BANDDIT: BAND-ALIGNED DIFFUSION TRANSFORMERS FOR PIXEL-SPACE IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Pixel-space diffusion models denoise pixel values directly, without the pretrained autoencoder that latent diffusion models use to compress images. At each denoising step, the pixel-space transformer must extract meaningful features from raw pixels, reason over these representations, and map its predictions back into image space. In contrast, latent diffusion models operate in a learned representation space that abstracts away some low-level detail, leaving the pretrained decoder to reconstruct fine details and textures. This places the full burden of spectral reconstruction on the transformer, raising a central question: how are different image frequencies represented across network depth, and how can this organization guide learning? We find that pixel-space diffusion transformers build the spectrum progressively across their blocks: low-frequency bands become decodable from shallow blocks, whereas high-frequency bands emerge near the output. This hierarchy lies in internal representations within a denoising step, distinct from the known coarse-to-fine evolution of samples along the sampling trajectory. We introduce BANDDIT, a band-aligned diffusion transformer that turns this hierarchy into explicit supervision. Wavelet bands of the clean image, which need no pretrained teacher, serve as targets for intermediate blocks at matching depths, from low frequencies in shallow blocks to high frequencies in deep ones. On ImageNet 256×256, BANDDIT-B/16 matches the baseline's FID and 5-crop FID with $1.3\times$ fewer epochs and reduces the high-frequency Fréchet Wavelet Distance by 17.4%, indicating improved global coherence, local structure, and texture fidelity.

1 INTRODUCTION

Image generation models learn image distributions over millions of correlated pixels. Latent-space image generation models compress images into the latent space using a pretrained autoencoder and learn the distribution in the smooth, lower-dimensional latent space using a transformer (Rombach et al., 2022; Peebles & Xie, 2023; Ma et al., 2024). As the autoencoder is trained for compression rather than generation, the transformer backbone in a latent diffusion model learns to generate within a compressed representation whose spatial-frequency content is constrained by the autoencoder (Skorokhodov et al., 2025; Yao et al., 2025; Rombach et al., 2022). Therefore, the transformer backbone primarily models coarse image structure with fine details added by the frozen decoder, leading to limited generation fidelity (Ning et al., 2026; Leng et al., 2025; Dai et al., 2023; Esser et al., 2024). Generating directly in pixel space removes the limitation (Chen et al., 2025; Yu et al., 2025b; Li & He, 2025), enabling the backbone to learn the entire frequency spectrum of the image, spanning from low-frequency components that govern global structure and coherence, to high-frequency components that capture fine details and textures.

The image spectrum is formed progressively. Along the sampling trajectory, diffusion models are known to resolve low-frequency components before high-frequency components (Dieleman, 2024; Rissanen et al., 2022). This frequency ordering along diffusion time is well explored through frequency-aware schedules or losses to improve training and sampling (Pavasovic et al., 2026; Hang et al., 2023; Ren et al., 2026; Schusterbauer et al., 2026). In contrast, whether such frequency-dependent ordering exists across network depth is not well established. We probe trained pixel-space and latent-space transformers and find that both construct the spectrum in a frequency-ordered manner: low-frequency bands are decodable from shallow blocks, whereas high-frequency bands become decodable only near the output (Sec. 3.2). While this hierarchy emerges in both model types,

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

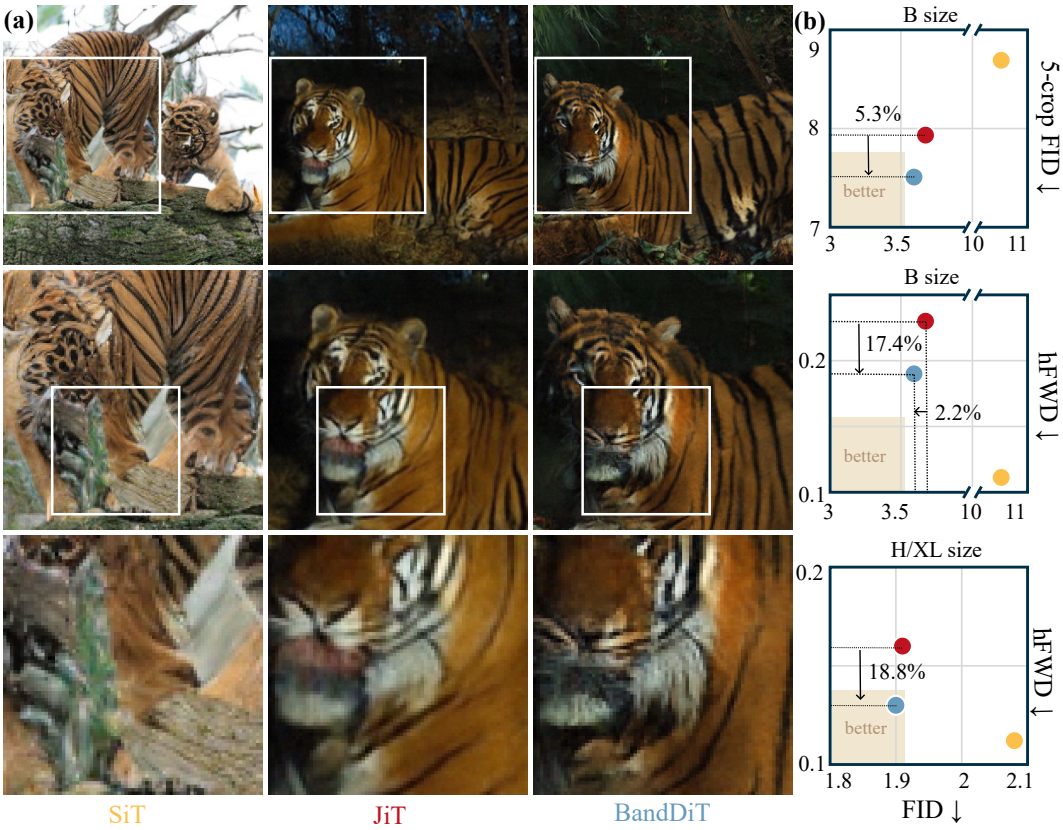


Figure 1: **BANDDIT Improves Pixel-Space Generation.** BANDDIT demonstrates better overall quality, local textures, and high-frequency details compared to the latent-space model (SiT) and the pixel-space model (JiT) (*left*). BANDDIT shows better FID, 5-crop FID and hFWD over both ST and JiT on ImageNet 256×256 (*right*).

pixel-space frequency bands contain richer spatial information at matched denoising times than their latent-space counterparts, consistent with prior findings on the latent-space generation bottleneck. We leverage this spatially informative decomposition to reinforce the pixel-space transformer's naturally emerging frequency ordering: the clean image's wavelet bands serve as teacher-free targets for intermediate blocks at the depths where the corresponding frequencies become decodable. This aligns supervision with the model's own coarse-to-fine reconstruction process.

Building on these findings, we propose BANDDIT, a teacher-free **Band**-aligned **Di**ffusion **T**ransformer that uses the clean image's wavelet-decomposed frequency bands to supervise intermediate transformer blocks at multiple depths. BANDDIT assigns each wavelet band to the depth at which its frequency content becomes decodable. This turns an emergent property of trained models into an explicit training objective, providing frequency-specific guidance throughout the network.

We show that band alignment accelerates convergence in pixel-space training on ImageNet 256 × 256. We evaluate generated images' global coherence using FID, local structure using 5-crop FID, and texture fidelity using the high-frequency Fréchet Wavelet Distance (hFWD). The B-size BANDDIT model matches the baseline's FID and 5-crop FID using 1.3× fewer training epochs and reduces hFWD by 17.4%. At larger scale, BANDDIT-H/16 reduce hFWD by 18.8%, 5-crop FID by 3.5% compared to the baseline, and achieves an FID of 1.90 at 500 epochs.

Our contributions are summarized below:

1. We show that a diffusion transformer constructs the image spectrum along network depth in coarse-to-fine order, with low-frequency components becoming decodable at shallow depth and high-frequency components emerging at greater depth. The ordering holds for pixel-

space and latent-space transformers, while the pixel-space model carries more information per band at the same denoising time.

2. We introduce BANDDIT, a teacher-free alignment loss that matches intermediate blocks to the wavelet bands of the clean image at different depths. The loss is used only during training and needs no pretrained encoder, no architectural change, and no sampler change.
3. BANDDIT improves pixel-space generation on ImageNet 256×256 over the JiT baseline under the same training and evaluation configuration. BANDDIT-B/16 reaches the baseline's FID and 5-crop FID with $1.3\times$ fewer training epochs and reduces hFWD by 17.4%. This indicates improved global coherence, local details, and texture fidelity of the generation. The improvement persists at larger model size: BANDDIT-H/16 reaches an FID of 1.90, 5-crop FID of 5.50, and hFWD of 0.13.

2 RELATED WORK

Latent- and pixel-space generation. Diffusion and flow-matching models (Ho et al., 2020; Song et al., 2022; Lipman et al., 2023; Liu et al., 2022; Albergo et al., 2025) are commonly trained in the latent space compressed by a pretrained autoencoder (Rombach et al., 2022; Esser et al., 2024; Dosovitskiy et al., 2021; Peebles & Xie, 2023; Ma et al., 2024). To address the loss of local detail in latent-space generation, later work widens the latent space, retrains the autoencoder, or aligns it with a internal or external representation encoder (Dai et al., 2023; Esser et al., 2024; Podell et al., 2023; Yao et al., 2025; Zheng et al., 2025; Leng et al., 2025).

Pixel-space generation removes the autoencoder. To model pixel inputs efficiently, cascaded models learn the image at increasing resolutions in stages (Ho et al., 2021; Chen et al., 2025), whereas single-network designs make long pixel-token sequences tractable (Hoogeboom et al., 2025; Yu et al., 2025b; Chen et al., 2026; Ma et al., 2026; Li & He, 2025). JiT (Li & He, 2025) shows that a plain transformer can operate in pixel space when it predicts the clean image and is trained with the velocity loss. BANDDIT adopts this backbone and adds only a training-time loss that supervises frequency bands at specific depths.

Representation alignment. Representation alignment (REPA) (Yu et al., 2025a) accelerates latent-space diffusion transformers by regressing intermediate features onto a frozen feature encoder such as DINO (Caron et al., 2021). Wang et al. (2025b) show that the gain plateaus or reverses late in training. Chefer et al. (2026) remove the external model and achieve better results by self-supervising with an EMA teacher. In pixel space, Shin et al. (2026) show that applying REPA directly worsens the FID of JiT over training. They replace the MLP projector with a shallow transformer adapter, mask part of its input tokens, and consistently outperform the baseline JiT method. BANDDIT retains this adapter design but replaces the semantic target with the energy-normalized wavelet bands of the clean image (Sec. 3.3), without a pretrained encoder or an EMA teacher.

Frequency-aware diffusion. Diffusion models resolve frequencies from the low band to the high band along the sampling trajectory (Dieleman, 2024; Falck et al., 2025). Methods exploit this ordering in three ways: by making the corruption process frequency-aware (Rissanen et al., 2022; Pavasovic et al., 2026; Lin et al., 2026), by adding frequency-specific modifications to the architecture (Ren et al., 2026; Ma et al., 2026; Chen et al., 2026; Cheng et al., 2024), or by reweighting the loss and the sampling steps across noise levels (Hang et al., 2023; Karras et al., 2022; Kingma & Gao, 2023; Schusterbauer et al., 2026). Coarse-to-fine division within each denoising step is less explored, although new architectures have been proposed to enforce such a division, with one part of the network producing global structure and another producing fine detail conditioned on it. DDT in latent space and PixelDiT in pixel space (Wang et al., 2025a; Yu et al., 2025b) both split the transformer into an encoder for low-frequency semantics and a decoder for high-frequency detail. BANDDIT leaves the architecture unchanged and imposes the coarse-to-fine order across depth with a training-only alignment loss.

3 BANDDIT

We propose BANDDIT, a pixel-space diffusion transformer that aligns its intermediate representations with the frequency structure of the clean ground-truth image, using wavelet bands as depth-

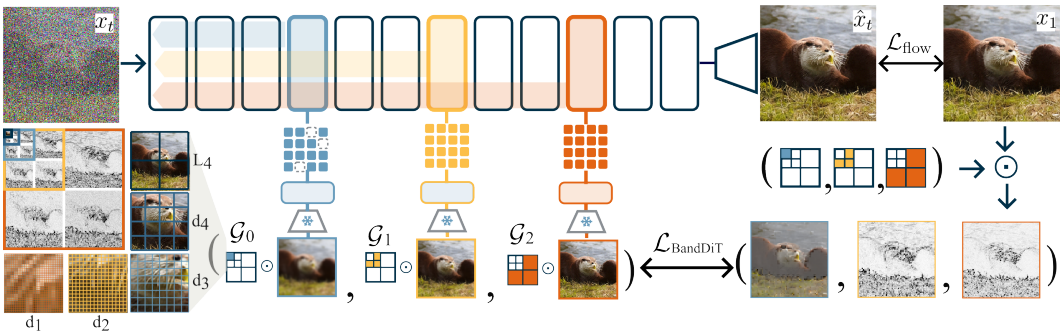


Figure 2: **BANDDIT Overview.** The clean image is split into low- ($\mathcal{G}_0$), mid- ($\mathcal{G}_1$), and high-frequency ($\mathcal{G}_2$) bands. An alignment branch decodes the tokens of selected blocks into images, and the alignment loss $\mathcal{L}_k$ matches each to one band group of the clean image, with shallow blocks matched to low frequency and deeper blocks to higher frequency.

specific supervision targets (Fig. 2). The design follows from the evidence in Sec. 3.2, where we observe a coarse-to-fine frequency structure emerging along the network depth. BANDDIT turns this hierarchy into a training-only, teacher-free loss on JiT to reinforce this structure and improve convergence, built from wavelet band targets (Sec. 3.3) and band alignment across depth (Sec. 3.4). Notation is summarized in Appendix A.

3.1 PRELIMINARIES

Diffusion Model. Diffusion models formulate image generation as a denoising process, transforming a noise sample $x_0 \sim \mathcal{N}(0, \sigma_0^2 I)$ into a data sample $x_1 \sim p_{\text{data}}$ along a continuous trajectory $\{x_t\}_{t\in[0,1]}$, where $t = 0$ denotes pure noise and $t = 1$ denotes clean data. The trajectory is defined by the linear interpolation:

$$x_t = t\, x_1 + (1 - t)\, x_0,$$

whose velocity $v = \mathrm{d}x_t/\mathrm{d}t = x_1 - x_0$ is constant along the path. A neural network is trained to predict this velocity from (x_t, t) under the prescribed noising process. During sampling, the learned probability-flow ODE is integrated from $t = 0$ to $t = 1$, transporting samples from the noise distribution to the data distribution.

Flow matching in pixel space. JiT (Li & He, 2025) finds that in pixel space the network achieves the best performance when net_θ predicts the image x_1 and is trained with the flow loss:

$$\mathcal{L}_{\text{flow}} = \mathbb{E}_{t,\, x_1,\, x_0} \left\| v_\theta(x_t, t) - v \right\|_2^2, \quad v_\theta(x_t, t) = \frac{\text{net}_\theta(x_t, t) - x_t}{1 - t}, \quad v = x_1 - x_0.$$

Since $v_\theta - v = (\text{net}_\theta(x_t, t) - x_1)/(1 - t)$, the loss equals $(1 - t)^{-2} \|\text{net}_\theta(x_t, t) - x_1\|_2^2$. The factor $(1 - t)^{-2}$ depends on t, so within one sample every pixel enters the loss with the same weight. The objective therefore expresses no preference over the coarse structure or the fine detail of x_1, but still a frequency coarse to fin pattern naturally emerge as seen in Sec. 3.2.

PixelREPA (Shin et al., 2026) aligns intermediate backbone tokens $z_t^{(l)}$ with features extracted by a frozen DINO encoder (Caron et al., 2021) using a masked transformer adapter (MTA). Specifically, MTA replaces a fraction r of the tokens with a learned mask token and processes the resulting sequence with a two-block transformer. The masking strategy, together with cross-token attention, encourages the backbone to reconstruct masked tokens from their visible neighbors.

3.2 FREQUENCY-SEQUENTIAL CONSTRUCTION ALONG DEPTH

We observe the frequency-sequentiality of intermediate transformer blocks in two trained publicly available models: the pixel-space model JiT-H/16 (Li & He, 2025) and the latent-space model SiT-XL/2 (Ma et al., 2024). To quantify the information overlap between the intermediate representation and the clean image, we define decodability $\rho_b^{(l)}$ for each frequency band b at layer index l as the average cosine similarity between the band-filtered $\hat{x}_t^{(l)}$ and the clean image x_1:

$$\rho_b^{(l)} = \cos\big(W_b(\hat{x}_t^{(l)}),\, W_b(x_1)\big), \tag{1}$$

where $\hat{x}_t^{(l)}$ denotes the image estimate decoded from the hidden state $z_t^{(l)}$ using the frozen linear map from the output layer at time t, and W_b denotes the coefficients of band b from a four-level orthonormal Haar Discrete Wavelet Transform (DWT). The frequency bands are indicated by their spatial period in the unit of pixel (px), measuring the pixel distance over which a band's pattern repeats. DWT results in five orthogonal bands $b \in \{d_1, d_2, d_3, d_4, \ell_4\}$, each detail band d_j contain coefficients $\{c_i\} \in d_j$, representing the three orientations at level j, from the coarsest detail band d_4: 16–32px to the finest detail d_1: 2–4 px, with each band's spatial period half in size from coarse to fine. The band with spatial period coarser than 32px is included as a single group denoted as ℓ_4. More details of the wavelet decomposition can be found in Appendix B.3.

The decodability of each frequency band b at layer index l is visualized in Fig. 3. Along depth, the decodability of each frequency band increases monotonically, with early blocks mostly capturing coarse structure and late blocks focusing on adding fine details. Moreover, pixel-space models preserve substantially more high-frequency information than latent-space models. Motivated by the frequency sequentiality of pixel space, BANDDIT supervises the coarse-to-fine bands of the clean image at corresponding depths. We ablate the necessity of the coarse-to-fine order in Sec. 4.3. Additional details of the spectral comparison between the latent and the pixel space are in Appendix C.

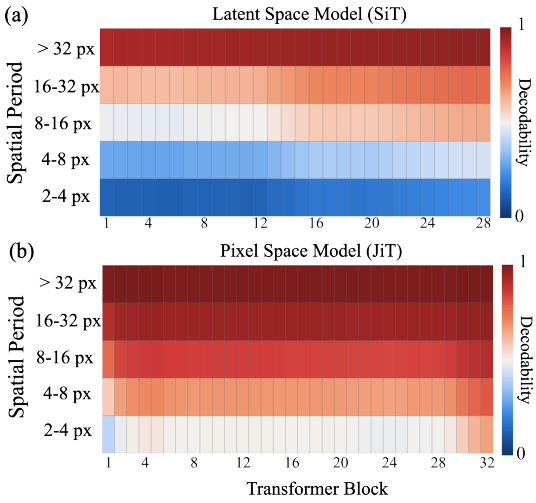


Figure 3: **Per-band decodability across depth.** Decodability (Eq. (1)) of each Haar band, labeled by spatial period, at every block output at $t = 0.5$ increases with depth in both models, and every band is more decodable in the pixel-space JiT (b) than in the latent-space SiT (a) at the same block.

3.3 WAVELET BAND TARGETS

We construct the alignment target directly from the clean image using a four-level orthonormal Haar transform as mentioned in Sec. 3.2. The wavelet bands are grouped into:

$$\mathcal{G}_0 = \{d_3, d_4, \ell_4\}, \qquad \mathcal{G}_1 = \{d_2\}, \qquad \mathcal{G}_2 = \{d_1\},$$

such that $\mathcal{G}_0$ covers spatial periods of 8 px and longer, $\mathcal{G}_1$ covers 4–8 px, and $\mathcal{G}_2$ covers 2–4 px. Each group $\mathcal{G}_k$ is aligned at block l_k for $k \in \{0, 1, 2\}$, with $l_0 < l_1 < l_2$ so that shallow blocks learn coarse-scale features and deeper blocks learn finer detail. We choose $\{l_0, l_1, l_2\} = \{D/4, 2D/4, 3D/4\}$ for a model with D layers of transformer blocks. Ablation study on the $\{l_0, l_1, l_2\}$ can be found in Sec. 4.3.

To make the BANDDIT target independent of the absolute scale of the image, we use normalized Haar band as the alignment target:

$$S_b(x) = \frac{W_b(x)}{\|x\|_2}, \tag{2}$$

where $\|x\|_2$ is the ℓ_2 norm of the image. Since the transform is orthonormal, this normalization gives the five bands unit total energy, so the target does not depend on the brightness or contrast of the image.

3.4 BAND-ALIGNED TRAINING OBJECTIVE

At time $t < 1$, the input x_t contains noise of power $(1-t)^2\sigma_0^2$ per coefficient, so any coefficient whose signal power falls below this noise floor cannot be distinguished from noise. We therefore weight each coefficient $c_i = [W_b(x_1)]_i$, $\{c_i\} \in b$, in the alignment loss by its Wiener gain:

$$\omega_i(t) = \frac{t^2 c_i^2}{t^2 c_i^2 + (1-t)^2\sigma_0^2}. \tag{3}$$

The Wiener gain is large for reliably observed structures and small for noise-dominated coefficients. More details of the derivation of the Wiener gain can be found in Appendix B.3.

For a frequency group $\mathcal{G}_k$, we define the BANDDIT alignment loss as:

$$\mathcal{L}_k = \frac{1}{N_k} \sum_{b \in \mathcal{G}_k} \sum_{i \in b} \tilde{\omega}_i(t) \left([S_b(\hat{x}^{(k)})]_i - [S_b(x_1)]_i \right)^2, \tag{4}$$

where $\hat{x}^{(k)} = \hat{x}_t^{(l_k)}$ is the image decoded from block l_k, N_k is the number of coefficients in $\mathcal{G}_k$, and $\tilde{\omega}_i(t)$ is the Wiener gain normalized within each band. Averaging over the N_k coefficients of each group in Eq. (4) keeps the group losses comparable despite their different sizes.

The overall objective combines the flow-matching loss with the frequency-group alignment losses:

$$\mathcal{L} = \mathcal{L}_{\text{flow}} + \lambda \sum_{k=0}^{K-1} \mathbb{E}_{t,x_1,x_0} \left[\mathcal{L}_k \right]. \tag{5}$$

where $K = 3$ is the number of band groups and λ weights the alignment losses relative to the flow loss. To compute $\hat{x}^{(k)}$, the hidden state at block l_k passes through a small two-block transformer adapter and is then decoded into an image by the output layer of the network. For the low-frequency group $\mathcal{G}_0$, a fraction of the input tokens is replaced by a learned mask token, which encourages the reconstruction to use context from neighboring tokens rather than only its own token. The adapters and masks are used only during training and are removed at sampling. Additional method details can be found in Appendix B.3.

4 EXPERIMENTS

4.1 SETUP

Implementation. The BANDDIT implementation uses JiT (Li & He, 2025) as its backbone and compares against JiT as the baseline model. For a D-layer transformer backbone, BANDDIT applies $K = 3$ alignment losses at blocks $(l_0, l_1, l_2) = (D/4, 2D/4, 3D/4)$. The spectral features are computed using a four-level orthonormal Haar pyramid with bands $\{d_1, d_2, d_3, d_4, \ell_4\}$, which are divided into the band groups $\mathcal{G}_0 = \{d_3, d_4, \ell_4\}$, $\mathcal{G}_1 = \{d_2\}$, and $\mathcal{G}_2 = \{d_1\}$. The mask ratios are $(r_0, r_1, r_2) = (0.2, 0, 0)$, where 20% of the tokens in the coarse band $\mathcal{G}_0$ are randomly masked to encourage the model to infer cross-patch information. As the fine bands $\mathcal{G}_1$, $\mathcal{G}_2$ have spatial periods smaller than the patch size, no tokens are masked. The alignment weight is set to $\lambda = 500$ and remains constant throughout the training process. We trained on $4\times$ NVIDIA H100 GPUs. Additional details of the training configuration are in Appendix B.1.

Evaluation. We evaluate our model on the ImageNet-1k (Russakovsky et al., 2015) dataset at a resolution of 256. Samples are generated using a 50-step Heun (Karras et al., 2022) ODE solver with the optimal classifier-free guidance (CFG) scale (Ho & Salimans, 2022) and interval (Kynkäänniemi et al., 2024), following JiT. We report FID-50K and IS to evaluate the overall generation quality, computed with the standard Fréchet Inception Distance (FID) (Heusel et al., 2017) and Inception Score (IS) (Salimans et al., 2016). Additionally, we report the 5-crop FID as a metric that measures local structure, and compute the Fréchet Wavelet Distance (Veeramacheneni et al., 2025) at the high-frequency band, which we call high-frequency FWD (hFWD), to evaluate texture fidelity, following WaiT (Pavasovic et al., 2026). All metrics are calculated over 50K samples and all ablation studies report FID-50K, 5-crop FID, and hFWD for the B/16 model on ImageNet at 256×256 resolution after 200 training epochs. More details of the evaluation metrics can be found in Appendix B.2.

4.2 MAIN RESULTS

ImageNet 256×256. Fig. 5 compares BANDDIT with JiT and PixelREPA across training epochs with the same backbone, training recipe, and sampler. BANDDIT achieves a lower FID at every evaluation milestone compared to JiT, with an improvement of 2.2% at 600 epochs, suggesting faster convergence and a persistent advantage with extended training. On 5-crop FID, which measures the fidelity of local structure, BANDDIT improves by 5.3% at 600 epochs over the JiT baseline. Moreover, BANDDIT reduces hFWD, a metric sensitive to texture detail, by 17.4%. BANDDIT also achieves a better 5-crop FID and a comparable hFWD to PixelREPA without any pretrained external feature encoder. At 600 epochs, BANDDIT yields a higher FID than PixelREPA. This gap

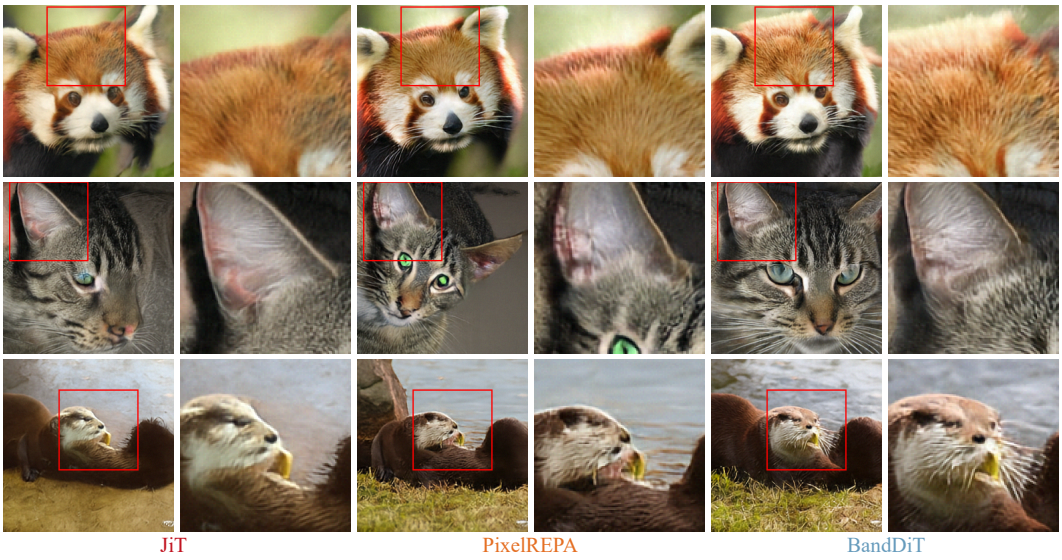


Figure 4: **Qualitative Comparisons.** Samples from the B/16 models of JiT, PixelREPA, and BANDDIT trained for 600 epochs, each paired with a zoomed view of the region in the red box. BANDDIT produces clearer and more realistic textures, such as fur and whiskers, than the two baselines.

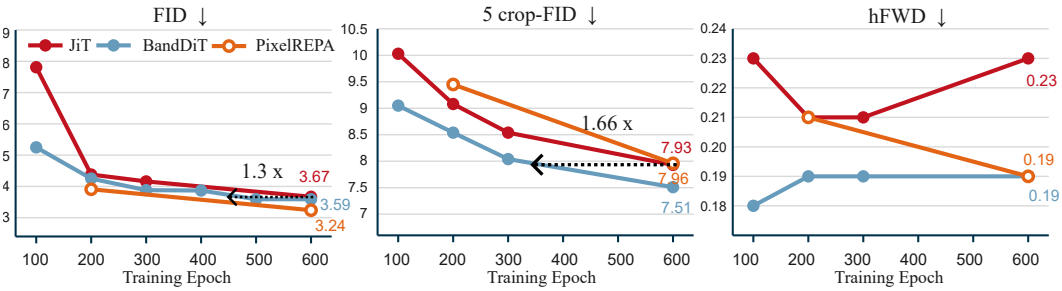


Figure 5: **Quantitative Comparisons.** FID, 5-crop FID, and hFWD (lower is better) over training epochs for the B/16 models of JiT, PixelREPA, and BANDDIT on ImageNet 256×256. The dotted arrows mark the epoch at which BANDDIT reaches the JiT value at 600 epochs, which takes 1.3× fewer epochs on FID and 1.66× fewer on 5-crop FID.

Method	Space	Backbone	Encoder	Params	Epoch	FID↓	IS↑	5-crop FID↓	hFWD↓
SiT-XL	latent	DiT	VAE	675M	1400	2.08	255.20	3.93	0.11
PixNerd-XL	pixel	DiT	DINOv2	700M	160	2.15	297.00	**2.96**	0.10
PixelDiT-XL	pixel	PixelDiT	–	797M	320	**1.67**	291.46	3.64	**0.08**
PixelREPA-H	pixel	JiT	DINOv2	953M	600	1.81	**315.96**	5.13	0.15
JiT-H	pixel	JiT	–	953M	600	1.91	297.57	5.70	0.16
BANDDIT-H	pixel	JiT	–	953M	500	1.90	311.81	5.50	0.13

Table 1: **Large-Scale Comparisons on ImageNet** 256×256**.** BANDDIT-H/16 against latent- and pixel-space models. Encoder lists any external model the method needs, and bold marks the best value in each column. Additional samples are in Appendix D.

may reflect PixelREPA's access to semantic priors encoded in pretrained DINO features, whereas our wavelet-band targets supervise image structure and texture without explicit class-level semantic guidance. Fig. 4 presents qualitative comparisons. BANDDIT produces competitive samples with coherent global structure, intricate local details, and rich high-frequency textures. PixelREPA more frequently reproduces class-appropriate textures, consistent with its DINO-based semantic supervision, while BANDDIT renders fine-scale structures such as fur and whiskers with more detail. These results suggest that recovering richer high-frequency content does not necessarily yield a lower FID.

$\mathcal{G}$	FID↓	5-crop FID↓	hFWD↓
None	4.60	9.08	0.26
$\mathcal{G}_0$	**4.23**	8.75	0.20
$\mathcal{G}_0 + \mathcal{G}_1$	4.30	8.79	0.21
$\mathcal{G}_0 + \mathcal{G}_1 + \mathcal{G}_2$	4.25	**8.54**	**0.19**

Table 3: **Ablation of Alignment Band.** FID, 5-crop FID and hFWD as the aligned band groups $\mathcal{G}$ include more band groups.

(l_0, l_1, l_2)	FID↓	5-crop FID↓	hFWD↓
None	4.60	9.08	0.26
$(9, 6, 3)$	4.73	8.65	0.21
$(6, 9, 12)$	4.58	8.64	**0.19**
$(3, 6, 9)$	**4.25**	**8.54**	**0.19**

Table 4: **Ablation of frequency hierarchy.** FID, 5-crop FID and hFWD for different block index (l_0, l_1, l_2) that align the coarse to fine band groups.

Additional samples are provided in Figs. 7 and 8. More qualitative comparison between the SiT, JiT, PixelREPA, and BandDiT can be found in Fig. 9.

Tab. 1 compares BANDDIT-H/16 with large-scale pixel-space and latent-space models. Among the methods that use no pretrained module and no architectural modification, BANDDIT achieves the best result on every metric, with the highest IS and the lowest FID, 5-crop FID, and hFWD. Among all the methods, BANDDIT reaches a lower hFWD than PixelREPA without its DINOv2 encoder, and a lower FID and a higher IS than the latent-space SiT without its VAE. PixNerd-XL achieves a better IS and 5-crop FID, while PixelDiT-XL achieves a lower FID and hFWD. The result of PixelDiT-XL may benefit from its patch-level encoder with a pixel-level decoder.

Scalability. We scale BANDDIT to the L and H models. As the model scales up, generation performance improves, with lower FID, hFWD, and 5-crop FID (Tab. 2). The improvement in local structure and texture persists across model sizes. More qualitative samples can be found in Appendix D.

Model	Epoch	FID↓	IS↑	5-crop FID↓	hFWD↓
JiT-L/16	600	2.39	289.74	6.29	0.23
BANDDIT-L/16	600	**2.20**	**294.41**	**6.16**	0.23
JiT-H/16	600	1.91	297.57	5.70	0.16
BANDDIT-H/16	500	**1.90**	**311.81**	**5.50**	**0.13**

Table 2: **Scaling in Model Size.** BANDDIT-L/16 and H/16 on ImageNet 256×256.

4.3 ABLATION STUDY

Ablation of Alignment Band. We test whether the gain arises from all band groups or primarily from one band group by adding groups in coarse-to-fine order. In Eq. (5), we change K from $\{\varnothing, 1, 2, 3\}$, indicating aligned groups $\mathcal{G} \in \{\varnothing, \mathcal{G}_0, \mathcal{G}_0 + \mathcal{G}_1, \mathcal{G}_0 + \mathcal{G}_1 + \mathcal{G}_2\}$. As shown in Tab. 3, the coarse band group $\mathcal{G}_0$ is mainly responsible for the FID gain, reducing FID from 4.60 to 4.23. This is consistent with the observation that FID is determined by global coherence, where the low-to-mid frequency bands dominate. Adding the detailed band $\mathcal{G}_2$ did not improve FID further but improved local structure and texture details, with lower 5-crop FID and hFWD. This is consistent with the observation that the coarse band group $\mathcal{G}_0$ determines global-structure quality (Pavasovic et al., 2026; Veeramacheneni et al., 2025), while the fine groups $\mathcal{G}_1$ and $\mathcal{G}_2$ contribute local detail and texture, enabling the full model to improve all metrics.

Ablation of Frequency Hierarchy. Sec. 3.2 shows that the transformer backbone constructs the bands in coarse-to-fine order along the depth dimension. Accordingly, BANDDIT assigns the band groups to blocks in the same hierarchy. We ablate whether this ordering is essential to the observed improvement in the metrics. Specifically, we change the blocks (l_0, l_1, l_2) used to align the three band groups, comparing BANDDIT's default assignment $(3, 6, 9)$ with the reversed assignment $(9, 6, 3)$ and with the same ordering shifted to deeper layers $(6, 9, 12)$.

As shown in Tab. 4, the coarse-to-fine assignment achieves the lowest FID and 5-crop FID, and matches the best hFWD. Shifting the assignment to deeper layers eliminates most of the improvement, while reversing the order gives a worse FID than the model without alignment. These results indicate that a band group is beneficial only when aligned at the depth where the backbone naturally represents it; imposing it on a mismatched block can instead be detrimental. Thus, BANDDIT's gains arise from the placement of the band groups, rather than from the additional supervision alone.

Ablation of Alignment Weights λ. We investigate the appropriate weighting of the alignment loss relative to the flow loss. We sweep the scalar weight λ in Eq. (5), which uniformly scales all three alignment losses, over $\{200, 500, 2000\}$ while keeping all other settings identical to those of BANDDIT. As shown in Tab. 5, $\lambda = 500$ achieves the lowest FID and 5-crop FID. This suggests

λ	FID↓	5-crop FID↓	hFWD↓
0	4.60	9.08	0.26
200	4.48	9.01	0.21
500	**4.25**	**8.54**	0.19
2000	4.38	8.74	**0.18**

Table 5: **Ablation of Alignment Weight λ.** FID, 5-crop FID and hFWD as the alignment weight λ sweep.

$g_b(t)$	FID↓	5-crop FID↓	hFWD↓
0	4.60	9.08	0.26
$2-t$	4.40	8.70	0.21
$\mathbf{1}[t > t_b^*]$	4.42	8.85	0.20
1	**4.25**	**8.54**	**0.19**

Table 6: **Ablation on the Time Dependence of the Alignment Weight.** FID, 5-crop FID and hFWD with different time-dependency factor $g_b(t)$.

that the alignment loss must receive sufficient weight to shape the intermediate representations, but that excessive weighting causes it to compete with the flow loss.

Ablation on the Time Dependence of the Weight. The Wiener gain in Eq. (3) introduces a time-dependent weighting of the alignment loss. Here, we investigate whether alternative weighting schemes that emphasize alignment at specific noise levels improve generation quality. To test this, we multiply the normalized gain by a per-band time factor,

$$\tilde{\omega}_i(t) \;\rightarrow\; \tilde{\omega}_i(t)\, g_b(t),$$

and compare three choices of g_b: BANDDIT uses $g_b(t) = 1$. The linear ramp $g_b(t) = 2 - t$ raises the weight at the noisy end, the gate $g_b(t) = \mathbf{1}[t > t_b^*]$ supervises band b only after its reveal time t_b^*, the time at which the band's power in x_t rises above the noise floor, see Eq. (3). At low t only the coarse bands are aligned and the fine bands are attenuated. Tab. 6 shows that both time-dependent factors underperform $g_b(t) = 1$ on FID, 5-crop FID, and hFWD. This indicates that Wiener gain already scales each coefficient by its SNR, and extra time factors worsen the supervision.

5 CONCLUSION

We observe that diffusion transformers exhibit a depth-wise spectral hierarchy: low-frequency structure becomes decodable in shallow blocks, whereas high-frequency details emerge near the output. BANDDIT turns this emergent coarse-to-fine organization into an explicit training objective for pixel-space generation, using wavelet bands extracted from the clean image as teacher-free supervision targets at corresponding depths. On ImageNet at 256×256 resolution, BANDDIT-B/16 matches the baseline's FID and 5-crop FID with $1.3\times$ fewer training epochs while reducing high-frequency Fréchet Wavelet Distance by 17.4%. These results demonstrate that aligning intermediate supervision with the model's own spectral reconstruction process improves training efficiency and high-frequency fidelity while preserving overall generation quality.

Limitations. BANDDIT exhibits a slight trade-off between global coherence, measured by FID, and fine detail, measured by 5-crop FID and hFWD. We observe that a lower 5-crop FID and hFWD sometimes come with a slightly worse FID. A possible explanation is that wavelet-band alignment weights all frequency bands equally, and improvements in generating more detail might distract class-guided generation, resulting in a worse FID. Nevertheless, without external semantic information or an additional learned module, simple frequency-based supervision targets improve fidelity in both global structure and texture generation over the baseline, offering a new perspective on the role of spectral representations in transformer-based generative models.

Future Work. A promising direction for future work is to extend BANDDIT to pixel-space image editing and inpainting. Its band-aligned supervision could promote consistency between synthesized regions and the surrounding content across spatial scales, from coarse structure to fine textures. The frequency-wise formulation may also provide a basis for selectively controlling the scale and richness of generated detail. Another direction is to combine band alignment with semantic representation alignment such as REPA, since the two may supervise complementary properties, frequency structure and class-level semantics. The band groups and their block depths are also fixed here, so learning them adaptively and scaling the approach to higher resolutions are natural next steps.

ETHICS STATEMENT

This work studies a training objective for class-conditional pixel-space image generation using ImageNet-1k. We collected no new personal data, conducted no human-subject research, and did not attempt to identify individuals in the training or evaluation data. ImageNet contains photographs of people and may contain personal information, and its images and labels reflect biases in online image collection and annotation. We use the dataset under its applicable terms and do not claim that the resulting model is suitable for biometric inference, surveillance, identity manipulation, impersonation, or other high-stakes applications.

As with other image generators, the model could be misused to produce misleading or deceptive images, and it may reproduce or amplify biases present in ImageNet. We did not perform a dedicated audit of memorization, privacy leakage, or subgroup-specific generation quality; these remain limitations of the present study. The model is evaluated only for research on class-conditional object generation, and the paper makes no claims about safe deployment for realistic person synthesis or consequential decision-making. The authors have no conflicts of interest to declare.

REPRODUCIBILITY STATEMENT

All experiments use the public ImageNet-1k dataset (Russakovsky et al., 2015) and build on the official JiT recipe (Li & He, 2025), with the same model sizes and training hyperparameters as the baseline. See Sec. 4 for the setup details. The supplementary material provides the wavelet band construction, the design details of the architecture, the training configuration, the pseudocode, and the full hyperparameter table. It also gives the metric protocols for FID-50K, 5-crop FID, and hFWD. We will release the training and evaluation code, configuration files, and result-generation scripts upon paper acceptance.

AI USE STATEMENT

In this work, we used generative AI tools for writing assistance, including language editing and polishing, and for suggesting potentially relevant prior work. The authors read the suggested papers and added references manually.

We used an AI coding agent to assist with research ideation, method and experiment implementation, code execution, and debugging. AI-assisted code was reviewed by the authors and checked with test functions and competing experiments designed to examine functionality and validate the implementation.

We used an AI chatbot to summarize and discuss experimental results and to provide critical feedback. The authors verified the underlying results, performed the scientific analysis, and made all decisions about the method, experiments, claims, and interpretation independently. We reviewed all AI-assisted text, literature suggestions, code, and analysis, and take full responsibility for the content of the paper.

REFERENCES

Michael S. Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Stochastic Interpolants: A Unifying Framework for Flows and Diffusions, October 2025. URL http://arxiv.org/abs/2303.08797. arXiv:2303.08797 [cs].

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers, May 2021. URL http://arxiv.org/abs/2104.14294. arXiv:2104.14294 [cs].

Hila Chefer, Patrick Esser, Dominik Lorenz, Dustin Podell, Vikash Raja, Vinh Tong, Antonio Torralba, and Robin Rombach. Self-supervised flow matching for scalable multi-modal synthesis. *arXiv preprint arXiv:2603.06507*, 2026.

Shoufa Chen, Chongjian Ge, Shilong Zhang, Peize Sun, and Ping Luo. PixelFlow: Pixel-Space Generative Models with Flow, April 2025. URL http://arxiv.org/abs/2504.07963. arXiv:2504.07963 [cs.CV].

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

Kun Cheng, Lei Yu, Zhijun Tu, Xiao He, Liyu Chen, Yong Guo, Mingrui Zhu, Nannan Wang, Xinbo Gao, and Jie Hu. Effective Diffusion Transformer Architecture for Image Super-Resolution, September 2024. URL http://arxiv.org/abs/2409.19589. arXiv:2409.19589 [cs.CV].

Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack, 2023.

Prafulla Dhariwal and Alex Nichol. Diffusion Models Beat GANs on Image Synthesis, June 2021. URL http://arxiv.org/abs/2105.05233. arXiv:2105.05233 [cs].

Sander Dieleman. Diffusion is spectral autoregression, September 2024. URL https://sander.ai/2024/09/02/spectral-autoregression.html. Blog post.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, June 2021.

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

Fabian Falck, Teodora Pandeva, Kiarash Zahirnia, Rachel Lawrence, Richard Turner, Edward Meeds, Javier Zazo, and Sushrut Karmalkar. A fourier space perspective on diffusion models. *arXiv preprint arXiv:2505.11278*, 2025.

Tiankai Hang, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo. Efficient diffusion training via min-snr weighting strategy. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7407–7417. IEEE, 2023.

Alex Henry, Prudhvi Raj Dachapally, Shubham Shantaram Pawar, and Yuxuan Chen. Query-key normalization for transformers. In *Findings of EMNLP*, 2020.

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

Jonathan Ho and Tim Salimans. Classifier-Free Diffusion Guidance, July 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models, December 2020. URL http://arxiv.org/abs/2006.11239. arXiv:2006.11239 [cs].

Jonathan Ho, Chitwan Saharia, William Chan, David J. Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded Diffusion Models for High Fidelity Image Generation, December 2021. URL http://arxiv.org/abs/2106.15282. arXiv:2106.15282 [cs].

Emiel Hoogeboom, Thomas Mensink, Jonathan Heek, Kay Lamerigts, Ruiqi Gao, and Tim Salimans. Simpler diffusion (SiD2): 1.5 FID on ImageNet512 with pixel-space diffusion. In *CVPR*, 2025.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Diederik Kingma and Ruiqi Gao. Understanding diffusion objectives as the elbo with simple data augmentation. *Advances in Neural Information Processing Systems*, 36:65484–65516, 2023.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017.

Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying Guidance in a Limited Interval Improves Sample and Distribution Quality in Diffusion Models, November 2024.

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 *ICCV*, 2025.

Tianhong Li and Kaiming He. Back to Basics: Let Denoising Generative Models Denoise, November 2025. URL http://arxiv.org/abs/2511.13720. arXiv:2511.13720 [cs].

Mingfeng Lin, Jiakun Chen, Liang Han, and Liqiang Nie. FREPix: Frequency-Heterogeneous Flow Matching for Pixel-Space Image Generation, May 2026. URL http://arxiv.org/abs/2605.06421. arXiv:2605.06421 [cs.CV].

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow Matching for Generative Modeling, February 2023. URL http://arxiv.org/abs/2210.02747. arXiv:2210.02747 [cs].

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow, September 2022. URL http://arxiv.org/abs/2209.03003. arXiv:2209.03003 [cs].

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

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, September 2024. URL http://arxiv.org/abs/2401.08740. arXiv:2401.08740 [cs].

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

Mang Ning, Mingxiao Li, Le Zhang, Lanmiao Liu, Matthew B. Blaschko, Albert Ali Salah, and Itir Onal Ertugrul. Spectrum matching: A unified perspective for superior diffusability in latent diffusion, 2026.

Anton Obukhov, Maximilian Seitzer, Po-Wei Wu, Semen Zhydenko, Jonathan Kyl, and Elvis Yu-Jing Lin. High-fidelity performance metrics for generative models in PyTorch, 2020. torch-fidelity library.

Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Pearson, 3rd edition, 2010.

Krunoslav Lehman Pavasovic, Théophane Vallaeys, Stéphane Mallat, Giulio Biroli, Luke Zettlemoyer, Brian Karrer, and Jakob Verbeek. WaiT for the Signal: Simple Frequency-Aware Flow-Matching, August 2026.

William Peebles and Saining Xie. Scalable Diffusion Models with Transformers, March 2023. URL http://arxiv.org/abs/2212.09748. arXiv:2212.09748 [cs].

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis, 2023.

Sucheng Ren, Qihang Yu, Ju He, Xiaohui Shen, Alan Yuille, and Liang-Chieh Chen. Frequency-Aware Flow Matching for High-Quality Image Generation, April 2026. URL http://arxiv.org/abs/2604.15521. arXiv:2604.15521 [cs.CV].

Severi Rissanen, Markus Heinonen, and Arno Solin. Generative modelling with inverse heat dissipation. *arXiv preprint arXiv:2206.13397*, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models, April 2022. URL http://arxiv.org/abs/2112.10752. arXiv:2112.10752 [cs].

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge, January 2015.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *NeurIPS*, 2016.

Johannes Schusterbauer, Ming Gui, Yusong Li, Pingchuan Ma, Felix Krause, and Björn Ommer. Denoising, Fast and Slow: Difficulty-Aware Adaptive Sampling for Image Generation, April 2026. URL http://arxiv.org/abs/2604.19141. arXiv:2604.19141 [cs.CV].

Noam Shazeer. GLU variants improve transformer, 2020.

Jaeyo Shin, Jiwook Kim, and Hyunjung Shim. Representation Alignment for Just Image Transformers is not Easier than You Think, March 2026.

Ivan Skorokhodov, Sharath Girish, Benran Hu, Willi Menapace, Yanyu Li, Rameen Abdal, Sergey Tulyakov, and Aliaksandr Siarohin. Improving the Diffusability of Autoencoders, June 2025.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising Diffusion Implicit Models, October 2022. URL http://arxiv.org/abs/2010.02502. arXiv:2010.02502 [cs].

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding, 2021.

Lokesh Veeramacheneni, Moritz Wolter, Hildegard Kuehne, and Juergen Gall. Fréchet Wavelet Distance: A Domain-Agnostic Metric for Image Generation, March 2025.

Shuai Wang, Zhi Tian, Weilin Huang, and Limin Wang. DDT: Decoupled Diffusion Transformer, April 2025a. URL http://arxiv.org/abs/2504.05741. arXiv:2504.05741 [cs].

Ziqiao Wang, Wangbo Zhao, Yuhao Zhou, Zekai Li, Zhiyuan Liang, Mingjia Shi, Xuanlei Zhao, Pengfei Zhou, Kaipeng Zhang, Zhangyang Wang, Kai Wang, and Yang You. REPA Works Until It Doesn't: Early-Stopped, Holistic Alignment Supercharges Diffusion Training, 2025b. URL http://arxiv.org/abs/2505.16792.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. Generation: Taming Optimization Dilemma in Latent Diffusion Models, March 2025.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation Alignment for Generation: Training Diffusion Transformers Is Easier Than You Think, June 2025a. URL http://arxiv.org/abs/2410.06940. arXiv:2410.06940 [cs].

Yongsheng Yu, Wei Xiong, Weili Nie, Yichen Sheng, Shiqiu Liu, and Jiebo Luo. PixelDiT: Pixel Diffusion Transformers for Image Generation, November 2025b. URL http://arxiv.org/abs/2511.20645. arXiv:2511.20645 [cs].

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In *NeurIPS*, 2019.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion Transformers with Representation Autoencoders, October 2025. URL http://arxiv.org/abs/2510.11690. arXiv:2510.11690 [cs].

Appendices

A Notation **15**

B Implementation and Experiment Details **16**

B.1 Training Configuration 16

B.2 Evaluation Metrics 17

B.3 Alignment Mechanism Design. 17

C Additional Analysis **19**

C.1 Spectral Compression Analysis of the Latent and Pixel Space. 19

D Additional Visualizations **20**

A Notation

Table 7: Glossary and Notation.

Symbol	Description
Diffusion and flow matching	
$x_0,\ \sigma_0$	Gaussian noise sample, $x_0 \sim \mathcal{N}(0, \sigma_0^2 I)$, and its noise scale
x_1	Clean image, $x_1 \sim p_{\text{data}}$
$t \in [0, 1]$	Time (noise level); $t = 0$ is pure noise and $t = 1$ is clean data
x_t	Linear interpolation $x_t = t\, x_1 + (1 - t)\, x_0$
v	Velocity of the path, $v = \mathrm{d}x_t/\mathrm{d}t = x_1 - x_0$
net_θ	Network, which predicts the clean image x_1 from (x_t, t)
v_θ	Velocity predicted from the network output, $v_\theta = (\text{net}_\theta(x_t, t) - x_t)/(1 - t)$
$\mathcal{L}_{\text{flow}}$	Flow-matching (velocity) loss of JiT
$\mathbb{E}$	Expectation over t, x_1, and x_0
Backbone and alignment	
D	Number of transformer blocks (depth) of the backbone
l	Block index, $1 \le l \le D$
$z_t^{(l)}$	Hidden state (tokens) at the output of block l for input x_t
K	Number of band groups, each with its own alignment loss ($K = 3$)
k	Index of a band group and its alignment loss, $k \in \{0, \ldots, K - 1\}$
l_k	Block at which group $\mathcal{G}_k$ is aligned, $l_0 < l_1 < l_2$
$h,\ h_k$	Two-block masked transformer adapter (from PixelREPA), and the adapter of group k
dec	Output layer of the backbone, shared by all groups, that decodes tokens into an image
$\text{Mask}(\cdot, r_k)$	Replaces a fraction r_k of the input tokens by a learned mask token

continued on next page

Tab. 7 (continued)

Symbol	Description
$r,\ r_k$	Fraction of tokens replaced by a learned mask token, and its value for group k ($(r_0, r_1, r_2) = (0.2, 0, 0)$)
$\hat{x}_t^{(l)},\ \hat{x}^{(k)}$	Image decoded from $z_t^{(l)}$ by the output layer; $\hat{x}^{(k)} = \hat{x}_t^{(l_k)}$ is the image decoded at block l_k
Wavelet bands and targets	
W	Four-level orthonormal Haar transform of an image
W_b	Coefficients of W in band b
b	Wavelet band, $b \in \{d_1, d_2, d_3, d_4, \ell_4\}$
d_j	Detail band at level j (three orientations pooled), holding spatial periods of 2^j to 2^{j+1} px
ℓ_4	Approximation band, holding spatial periods of 32 px and longer
$\mathcal{G}_k$	Band group: $\mathcal{G}_0 = \{d_3, d_4, \ell_4\}$ (low), $\mathcal{G}_1 = \{d_2\}$ (mid), $\mathcal{G}_2 = \{d_1\}$ (high frequency)
$\rho_b^{(l)}$	Decodability of band b at block l, the cosine similarity between $W_b(\hat{x}_t^{(l)})$ and $W_b(x_1)$ (Eq. (1))
$S_b(x)$	Band feature normalized by the image norm, $S_b(x) = W_b(x)/\|x\|_2$ (Eq. (2))
$\|\cdot\|_2$	Euclidean (ℓ_2) norm
i	Index of a wavelet coefficient within a band
c_i	Clean coefficient, $c_i = [W_b(x_1)]_i$
N_k	Number of coefficients in group $\mathcal{G}_k$
Alignment loss	
$\omega_i(t)$	Wiener gain of coefficient i, the fraction of its power attributable to the clean signal (Eq. (3))
$\tilde{\omega}_i(t)$	Wiener gain normalized to mean one within each band
$\mathcal{L}_k$	Alignment loss of group $\mathcal{G}_k$ (Eq. (4))
$g_b(t),\ t_b^*$	Per-band time factor of the time-weighting ablation, and the reveal time of band b at which its power exceeds the noise floor
λ	Weight of the alignment losses relative to the flow loss ($\lambda = 500$)
$\mathcal{L}$	Overall training objective, $\mathcal{L}_{\text{flow}} + \lambda \sum_k \mathbb{E}[\mathcal{L}_k]$ (Eq. (5))
Evaluation metrics	
FID, IS	Fréchet Inception Distance and Inception Score (lower FID and higher IS are better)
5-crop FID	FID computed on five crops per image, measuring local structure
hFWD	High-frequency Fréchet Wavelet Distance, measuring texture fidelity

B Implementation and Experiment Details

B.1 Training Configuration

We train BandDiT for class-conditional generation directly in pixel space on ImageNet (Russakovsky et al., 2015). The model uses JiT (Li & He, 2025) as its backbone, with the same parameter size and structure. Each transformer block applies RMSNorm (Zhang & Sennrich, 2019) before attention and before the feed-forward network, normalizes the query and key vectors (Henry et al., 2020), uses rotary position embedding (Su et al., 2021) in attention, and uses a SwiGLU feed-forward network (Shazeer, 2020). The adapter blocks of the alignment branches share this design. BandDiT adds the alignment loss on the outputs of three intermediate blocks during training and re-

	BANDDIT-B	BANDDIT-L	BANDDIT-H
alignment block indices l_k	{3, 6, 9}	{6, 12, 18}	{8, 16, 24}
band groups $\mathcal{G}_k$	$\mathcal{G}_0 = \{\ell_4, d_4, d_3\}, \mathcal{G}_1 = \{d_2\}, \mathcal{G}_2 = \{d_1\}$		
weight λ	500		
per-group mask ratio r_k	(0.2, 0, 0)		
adapter layers	2		

Table 8: **BANDDIT hyperparameters.** The table lists the hyperparameters for the BANDDIT alignment loss. All other training hyperparameters follow the official JiT code.

moves the auxiliary head at sampling time. We train with the official JiT recipe on 4×H100 GPUs, using the AdamW (Kingma & Ba, 2017; Loshchilov & Hutter, 2019) optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and no weight decay. The learning rate is 5×10^{-5} scaled by the global batch size divided by 256, with a linear warm-up over the first 5 epochs. Training details are in Tab. 8.

B.2 EVALUATION METRICS

The 5-crop FID and hFWD defined below are computed on the same 50K generated images used to calculate the FID reported in the corresponding row. The reference set for both consists of the 50K ImageNet validation images, center-cropped and resized to 256×256 following ADM (Dhariwal & Nichol, 2021). We use the Fréchet distance between two Gaussians, $\mathrm{FD}(\mu_g, \Sigma_g; \mu_r, \Sigma_r) = \|\mu_g - \mu_r\|_2^2 + \mathrm{tr}\big(\Sigma_g + \Sigma_r - 2(\Sigma_g\Sigma_r)^{1/2}\big)$, with the same implementation as the FID. FID and IS follow the JiT evaluation protocol (Li & He, 2025) and are computed with torch-fidelity (Obukhov et al., 2020) against the same ImageNet reference statistics.

5-crop FID. 5-crop FID measures how closely the local structure of the generated images matches the local structure of real images, and follows the definition of WaiT (Pavasovic et al., 2026). 5-crop FID takes five square crops of 128 pixels from each image, at the four corners and the center, and computes the FID between all crops of the generated images and all crops of the real images.

High-Frequency Fréchet Wavelet Distance (hFWD). hFWD measures how closely the texture of the generated images matches the texture of real images, and follows the definition of WaiT (Pavasovic et al., 2026). hFWD applies a four-level Haar wavelet packet transform to each image, then calculates the Fréchet distance between the generated and the real images measured on Haar wavelet packet coefficients, averaged over every frequency band except the DC band, which holds the mean intensity of each 16×16 pixel block.

B.3 ALIGNMENT MECHANISM DESIGN

Wavelet Bands. Tab. 9 lists the five bands of the four-level Haar band at 256×256 image size. The wavelet-band decomposition uses a four-level orthonormal Haar basis. Each band contain coefficients about the horizontal, vertical, and diagonal detail components in the corresponding spatial period. The energy of each band is computed as the sum of squared wavelet coefficients,

$$E_s = \|W_s(x)\|_2^2.$$

The coarse group $\mathcal{G}_0$ captures most of a natural image's energy in only 6.25% of the coefficients, whereas the finest-octave group $\mathcal{G}_2$ contains 75% of the coefficients. To account for the differences in group size, we normalize the the total energies to unit to make the losses comparable.

Wiener Gain. Let $c_i = [W_b(x_1)]_i$ be a clean coefficient and let y_i be the corresponding coefficient of the interpolated input x_t. The Haar transform is orthonormal, so white image-space noise stays white in the wavelet domain, and $y_i = tc_i + (1-t)\epsilon_i$ with $\epsilon_i \sim \mathcal{N}(0, \sigma_0^2)$. The clean signal contributes power $t^2c_i^2$ and the noise contributes power $(1-t)^2\sigma_0^2$, and their signal-to-total-power ratio is the gain of Eq. (3). As t increases, the noise term falls and the gains approach one, so the objective follows the evidence available in the corrupted input.

The raw gains have different average magnitudes across time and across bands, which would confound the reweighting with a change in the total strength of the alignment loss. The gains are

Algorithm 1 BANDDIT Training Objective

Require: clean image x_1, noise x_0, time t; network net_θ with output layer dec; band groups $\mathcal{G}_k$, blocks l_k, adapters h_k, and mask ratios r_k for $k = 0, \dots, K-1$; weight λ; noise scale σ_0

$x_t \leftarrow t\, x_1 + (1-t)\, x_0$ ▷ interpolant
run net_θ on (x_t, t) to obtain its prediction and the hidden states $z_t^{(l_k)}$
$\mathcal{L}_{\text{flow}} \leftarrow \|v_\theta(x_t, t) - (x_1 - x_0)\|_2^2$ ▷ flow-matching loss
$S_b(x_1) \leftarrow W_b(x_1)/\|x_1\|_2$ for every band b ▷ Eq. (2)
$\tilde{\omega}_i(t) \leftarrow$ mean-one Wiener gain of each coefficient of $W(x_1)$ ▷ Eq. (3)
for $k = 0$ **to** $K-1$ **do**
  $\hat{x}^{(k)} \leftarrow \mathrm{dec}\big(h_k(\mathrm{Mask}(z_t^{(l_k)}, r_k))\big)$ ▷ mask a fraction r_k of tokens, adapt, decode
  $\mathcal{L}_k \leftarrow \frac{1}{N_k} \sum_{b \in \mathcal{G}_k} \sum_{i \in b} \tilde{\omega}_i(t) \big([S_b(\hat{x}^{(k)})]_i - [S_b(x_1)]_i\big)^2$ ▷ Eq. (4)
end for
return $\mathcal{L}_{\text{flow}} + \lambda \sum_k \mathcal{L}_k$ ▷ Eq. (5)

band	Level	Period (px)	Coefficients	Group
d_1	1	2–4	$3{\times}128^2 = 49{,}152$	$\mathcal{G}_2$
d_2	2	4–8	$3{\times}64^2 = 12{,}288$	$\mathcal{G}_1$
d_3	3	8–16	$3{\times}32^2 = 3{,}072$	$\mathcal{G}_0$
d_4	4	16–32	$3{\times}16^2 = 768$	$\mathcal{G}_0$
ℓ_4	> 4	≥ 32	$16^2 = 256$	$\mathcal{G}_0$

Table 9: **Haar Bands of a** $256{\times}256$ **Image.** Haar wavelet band and related information.

therefore normalized within each orientation band:

$$\widetilde{\omega}_i(t) = \frac{\omega_i(t)}{\mathrm{mean}_{i' \in s(i)}\, \omega_{i'}(t)},$$

where $s(i)$ collects the spatial positions with the same orientation, band, and color channel as coefficient i, so that the gains have mean one within each such slice.

Decoder and Adapter. For each selected layer l_k, the hidden state $z_t^{(l_k)}$ passes through an adapter h_k and is decoded by the output layer dec of the network:

$$\hat{x}^{(k)} = \mathrm{dec}\big(h_k(\mathrm{Mask}(z_t^{(l_k)}, r_k))\big).$$

The decoder is shared with the backbone and has frozen weights. The adapter is built from two transformer blocks with the same time and class conditioning as the backbone.

Mask Design. The groups are assigned from coarse to fine, and Tab. 8 gives the block index of each model size (B,L,H). The output decoder is linear per token, so without masking a group prediction could reconstruct each patch from its own token and repeat the mapping the flow objective has already learned. We therefore replace a fraction of the tokens in the alignment input with a learned mask token, which encourages the reconstruction of noisy tokens to depend on context. The mask is applied to the $\mathcal{G}_0$ group alignment only, because $\mathcal{G}_1$ and $\mathcal{G}_2$ have spatial periods shorter than a patch, so masking a token would bottleneck the learning of details.

Pseudocode. Algorithm 1 computes the BANDDIT loss for one training sample. It forms the interpolated input, runs the network to obtain its prediction and the intermediate hidden states, and calculates the flow-matching loss. It then builds the target from the clean image by a Haar transform and a division by the image norm, and evaluates the normalized Wiener gain of every clean coefficient at time t. For each alignment loss, it replaces a fraction of the tokens of the selected hidden state with a learned mask token, passes the result through the adapter, and decodes it with the output layer. It transforms the resulting estimate, scores its squared difference from the target on the band group under the gain, and divides by the coefficient count. It returns the flow-matching loss plus the group losses scaled by λ.

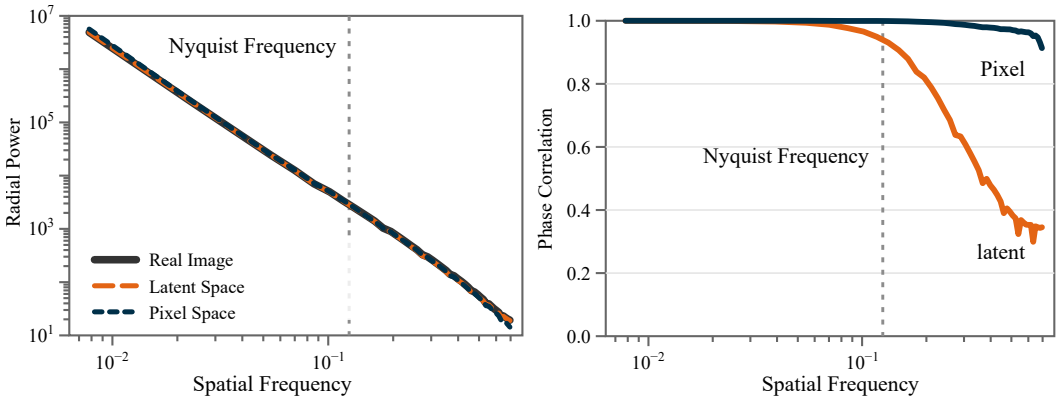


Figure 6: **Energy and Phase Preservation for Latent and Pixel Space Models.** (a) Energy spectrum and (b) phase correlation between the clean image and its reconstruction by the autoencoder of the latent-space model and by the pixel-space model. The latent-space autoencoder loses fine detail during compression, whereas the pixel-space model does not.

C Additional Analysis

C.1 Spectral Compression Analysis of the Latent and Pixel Space

To analyze the compression loss of the autoencoder, we compute the information loss by passing images through the autoencoder of the latent-space model (Rombach et al., 2022), and through the patch embedding and the output layer of the pixel-space model (Li & He, 2025). We measure the difference between the input and output in energy spectrum in Fig. 6(a) and the phase correlation in Fig. 6(b). The phase correlation is calculated between input and output frequency coefficients.

Fig. 6(a) shows that the autoencoder preserves energy across the whole spectrum. However, as the frequency increases, the phase correlation drops considerably for the latent-space model. The drop aligns with the Nyquist frequency (Oppenheim & Schafer, 2010) of the latent patch size, the highest frequency the latent grid can represent, corresponding to a period of twice the patch size. The correlation falls to 0.06 in the highest quarter of the spectrum. For the pixel space, the phase correlation overlaps well with the original image, indicating nearly zero information loss. These results indicate that the autoencoder struggles to preserve details at scales finer than the latent patch size, limiting the ability of latent-space targets to supervise fine-scale structure. Pixel-space targets can better preserve the information, a property that benefits the wavelet-band supervision of Sec. 3.3.

D ADDITIONAL VISUALIZATIONS

Figs. 7 and 8 show further class-conditional samples at 256×256 resolution from BANDDIT-H/16, trained for 500 epochs and sampled with a CFG scale of 2.5, and from BANDDIT-L/16, trained for 600 epochs and sampled with a CFG scale of 2.45, respectively. The samples cover diverse ImageNet categories and illustrate the global structure, local detail, and texture of the generated images at both model sizes. Fig. 9 presents additional qualitative comparisons between SiT-XL Ma et al. (2024), JiT-H Li & He (2025), PixelREPA-H Shin et al. (2026), and BandDiT-H. Each example spans two rows, with the full images shown above and corresponding zoomed-in crops below. Across these examples, BandDiT produces richer high-frequency textures with more coherent fine-scale structure. PixelREPA captures class-specific textures but often lacks finer details and can exhibit inconsistencies in their spatial organization. The JiT baseline likewise tends to underrepresent high-frequency content. SiT achieves strong overall image quality but loses some fine-scale texture, as illustrated by the missing fur details on the panda's head in the final example, consistent with the loss of fine-scale information through latent-space compression.



Figure 7: **Qualitative Samples from BANDDIT-H/16.** Class-conditional 256×256 samples from BANDDIT-H/16 trained for 500 epochs on ImageNet and generated with classifier-free guidance at scale 2.5.



Figure 8: **Qualitative Samples from BANDDIT-L/16.** Class-conditional 256×256 samples from BANDDIT-L/16 trained for 600 epochs on ImageNet and generated with classifier-free guidance at scale 2.45.

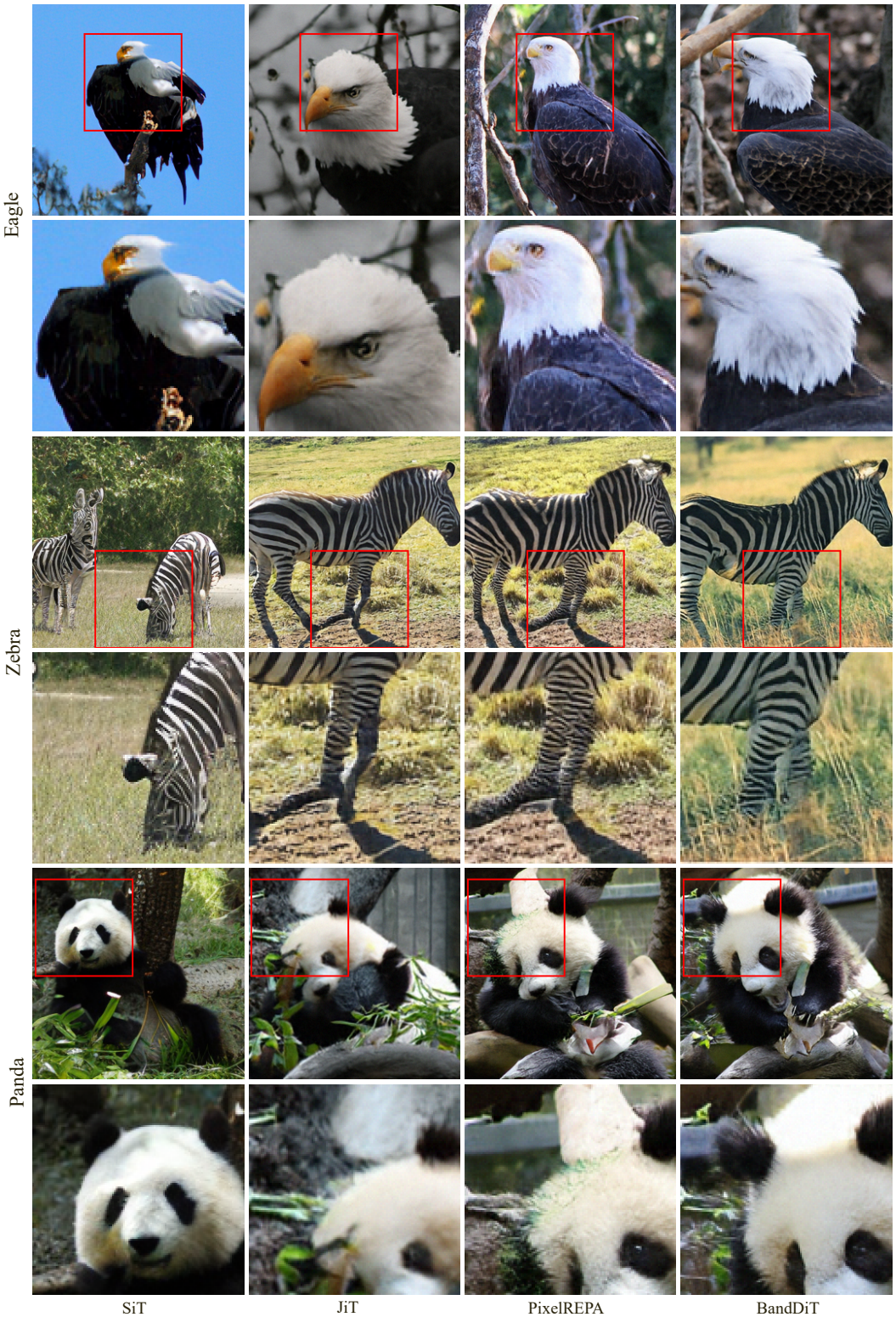


Figure 9: **Qualitative Comparison between SiT, JiT, PixelREPA and BandDiT.** Each pair of rows shows full images followed by corresponding zoomed-in crops. BandDiT produces richer high-frequency textures with more coherent fine-scale structure. PixelREPA captures class-specific textures but can lack fine detail and spatial consistency, while JiT tends to underrepresent high-frequency content. SiT produces strong overall results but misses some fine-scale textures, as illustrated by the panda fur in the final example.