigLIP supervision, the VLM-NLL objective does not depend on cross-sample negatives and therefore does not require a large global batch.

Evaluation. We evaluate reconstruction using posterior-mean reconstructions on 50K ImageNet validation images and report rFID and PSNR. Semantic accessibility is measured by an ImageNet ridge linear probe trained on 100K images and evaluated on 10K validation images. For final generation evaluation, we freeze each trained visual tokenizer and train the same MMDiT-XL pipeline used in our text-to-image experiments before evaluating GenEval and DPG-Bench.

D.4 Unified Text–Image Modeling via End-to-End Diffusion

As discussed in section 5.4, our end-to-end formulation is not restricted to image latents and can be applied to other modalities. We therefore extend our framework to text and jointly model text and images within a single diffusion backbone, allowing the generative objective to shape both text and image latent spaces. Detailed experiment results and qualitative results are shown in Table 20 and Figure 8.

Text latent representation. We initialize the text VAE from Cola Guo et al. (2026). Each text token is encoded into a 16-dimensional continuous latent, and the decoder reconstructs the original tokens with a cross-entropy loss. We train the encoder and decoder trunks together with the latent projections ($\sim$194M trainable parameters), while freezing the token embedding and output vocabulary head. The encoder applies a non-affine LayerNorm to the posterior mean, so we do not apply the additional latent normalization used for image latents. To handle variable-length captions efficiently, text sequences are packed without padding and converted to a dense (B, L, d) representation with a validity mask only before entering the diffusion model.

Block-causal joint diffusion. During the end-to-end phase, we jointly model text and image latents using a block-causal MMDiT-L/2 Esser et al. (2024) with 24 blocks, width 1024, 16 attention heads, and patch size 2. The model maintains separate streams for the two modalities and couples them through joint attention. At 256×256 resolution, each image contains $16 \times 16 = 256$ latent tokens, while text contains at most 256 latent tokens and is partitioned into blocks of 16 tokens. Image tokens use bidirectional attention, whereas text tokens use block-causal attention, enabling block-wise generation of $p(t|i)$.

We train text-to-image $p(i|t)$ and image-to-text $p(t|i)$ with equal sampling probability. Following the convention $z_t = (1-t)\epsilon + tz$, where $\epsilon \sim \mathcal{N}(0, I)$, we apply noise and diffusion supervision only to the target modality, while keeping the conditioning modality clean at $t_{\text{cond}} = 1$. We stop gradients through the clean conditioning modality, so the generative objective updates only the target-modality encoder in each direction; each VAE is nevertheless optimized end-to-end when its modality serves as the generation target. Image timesteps are sampled independently per example, whereas text timesteps are sampled independently for each text block. For text generation, each noisy block attends to all preceding clean blocks and to the noisy tokens within the current block, allowing all blocks to be trained in parallel with clean prefix conditioning. Cross-modal attention is direction-dependent: image tokens attend to clean text for $p(i|t)$, while text tokens attend to clean image features for $p(t|i)$. For classifier-free guidance, we drop the conditioning modality with probability 0.1 by disabling the corresponding cross-modal attention.

Training objective. The text VAE follows the same end-to-end formulation as the image branch:

$$\mathcal{L}_{\text{text}} = \mathcal{L}_{\text{CE}} + \lambda_{\text{KL}} D_{\text{KL}}(q_{\phi_t}(z_t|x) \,\|\, \mathcal{N}(0, I)) - \lambda_{\text{ent}} \mathcal{H}(q_{\phi_t}(z|x)) + \lambda_{\text{prior}} \mathcal{L}_{\text{prior}}, \tag{12}$$

Table 20: Text-to-image and image-to-text performance of frozen text/image VAEs versus our end-to-end trained counterparts, all with the same block-causal MMDiT and the same three-stage recipe. Image-to-text is evaluated on the COCO Lin et al. (2014) Karpathy test split with pycocoeval-cap Chen et al. (2020), following (Team, 2024; Zhou et al., 2025).

Setting	Text-to-image		Image-to-text					
	GenEval↑	DPG↑	CIDEr↑	BLEU-4↑	METEOR↑	ROUGE-L↑	SPICE↑	CLIPScore↑
Frozen text VAE + frozen image VAE	0.80	63.96	20.03	5.36	13.72	29.01	8.03	53.21
E2E-trained text VAE + frozen image VAE	0.82	68.24	30.19	7.46	16.81	32.95	11.12	55.96
Jointly E2E-trained text + image VAEs	**0.87**	**72.68**	**33.28**	**7.76**	**17.74**	**34.16**	**11.81**	**57.97**

where $\mathcal{L}_{\text{prior}}$ is the flow-matching objective of the joint diffusion model and $\lambda_{\text{KL}} = 10^{-6}$. Following our analysis in section 3.1, the entropy term counteracts the prior-fitting pressure and prevents latent collapse. We apply the entropy regularizer only to the target modality in each training direction. For text-only end-to-end training, the image VAE remains frozen and only the text encoder receives generative gradients. For joint end-to-end training, both text and image VAEs are optimized under the same GenFirst schedule, allowing the bidirectional diffusion objectives to shape both latent spaces.

Training and evaluation. We compare three latent-training settings. In the first setting, both the Cola text VAE and the FLUX.1-dev image VAE black-forest labs (2024) are kept frozen, providing a fully frozen latent baseline. In the second setting, we end-to-end train the text VAE together with the block-causal MMDiT while keeping the FLUX.1-dev image VAE frozen, isolating the effect of reshaping the text latent space. In the third setting, both the Cola-initialized text VAE and the FLUX.1-dev image VAE are jointly optimized with the block-causal MMDiT, allowing the bidirectional generative objectives to reshape both latent spaces. End-to-end training is conducted at 256×256 resolution on the same dataset as our text-to-image experiments, with captions truncated to 256 tokens. We use a batch size of 256, AdamW Loshchilov & Hutter (2017) with learning rates of 10^{-4} for MMDiT and 10^{-5} for the text VAE, and gradient clipping at 1.0. We use $\lambda_{\text{prior}} = \lambda_{\text{ent}} = 0.1$ for the first 200K steps and reduce both to 0.01 for another 100K steps. For joint end-to-end training, the same GenFirst schedule is applied to both text and image latent optimization.

After end-to-end latent training, we freeze the resulting text and image VAEs and train a new block-causal MMDiT-XL/2 from scratch. The prior-only model uses 28 blocks, width 1152, and 16 attention heads, matching the model scale of MMDiT-XL/2 in our main text-to-image experiments. We follow the same training protocol as our text-to-image experiments: 200K steps at 256×256, 80K steps at 512×512, followed by 40K steps of fine-tuning on GPT-4o data. We evaluate text-to-image generation using GenEval and DPG-Bench. We evaluate image-to-text generation on the COCO Karpathy test split using CIDEr, BLEU-4, METEOR, ROUGE-L, SPICE, and CLIPScore.

E More Qualitative Results

Figure 9 presents class-conditional samples generated by EAR and EiT. Both models produce diverse and visually coherent images with fine-grained details and strong correspondence to the input conditions. Figure 10 compares text-to-image generation using the same MMDiT Esser et al. (2024) and training data, differing only in the VAE; the left uses the FLUX.1-dev VAE and the right our end-to-end trained VAE. End-to-end training improves text-to-image generation in three aspects. First, it substantially improves perceptual quality and fine-grained textures. For example, the fisherman exhibits more realistic skin details and beard texture, while the cat shows finer and more natural fur; similar improvements are also visible in the butterfly, sandstone, and reflective material examples. Second, it improves generation plausibility and structural coherence. For example, the baseline generates a horse with three hind legs, whereas our end-to-end trained VAE produces a more anatomically plausible structure. Third, our end-to-end trained VAE improves text alignment. For example, it more faithfully captures the blooming tea inside the transparent teapot and correctly renders the specified count of three moons.

Figure 11, 12, 13, 14 present additional uncurated examples on ImageNet 256×256 with EiT. Figure 15, 16, 17, 18 present additional uncurated examples on ImageNet 256×256 with EAR.

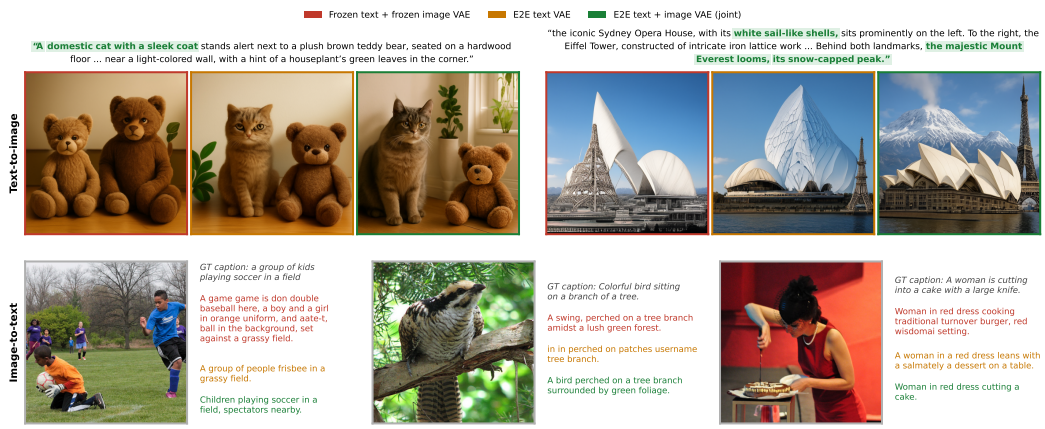


Figure 8: **Qualitative comparison of unified text–image latent settings.** All settings use the same MMDiT and sampling configuration. **Top:** text-to-image generation on DPG-Bench. **Bottom:** image-to-text generation on the COCO Karpathy test split. End-to-end trained latents improve prompt fidelity and caption coherence, with joint text–image training giving the strongest results.

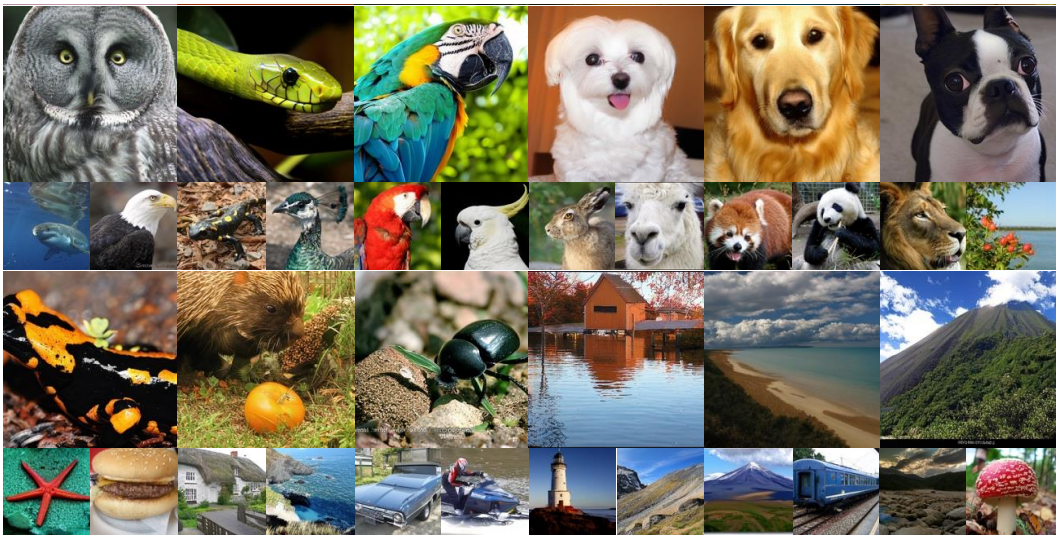


Figure 9: Class-to-image samples from EiT (row 1,2) and EAR (row 3,4) with guidance(CFG scale=4).

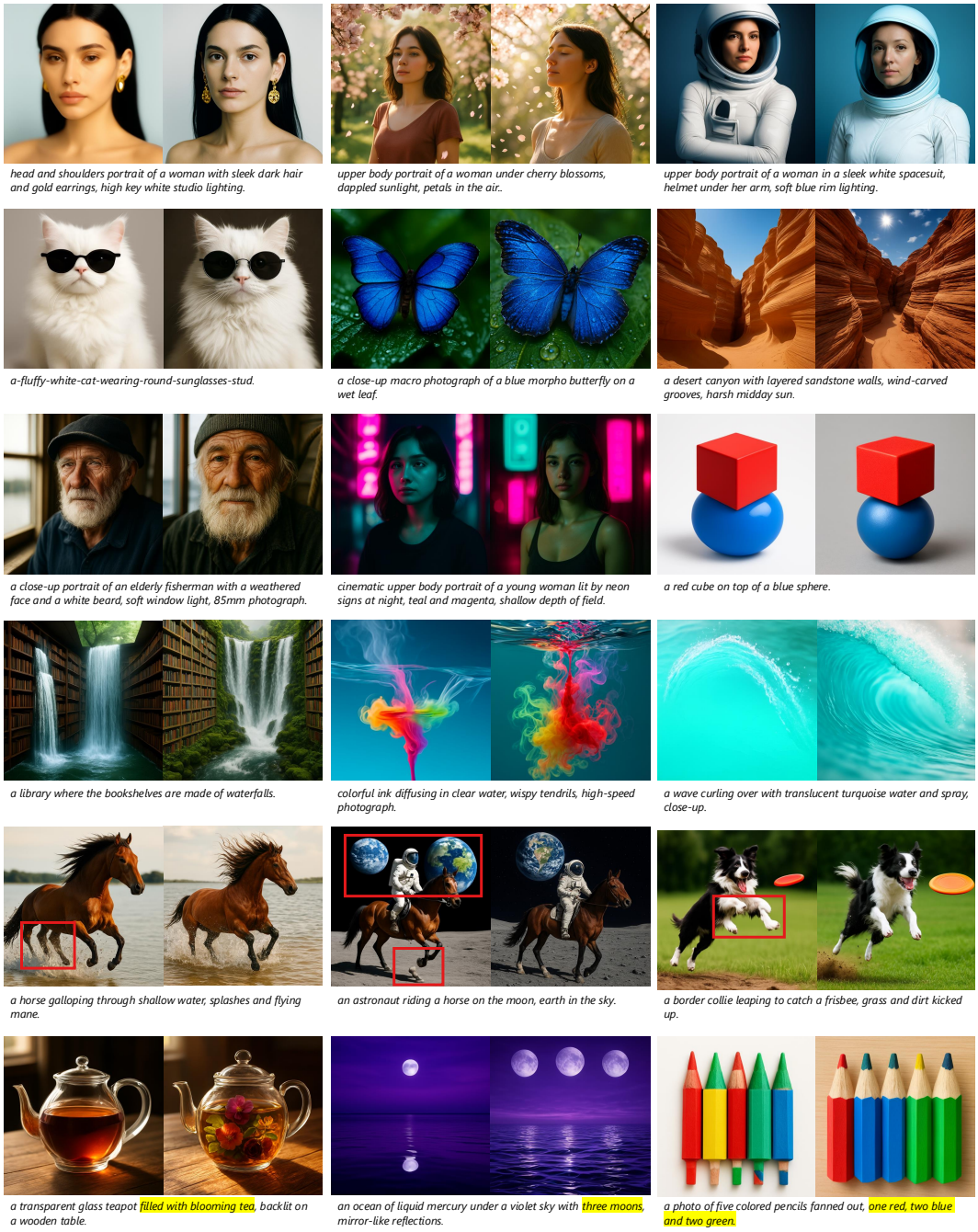


Figure 10: **Qualitative comparison of text-to-image generation.** Both settings use the same model and training data; the **left** uses the FLUX.1-dev VAE and the **right** our end-to-end trained VAE. End-to-end training improves perceptual quality and fine-grained textures (rows 1–3), produces more plausible and coherent images (rows 4–5), and improves text alignment (row 6).



Class 011, goldfinch, Carduelis carduelis



Class 018, magpie



Class 025, European fire salamander



Class 074, garden spider, Aranea diadematа



Class 096, toucan



Class 107, jellyfish



Class 127, white stork, Ciconia ciconia



Class 153, Maltese dog, Maltese terrier, Maltese

Figure 11: **Uncurated samples from EiT (480 epochs) with guidance on ImageNet** 256×256**.**



Class 222, kuvasz



Class 271, red wolf, maned wolf, Canis rufus



Class 286, cougar, puma, catamount, mountain lion



Class 294, brown bear, bruin, Ursus arctos



Class 302, ground beetle, carabid beetle



Class 309, bee



Class 326, lycaenid, lycaenid butterfly



Class 388, giant panda, panda, panda bear

Figure 12: **Uncurated samples from EiT (480 epochs) with guidance on ImageNet** 256×256**.**



Class 402, acoustic guitar



Class 425, barn



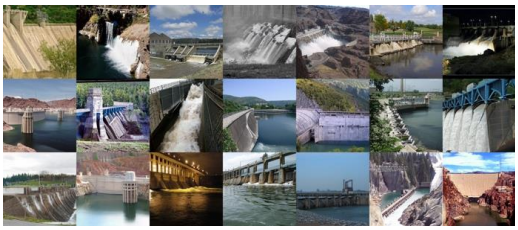
Class 437, beacon, lighthouse, beacon light, pharos



Class 466, bullet train, bullet



Class 505, coffeepot



Class 525, dam, dike, dyke



Class 538, dome



Class 572, goblet

Figure 13: **Uncurated samples from EiT (480 epochs) with guidance on ImageNet** 256×256**.**



Class 698, palace



Class 718, pier



Class 757, recreational vehicle



Class 831, studio couch, day bed



Class 945, bell pepper



Class 963, pizza, pizza pie



Class 976, promontory, headland, head, foreland



Class 985, daisy

Figure 14: **Uncurated samples from EiT (480 epochs) with guidance on ImageNet** 256×256**.**

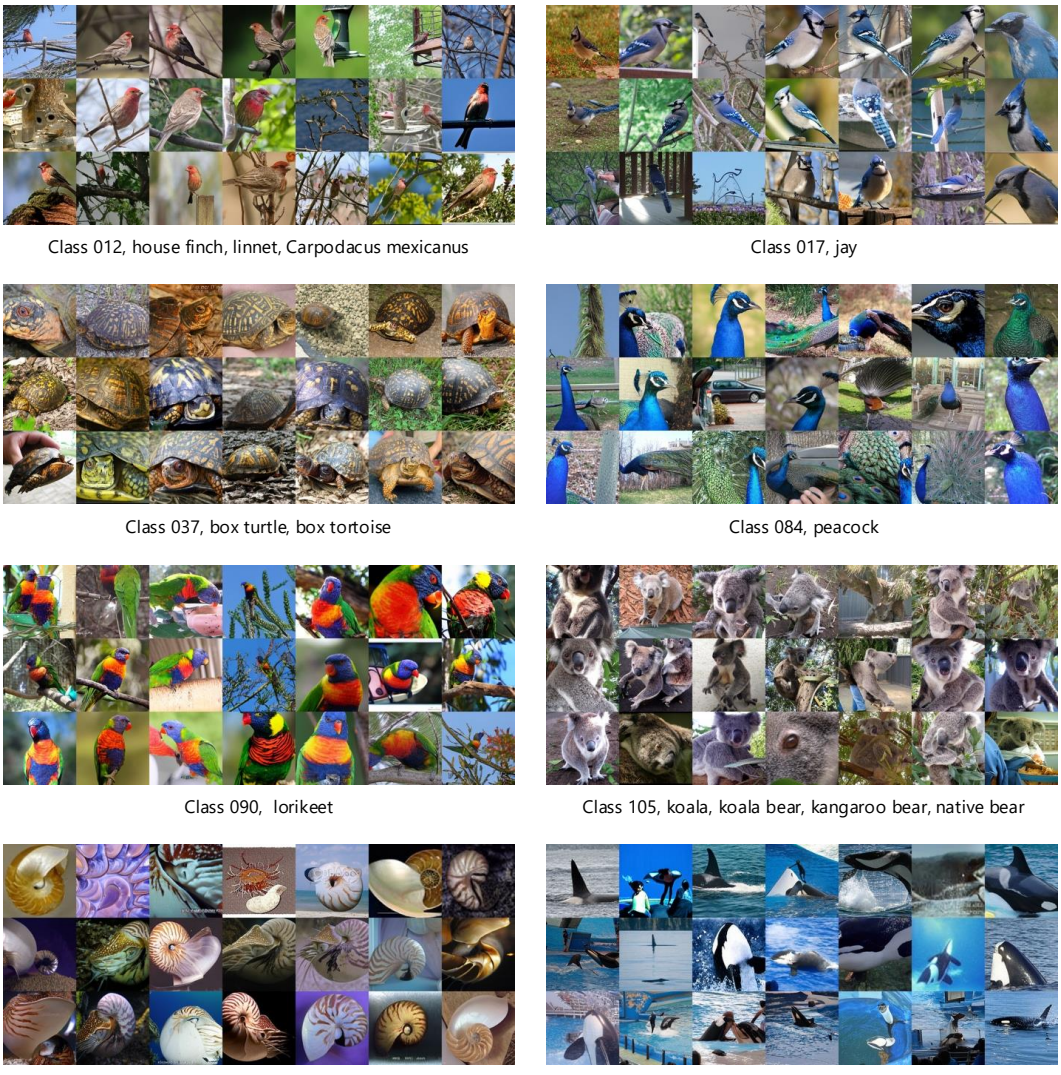


Figure 15: **Uncurated samples from EAR (640 epochs) with guidance on ImageNet** 256×256**.**



Class 207, golden retriever



Class 232, Border collie



Class 277, red fox, Vulpes vulpes



Class 292, tiger, Panthera tigris



Class 323, monarch, monarch butterfly, milkweed butterfly



Class 348, ram, tup



Class 387, lesser panda, red panda, panda



Class 393, anemone fish

Figure 16: **Uncurated samples from EAR (640 epochs) with guidance on ImageNet** 256×256**.**



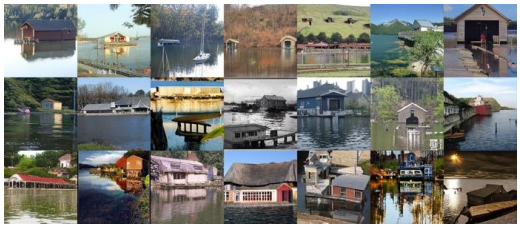
Class 407, check, ambulance



Class 417, balloon



Class 438, beaker



Class 449, boathouse



Class 475, car mirror



Class 556, fire screen, fireguard



Class 579, grand piano, grand



Class 598, home theater, home theatre

Figure 17: **Uncurated samples from EAR (640 epochs) with guidance on ImageNet** 256×256**.**



Class 628, liner, ocean liner



Class 658, mitten



Class 780, schooner



Class 832, stupa, tope



Class 933, cheeseburger



Class 946, cardoon



Class 973, coral reef



Class 980, volcano

Figure 18: **Uncurated samples from EAR (640 epochs) with guidance on ImageNet** 256×256**.**