FEATOK: FEATURE-LOSS TOKENIZATION FOR IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Discrete image generators learn the distribution of tokens produced by an image tokenizer. These token representations should be easy for generators to model, not merely enable faithful image reconstruction. Yet these tokenizers are commonly trained to reconstruct pixels, which prioritizes low-level fidelity and may disrupt representation structure useful for generation. Motivated by feature-predictive self-supervised learning, we introduce feature-loss tokenization (FeaTok), which trains discrete tokens by reconstructing pretrained features rather than pixels. Ablations across SigLIP, SigLIP2, DINOv2, and DINOv3 show better generative performance with FeaTok. In class-conditional generation, FeaTok advances the quality–compute Pareto frontier, reducing gFDr6 at 256×256 by up to 29% over the strongest discrete baselines and outperforming the strongest unguided continuous baseline (RAEv2) with $38\times$ less sampling compute. FeaTok also achieves higher human-preference reward-model scores than prior art without reward-based training and surpasses continuous and discrete text-conditional baselines on three benchmarks with only 500 fine-tuning updates after pre-training.

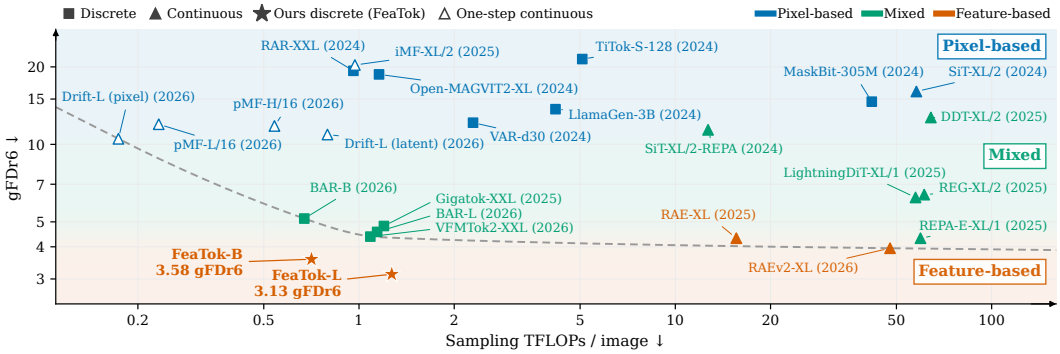


Figure 1: **Feature-based discrete tokens enable strong generation.** FeaTok models advance the quality–compute Pareto frontier for image generation. We categorize generative image models by how pretrained feature-predictive self-supervised learning representations (Def. 1) shape the generated representation or the generator: not at all (pixel-based), together with pixel objectives (mixed), or alone (feature-based). Generation performance on ImageNet 256×256, where all models sample without guidance. Both axes are log-scaled.

1 INTRODUCTION

Image generation enables machines to create new visual content, supporting applications ranging from creative design to multimodal interaction. A growing body of work represents images as discrete tokens and models them autoregressively (Sun et al., 2024; Tian et al., 2024; Yu et al., 2026). Discrete tokens share the interface of language models, which makes them a natural choice for unified multimodal models. They also sample an image with an order of magnitude less compute than continuous diffusion models, as shown in Fig. 1.

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.

Image generation quality depends on discrete tokens that are both expressive and easy for a generator to model. Previous discrete tokenizers focus on reconstruction performance, which targets the first property, *expressivity*. However, generation also depends on the second property, which we call *generative learnability*: how easily a generator fits the token distribution at a given budget. These two properties do not always align, as observed for both continuous and discrete image tokenizers and generators (Gu et al., 2024; Yao et al., 2025; Ramanujan et al., 2025).

Generative learnability is related to the structure of a representation (Skorokhodov et al., 2025; Chu et al., 2026), which is in turn induced by the training target. Pixel reconstruction prioritizes high-variance image details over features useful for perception (Balestriero & Lecun, 2024), and using self-supervised learning (SSL) objectives to predict features instead of pixels yields more semantically meaningful representations (Assran et al., 2023). We refer to the representations of encoders trained with SSL objectives in feature space as *pretrained visual features* (see Def. 1). Moreover, RAE (Zheng et al., 2026b) and RAEv2 (Singh et al., 2026) show that the benefits of pretrained visual features go beyond representation learning and successfully transfer to image generation.

Wang et al. (2024) trained discrete tokens by reconstructing the features of image-understanding encoders instead of pixels, and showed that these tokens can outperform pixel-trained VQ tokens for image generation. Yet the discrete tokenizers that followed still use pixel reconstruction as the central loss, and add pretrained visual features with additional losses (Xiong et al., 2025; Du et al., 2026; Yu et al., 2026; Zheng et al., 2026a; Chen et al., 2026b). This raises the question of whether *pixel reconstruction loss is essential for modern discrete tokenizer training used for downstream image generation*. We argue that it is not. Feature objectives should shape both the encoder, pretrained in feature space, and the discrete tokens.

We introduce feature-loss tokenization (**FeaTok**), which separates discrete representation learning from image reconstruction. Instead of a pixel decoder and a pixel reconstruction loss, a feature decoder that reconstructs pretrained visual features shapes the discrete tokens. A separate image decoder is trained with pixel losses on *detached* tokens and only renders generated token sequences, so pixel losses never shape the discrete representation. We additionally discretize the encoder's global representation (*e.g.*, CLS token), and generate it before the patch tokens, giving them image-level context. We call this *global-augmented* generative modelling. Ablations in Sec. 2.2 show that FeaTok has better spatial structure and generates better than tokenizers trained with pixel or combined losses for all four encoders we study, and that global-augmented modelling contributes to this improvement.

On class-conditional ImageNet-1k (Russakovsky et al., 2015) at 256×256, FeaTok advances the quality–compute Pareto front for image generation (Fig. 1). We measure quality by gFDr6, the mean Fréchet-distance ratio across six feature spaces rather than the Inception space alone (Yang et al., 2026). FeaTok improves on the strongest discrete baselines by 29% without guidance and by 25% with it, and outperforms recent continuous generators at a matched sampling cost in both settings. The gains grow at 512×512: relative to the previously best discrete model (BAR (Yu et al., 2026)), FeaTok reduces gFDr6 by 43% without guidance and 39% with guidance, up from 32% and 25% at 256×256, and it also outperforms RAE. FeaTok further achieves higher rewards than discrete and continuous baselines on three human-preference models (Kirstain et al., 2023; Wu et al., 2023; Ma et al., 2025) without reward-based training. Finally, reusing the ImageNet-trained tokenizer, FeaTok surpasses the baselines on three text-to-image benchmarks (Ghosh et al., 2023; Li et al., 2024a; Hu et al., 2024) after 75K pretraining updates and only 500 fine-tuning updates. Discrete tokens shaped by feature losses alone are therefore competitive with continuous representations while sampling at a fraction of the cost, which makes them a practical substrate for unified multimodal models.

2 FEATURE-LOSS TOKENIZATION

In this section, we present **FeaTok**, which learns discrete tokens by reconstructing pretrained visual features. We use this term to distinguish representations by their pretraining objective.

Definition 1. We define *pretrained visual features* as representations of encoders learned by self-supervision in feature space, including language–image contrastive learning (Radford et al., 2021; Zhai et al., 2023; Tschannen et al., 2025) and self-distillation (Caron et al., 2021; Oquab et al., 2024; Siméoni et al., 2026; Tschannen et al., 2025). Encoders trained by pixel reconstruction (He et al., 2022) are excluded.

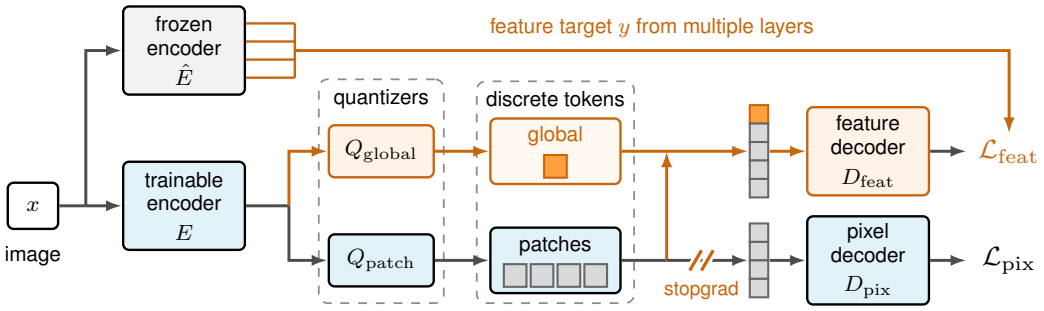


Figure 2: **FeaTok tokenizer and generation pipeline.** Blue denotes the conventional discrete tokenizer and orange our contributions. Only feature losses shape the discrete tokens: $\mathcal{L}_{\text{feat}}$ trains D_{feat} to reconstruct the multi-layer target y of the frozen encoder $\hat{E}$ from both global and patch tokens, while $\mathcal{L}_{\text{pix}}$ trains only D_{pix} on detached patch tokens. The generator (not shown) models the global token from Q_{global} together with the patch tokens from Q_{patch}.

We next describe how FeaTok uses these features to train the tokenizer with feature losses (Fig. 2).

2.1 Learning discrete tokens with feature loss only

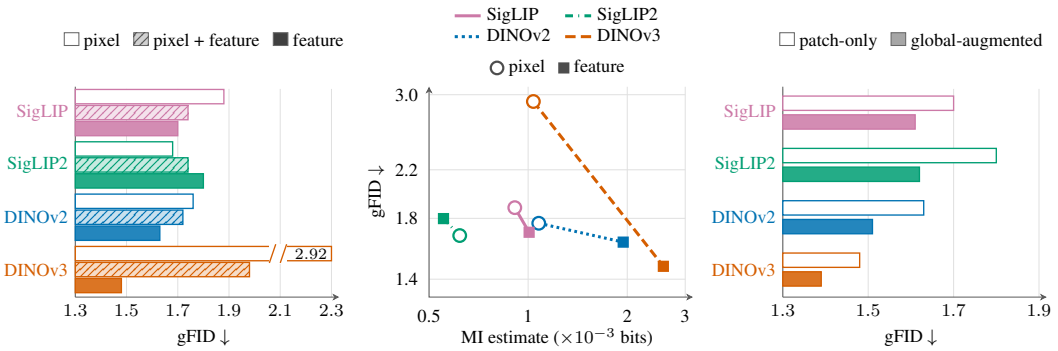
Conventional discrete tokenizers (Van Den Oord et al., 2017; Jia et al., 2025; Bachmann et al., 2025; Zheng et al., 2026a; Yu et al., 2026) consist of an encoder E, a quantizer Q, and a pixel decoder D_{pix}. Given an image x partitioned into N patches, the encoder E produces N feature vectors, one per patch, which the quantizer Q maps to discrete tokens $z_{1:N} = Q(E(x))$. The pixel decoder D_{pix} then reconstructs the image as $\hat{x} = D_{\text{pix}}(z_{1:N})$. Tokenizers are typically trained as an autoencoder with a pixel-level objective $\mathcal{L}_{\text{pix}}$ that combines pixel reconstruction losses (ℓ_1, ℓ_2), perceptual losses (LPIPS (Zhang et al., 2018), GRAM (Lu et al., 2026a)), and adversarial losses (Goodfellow et al., 2014). Once the tokenizer is trained, an autoregressive model conditioned on class labels or text prompts c learns the distribution $p_\phi(z_{1:N} \mid c)$ of the resulting token sequences using the cross-entropy loss.

Shaping discrete tokens with feature losses. FeaTok follows the autoencoder-style tokenizer design, but learns the discrete tokens by reconstructing features rather than pixels (see Fig. 2). Concretely, we add a feature decoder D_{feat} that maps the discrete tokens $z_{1:N}$ to a prediction $\hat{y}_{1:N} = D_{\text{feat}}(z_{1:N})$ of a feature target $y_{1:N}$. We train the encoder, quantizer, and feature decoder to match this prediction to the target using a loss $\mathcal{L}_{\text{feat}}$ that combines ℓ_1, ℓ_2, and cosine distances.

For the feature target, following the insights in Singh et al. (2026), we use an average of features from multiple layers of the frozen vision encoder $\hat{E}$, $y_{1:N}(x) = |\mathcal{K}|^{-1} \sum_{k \in \mathcal{K}} \text{LN}_k(\hat{E}^{(k)}(x)_{1:N})$, where $\mathcal{K}$ contains four evenly spaced encoder layers and LayerNorm (Ba et al., 2016) normalizes each layer's features before averaging. This target is intended to combine visual detail from earlier layers with semantic information from later layers, balancing expressivity and generative learnability.

While we retain the pixel decoder D_{pix} and its pixel-level loss $\mathcal{L}_{\text{pix}}$ during training, we stop the gradients at the decoder input. This preserves the ability to decode tokens back into images while preventing the pixel reconstruction losses from shaping the token distribution.

Leveraging global tokens. In addition to patch features, various vision encoders supply a global representation, *e.g.*, a CLS token in DINO (Caron et al., 2021; Oquab et al., 2024; Siméoni et al., 2026) or a global pooling token in SigLIP (Zhai et al., 2023; Tschannen et al., 2025). Replacing pixel reconstruction with feature reconstruction losses naturally suggests discretizing this global feature along with patches. The benefit of doing so is that it allows us to define a hierarchical generative model that first generates the global token and then the patch tokens conditioned on it which simplifies the conditional distributions of the patch tokens by providing a global context. This global representation is generally ignored by discrete tokenizers trained using pixel-reconstruction losses, as the pixel reconstruction over patches does not provide direct supervision on the global feature discretization. Prior work follows a similar global-to-local strategy, using learned holistic tokens (Zheng et al., 2025a), coarse-to-fine token sequences (Bachmann et al., 2025), or separately



(a) Tokenizer targets. (b) Spatial dependence. (c) Global augmentation.

Figure 3: **Tokenizer-target and generator-type ablations.** (a) With patch-only generation, feature targets improve gFID for SigLIP, DINOv2, and DINOv3. The DINOv3 pixel bar has a break to accommodate its gFID of 2.92. (b) Spatial dependence between adjacent patch tokens versus patch-only gFID. The feature target raises the dependence and lowers gFID for SigLIP, DINOv2, and DINOv3. Both axes are log-scaled. (c) With the feature target fixed, global-augmented generation improves gFID for all four encoders, achieving better generation than pixel-target baselines.

generated semantic codes (Lavoie et al., 2025). In our case, the global token directly discretizes the pretrained encoder's global representation.

Specifically, we use separate quantizers, Q_{global} and Q_{patch}, for global and patch features, respectively. The feature decoder D_{feat} takes both global and patch tokens and reconstructs the combined target $y = (y_{\text{global}}, y_{1:N})$ using the feature loss $\mathcal{L}_{\text{feat}}$. This yields a discrete global token z_{global}, which we model jointly with the patch tokens using the following factorization:

$$p_\phi(z_{\text{global}}, z_{1:N} \mid c) = p_\phi(z_{\text{global}} \mid c) \prod_{i=1}^{N} p_\phi(z_i \mid z_{\text{global}}, z_{<i}, c). \tag{1}$$

where c denotes the class label or text prompt. During inference, the generator first samples the global token and then samples patch tokens conditioned on both global and external conditions.

Combined training objective. We train the tokenizer and pixel decoder jointly using the feature and pixel losses described above. Following prior work on semantic preservation during tokenizer training (Chen et al., 2026a; Du et al., 2026), we also include an ℓ_2 regularizer $\mathcal{L}_{\text{reg}}$ between the last-layer features h and $\hat{h}$ of the trainable and frozen encoders. With $z = (z_{\text{global}}, z_{1:N})$, the complete objective is

$$\mathcal{L}_{\text{FeaTok}} = \mathbb{E}_x \Bigg[\underbrace{\mathcal{L}_{\text{feat}}(D_{\text{feat}}(z), y) + \lambda_{\text{reg}} \mathcal{L}_{\text{reg}}(h, \hat{h})}_{\text{trains encoder, quantizers, feature decoder}} + \underbrace{\lambda_{\text{pix}} \mathcal{L}_{\text{pix}}(D_{\text{pix}}(\text{sg}(z_{1:N})), x)}_{\text{trains only the pixel decoder}} \Bigg], \tag{2}$$

where the expectation is over training images x. The weights λ_{reg} and λ_{pix} balance the loss terms, and sg stops gradients at the pixel decoder input.

Once the tokenizer is trained, we keep it fixed and train an autoregressive generator on its tokens using teacher forcing. We combine the cross-entropy loss for the global token with the average loss over patch tokens, weighted by λ_{global} and λ_{patch}, respectively. These weights control the emphasis on global and patch prediction independently of the number of patches.

2.2 Controlled ablations across pretrained vision encoders

We verify FeaTok, feature-loss tokenization, against pixel-loss tokenization using four pretrained vision encoders: SigLIP (Zhai et al., 2023), SigLIP2 (Tschannen et al., 2025), DINOv2 (Oquab et al., 2024), and DINOv3 (Siméoni et al., 2026). We compare the generation performance of

a generative model trained on ImageNet-1k (Russakovsky et al., 2015) images at 256×256 resolution using different tokenizers. Details of ablation setups are given in App. C. Additionally, App. C.2 shows that tokenizers built on MAE (He et al., 2022), whose features are learned by pixel reconstruction, generate poorly under every supervision.

Feature targets work better than pixel-only or mixed feature–pixel targets. Feature supervision improves the generation over pixel supervision for three of the four encoders, as shown in Fig. 3a. With patch-only generation, replacing pixel targets with feature targets lowers gFID for SigLIP, DINOv2, and DINOv3, while combining pixel and feature losses, *i.e.*, letting gradients from both decoders update the discrete tokens, gives intermediate results. Fig. 3b relates these gFID changes to the spatial dependence between adjacent discrete patch tokens. The feature target raises this dependence and lowers gFID for SigLIP, DINOv2, and DINOv3, showing that the feature target improves the token structure for generation. SigLIP2 is the exception, possibly because, unlike the other three encoders, it is not trained only in feature space. Its training adds a text decoder that generates captions of whole images and of specific regions (Tschannen et al., 2025). App. C.4 details the estimator and presents other token-structure measures.

Global-augmented generation facilitates generative modelling. Holding the tokenizer fixed and adding the global conditioning to generative modeling lowers gFID from 1.70 to 1.61 for SigLIP, 1.80 to 1.62 for SigLIP2, 1.63 to 1.51 for DINOv2, and 1.48 to 1.39 for DINOv3, as shown in Fig. 3c. With the global token, FeaTok with SigLIP2 overtakes the counterpart trained with pixel reconstruction losses.

Leveraging features from multiple layers improves generation. The four-layer average gives the best measured gFID for each encoder in Fig. 4, with global-augmented generation throughout. Compared with the first and last layers, respectively, averaging yields the best generation performance consistently across all encoders. Tab. 11 in App. C reports the numbers presented in this section in detail, and with the accompanying reconstruction results for each ablation.

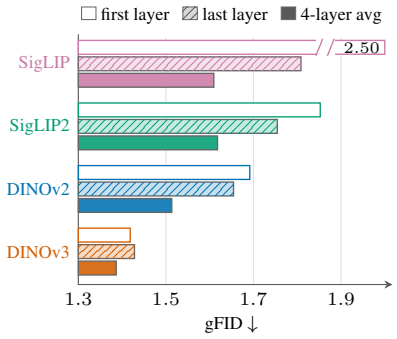


Figure 4: **Feature-target depth.** First-layer, last-layer and four-layer-average targets with global-augmented generation. The SigLIP first-layer bar has a break to accommodate its gFID of 2.50.

3 State-of-the-art image generation with discrete tokens

To demonstrate the effectiveness of our approach, we evaluate it on ImageNet-1k (Russakovsky et al., 2015) at 256×256 and 512×512 resolution, as well as text-to-image (T2I) generation. Our models achieve state-of-the-art performance among methods using discrete tokenizers and outperform recent continuous generative models at a matched inference cost on ImageNet-1k 256×256. FeaTok achieves better performance on ImageNet-1k 512×512 than both continuous and discrete baselines. On T2I, FeaTok outperforms RAEv2 (Singh et al., 2026) after a short fine-tuning on three T2I benchmarks, including GenEval (Ghosh et al., 2023), DPG-Bench (Hu et al., 2024), and GenAI-Bench (Li et al., 2024a).

3.1 ImageNet-1k generation at 256×256

We train our 256×256 FeaTok tokenizers on ImageNet-1k using SigLIP (Zhai et al., 2023), SigLIP2 (Tschannen et al., 2025), DINOv2 (Oquab et al., 2024), and DINOv3 (Siméoni et al., 2026) encoders, with a vocabulary size of $65,536$. For each tokenizer, we train the same autoregressive generator as BAR (Yu et al., 2026), the previous state-of-the-art discrete model for ImageNet-1k, with our global-augmented modelling. App. A specifies the implementation details including vision encoder versions, tokenizer and generator instantiations, *etc.*

Fréchet distance in the Inception space can misrank modern generative models, and measuring it in stronger feature spaces such as DINOv2 agrees better with human judgments (Stein et al., 2023). We therefore report gFDr6, the mean Fréchet-distance ratio across Inception-v3 (Szegedy et al., 2016), ConvNeXt V2 (Woo et al., 2023), DINOv2 (Oquab et al., 2024), MAE (He et al.,

Table 1: Comparison with discrete baselines on ImageNet-1k 256×256. FeaTok outperforms all discrete baselines with and without guidance. Size denotes the generator parameter count. TFLOPs reports the sampling cost per image without guidance. Baselines use their released sampling settings, and App. F reports FeaTok at gFID-selected settings. ‡VAR's unguided generator cost is estimated as half its guided cost. **Bold** and underline mark the best and second-best values across all models in each metric column.

			without guidance							with guidance						
method	size	TFLOPs	gFDr6	incep.	cvnx2	dino2	mae	siglip2	clip	gFDr6	incep.	cvnx2	dino2	mae	siglip2	clip
Pixel-based																
VAR-d30 (Tian et al., 2024)	2.0B	‡2.29	12.14	3.09	5.09	11.92	10.21	22.60	19.91	6.75	1.16	1.64	5.30	7.25	11.95	13.22
LlamaGen-3B (Sun et al., 2024)	3.1B	4.18	13.70	5.62	8.70	13.33	9.34	23.32	21.91	5.97	1.26	1.40	4.75	5.35	10.46	12.59
TiTok-S-128 (Yu et al., 2024b)	0.3B	5.08	21.45	5.11	10.51	22.62	14.22	41.96	34.27	15.32	1.19	1.98	14.04	13.53	30.23	30.96
Open-MAGVIT2-XL (Luo et al., 2024)	1.5B	1.16	18.70	6.85	12.83	19.89	10.29	34.18	28.13	7.72	1.39	2.24	7.15	5.53	14.98	15.03
MaskBit-305M (Weber et al., 2024)	0.3B	41.79	14.65	2.36	6.84	13.23	10.45	27.15	27.84	11.68	0.93	3.06	9.03	9.71	21.43	25.94
RAR-XXL (Yu et al., 2025a)	1.5B	0.96	19.31	2.50	7.07	16.95	12.50	32.90	43.93	15.07	0.88	2.71	10.23	11.39	24.73	40.48
Mixed																
Gigatok-XXL (Xiong et al., 2025)	1.4B	1.20	4.82	1.18	1.24	3.78	4.84	8.78	9.10	4.24	1.72	1.44	2.96	4.30	7.27	7.76
VFMTok2-XXL (Zheng et al., 2026a)	1.4B	1.09	4.38	1.22	0.99	3.25	4.82	7.73	8.29	4.37	2.32	1.79	2.98	4.67	7.01	7.44
BAR-B (Yu et al., 2026)	0.4B	0.67	5.15	1.01	1.70	5.21	5.82	9.40	7.73	4.89	0.69	0.95	4.63	5.89	9.32	7.86
BAR-L (Yu et al., 2026)	1.1B	1.14	4.57	**0.83**	1.38	4.59	5.22	8.14	7.26	4.11	**0.61**	0.79	3.72	5.02	7.49	7.06
Feature-based																
FeaTok-B-SigLIP	0.4B	0.71	4.46	1.58	0.89	4.49	5.86	8.00	5.94	4.14	1.28	0.75	3.96	5.51	7.71	5.63
FeaTok-B-SigLIP2	0.4B	0.71	4.00	1.42	0.98	3.79	4.87	7.52	5.41	3.46	1.30	0.86	3.09	4.14	6.29	5.05
FeaTok-B-DINOv2	0.4B	0.71	4.40	0.92	1.36	4.12	4.93	8.31	6.74	3.56	1.15	0.89	3.00	4.21	6.39	5.71
FeaTok-B-DINOv3	0.4B	0.71	3.58	1.09	0.82	3.30	4.62	6.50	5.15	3.39	1.06	**0.70**	3.00	4.43	6.28	4.91
FeaTok-L-SigLIP	1.1B	1.27	4.00	1.49	0.80	3.94	5.09	7.12	5.56	3.92	1.52	0.85	3.80	4.82	7.01	5.54
FeaTok-L-SigLIP2	1.1B	1.27	3.45	1.11	0.82	3.37	4.04	6.29	5.11	3.27	1.12	0.78	2.98	3.75	6.07	4.94
FeaTok-L-DINOv2	1.1B	1.27	3.47	0.87	0.83	3.11	4.13	6.33	5.52	3.20	0.95	0.74	**2.69**	3.77	5.79	5.28
FeaTok-L-DINOv3	1.1B	1.27	**3.13**	0.96	**0.68**	**2.95**	**4.00**	**5.57**	**4.62**	**3.08**	1.09	0.82	2.93	**3.73**	**5.28**	**4.65**

2022), SigLIP2 (Tschannen et al., 2025), and CLIP (Radford et al., 2021) feature spaces, following Yang et al. (2026). In App. E we detail the sampling configurations for each model. In App. F we report gFID (Heusel et al., 2017) and precision, recall (Kynkäänniemi et al., 2019), density and coverage (Naeem et al., 2020) in Inception space as complementary diagnostics. Unless stated otherwise, baselines are evaluated by us using their released setups.

FeaTok outperforms discrete models. In Tab. 1 we compare FeaTok using different vision encoders and two model sizes to existing discrete image generative models in terms of gFDr6 and its per-feature metrics. All four FeaTok-L variants outperform the strongest prior discrete baseline in gFDr6, showing consistent gains across four pretrained visual feature encoders satisfying Def. 1. FeaTok-DINOv3 is the strongest discrete model at both generator scales, with and without guidance. The largest improvements appear without guidance, where we reduce the prior baseline gFDr6 of 4.38 from VFMTok2-XXL (Zheng et al., 2026a) to 3.13. With guidance, we reduce the 4.11 gFDr6 of BAR-L to 3.08. Increasing the generator from base to large consistently improves gFDr6; for DINOv3, it decreases from 3.58 to 3.13 without guidance and from 3.39 to 3.08 with guidance. Importantly, these generation gains do not coincide with improved reconstruction: rFID remains nearly constant across FeaTok tokenizers (0.92–1.01; App. C.3), reinforcing that reconstruction fidelity alone does not explain generative performance. Figs. 8–11 in App. H show qualitative samples.

FeaTok is competitive with continuous models at lower sampling cost. Tab. 2 compares against continuous baselines at their standard sampling settings. Without guidance, FeaTok-L-DINOv3 achieves gFDr6 3.13, outperforming RAEv2 (3.95) while using 1.27 rather than 47.72 TFLOPs per image. With guidance, our 3.08 is competitive with RAE-XL (3.26) (Zheng et al., 2026b) and REPA-E (3.04) (Leng et al., 2025), although RAEv2 retains the best gFDr6 (2.17).

To compare at similar sampling costs, Tab. 3 evaluates RAEv2 with fewer sampling steps, tuning its internal guidance at each step count for the best gFDr6 (App. E). With guidance, FeaTok achieves gFDr6 3.08 at 2.31 TFLOPs per image, compared with 3.43 at 3.07 TFLOPs for six-step RAEv2. RAEv2 is not trained to perform few-step sampling. We therefore also compare with Drift (Deng et al., 2026), iMF (Geng et al., 2026), and pMF (Lu et al., 2026b), which are explicitly trained for one-step generation. Without guidance, their gFDr6 values range from 10.51 to 20.34, and even the best guided value is 6.87, whereas unguided FeaTok achieves 3.13. App. G provides detailed cost measurements, including FLOPs, wall-clock throughput, and peak memory.

Table 2: Comparison with baselines using continuous features on ImageNet-1k at 256×256. Without guidance, FeaTok achieves lower gFDr6 than all listed baselines at a lower sampling cost. Size denotes the generator parameter count, and TFLOPs reports the sampling cost per image without guidance. **Bold** and underline mark the best and second-best values in each metric column. Guided continuous-baseline results are transcribed from Singh et al. (2026).

			without guidance							with guidance						
method	size	TFLOPs	gFDr6	incep.	cvnx2	dino2	mae	siglip2	clip	gFDr6	incep.	cvnx2	dino2	mae	siglip2	clip
Pixel-based																
SiT-XL/2 (Ma et al., 2024)	675M	57.82	16.05	5.24	8.34	16.77	9.40	30.05	26.52	8.44	1.26	2.02	7.89	5.62	16.14	17.69
Mixed																
SiT-XL/2-REPA (Yu et al., 2025b)	675M	12.69	11.39	3.57	5.10	10.81	7.34	20.70	20.80	5.45	0.85	1.22	4.27	3.85	9.87	12.65
LightningDiT-XL/1 (Yao et al., 2025)	675M	57.45	6.22	1.33	2.42	5.77	4.08	11.77	11.95	4.57	0.85	1.09	3.76	3.02	8.47	10.21
DDT-XL/2 (Wang et al., 2026)	675M	64.25	12.72	3.86	5.52	11.89	8.20	23.06	23.78	5.70	0.75	1.02	4.26	4.11	10.16	13.86
REPA-E-XL/1 (Leng et al., 2025)	675M	59.57	4.32	1.03	2.35	3.81	**3.48**	7.38	7.84	3.04	0.70	1.28	2.44	**2.52**	5.04	6.28
REG-XL/2 (Wu et al., 2025)	675M	61.22	6.38	1.05	1.74	4.98	4.96	11.72	13.81	4.64	0.92	1.14	3.45	3.02	8.42	10.86
Feature-based																
RAE-XL (Zheng et al., 2026b)	839M	15.59	4.32	**0.91**	2.23	3.01	4.79	5.66	9.36	3.26	0.69	1.79	2.11	3.30	3.79	7.87
RAEv2-XL (Singh et al., 2026)	875M	47.72	3.95	1.00	1.41	**2.39**	5.83	5.66	7.41	**2.17**	**0.64**	**0.77**	**1.15**	2.67	**2.54**	5.21
FeaTok-L-DINOv3	1.1B	1.27	**3.13**	0.96	**0.68**	2.95	4.00	**5.57**	**4.62**	3.08	1.09	0.82	2.93	3.73	5.28	**4.65**

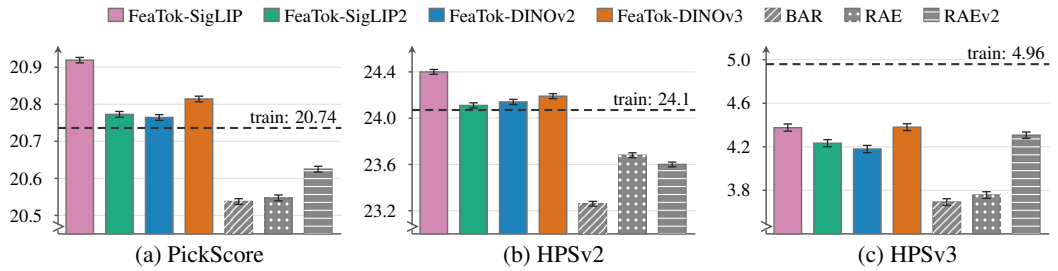
Table 3: Generation with matched inference cost on ImageNet 256×256. TFLOPs are computed without guidance per image. Guided one-step results are from Yang et al. (2026). Drift models share the same setup w and w/o guidance. **Bold** and underline mark the best and second-best values in each metric column.

			without guidance							with guidance						
method	steps	TFLOPs	gFDr6	incep.	cvnx2	dino2	mae	siglip2	clip	gFDr6	incep.	cvnx2	dino2	mae	siglip2	clip
Baselines with continuous features																
RAEv2-XL (Singh et al., 2026)	3	1.65	38.90	14.12	11.69	16.68	69.14	78.74	43.06	8.95	2.98	3.15	6.41	10.49	16.15	14.50
RAEv2-XL (Singh et al., 2026)	6	3.07	8.49	2.52	2.90	5.32	12.75	14.45	13.01	3.43	1.27	1.29	**2.11**	4.54	**4.92**	6.42
Baselines of one-step flow models																
Drift-L (latent) (Deng et al., 2026)	1	0.79	10.92	0.91	2.03	10.35	6.51	24.12	21.59	10.92	0.91	2.03	10.35	6.51	24.12	21.59
Drift-L (pixel) (Deng et al., 2026)	1	0.17	10.51	**0.85**	0.73	6.33	6.51	22.13	26.49	10.51	**0.85**	**0.73**	6.33	6.51	22.13	26.49
iMF-XL/2 (Geng et al., 2026)	1	0.97	20.34	6.33	9.26	19.10	13.28	41.90	32.15	8.39	1.09	1.72	7.30	6.09	17.02	17.14
pMF-L/16 (Lu et al., 2026b)	1	0.23	11.97	1.86	1.68	8.76	13.12	27.75	18.67	9.09	1.62	1.36	6.70	9.72	20.34	14.81
pMF-H/16 (Lu et al., 2026b)	1	0.54	11.79	2.37	2.38	9.58	10.28	27.79	18.34	6.87	1.37	1.15	5.43	6.25	15.33	11.68
FeaTok-L-DINOv3	256	1.27	**3.13**	0.96	**0.68**	**2.95**	**4.00**	**5.57**	**4.62**	**3.08**	1.09	0.82	2.93	**3.73**	5.28	**4.65**

FeaTok reaches higher human-preference rewards, closer to those of the training data. Fig. 5 compares generated and real ImageNet-1k data using PickScore (Kirstain et al., 2023), HPSv2 (Wu et al., 2023), and HPSv3 (Ma et al., 2025), all learned from human preference comparisons. These scores are automatic proxies for human evaluations, and complement the distributional measurements in Tabs. 1 and 2. FeaTok performs better than baselines both in terms of the raw value of the rewards and the closeness to the training data. On PickScore and HPSv2, the SigLIP2, DINOv2, and DINOv3 variants are closer to the real-data reference than every baseline. On the more discriminative HPSv3, FeaTok-DINOv3 and FeaTok-SigLIP have the reward scores closest to those of the training set, followed by RAEv2, all substantially closer than BAR-L and RAE.

3.2 FeaTok outperforms BAR-L and RAE at 512×512

Following Yu et al. (2026), we train a tokenizer for ImageNet-1k generation at 512×512 with the DINOv3 vision encoder, using vocabulary size $1,024$ per token. Implementation details are in Apps. A and B. We show the gFDr6 results for 512 resolution in Tab. 4. The benefit of FeaTok over baseline is larger at higher resolution. FeaTok-L-DINOv3 reduces the unguided gFDr6 of BAR-L from 3.82 to 2.17 and the guided result from 2.94 to 1.80. These are reductions of 43% and 39%, compared with 32% and 25% for the corresponding large-scale comparison at 256 resolution. Our method outperforms BAR-L in all feature spaces without guidance, and in all feature spaces except Inception with guidance. FeaTok also outperforms RAE in gFDr6, with 2.17 against 3.03 without guidance and 1.80 against 2.13 with guidance, although RAE is better in the Inception and SigLIP2 spaces in both settings. RAEv2 does not provide a model at 512 resolution.



(a) PickScore (b) HPSv2 (c) HPSv3

Figure 5: Human-preference rewards on 50K generated ImageNet images with 95% confidence intervals. FeaTok achieves the highest rewards compared to baselines, and also closer to the rewards measured on the training set. Dashed lines represent the mean of the rewards over training-data. HPSv2 scores are displayed multiplied by 100. FeaTok uses large generators sampled with guidance.

Table 4: Generation on ImageNet-1k 512 × 512. FeaTok outperforms BAR-L and RAE with and without guidance. **Bold** and underline mark the best and second-best values in each metric column. †RAE uses autoguidance instead of classifier-free guidance, and its unguided row uses the released configuration with autoguidance disabled.

	without guidance							with guidance						
method	gFDr6	incep.	cvnx2	dino2	mae	siglip2	clip	gFDr6	incep.	cvnx2	dino2	mae	siglip2	clip
Baselines														
RAE-XL† (Zheng et al., 2026b)	3.03	**0.97**	1.42	2.07	3.11	**4.40**	6.19	2.13	**0.75**	1.10	1.34	2.02	**2.74**	4.84
BAR-L (Yu et al., 2026)	3.82	1.19	1.77	2.86	2.80	8.98	5.30	2.94	0.78	1.02	2.24	2.54	5.81	5.22
FeaTok-L-DINOv3	**2.17**	1.16	**0.74**	**1.41**	**1.69**	4.63	**3.38**	**1.80**	0.94	**0.65**	**1.26**	**1.30**	3.51	**3.17**

3.3 Extension to text-to-image generation

Finally, we demonstrate the usefulness of FeaTok by applying it to text-to-image (T2I) generation. Following RAEv2 (Singh et al., 2026), we use BLIP-3o datasets (Chen et al., 2025) (Long-Caption, Short-Caption, and JourneyDB) for pre-training and BLIP-3o-60K for fine-tuning. We replace the class conditioning in FeaTok with a sequence of text embeddings provided by Qwen3-0.6B (Team, 2025) to enable T2I generation, re-using our FeaTok-SigLIP2 tokenizer trained on ImageNet-1k. We pre-train 75k updates with batch size 2048, while baselines use 150k updates with batch size 1024, giving the same total number of training examples processed. Then we further fine-tune the pre-trained model for 500 updates, while baselines are fine-tuned for approximately 3k updates.

After pre-training, FeaTok is broadly comparable to RAEv2. It is slightly lower on GenEval (Ghosh et al., 2023) (0.611 versus 0.624) and DPG-Bench (Hu et al., 2024) (79.8 versus 81.7), but higher on GenAI-Bench (Li et al., 2024a) (0.668 versus 0.638). After only 500 additional updates, FeaTok exceeds RAEv2 on all three benchmarks: 0.837 versus 0.827 on GenEval, 0.698 versus 0.680 on GenAI-Bench, and 83.4 versus 82.3 on DPG-Bench. The comparison with BAR (Yu et al., 2026) isolates the effect of the tokenizer directly, whose tokenizer is trained with pixel losses and a feature-alignment loss using SigLIP2. After pre-training, FeaTok is ahead of BAR on GenEval, slightly ahead on GenAI-Bench, and tied on DPG-Bench. After fine-tuning, all three benchmarks rank FeaTok above BAR, with 0.837 versus 0.817 on GenEval, 0.698 versus 0.688 on GenAI-Bench, and 83.4 versus 81.7 on DPG-Bench. The advantage of feature-loss tokens over mixed tokens thus carries over from class-conditional to text-to-image generation. We provide several T2I qualitative examples in Fig. 6. More qualitative samples are presented in App. I (Fig. 12).

4 Related work

Pretrained representations for continuous generation. Continuous generators show that the pretraining objective of the features matters. RAE (Zheng et al., 2026b) and RAEv2 (Singh et al., 2026) train diffusion models in the feature spaces of pretrained vision encoders, paired with learned image decoders, and RAE finds that MAE features (He et al., 2022) reconstruct best but generate

Table 5: Text-to-image results after pre-training and fine-tuning. FeaTok is broadly comparable to RAEv2, and surpasses all baselines after 500 fine-tuning updates. Continuous-latent baseline results are taken from Singh et al. (2026).

	after pre-training			after fine-tuning		
model	GenEval ↑	GenAI ↑	DPG ↑	GenEval ↑	GenAI ↑	DPG ↑
continuous models						
Flux-VAE (Labs, 2024)	0.417	0.573	77.6	0.783	0.639	79.2
RAE (Zheng et al., 2026b)	0.584	0.632	80.1	0.815	0.672	80.6
RAEv2 (Singh et al., 2026)	**0.624**	0.638	**81.7**	0.827	0.680	82.3
discrete models						
BAR (Yu et al., 2026)	0.564	0.664	79.8	0.817	0.688	81.7
FeaTok	0.611	**0.668**	79.8	**0.837**	**0.698**	**83.4**



a butterfly gently touching the vast surface of the lake, peace, serenity,

a photo of a cat and a dog on the beach wearing fancy hats

grizzly bear wearing medieval armor looking fierce and roaring

a realistic photo of a margarita cocktail

a hyperrealistic image of half a lemon dropping into clear water…

Figure 6: Text-to-image samples with abbreviated prompts. Full prompts are available in Tab. 17.

worse than DINOv2 features. REPA (Yu et al., 2025b) aligns intermediate SiT (Ma et al., 2024) features with pretrained representations, where MAE is the weakest of the tested targets, and DDT (Wang et al., 2026) adopts this alignment. REG (Wu et al., 2025) denoises a pretrained class token together with the image latents, VA-VAE (Yao et al., 2025) aligns VAE latents with a pretrained vision encoder, and REPA-E (Leng et al., 2025) tunes the VAE and the diffusion model end to end with the REPA loss.

Discrete tokenization with pretrained representations. VQ-VAE (Van Den Oord et al., 2017) learns discrete image tokens through pixel reconstruction, and later tokenizers simplify the quantizer (Yu et al., 2024a; Mentzer et al., 2024). Wang et al. (2024) train tokenizers to reconstruct the features from image understanding encoders, and in autoregressive generation all but the MAE teacher outperform pixel-trained VQGAN tokens. Recent tokenizers keep pixel reconstruction as the central objective and add pretrained encoder features through feature alignment (Yu et al., 2026), semantic regularization (Xiong et al., 2025), self-distillation (Du et al., 2026), or feature reconstruction from several encoder depths (Jia et al., 2025; Chen et al., 2026b; Zheng et al., 2025b; 2026a). TA-Tok (Han et al., 2025) trains feature-reconstructed tokens for unified multimodal models, but decodes images through pixel-trained VQ tokens or latents. In FeaTok, only pretrained visual features (Def. 1) shape the tokens, which reconstruct a multi-layer target and include a quantized global token.

5 Conclusion

We presented FeaTok, a discrete image tokenizer whose tokens are shaped only by feature reconstruction, while a pixel decoder trained on detached tokens renders them into images. We also discretize the encoder's global representation and model it together with the patch tokens. Across four pretrained encoders, FeaTok generates better than tokenizers trained with pixel or combined losses. FeaTok achieves the best gFDr6 among the evaluated discrete generators on ImageNet at 256×256 and 512×512, outperforms continuous generators at a matched sampling cost, achieves higher human-preference reward-model scores, and transfers to text-to-image generation after pretraining and only 500 fine-tuning updates. We believe these results are a step toward effective discrete representations for unified multimodal modeling, and that the low sampling cost of FeaTok makes it attractive for scaling multimodal training.

AI USE STATEMENT

In this work, we used generative AI tools for implementing methods, designing or providing feedback on research methodology or experiments, and interpreting results.

Generating synthetic datasets, helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, assisting with translation, cleaning and reformatting datasets, and supporting qualitative and thematic data analysis are not applicable to this work.

Additionally, we used generative AI tools for creating or modifying scientific figures or images, suggesting experimental parameters, creating or editing software code, drafting parts of a research paper, transcribing recordings of research material, summarizing or analysing existing literature, brainstorming, sourcing/searching for information, editing a research paper to improve readability, and identifying relevant literature.

We have reviewed all AI-assisted work. LLM-generated code was checked by the authors and validated by running the experiments, including reproducing the published results of the released baselines with our evaluation pipeline. All reported numbers were transcribed from the evaluation outputs and cross-checked against them. Figures are generated by scripts that read these outputs directly. AI-assisted interpretations of results and AI-drafted or AI-edited text were revised by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15619–15629. IEEE, 2023.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Roman Bachmann, Jesse Allardice, David Mizrahi, Enrico Fini, Oğuzhan Fatih Kar, Elmira Amirloo, Alaaeldin El-Nouby, Amir Zamir, and Afshin Dehghan. Flextok: Resampling images into 1d token sequences of flexible length. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=DgdOkUUBzf.

Randall Balestriero and Yann Lecun. How learning by reconstruction produces uninformative features for perception. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 2566–2585. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/balestriero24b.html.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jegou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9630–9640, 2021. doi: 10.1109/ICCV48922.2021.00951.

Bowei Chen, Sai Bi, Hao Tan, He Zhang, Tianyuan Zhang, Zhengqi Li, Yuanjun Xiong, Jianming Zhang, and Kai Zhang. Aligning visual foundation encoders to tokenizers for diffusion models. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=ajnBafpqmE.

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*, 2025.

Yitong Chen, Zijie Diao, Junke Wang, Lingyu Kong, Yixuan Ren, Bo He, Yu-Gang Jiang, and Zuxuan Wu. IDEAL: in-depth alignment makes A discrete representation autoencoder. *CoRR*, abs/2606.11096, 2026b.

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. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=fSnxuRhWoC.

Mingyang Deng, He Li, Tianhong Li, Yilun Du, and Kaiming He. Generative modeling via drifting. *arXiv preprint arXiv:2602.04770*, 2026.

Sinan Du, Jiahao Guo, Bo Li, Shuhao Cui, Zhengzhuo Xu, Yifu Luo, Yongxian Wei, Kun Gai, Xinggang Wang, Kai Wu, et al. Vqrae: Representation quantization autoencoders for multimodal understanding, generation and reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 30322–30334, 2026.

Zhengyang Geng, Yiyang Lu, Zongze Wu, Eli Shechtman, J. Zico Kolter, and Kaiming He. Improved mean flows: On the challenges of fastforward generative models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 30467–30476. Computer Vision Foundation, 2026.

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

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

Yuchao Gu, Xintao Wang, Yixiao Ge, Ying Shan, and Mike Zheng Shou. Rethinking the objectives of vector-quantized tokenizers for image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7631–7640. IEEE, 2024.

Jiaming Han, Hao Chen, Yang Zhao, Hanyu Wang, Qi Zhao, Ziyan Yang, Hao He, Xiangyu Yue, and Lu Jiang. Vision as a dialect: Unifying visual understanding and generation via text-aligned representations. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=ILr4UNiZcQ.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 15979–15988. IEEE, 2022.

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. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems*, pp. 6626–6637, 2017.

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.

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking FID: towards a better evaluation metric for image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9307–9315. IEEE, 2024.

Mingkai Jia, Mingxiao Li, Zhijian Shu, Anlin Zheng, Liaoyuan Fan, Jiaxin Guo, Tianxing Shi, Dongyue Lu, Zeming Li, Xiaoyang Guo, et al. Dino-tok: Adapting dino for visual tokenizers. *arXiv preprint arXiv:2511.20565*, 2025.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems*, pp. 3929–3938, 2019.

Tuomas Kynkäänniemi, Tero Karras, Miika Aittala, Timo Aila, and Jaakko Lehtinen. The role of imagenet classes in fréchet inception distance. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=4oXTQ6m_ws8.

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

Samuel Lavoie, Michael Noukhovitch, and Aaron Courville. Compositional discrete latent code for high fidelity, productive diffusion models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=1iSnpztjbD.

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. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18262–18272. IEEE, 2025.

Baiqi Li, Zhiqiu Lin, Deepak Pathak, Jiayao Li, Yixin Fei, Kewen Wu, Tiffany Ling, Xide Xia, Pengchuan Zhang, Graham Neubig, et al. Genai-bench: Evaluating and improving compositional text-to-visual generation. *arXiv preprint arXiv:2406.13743*, 2024a.

Daiqing Li, Aleks Kamko, Ehsan Akhgari, Ali Sabet, Linmiao Xu, and Suhail Doshi. Playground v2.5: Three insights towards enhancing aesthetic quality in text-to-image generation, 2024b.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36115–36125. Computer Vision Foundation, 2026.

Jiasen Lu, Liangchen Song, Mingze Xu, Byeongjoo Ahn, Yanjun Wang, Chen Chen, Afshin Dehghan, and Yinfei Yang. Atoken: A unified tokenizer for vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28701–28711, 2026a.

Yiyang Lu, Susie Lu, Qiao Sun, Hanhong Zhao, Zhicheng Jiang, Xianbang Wang, Tianhong Li, Zhengyang Geng, and Kaiming He. One-step latent-free image generation with pixel mean flows. *CoRR*, abs/2601.22158, 2026b.

Zhuoyan Luo, Fengyuan Shi, Yixiao Ge, Yujiu Yang, Limin Wang, and Ying Shan. Open-magvit2: An open-source project toward democratizing auto-regressive visual generation. *arXiv preprint arXiv:2409.04410*, 2024.

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. In *European Conference on Computer Vision*, pp. 23–40. Springer, 2024.

Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. HPSv3: Towards wide-spectrum human preference score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15086–15095, 2025.

Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: Vq-vae made simple. In *International Conference on Learning Representations*, volume 2024, pp. 51772–51783, 2024.

Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7176–7185. PMLR, 2020.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

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.

Vivek Ramanujan, Kushal Tirumala, Armen Aghajanyan, Luke Zettlemoyer, and Ali Farhadi. When worse is better: Navigating the compression generation trade-off in visual tokenization. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=o8hWyJIgAV.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seung Eun Yi, Michael Ramamonjisoa, Francisco Massa, Daniel HAZIZA, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Herve Jegou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=2NlGyqNjns. Featured Certification.

Jaskirat Singh, Boyang Zheng, Zongze Wu, Richard Zhang, Eli Shechtman, and Saining Xie. Improved baselines with representation autoencoders. *arXiv preprint arXiv:2605.18324*, 2026.

Ivan Skorokhodov, Sharath Girish, Benran Hu, Willi Menapace, Yanyu Li, Rameen Abdal, Sergey Tulyakov, and Aliaksandr Siarohin. Improving the diffusability of autoencoders. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=2hEDcA7xy4.

George Stein, Jesse C. Cresswell, Rasa Hosseinzadeh, Yi Sui, Brendan Leigh Ross, Valentin Villecroze, Zhaoyan Liu, Anthony L. Caterini, Eric Taylor, and Gabriel Loaiza-Ganem. Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=08zf7kTOoh.

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.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826. IEEE Computer Society, 2016.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

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, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Luting Wang, Yang Zhao, Zijian Zhang, Jiashi Feng, Si Liu, and Bingyi Kang. Image understanding makes for a good tokenizer for image generation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=RMmgu49lwn.

Shuai Wang, Zhi Tian, Weilin Huang, and Limin Wang. Ddt: Decoupled diffusion transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 40633–40642, 2026.

Mark Weber, Lijun Yu, Qihang Yu, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. Maskbit: Embedding-free image generation via bit tokens. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=NYe2JuN3v3. Featured Certification, Reproducibility Certification.

Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. Convnext V2: co-designing and scaling convnets with masked autoencoders. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16133–16142. IEEE, 2023.

Ge Wu, Shen Zhang, Ruijing Shi, Shanghua Gao, Zhenyuan Chen, Lei Wang, Zhaowei Chen, Hongcheng Gao, Yao Tang, jian Yang, Ming-Ming Cheng, and Xiang Li. Representation entanglement for generation: Training diffusion transformers is much easier than you think. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=koEALFNBj1.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

Tianwei Xiong, Jun Hao Liew, Zilong Huang, Jiashi Feng, and Xihui Liu. Gigatok: Scaling visual tokenizers to 3 billion parameters for autoregressive image generation. In *IEEE/CVF International Conference on Computer Vision*, pp. 18770–18780. IEEE, 2025.

Jiawei Yang, Zhengyang Geng, Xuan Ju, Yonglong Tian, and Yue Wang. Representation fréchet loss for visual generation. *arXiv preprint arXiv:2604.28190*, 2026. URL https://arxiv.org/abs/2604.28190.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15703–15712. IEEE, 2025.

Lijun Yu, José Lezama, Nitesh Bharadwaj Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, et al. Language model beats diffusion-tokenizer is key to visual generation. In *International Conference on Learning Representations*, volume 2024, pp. 765–783, 2024a.

Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation. *Advances in Neural Information Processing Systems*, 37:128940–128966, 2024b.

Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Randomized autoregressive visual generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18431–18441. IEEE, 2025a.

Qihang Yu, Qihao Liu, Ju He, Xinyang Zhang, Yang Liu, Liang-Chieh Chen, and Xi Chen. Autoregressive image generation with masked bit modeling. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=jALSox02QD.

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. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=DJSZGGZYVi.

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.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595. Computer Vision Foundation / IEEE Computer Society, 2018.

Anlin Zheng, Haochen Wang, Yucheng Zhao, Weipeng Deng, Tiancai Wang, Xiangyu Zhang, and Xiaojuan Qi. Holistic tokenizer for autoregressive image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16916–16926, 2025a.

Anlin Zheng, Xin Wen, Xuanyang Zhang, Chuofan Ma, Tiancai Wang, Gang YU, Xiangyu Zhang, and XIAOJUAN QI. Vision foundation models as effective visual tokenizers for autoregressive generation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b. URL https://openreview.net/forum?id=PESrAH82Zh.

Anlin Zheng, Qi Han, Xin Wen, Chuofan Ma, Lanxi Gong, Gang Yu, Xiangyu Zhang, and Xiaojuan Qi. Vision foundation models as generalist tokenizers for image generation. *arXiv preprint arXiv:2605.18390*, 2026a.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. In *International Conference on Learning Representations*, volume 2026, pp. 35791–35820, 2026b.

A TOKENIZER IMPLEMENTATION DETAILS

This section specifies the final FeaTok tokenizer used in Sec. 3. We follow the notation presented in Sec. 2. We follow BAR (Yu et al., 2026), the previous state-of-the-art method, in our tokenizer and generator training. BAR uses bit representations for its tokens and an autoregressive model with a masked modeling head to predict the bits within each token. Ablation protocols and variants are described in App. C.

Which pretrained encoders we use. FeaTok uses SigLIP-L, SigLIP2-SO400M, DINOv2-L with registers, or DINOv3-L. Tab. 6 identifies their pretrained checkpoints. Note that DINOv2-L uses a patch size of 14, whereas the other encoders use a patch size of 16. For DINOv2-L, we resize the image to 224×224 so that all encoders encode each image into 16×16 patches.

Table 6: Pretrained source encoders and checkpoints, shared by the frozen target encoder $\hat{E}$ and the initialization of the trainable encoder E.

name in main text	variant	checkpoint	hidden dim	patch size
SigLIP	ViT-L/16	`google/siglip-large-patch16-256`	1024	16
SigLIP2	SO400M/16, NaFlex	`google/siglip2-so400m-patch16-naflex`	1152	16
DINOv2	ViT-L/14, registers	`facebook/dinov2-with-registers-large`	1024	14
DINOv3	ViT-L/16	`facebook/dinov3-vitl16-pretrain-lvd1689m`	1024	16

Quantizer implementation. We follow BAR and use finite scalar quantization (FSQ) (Mentzer et al., 2024) with 16 2-level slots for tokenizers trained on ImageNet-1k at 256×256. A trainable input projector $P_{\text{in},\theta} : \mathbb{R}^D \to \mathbb{R}^d$ maps the adapted backbone features h to $d = 16$ scalar latents, $u_i = P_{\text{in},\theta}(E^{(L)}(x)_i)$, where L denotes the final encoder layer and i indexes the N image patches. These latents are quantized coordinatewise to two levels. For coordinate ℓ with output levels $a_\ell^- < a_\ell^+$ and threshold b_ℓ,

$$B_{i\ell} = \mathbf{1}[u_{i\ell} \geq b_\ell], \qquad q_{i\ell} = a_\ell^- + (a_\ell^+ - a_\ell^-)B_{i\ell}. \tag{3}$$

The bit vector $B_i \in \{0,1\}^d$ indexes one of 2^{16} patch tokens. We use the straight-through variable $q_i^{\text{st}} = u_i + \text{sg}(q_i - u_i)$, where sg is stop-gradient. The feature decoder reads the straight-through codes through its own input projections, which carry the feature-loss gradient to $P_{\text{in},\theta}$ and the encoder. A separate trainable output projector $P_{\text{out},\theta}$ maps the codes to the pixel-decoder width and belongs to the pixel branch only. Its input is detached, so the two decoders share only the discrete codes. Both projections are learned; only the coordinatewise rounding rule is fixed. Other differentiable quantizers can replace this module without changing the FeaTok framework.

Discrete global token. In addition to its N patch features, E emits a global feature. A separate global quantizer Q_{global} maps this feature to a 16-bit global code z_{global}, which is prepended to the patch codes $z_{1:N}$. Given the external class label c, the generator models $p(z_{\text{global}}, z_{1:N} \mid c)$ and samples z_{global} first, so it supplies global context to the patch tokens. The feature decoder reconstructs both global and patch feature targets, so the global code receives direct reconstruction supervision. The patch feature target is the average of representations from four layers, whereas the global target is the last-layer representation. The pixel decoder consumes only the sampled patch tokens, allowing a fair comparison between feature-loss and pixel-loss targets.

Architecture and training. The trainable encoder E and the frozen encoder $\hat{E}$ are initialized from the same pretrained checkpoint (Tab. 6). The feature decoder D_{feat} is a ViT matched to the encoder, and the pixel decoder D_{pix} is the ViT-L decoder of BAR (Yu et al., 2026), shared across all encoders. Tab. 7 lists the architectures and training hyperparameters. All tokenizers are trained on NVIDIA H200 GPUs.

Losses. $\mathcal{L}_{\text{feat}}$ sums ℓ_1, ℓ_2, and cosine distances and averages the patch and global terms. $\mathcal{L}_{\text{reg}}$ is the mean squared error between the last-layer features of E and $\hat{E}$. Following BAR, $\mathcal{L}_{\text{pix}}$ combines pixel, perceptual, Gram, and adversarial losses with an alignment loss that regresses the frozen last-layer features from the third pixel-decoder block through an MLP. All pixel-branch terms act on stop-gradient tokens, so they update only D_{pix} and the output projector $P_{\text{out},\theta}$, which no other module uses. The encoder, the input projector, and the feature decoder receive gradients only from $\mathcal{L}_{\text{feat}}$ and $\mathcal{L}_{\text{reg}}$. Because FSQ rounds each coordinate to fixed levels without a learned codebook, it needs no commitment or codebook loss. Tab. 7 gives all loss weights.

Table 7: FeaTok tokenizer architecture and training hyperparameters at 256×256, shared across encoders unless noted. We follow the tokenizer setup of BAR (Yu et al., 2026) and change only the parts specific to FeaTok: the feature decoder and feature losses, the encoder regularizer, the discrete global token, and the stop-gradient that restricts the pixel losses to the pixel decoder. Epochs are over ImageNet-1k, with optimizer steps in parentheses.

category	parameter	value
architecture	feature decoder D_{feat}	ViT, 24 layers, width 1024 (ViT-L encoders), 27 layers, width 1152 (SigLIP2)
	pixel decoder D_{pix}	ViT-L, 24 layers, width 1024, patch size 16
	quantizer	FSQ, 16 bits per patch token and 16 bits for the global token
optimization	optimizer	AdamW, $\beta = (0.9, 0.999)$
	learning rate	10^{-4} to 10^{-5}, cosine decay
	encoder learning rate	$0.1\times$ the learning rate
	weight decay	0
	warmup	2 epochs (10K steps)
	training length	40 epochs (200K steps)
	batch size	256
	precision, gradient clipping	bf16, max grad norm 1.0
	weight averaging	EMA, decay 0.999
feature losses	$\mathcal{L}_{\text{feat}}$ (ℓ_1, ℓ_2, cosine)	1.0, 1.0, 1.0
	λ_{reg}	0.25 (SigLIP), 0.25 (SigLIP2), 0.5 (DINOv2), 0.25 (DINOv3)
pixel losses	ℓ_2, ℓ_1 reconstruction	1.0, 0.1
	perceptual	1.1 (LPIPS and ConvNeXt-S output loss, $1.0 : 0.1$)
	Gram	10
	adversarial	0.05, discriminator on a frozen DINO ViT-S/8 backbone, from 20 epochs
	decoder feature alignment	0.7
	λ_{pix}	1.0
data	augmentation	resize to 256, random 256×256 crop, random flip

Table 8: FeaTok tokenizer architecture and training hyperparameters at 512×512, used for FeaTok-L-DINOv3 in Tab. 4. *Entries that differ from the 256×256 recipe in Tab. 7. Epochs are over ImageNet-1k, with optimizer steps in parentheses.

category	parameter	value
architecture	encoder input*	DINOv3-L on the full 512×512 image, 32×32 grid of 1024 patch tokens
	feature decoder D_{feat}	ViT, 24 layers, width 1024
	pixel decoder D_{pix}	ViT-L, 24 layers, width 1024, patch size 16
	quantizer*	FSQ, 10 bits per patch token and 10 bits for the global token
	memory*	gradient checkpointing
optimization	optimizer	AdamW, $\beta = (0.9, 0.999)$
	learning rate	10^{-4} to 10^{-5}, cosine decay
	encoder learning rate	$0.1\times$ the learning rate
	weight decay	0
	warmup	2 epochs (10K steps)
	training length	40 epochs (200K steps)
	batch size	256, from 4 gradient-accumulation steps*
	precision, gradient clipping	bf16, max grad norm 1.0
	weight averaging	EMA, decay 0.999
feature losses	$\mathcal{L}_{\text{feat}}$ (ℓ_1, ℓ_2, cosine)	1.0, 1.0, 1.0
	λ_{reg}	0.25
pixel losses	ℓ_2, ℓ_1 reconstruction	1.0, 0.1
	perceptual	1.1 (LPIPS and ConvNeXt-S output loss, $1.0 : 0.1$)
	Gram	10
	adversarial	0.05, frozen DINO ViT-S/8 discriminator at 448×448 input*, from 20 epochs
	decoder feature alignment	0.7
	λ_{pix}	1.0
data	augmentation*	resize to 512, random 512×512 crop, random flip

Tokenizer at 512×512. For ImageNet-1k at 512×512 (Sec. 3.2), we train a separate tokenizer from DINOv3-L rather than fine-tuning the 256×256 one. The encoder processes the full 512×512 image, giving a 32×32 grid of 1024 patch tokens. Following BAR (Yu et al., 2026), each patch token and the global token carry 10 bits, so the quantizer has 2^{10} possible tokens. The feature target, losses, and optimization follow the 256×256 recipe. Tab. 8 lists all settings and marks those that differ from Tab. 7.

B GENERATOR IMPLEMENTATION DETAILS

We follow BAR (Yu et al., 2026) to implement the generative model, with our global-augmented generative modelling.

BAR generative model. Recall that BAR learns tokens with FSQ in a bit representation and models their distribution with an adapted autoregressive model. Instead of a single categorical prediction over the entire codebook, BAR uses a lightweight masked bit modeling head. During training, some bits of the target token are masked, and the head learns to recover them with a bit-wise cross-entropy objective. During sampling, it starts with all bits masked and progressively fills them in, conditioning later predictions on the bits already revealed. The autoregressive context stays fixed during these within-token steps. Once the token is complete, it becomes part of the context for the next token. This avoids an output distribution with one logit per codebook entry while allowing dependencies among a token's bits.

The token-level training objective described in Sec. 2.1 is

$$\mathcal{L}_{\text{gen}} = \mathbb{E}_{(z,c)}\left[\lambda_{\text{global}}\,\ell_{\text{CE}}(z_{\text{global}} \mid c) + \lambda_{\text{patch}}\,\frac{1}{N}\sum_{i=1}^{N}\ell_{\text{CE}}(z_i \mid z_{\text{global}}, z_{<i}, c)\right], \tag{4}$$

where ℓ_{CE} is the conditional cross-entropy prediction loss.

Our implementation. We instantiate the conditional cross-entropy in Eq. 4 with BAR's masked bit modeling objective (Yu et al., 2026). Let $B_j = (B_{j1}, \ldots, B_{jd}) \in \{0,1\}^d$ be the clean bit representation of token z_j, with $d = 16$, where $\ell \in \{1, \ldots, d\}$ indexes the bits. For each token, we sample a masking ratio r_j and mask $n_j = \text{round}(d\,r_j)$ of its bits, chosen uniformly at random, giving a mask $m_j \in \{0,1\}^d$ with $m_{j\ell} = 1$ if bit ℓ is masked. Following BAR, patch tokens draw $r_j = \max(1-t,\, 1/d)$ with $t = \sigma(\epsilon)$, $\epsilon \sim \mathcal{N}(0, 0.8^2)$, and σ the sigmoid. The global token, which BAR does not have, draws $t \sim \mathcal{U}[0,1]$ instead. The lower bound $r_j \geq 1/d$ ensures $n_j \geq 1$, so every token has at least one masked bit to predict. Replacing each masked bit $B_{j\ell}$ by a mask symbol gives $\widetilde{B}_j$. The prediction head estimates the likelihood of the clean bits conditioned on $\widetilde{B}_j$, the masking ratio r_j, and the autoregressive context C_j. The mask positions are already encoded by the mask symbols in $\widetilde{B}_j$. During teacher forcing, $C_{\text{global}} = c$ and $C_i = (z_{\text{global}}, z_{<i}, c)$ for patch i.

Let $\mathcal{S}$ be a set of token positions, either the global token $\mathcal{S}_{\text{global}} = \{\text{global}\}$ or the patch tokens $\mathcal{S}_{\text{patch}} = \{1, \ldots, N\}$. For a training batch of M sequences, indexed by b, the masked-bit loss over $\mathcal{S}$ is the weighted cross-entropy over all bits of the tokens in $\mathcal{S}$:

$$\ell_{\text{MBM}}(\mathcal{S}) = -\frac{\displaystyle\sum_{b=1}^{M}\sum_{j\in\mathcal{S}}\sum_{\ell=1}^{d} w_{j\ell}^{(b)} \log p_{\phi,\ell}\Big(B_{j\ell}^{(b)} \mid \widetilde{B}_j^{(b)}, r_j^{(b)}, C_j^{(b)}\Big)}{\displaystyle\sum_{b=1}^{M}\sum_{j\in\mathcal{S}}\sum_{\ell=1}^{d} w_{j\ell}^{(b)}}, \tag{5}$$

where $p_{\phi,\ell}$ is the head's two-class prediction at bit position ℓ and $w_{j\ell} = m_{j\ell} + \eta(1 - m_{j\ell})$ weights masked bits by 1 and visible bits by $\eta = 0.1$. The complete objective combines the global and patch losses:

$$\mathcal{L}_{\text{gen}}^{\text{MBM}} = \lambda_{\text{global}}\,\ell_{\text{MBM}}(\mathcal{S}_{\text{global}}) + \lambda_{\text{patch}}\,\ell_{\text{MBM}}(\mathcal{S}_{\text{patch}}), \tag{6}$$

with $\lambda_{\text{patch}} = 1 - \lambda_{\text{global}}$, averaged over training batches and over the sampled masks. Each term $-\log p_{\phi,\ell}(B_{j\ell} \mid \cdot)$ is the cross-entropy between the observed bit and the head's two-class prediction, so Eq. 6 specifies the token prediction loss ℓ_{CE} in Eq. 4 for our masked bit modeling head. Because the normalization pools the bits of all tokens in $\mathcal{S}$, a token with more masked bits contributes more, so the patch loss is a bit-weighted average over patch positions rather than exactly the uniform average $\frac{1}{N}\sum_{i=1}^{N}$ of Eq. 4. The bit prediction task additionally conditions on the visible bits within the current token, making it a denoising cross-entropy objective rather than an exact evaluation of the complete token's autoregressive negative log-likelihood.

FeaTok retains the BAR generator backbone and masked-bit prediction mechanism while changing the tokenizer that supplies its training codes. For global-augmented generation, we additionally prepend a sampled global token to the patch token sequence. This global token is image-dependent

Table 9: Generator architecture and training hyperparameters at 256×256. We follow the generator setup of BAR (Yu et al., 2026) and add only the global token: its conditioning, its joint condition dropout, and the loss weight λ_{global}. Epochs are over ImageNet-1k, with optimizer steps in parentheses.

category	parameter	BAR-B	BAR-L
architecture	layers	24	32
	width	768	1280
	attention heads	12	16
	masked bit modeling head	3 layers, width 2048	
conditioning	class condition tokens	32	
	global token repeats	32	
	condition dropout	10% of samples	
	dropped conditions	60% both, 20% class only, 20% global only	
optimization	optimizer	AdamW, $\beta = (0.9, 0.96)$	
	learning rate	4×10^{-4} to 10^{-5}, cosine decay	
	weight decay	0.03	
	warmup	100 epochs (62.5K steps)	
	random-order annealing	200 to 300 epochs (125K to 187.5K steps)	
	training length	400 epochs (250K steps)	
	batch size	2048	
	precision, gradient clipping	bf16, max grad norm 1.0	
	dropout, attention dropout	0.0, 0.0	0.2, 0.2
	weight averaging	none	
loss	visible-bit weight η	0.1	
	patch mask ratio	$\max(1-t,\ 1/16),\ t = \sigma(\epsilon),\ \epsilon \sim \mathcal{N}(0, 0.8^2)$	
	global mask ratio	$\max(1-t,\ 1/16),\ t \sim \mathcal{U}[0, 1]$	
	λ_{global}, SigLIP	0.5	$1/257$
	λ_{global}, SigLIP2	$1/257$	$1/257$
	λ_{global}, DINOv2	0.2	0.2
	λ_{global}, DINOv3	0.2	0.2

and thus distinct from the external class label or text condition c. BAR-B and BAR-L denote the base and large generator configurations. In our experiments for ImageNet-1k at 256×256, each patch token contains 16 bits, corresponding to 2^{16} possible tokens. For ImageNet-1k at 512×512, we use 10 bits per token following BAR (Yu et al., 2026). We use allocation to specify the unmasking process for each token during the sampling, *e.g.*, $[4, 4, 4, 4]$ specifies four within-token sampling steps, each revealing four bits. These within token sampling steps are distinct from the autoregressive steps over patch positions. App. C.1 gives the fixed ablation sampling protocol, and App. E lists the sampling settings used for the main comparisons.

Architecture and training. Tab. 9 lists the generator architectures and training hyperparameters, which follow BAR (Yu et al., 2026). The class label enters as repeated condition tokens, and for global-augmented generation the global token is likewise repeated after them. For classifier-free guidance, a fixed fraction of training samples drop their conditions. Most of these drop both the class label and the global token, and the rest drop only one of them, so the generator also learns each condition on its own. Following BAR, training starts with randomly permuted token orders, linearly anneals the probability of a random order to zero, and ends with raster order. The global-token loss weight λ_{global} of Eq. 6 depends on the encoder and, for SigLIP, on the generator size, with $\lambda_{\text{patch}} = 1 - \lambda_{\text{global}}$. The value $1/257$ weights the global token roughly like each of the 256 patch tokens, and the ablation generators use $\lambda_{\text{global}} = 0.5$. Generators train on tokens precomputed with the EMA weights of their tokenizers, on NVIDIA H200 GPUs.

Generator at 512×512. At 512×512, we train a single BAR-L generator on the tokens of the 512×512 tokenizer (Tab. 8). It models a sequence of one global token followed by 1024 patch tokens of 10 bits each. Following the 512×512 recipe of BAR (Yu et al., 2026), it trains for 200 epochs, half the 256×256 schedule, with warmup and random-order annealing scaled accordingly. We set $\lambda_{\text{global}} = 1/1025$, which weights the global token like each of the 1024 patch tokens. Tab. 10 lists all settings and marks those that differ from the BAR-L column of Tab. 9.

Table 10: Generator architecture and training hyperparameters at 512×512, used for FeaTok-L-DINOv3 in Tab. 4. *Entries that differ from the BAR-L column of the 256×256 recipe in Tab. 9. Epochs are over ImageNet-1k, with optimizer steps in parentheses.

category	parameter	BAR-L
architecture	layers	32
	width	1280
	attention heads	16
	masked bit modeling head	3 layers, width 2048
	sequence*	1 global token and 1024 patch tokens, 10 bits each
conditioning	class condition tokens	32
	global token repeats	32
	condition dropout	10% of samples
	dropped conditions	60% both, 20% class only, 20% global only
optimization	optimizer	AdamW, $\beta = (0.9, 0.96)$
	learning rate	4×10^{-4} to 10^{-5}, cosine decay
	weight decay	0.03
	warmup*	50 epochs (31.25K steps)
	random-order annealing*	100 to 150 epochs (62.5K to 93.75K steps)
	training length*	200 epochs (125K steps)
	batch size	2048
	precision, gradient clipping	bf16, max grad norm 1.0
	dropout, attention dropout	0.2, 0.2
	weight averaging	none
loss	visible-bit weight η	0.1
	patch mask ratio*	$\max(1-t,\ 1/10), t = \sigma(\epsilon), \epsilon \sim \mathcal{N}(0, 0.8^2)$
	global mask ratio*	$\max(1-t,\ 1/10), t \sim \mathcal{U}[0, 1]$
	λ_{global}*	$1/1025$

Table 11: **Ablations for FeaTok.** Feature-only loss outperforms pixel and combined pixel–feature losses for three pretrained encoders. Jointly modeling global and patch tokens with feature-only supervision achieves the best gFID for all four encoders. We report rFID on the ImageNet validation set and gFID on 50K ImageNet samples without guidance, with lower values indicating better performance. The tokenizer target specifies which losses train the discrete tokens, and the generator type specifies whether global-augmented generative modeling is used. Feature denotes the normalized average of four encoder layers, whereas first layer and last layer use a single encoder layer as the target. rFID depends only on the tokenizer, so rows that share a tokenizer report the same value. **Bold** and underline indicate the best and second-best result in each metric column.

		SigLIP		SigLIP2		DINOv2		DINOv3	
tokenizer target	generator type	rFID	gFID	rFID	gFID	rFID	gFID	rFID	gFID
pixel	patch-only	1.40	1.88	1.16	1.68	<u>0.85</u>	1.76	**0.72**	2.92
pixel + feature	patch-only	1.26	1.74	1.08	1.74	**0.83**	1.72	<u>0.73</u>	1.98
feature	patch-only	<u>1.01</u>	<u>1.70</u>	**0.92**	1.80	0.93	1.63	0.99	1.48
pixel + feature	global-augmented	1.26	1.77	1.08	<u>1.65</u>	**0.83**	<u>1.60</u>	<u>0.73</u>	1.67
first layer	global-augmented	**0.93**	2.50	<u>1.03</u>	1.85	1.26	1.69	1.03	<u>1.42</u>
last layer	global-augmented	1.56	1.81	1.57	1.75	1.65	1.65	1.26	1.43
feature (FeaTok)	global-augmented	<u>1.01</u>	**1.61**	**0.92**	**1.62**	0.93	**1.51**	0.99	**1.39**

C Ablation details

C.1 Ablation protocol and sampling without guidance

The controlled ablations in Sec. 2.2 compare tokenizer design choices using BAR generators (Yu et al., 2026) specified in App. B at 256×256 resolution. We evaluate 50K generated images under a fixed sampling configuration, rather than choosing a separate best sampling configuration for each ablation. We use guidance scale 1.0, *i.e.*, no guidance, sampling temperature 2.0, and bit-group allocation $[4, 4, 4, 4]$. For more details about the sampling hyperparameters, check App. E. For global-augmented models, the global token is sampled first with the same sampling hyperparameters, and then conditions patch generation. Patch-only models omit the global token, but retain class conditioning. We use the unguided protocol for ablation because it avoids the additional

Table 12: Reconstruction of regularized tokenizers on ImageNet at 256×256. Feature denotes the four-layer-average target. **Bold** marks the better result within each source block.

source	supervision	rFID ↓	rFDr6 ↓	PSNR ↑	SSIM ↑
SigLIP	pixel	1.40	2.68	**17.89**	**0.426**
	feature	**1.01**	**2.45**	17.27	0.418
SigLIP2	pixel	1.16	2.09	**18.81**	**0.466**
	feature	**0.92**	**2.04**	17.67	0.433
DINOv2	pixel	**0.85**	**1.82**	**20.04**	**0.519**
	feature	0.93	2.06	17.50	0.433
DINOv3	pixel	**0.72**	2.24	**20.61**	**0.545**
	feature	0.99	**2.08**	17.45	0.424

unconditional prediction used for classifier-free guidance and removes guidance strength as an additional sampling hyperparameter. The full results presented in Sec. 2.2 are detailed in Tab. 11.

C.2 Tokenizers from an MAE encoder

We also trained tokenizers from MAE-L/16 (He et al., 2022), whose features are learned by pixel reconstruction and therefore fall outside Def. 1. These runs use the same tokenizer and BAR-B generator recipes, with 250K generator steps and 50K samples without guidance at patch temperature 2.0 and allocation $[4, 4, 4, 4]$. They differ in three respects: they omit source-preservation regularization, use the last MAE layer as the feature target, and use MAE's auxiliary class token as the global token, sampled at temperature 2.5 and allocation $[2, 2, 5, 7]$. MAE-based tokenizers generate poorly under every supervision. With patch-only generation, gFID is 5.74 for the pixel target, 5.94 for combined pixel and feature targets, and 6.40 for the feature target, and global-augmented generation reaches 5.55 and 5.75 with the combined and feature targets. Feature supervision thus does not improve MAE-based tokens, and every MAE configuration stays far above the gFID range of Tab. 11, so we do not pursue MAE further.

C.3 Reconstruction results

Tab. 12 reports reconstruction for four source encoders under pixel and feature supervision, using regularized tokenizers throughout. With pixel supervision, the pixel loss trains the encoder and quantizer. With feature supervision, the pixel gradient is cut at the discrete representation, and the feature decoder reconstructs the four-layer-average target.

Across the four encoders, rFID ranges from 0.72 to 1.40. The six-space view complements rFID: feature targets improve rFDr6 over pixel targets for SigLIP (2.68 to 2.45), SigLIP2 (2.09 to 2.04), and DINOv3 (2.24 to 2.08), but not DINOv2 (1.82 to 2.06). These reconstruction scores alone do not explain the generation differences in Sec. 3.1.

Pixel fidelity versus perceptual quality. Feature supervision lowers PSNR and SSIM for all four encoders, *e.g.*, from 20.61 to 17.45 PSNR for DINOv3, so the tokens no longer reproduce each input pixel by pixel. For image generation, however, the goal is images that look real to humans, not pixel-exact copies of a given input. PSNR and SSIM compare each reconstruction with its input pixel by pixel, so they penalize small shifts of texture and detail that humans barely notice, and they correlate poorly with human perceptual judgments (Zhang et al., 2018). Distribution-level metrics instead ask whether the images look like real images, and they favor feature supervision. Feature targets improve rFDr6 for three of the four encoders, and the same feature-supervised tokenizers give lower gFID in the controlled ablations (Tab. 11), the best gFDr6 among discrete generators (Tab. 1), and higher human-preference reward-model scores (Fig. 5). The lower paired pixel fidelity is therefore a trade-off rather than a free improvement. Feature-supervised tokens discard some input detail, and not all of it need be perceptually irrelevant, but for generation the trade-off favors feature supervision in the distribution-level and reward-model evaluations above.

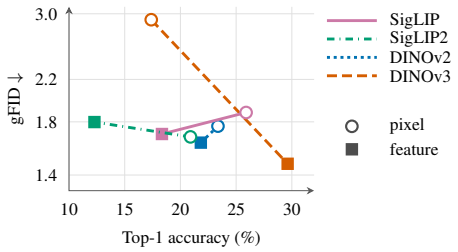


Figure 7: **Linear separability and generation.** Discrete-token probe accuracy on 25K test images versus patch-only gFID on 50K samples without guidance. The gFID axis is log-scaled.

C.4 Token Structure Measurement

Spatial-dependence diagnostic. The spatial dependence in Fig. 3b measures how informative one patch token is about an adjacent patch token in the same image. Rather than estimating MI between complete categorical token IDs, we use a component-wise proxy based on the tokenizer's binary code representation. Each patch token is represented by 16 bits. For each pair of patch positions at a given offset, we estimate the MI, in bits, between pairs of these bits from their joint occurrence frequencies across 25K class-balanced ImageNet training images. We average these estimates over the 256 bit pairs, over all position pairs of the offset, and over the four cardinal directions. Higher values indicate stronger statistical dependence between the bits of spatially separated tokens. The same estimator is applied to all eight tokenizers. Each point in Fig. 3b pairs the estimate for adjacent patches with the gFID of the patch-only generator from Tab. 11. Replacing the pixel target by the feature target raises the dependence for SigLIP, DINOv2, and DINOv3, but it lowers the dependence for SigLIP2. This diagnostic measures component-wise dependence in the tokenizer's code parameterization, not MI between full token IDs or the prediction loss of a learned generator.

Discrete-token linear probes. We fit linear classifiers on mean-pooled patch bits using 100K training images, select the learning rate and weight decay on 10K validation images, and evaluate on 25K test images held out from probe fitting, averaging accuracy over three training seeds. As in Fig. 3b, Fig. 7 pairs each tokenizer with the gFID of its patch-only generator from Tab. 11. Replacing the pixel target by the feature target lowers probe accuracy but improves gFID for SigLIP and DINOv2. For SigLIP2, probe accuracy and gFID both worsen, and for DINOv3, both improve. Thus, this measure of linear separability does not consistently track class-conditional generation quality, unlike the spatial dependence in Fig. 3b.

D Evaluation Protocol

This section collects the experimental setup and measurement conventions used in Sec. 3.

Experimental setup. We evaluate class-conditioned image generation on ImageNet-1k (Russakovsky et al., 2015) at 256×256 and 512×512, and text-conditioned generation on GenEval (Ghosh et al., 2023), GenAI-Bench (Li et al., 2024a), and DPG-Bench (Hu et al., 2024).

Reconstruction and generation metrics. For a fixed feature extractor, Fréchet distance compares the empirical means and covariances of real and reconstructed or generated image features. rFID is the tokenizer's reconstruction measurement that compares the reconstructed images and original images in the Inception space using the ImageNet-1k validation set. gFID compares generated images with the original training images using Fréchet distance in Inception space. We re-use the released reference statistics from Li & He (2026). We also report gFDr6 introduced by Yang et al. (2026), which reports generative Fréchet distance in six different feature spaces. The six-space evaluation uses Inception-v3 (Szegedy et al., 2016), ConvNeXt V2 (Woo et al., 2023), DINOv2 (Oquab et al., 2024), MAE (He et al., 2022), SigLIP2 (Tschannen et al., 2025), and CLIP (Radford et al., 2021), with the checkpoints of Yang et al. (2026): ConvNeXt V2-B, DINOv2 ViT-L/14, MAE ViT-L/16, SigLIP2 So400m/16, and CLIP ViT-L/14. In each space, the Fréchet-distance ratio divides the measured distance by that space's fixed reference normalization constant, *i.e.*, the distance computed between the training and validation sets. gFDr6 averages

Table 13: Sampling configurations of FeaTok for the 256×256 results in Tab. 1.

		patch branch			global branch		
model	guidance	alloc.	gs	temp	alloc.	gs	temp
base generator (0.4B)							
FeaTok-SigLIP	without	[4, 4, 4, 4]	1.0	1.8	[4, 4, 4, 4]	1.0	1.4
	with	[4, 4, 4, 4]	4.0	1.6	[4, 4, 4, 4]	1.0	2.0
FeaTok-SigLIP2	without	[4, 4, 4, 4]	1.0	1.6	[3, 3, 5, 5]	1.0	2.0
	with	[4, 4, 4, 4]	4.0	1.6	[4, 4, 4, 4]	1.0	2.0
FeaTok-DINOv2	without	[4, 4, 4, 4]	1.0	2.0	[4, 4, 4, 4]	1.0	2.0
	with	[4, 4, 4, 4]	3.0	2.0	[4, 4, 4, 4]	1.5	2.0
FeaTok-DINOv3	without	[4, 4, 4, 4]	1.0	1.75	[3, 3, 5, 5]	1.0	2.0
	with	[4, 4, 4, 4]	3.0	1.6	[4, 4, 4, 4]	1.0	2.0
large generator (1.1B)							
FeaTok-SigLIP	without	[4, 4, 4, 4]	1.0	1.8	[2, 2, 5, 7]	1.0	2.0
	with	[4, 4, 4, 4]	2.0	1.8	[4, 4, 4, 4]	1.5	2.4
FeaTok-SigLIP2	without	[4, 4, 4, 4]	1.0	1.75	[4, 4, 4, 4]	1.0	2.0
	with	[4, 4, 4, 4]	2.5	1.6	[4, 4, 4, 4]	1.0	2.0
FeaTok-DINOv2	without	[4, 4, 4, 4]	1.0	2.0	[4, 4, 4, 4]	1.0	2.0
	with	[2, 2, 5, 7]	3.0	2.5	[4, 4, 4, 4]	1.5	2.5
FeaTok-DINOv3	without	[3, 3, 5, 5]	1.0	2.0	[3, 3, 5, 5]	1.0	2.0
	with	[4, 4, 4, 4]	2.0	1.9	[4, 4, 4, 4]	1.5	2.0

these six ratios for generated images, and rFDr6 averages them for reconstructions. We re-use the released reference statistics from Yang et al. (2026). All generation distribution metrics use 50K generated samples, and PRDC compares them with 10K real reference images.

Measured here versus transcribed. For the ImageNet comparisons in Tabs. 1, 2, 3, and 4, all per-representation gFDr and gFDr6 values are evaluated by us using 50k samples, except the guided continuous-baseline results in Tab. 2, transcribed from Tab. 7 of Singh et al. (2026), and the Drift results and guided iMF and pMF results in Tab. 3, transcribed from Yang et al. (2026) for the original models without FD-loss post-training. We evaluate the original iMF and pMF checkpoints without guidance ourselves, at guidance scale 1 with their released guidance intervals. Drift's released models already sample without guidance, so its transcribed results fill both column groups of Tab. 3. Fig. 1 uses the same unguided one-step values. For text-to-image generation, the continuous-baseline results in Tab. 5 are taken from Singh et al. (2026), whereas the discrete-baseline (BAR-L) results use our own implementations and evaluations. All FeaTok results, reward scores in Fig. 5, and appendix ablations are evaluated by us. For its final models, BAR (Yu et al., 2026) fine-tunes the pixel decoder for 40 additional epochs. Our primary BAR comparison excludes this decoder-only post-training for both methods, which separates the effect of the proposed tokenization and generation recipe from an additional decoder-refinement stage. BAR's original and refined decoders consume the same discrete tokens. The original decoder is used in Tabs. 1 and 14, Figs. 1 and 5, and the qualitative samples in App. H. Tab. 15 additionally reports BAR with the refined decoder at the same sampling settings. Baselines use their released sampling settings when available, and otherwise settings selected by a sweep (App. E), and App. F additionally reports gFDr6 for FeaTok at gFID-selected configurations (Tab. 15).

E Sampling hyperparameters

We report the sampling hyperparameters for all the main results we presented in the paper.

The knobs. FeaTok samples two branches, and they are configured independently: the 256 patch positions and the single global position each carry their own guidance scale, temperature and bit-group allocation. We write an allocation as its list of group sizes, so $[2, 2, 5, 7]$ unmasks a token's 16 bits in four groups of 2, 2, 5 and 7 bits, and $[b^k]$ abbreviates k groups of b bits. The list sums to the bits per token (16 at 256^2, 10 at 512^2) and its *length* sets the sampling cost, since a k-group schedule issues $k+1$ forwards per position. A guidance scale of 1.0 is no-guidance, so every row without guidance carries patch guidance 1.0. The global branch is unguided in every configuration without guidance and in half of the guided ones.

Table 14: ImageNet-1k generation at 256×256, without and with guidance, evaluated by gFID and PRDC. **Bold** and underline mark the highest and second-highest precision, recall, density, and coverage within each discrete generator-scale block, with ties at the displayed precision sharing a rank.

	generation@256 w/o guidance					generation@256 w/ guidance				
method	gFID ↓	Prec. ↑	Rec. ↑	Dens. ↑	Cov. ↑	gFID ↓	Prec. ↑	Rec. ↑	Dens. ↑	Cov. ↑
base generator										
BAR-B (Yu et al., 2026)	1.64	0.800	0.614	**1.153**	**0.997**	1.13	0.775	**0.660**	1.060	0.994
FeaTok-B-SigLIP	1.61	**0.807**	0.621	1.146	0.996	1.27	0.778	0.654	1.068	0.994
FeaTok-B-SigLIP2	1.53	0.802	0.621	1.142	0.995	1.12	0.782	0.655	1.067	**0.996**
FeaTok-B-DINOv2	1.47	0.782	0.645	1.085	0.996	1.10	0.784	0.659	1.069	0.995
FeaTok-B-DINOv3	1.34	0.783	**0.647**	1.082	0.996	1.07	**0.789**	0.652	**1.088**	**0.996**
large generator										
BAR-L (Yu et al., 2026)	1.36	0.798	0.631	1.121	0.995	1.02	0.780	**0.667**	1.059	**0.996**
FeaTok-L-SigLIP	1.37	0.800	0.633	**1.134**	0.995	1.26	**0.801**	0.636	**1.128**	**0.996**
FeaTok-L-SigLIP2	1.33	**0.801**	0.631	1.127	**0.996**	1.07	0.785	0.656	1.070	0.994
FeaTok-L-DINOv2	1.26	0.787	0.643	1.091	**0.996**	1.15	0.784	0.652	1.070	0.994
FeaTok-L-DINOv3	1.15	0.791	**0.651**	1.092	0.995	1.05	0.781	0.665	1.053	**0.996**
continuous latents										
RAEv2 (Singh et al., 2026)	1.61	0.790	0.634	1.116	0.995	1.04	0.792	0.652	1.099	0.996

Reduced-step RAEv2. The released RAEv2 sampling configuration for ImageNet 256×256 (`https://github.com/nanovisionx/RAEv2/blob/8a0d238/configs/stage2/sampling/imagenet-dinov3l-k7.yaml`) uses 100 ODE steps and an internal-guidance scale of 1.78, which we use for all other RAEv2 results. For the 3- and 6-step RAEv2 rows of Tab. 3, we select the internal-guidance scale with the lowest gFDr6 among the evaluated settings, rather than keeping the released scale of 1.78. The selected scales are 4.0 and 2.6, respectively.

Sampling hyperparameters for resolution 512×512. At 512 a token carries 10 bits, so the allocations sum to 10 rather than 16. Without guidance, FeaTok-L-DINOv3 in Tab. 4 samples the patch branch with allocation $[2, 2, 3, 3]$, scale 1.0, and temperature 2.4, and the global branch with allocation $[2, 2, 2, 4]$, scale 1.0, and temperature 2.0. With guidance, the patch branch uses allocation $[2, 2, 2, 4]$, scale 3.5, and temperature 2.6, and the global branch uses allocation $[2, 2, 2, 4]$, scale 1.0, and temperature 2.5. Guided BAR-L (Yu et al., 2026) and RAE (Zheng et al., 2026b) use the sampling settings released with those models. BAR does not release an unguided 512 configuration, so we did a hyperparameter sweep and set its unguided sampling configuration as allocation $[2, 2, 3, 3]$, scale 1.0, and temperature 2.4.

F Comparison using conventional gFID and PRDC metrics.

In the main paper, we report results in gFDr6 (Yang et al., 2026). We compare our method with baselines using gFID (Heusel et al., 2017) and Precision, Recall (Kynkäänniemi et al., 2019), Density, and Coverage (Naeem et al., 2020) in Tab. 14. Per conventions, we use different hyperparameters during sampling for gFID evaluation than gFDr6.

gFID measures the Fréchet distance in the feature space of an Inception-v3 network (Szegedy et al., 2016) trained for ImageNet classification. Its sensitivity to ImageNet class information means that it can be lowered substantially without improving the images (Kynkäänniemi et al., 2023), and Inception features represent the content of modern generators poorly, so gFID often disagrees with human raters (Stein et al., 2023; Jayasumana et al., 2024). These problems are most severe at the low values reached by current ImageNet generators. ImageNet validation images score a gFID of 1.68 against the training set (Yang et al., 2026), so generators below 2.0, and especially below 1.68, appear as close to the training data as real images, although their samples remain clearly distinguishable from real ones (Yang et al., 2026). In this regime, gFID can also be optimized directly: post-training a generator on Inception features lowers its gFID from 3.31 to 0.81 without producing the best samples, whereas post-training on modern representations improves object structure while gFID becomes worse (Yang et al., 2026). gFDr6 addresses these issues by averaging Fréchet distances over six representation spaces, each normalized by the distance between the training and validation sets, so no single feature space dominates and a value of 1 means as close to the training data as real validation images.

Table 15: gFDr6 on ImageNet-1k 256×256 at gFID-selected sampling configurations. BAR uses its released settings, and each FeaTok model uses the configuration of Tab. 14. Refined-decoder rows replace BAR's pixel decoder by the one fine-tuned for 40 additional epochs, with the same tokens and sampling settings. Columns follow Tab. 1. **Bold** and underline mark the best and second-best values within each generator size in each metric column.

	without guidance							with guidance						
method	gFDr6	incep.	cvnx2	dino2	mae	siglip2	clip	gFDr6	incep.	cvnx2	dino2	mae	siglip2	clip
BAR-B (Yu et al., 2026)	5.15	1.01	1.70	5.21	5.82	9.40	7.73	4.89	0.69	0.95	4.63	5.89	9.32	7.86
BAR-B (refined decoder)	4.62	0.99	1.65	4.71	5.23	8.39	6.76	4.36	**0.67**	0.90	4.08	5.35	8.32	6.81
FeaTok-B-SigLIP	5.08	1.01	1.31	5.63	6.11	9.28	7.15	4.63	0.79	1.03	4.77	5.77	8.91	6.51
FeaTok-B-SigLIP2	4.66	0.96	1.29	4.97	5.14	8.85	6.73	4.03	0.70	0.85	3.88	4.71	7.65	6.39
FeaTok-B-DINOv2	4.40	0.92	1.36	**4.12**	4.93	8.31	6.74	3.87	0.69	0.85	**3.37**	**4.51**	7.34	6.43
FeaTok-B-DINOv3	**4.21**	**0.87**	**1.24**	4.26	**4.87**	**7.73**	**6.30**	**3.80**	0.68	**0.75**	3.66	4.61	**7.01**	**6.07**
BAR-L (Yu et al., 2026)	4.57	0.83	1.38	4.59	5.22	8.14	7.26	4.11	0.61	0.79	3.72	5.02	7.49	7.06
BAR-L (refined decoder)	4.10	0.81	1.34	4.11	4.72	7.34	6.26	3.67	**0.60**	0.76	3.29	4.56	6.77	6.00
FeaTok-L-SigLIP	4.75	0.85	1.09	5.23	5.59	8.69	7.08	4.42	0.78	0.92	4.73	5.14	8.19	6.74
FeaTok-L-SigLIP2	4.02	0.84	1.13	4.30	4.33	7.47	6.05	3.91	0.66	0.89	3.83	4.17	7.52	6.41
FeaTok-L-DINOv2	3.57	0.78	0.88	**3.25**	4.21	6.58	5.72	3.91	0.71	0.86	**3.17**	4.43	7.38	6.90
FeaTok-L-DINOv3	**3.35**	**0.74**	**0.75**	3.36	**4.03**	**5.98**	**5.25**	**3.45**	0.65	**0.68**	3.39	**4.00**	**6.34**	**5.63**

Tab. 14 provides additional evidence. At these low-gFID operating points, improvements in Inception Fréchet distance do not consistently agree with the fidelity diagnostics. Across the ten BAR and FeaTok models, the guided configuration has lower gFID and higher recall in every case, but lower precision in seven and lower density in nine. For example, BAR-B's gFID improves from 1.64 to 1.13, while precision falls from 0.800 to 0.775 and density from 1.153 to 1.060. Recall rises from 0.614 to 0.660. Thus, rather than demonstrating a uniform fidelity gain, these selected configurations often show a fidelity–recall trade-off in the opposite direction to the usual motivation for guidance. Coverage remains near saturation (0.994–0.997). The results caution against interpreting a small gFID improvement as an unambiguous improvement in generation quality, motivating evaluation across multiple representations and complementary reward measures.

Without guidance, every FeaTok-B model has a lower gFID than BAR-B (1.34–1.61 against 1.64), three of the four FeaTok-L models improve on BAR-L (1.15–1.33 against 1.36, with FeaTok-L-SigLIP at 1.37), and FeaTok-L-DINOv3 reaches 1.15 against 1.61 for RAEv2. With guidance, three of the four FeaTok-B models improve over BAR-B, while BAR-L reaches the lowest large-model gFID, 1.02, against 1.04 for RAEv2 and 1.05 for FeaTok-L-DINOv3. These guided values are all well below the 1.68 of real validation images, in the regime where gFID no longer tracks sample quality. Under gFDr6, which accounts for all six representation spaces, FeaTok-L-DINOv3 improves over BAR-L by 25% with guidance (Tab. 1).

The baselines in Tab. 1 use their released sampling settings, which were selected for gFID, whereas the FeaTok settings there are selected for gFDr6. Tab. 15 therefore also reports gFDr6 for every FeaTok model at its gFID-selected configuration, the one used in Tab. 14, so that both sides follow the same selection criterion. Under this criterion, all four FeaTok-B models and three of the four FeaTok-L models still improve on the original-decoder BAR model of the same size, with and without guidance. FeaTok-L-DINOv3 reduces gFDr6 from 4.57 to 3.35 without guidance and from 4.11 to 3.45 with guidance, while FeaTok-L-SigLIP is the exception, at 4.75 and 4.42. FeaTok does not use the additional 40 epochs of decoder-only fine-tuning, yet its DINOv3 tokenizers still improve on BAR with the refined decoder in all four settings, *e.g.*, FeaTok-L-DINOv3 reduces gFDr6 from 4.10 to 3.35 without guidance and from 3.67 to 3.45 with guidance, and its DINOv2 tokenizers do so in three of the four settings.

G SAMPLING-COST MEASUREMENT

TFLOPs. TFLOPs is arithmetic per image, counted with `torch.utils.flop_counter` in the $2\times$MAC convention, and is therefore hardware-independent unlike wall-clock throughput. It covers the full sampling loop of the generator plus one decoder pass that maps the generated tokens or latents to pixels. Classifier-free guidance duplicates the generator batch and roughly doubles its cost (BAR, FeaTok, SiT, LightningDiT, DDT and the discrete baselines), whereas REPA-E and REG gate the doubled forward to a guidance interval and pay $1.8\times$, RAEv2's internal guidance reuses a sin-

Table 16: Sampling throughput and peak GPU memory on one L40S (bf16, uncompiled). Throughput is in generated images per second (img/s, higher is better), and memory is the peak GPU memory in GiB (lower is better). *gen.* counts generation only, and *e2e* includes pixel decoding. We report a matched batch size of 256, and each model/setting's best end-to-end throughput over the batch sweep, with the batch size (bsz) at which it is reached and its memory. Guided rows use classifier-free guidance for BAR and FeaTok (scale 5.0 for base and 5.3 for large models) and internal guidance (1.78) for RAEv2, which needs no extra forward pass. FeaTok rows use the DINOv3 tokenizer, and throughput does not depend on the encoder. RAEv2 uses 100 ODE steps, the setting of its released sampling configuration (`https://github.com/nanovisionx/RAEv2/blob/8a0d238/configs/stage2/sampling/imagenet-dinov3l-k7.yaml`).

		matched batch 256			peak e2e over the batch sweep		
system	guidance	gen. img/s ↑	e2e img/s ↑	GiB ↓	img/s ↑	bsz	GiB ↓
BAR-B (Yu et al., 2026)	without	28.93	27.69	10.6	34.32	1024	32.2
	with	17.06	16.59	17.8	17.49	512	32.2
FeaTok-B	without	27.49	26.36	10.4	31.82	1024	31.3
	with	15.97	15.57	17.4	16.31	512	31.4
BAR-L (Yu et al., 2026)	without	16.02	15.61	22.1	17.02	512	36.8
	with	8.71	8.59	36.8	8.59	256	36.8
FeaTok-L	without	14.69	14.36	22.0	15.35	512	36.7
	with	7.85	7.75	36.7	7.75	256	36.7
RAEv2 (Singh et al., 2026)	without	1.10	1.09	13.4	1.89	8	7.5
	with	1.09	1.09	14.2	1.85	8	7.5

gle forward and costs nothing, and RAE's autoguidance adds a smaller second network for $+30\%$. Guidance never duplicates the decoder, since only the conditional sample is decoded. Each decoder is built from its official release and driven through the released sampler's decode call. Decoder costs range from 0.16 TFLOPs for the pixel decoder shared by BAR and FeaTok to 0.62 for the SD-VAE decoder of the latent diffusion baselines. They are small relative to multi-step continuous generators, but not negligible for the cheapest samplers: the pixel decoder adds 15–32% to the generator cost of BAR and FeaTok, and the SD-VAE decoder exceeds the generator cost of the latent one-step models.

Throughput and memory. Tab. 16 measures sampling on one NVIDIA L40S GPU in bf16 without compilation. Each run times the generation of the discrete tokens or continuous latents and the subsequent pixel decoding separately, so we report generation-only and end-to-end throughput from the same run. *Throughput* is the number of images produced per second, computed as batch size divided by the generation time, or by the generation and decoding time for end-to-end throughput. *Memory* is peak GPU memory usage during sampling and decoding at the stated batch size, reported in GiB, not memory per image.

For peak throughput, we compare each model and guidance setting at its own best end-to-end batch size, selected from $\{1, 8, 32, 128, 256, 512, 1024, 2048\}$. Larger batches benefit autoregressive models, whose end-to-end peaks occur at batches 256–1024. RAEv2 instead peaks at batch 8 with 1.89 images/s and slows to about 1.1 images/s at larger batches on this GPU. FeaTok-B peaks at 31.82 images/s without guidance and 16.31 with guidance. FeaTok-L reaches 15.35 and 7.75 images/s, which is $8.1\times$ and $4.2\times$ the corresponding peak of RAEv2. Decoding takes 1.3–4.3% of end-to-end time at batch 256 for BAR and FeaTok and 0.4% for RAEv2, and it never raises peak memory. We report the corresponding batch size and memory alongside each peak.

H Qualitative samples across tokenizer encoders

Figs. 8–11 compare the released BAR-L model against our FeaTok models with multiple encoders. For each block, the five rows are labelled on the left in the order BAR-L, then FeaTok-L with SigLIP2, DINOv2, DINOv3, and SigLIP, top to bottom. The eight images in each row are drawn with a fixed random seed from that class's 50 generated samples.



macaw (88)



goose (99)



golden retriever (207)

Figure 8: ImageNet qualitative samples (1/4). Rows: BAR-L, then FeaTok-L with SigLIP2, DINOv2, DINOv3, and SigLIP, at their respective guided settings (App. E).



monarch butterfly (323)



giant panda (388)



balloon (417)

Figure 9: ImageNet qualitative samples (2/4). Rows: BAR-L, then FeaTok-L with SigLIP2, DINOv2, DINOv3, and SigLIP, at their respective guided settings (App. E).



lighthouse (437)

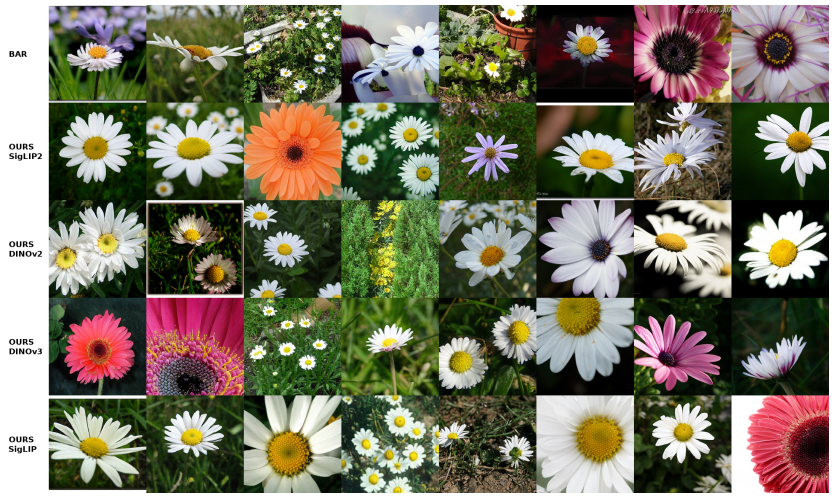


pickup truck (717)



school bus (779)

Figure 10: ImageNet qualitative samples (3/4). Rows: BAR-L, then FeaTok-L with SigLIP2, DINOv2, DINOv3, and SigLIP, at their respective guided settings (App. E).



daisy (985)



coral fungus (991)

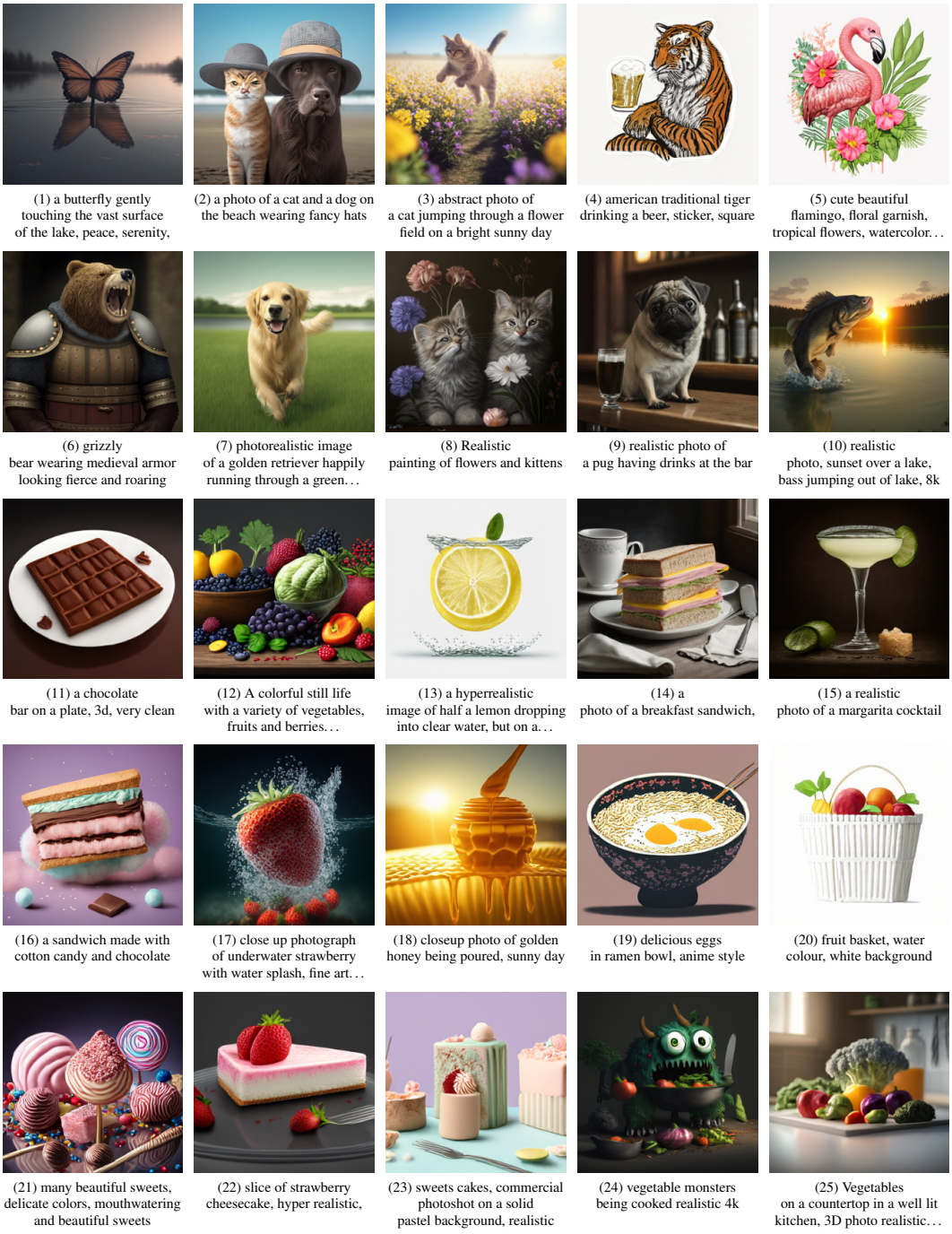


volcano (980)

Figure 11: ImageNet qualitative samples (4/4). Rows: BAR-L, then FeaTok-L with SigLIP2, DINOv2, DINOv3, and SigLIP, at their respective guided settings (App. E).

I QUALITATIVE TEXT-TO-IMAGE SAMPLES

Fig. 12 shows samples for MJHQ-30K (Li et al., 2024b) prompts in the *animals* and *food* categories. Prompts beneath the images are abbreviated for space. The full prompts used for the T2I visuals are in Tab. 17.



(1) a butterfly gently touching the vast surface of the lake, peace, serenity,
(2) a photo of a cat and a dog on the beach wearing fancy hats
(3) abstract photo of a cat jumping through a flower field on a bright sunny day
(4) american traditional tiger drinking a beer, sticker, square
(5) cute beautiful flamingo, floral garnish, tropical flowers, watercolor...
(6) grizzly bear wearing medieval armor looking fierce and roaring
(7) photorealistic image of a golden retriever happily running through a green...
(8) Realistic painting of flowers and kittens
(9) realistic photo of a pug having drinks at the bar
(10) realistic photo, sunset over a lake, bass jumping out of lake, 8k
(11) a chocolate bar on a plate, 3d, very clean
(12) A colorful still life with a variety of vegetables, fruits and berries...
(13) a hyperrealistic image of half a lemon dropping into clear water, but on a...
(14) a photo of a breakfast sandwich,
(15) a realistic photo of a margarita cocktail
(16) a sandwich made with cotton candy and chocolate
(17) close up photograph of underwater strawberry with water splash, fine art...
(18) closeup photo of golden honey being poured, sunny day
(19) delicious eggs in ramen bowl, anime style
(20) fruit basket, water colour, white background
(21) many beautiful sweets, delicate colors, mouthwatering and beautiful sweets
(22) slice of strawberry cheesecake, hyper realistic,
(23) sweets cakes, commercial photoshot on a solid pastel background, realistic
(24) vegetable monsters being cooked realistic 4k
(25) Vegetables on a countertop in a well lit kitchen, 3D photo realistic...

Figure 12: Text-to-image samples at 256×256 from MJHQ-30K animal (first two rows) and food prompts. Full numbered prompts are in Tab. 17.

Table 17: Full MJHQ-30K (Li et al., 2024b) prompts for the panels of Fig. 12, numbered in the figure's reading order (left to right, top to bottom); the number appears in front of each panel's truncated caption. Prompts are reproduced verbatim, including the trailing keyword lists and the original capitalisation and punctuation, since these are part of what the model was conditioned on. Panels 1, 2, 6, 15, 13 are the five reproduced in Fig. 6.

#	prompt
1	a butterfly gently touching the vast surface of the lake, peace, serenity,
2	a photo of a cat and a dog on the beach wearing fancy hats
3	abstract photo of a cat jumping through a flower field on a bright sunny day
4	american traditional tiger drinking a beer, sticker, square
5	cute beautiful flamingo, floral garnish, tropical flowers, watercolor vibrant illustration, png
6	grizzly bear wearing medieval armor looking fierce and roaring
7	photorealistic image of a golden retriever happily running through a green field with a lake in the background with cinematic lighting, high definition, depth of field superresolution, insanely detailed
8	Realistic painting of flowers and kittens
9	realistic photo of a pug having drinks at the bar
10	realistic photo, sunset over a lake, bass jumping out of lake, 8k
11	a chocolate bar on a plate, 3d, very clean
12	A colorful still life with a variety of vegetables, fruits and berries arranged on a wooden table, creating an atmosphere of a healthy and balanced diet.
13	a hyperrealistic image of half a lemon dropping into clear water, but on a transparent background
14	a photo of a breakfast sandwich,
15	a realistic photo of a margarita cocktail
16	a sandwich made with cotton candy and chocolate
17	close up photograph of underwater strawberry with water splash, fine art food photography, low angle perspective, dramatic studio lighting, color graded, strong contrast, sharp focus, hyper detailed, realistic, highquality retouching, food, awardwinning photography, editorial photography,
18	closeup photo of golden honey being poured, sunny day
19	delicious eggs in ramen bowl, anime style
20	fruit basket, water colour, white background
21	many beautiful sweets, delicate colors, mouthwatering and beautiful sweets
22	slice of strawberry cheesecake, hyper realistic,
23	sweets cakes, commercial photoshot on a solid pastel background, realistic
24	vegetable monsters being cooked realistic 4k
25	Vegetables on a countertop in a well lit kitchen, 3D photo realistic, cinematic lighting