Supplementary Material "PixelDiT: Pixel Diffusion Transformers for Image Generation"

1. Architecture and System Details

1.1. Summary of Model Size

To study the impact of model size, we evaluate the base (B), large (L), and extra-large (XL) variants of PixelDiT on ImageNet-256. Tables 1 and 2 summarize the detailed architectural specifications and training settings for B, L, XL and T2I variants. **Note that the default configuration of all experiments in the main paper is PixelDiT-XL.** If not otherwise specified, we use the XL configuration for all ImageNet-256 experiments in this appendix.

	PixelDiT-B	PixelDiT-L	PixelDiT-XL
Architecture			
Input dim.		$256 \times 256 \times 3$	
Patch-level depth N	12	22	26
Pixel-level depth M	2	4	4
Hidden size D	768	1024	1152
Heads	12	16	16
Pixel hidden size D_{pix}	16	16	16
Patch size p	16	16	16
#Params (M)	184	569	797
Representation Alignment [27]			
Alignment weight		0.5	
Alignment depth		Layer 8 of the patch-level pathway	
sim$(\cdot,\cdot)$		Cosine Similarity	
Encoder $f(x)$		Frozen DINOv2 [20]	
Optimization			
Training iteration	800K	1M	1.6M
Batch Size		256	
Timestep Sampler		Logit-normal [6]	
Optimizer		AdamW [16], β_1=0.9, β_2=0.999	
EMA Decay		0.9999	
Class Drop Prob.		0.1	
Gradient Clipping	1.0	1.0	1.0 $\rightarrow$ 0.5
Learning Rate	1e-4	1e-4	1e-4 $\rightarrow$ 1e-5
Weight Decay	0	0	0
Sampling			
Sampler		FlowDPMSolver [17, 25]	
Sampling Steps		100	
CFG scale (80-ep)	3.25	3.25	3.25
CFG interval (80-ep)		[0.10, 1.00]	
CFG scale (320-ep)	–	–	2.75
CFG interval (320-ep)	–	–	[0.10, 0.90]
Guidance for Figs. 3–6		3.25, [0.10, 1.00] for all checkpoints	

Table 1. Detailed architecture and training configurations for PixelDiT B/L/XL models on ImageNet-256.

1.2. Text-to-Image Architecture with MM-DiT

Figure 1 illustrates the T2I variant of PixelDiT, where the patch-level pathway is extended with MM-DiT blocks [6] to fuse text embeddings, while the pixel-level pathway remains unchanged. The figure emphasizes stream separation, conditioning flow, and the pixel-wise modulation interface used by the pixel-level pathway.

2. Solvers and Guidance Scales

2.1. Ablation of Solvers

We compare three diffusion samplers for denoising on ImageNet-256: FlowDPMSolver [17, 25], Euler, and Heun, all run for 100 steps without classifier-free guidance. Figure 2 plots gFID, sFID, Inception Score (IS), precision, and recall as training progresses from 100K to 1,200K iterations. Across most of the training trajectory, FlowDPMSolver achieves lower or comparable gFID and sFID than

Hyperparameter	PixelDiT-T2I
Architecture	
Input dim.	$512^2 \times 3$ / $1024^2 \times 3$
Patch-level depth N	14
Pixel-level depth M	2
Hidden size D	1536
Heads	24
Pixel hidden size D_{pix}	16
Patch size p	16
#Params (M)	1311
Optimization	
Training iteration	400K (512px) $\rightarrow$ 100K (1024px)
Batch Size	1024 (512px) $\rightarrow$ 768 (1024px)
Learning Rate	1e-4 (512px) $\rightarrow$ 2e-5 (1024px)
Gradient Clipping	0.5 (512px) $\rightarrow$ 0.1 (1024px)
Text Drop Prob.	0.1
Inference	
Sampler	FlowDPMSolver
Sampling Steps	25
CFG scale	4.5

Table 2. Implementation details for PixelDiT-T2I model (1024^2).

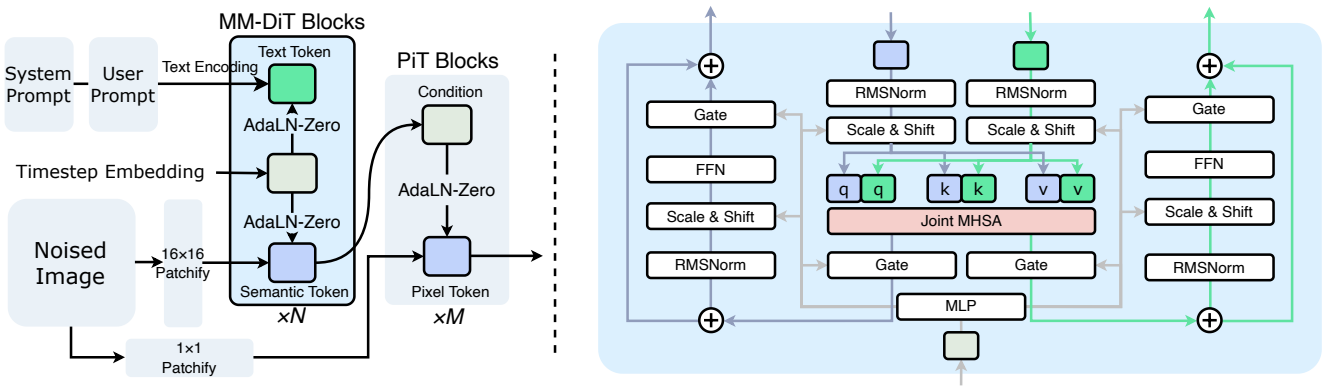


Figure 1. T2I architecture of PixelDiT with MM-DiT blocks on the patch-level pathway. The pixel-level pathway performs dense per-pixel modeling conditioned on semantic tokens.

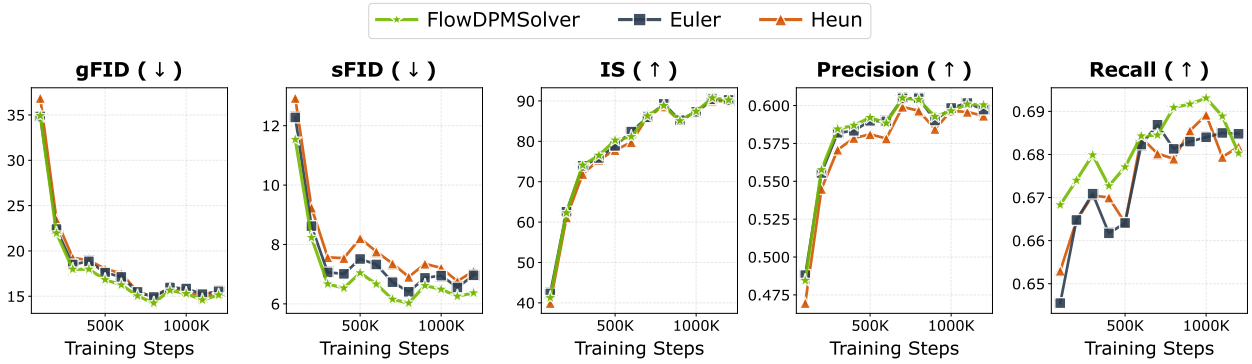


Figure 2. Comparison of FlowDPMSolver, Euler, and Heun samplers on ImageNet-256 with 100 inference steps and no classifier-free guidance. FlowDPMSolver achieves the best combined trade-off between fidelity in gFID and sFID and diversity in IS, precision, and recall, which motivates its use as our default sampler.

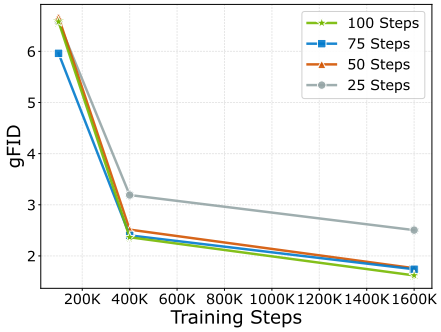
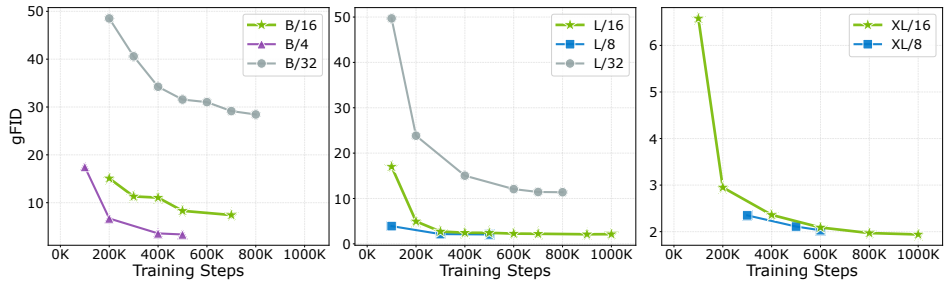


Figure 3. Effect of the number of FlowDPMSolver inference steps on gFID for PixelDiT-XL at different training stages on ImageNet-256. Increasing the number of steps is most beneficial once the model is moderately or fully trained (400K and 1.6M iterations), with diminishing returns beyond 50 steps; we adopt 100 steps as the default for ImageNet experiments.

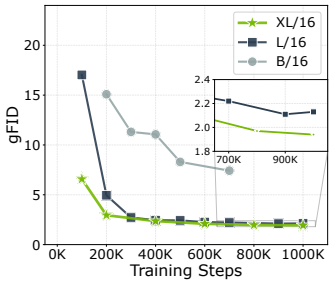
Euler and Heun, with the gap particularly pronounced in the low- and mid-epoch regimes (up to roughly 1–2 gFID points around 400K–800K iterations). FlowDPMSolver maintains the best overall trade-off: it matches or exceeds the competing solvers on sFID and IS while keeping precision and recall high. These results motivate our choice of FlowDPMSolver as the default sampler for all main ImageNet and text-to-image evaluations.

2.2. Inference Steps

We further analyze the impact of the number of inference steps when using FlowDPMSolver. Figure 3 shows gFID for PixelDiT-XL at three training stages (100K, 400K, and 1.6M iterations) as we vary the sampling budget from 25 to 100 steps. At 100K iterations the model is undertrained and additional steps give only modest improvements, with gFID remaining in the 6–7 range. Once the model has learned reasonable global structure (400K iterations), increasing the budget from 25 to 50 steps reduces gFID from about 3.19 to 2.51, and 100 steps further improves it to 2.36. For the fully converged checkpoint at 1.6M iterations, 25 steps already achieve gFID $\approx$ 2.50, but 50–75 steps lower it to around 1.76–1.74, and 100 steps obtain the best score of approximately 1.61 gFID. Overall, more inference steps consis-



(a) Patch-size ablations for B/L/XL models on ImageNet-256.



(b) Comparison at fixed patch size.

Figure 4. Convergence analysis of PixelDiT on ImageNet-256. (a) gFID vs. training iterations for B, L, and XL models with varying patch sizes. (b) Comparison of B/L/XL models at a fixed patch size p=16.

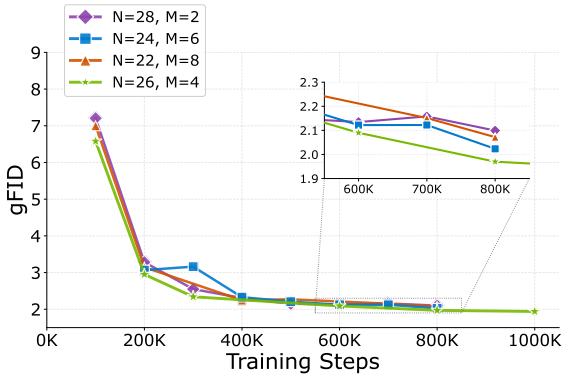


Figure 5. Ablation of depth allocation between patch-level (N) and pixel-level (M) pathways on ImageNet-256. Our chosen configuration (N=26, M=4), highlighted in brown, offers the best trade-off between early convergence and final image quality.

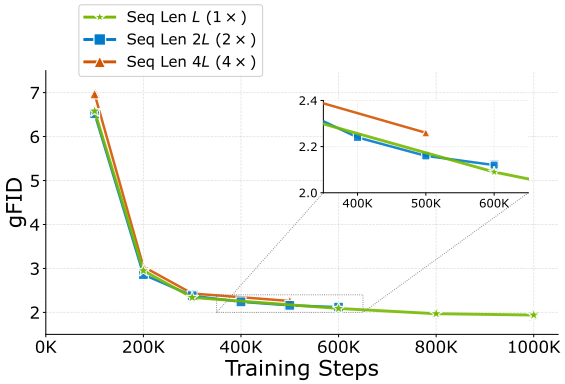


Figure 6. Ablation of Pixel Token Compaction rates on ImageNet-256. The curves compare three post-compaction sequence lengths, denoted as "Seq Len L (1×)", "Seq Len $2L$ (2×)", and "Seq Len $4L$ (4×)", where L is the number of patch tokens after compaction. "Seq Len L (1×)" corresponds to our default configuration in which each p^2 pixel block is compacted into a single token.

tently benefit well-trained models, though the marginal gain beyond 50 steps becomes small. In practice we therefore use 100 steps for class-conditional ImageNet experiments to match the strongest quality, and 25 steps for text-to-image generation where sampling latency is more critical.

2.3. Guidance Scale and Interval

We report the classifier-free guidance (CFG) settings used for PixelDiT-XL at 80 and 320 epochs on ImageNet-256. Table 3 lists the CFG scale, active time interval, and the resulting gFID, sFID, IS, precision, and recall. For the 80-epoch checkpoint, the best gFID is 2.36, obtained with a relatively strong guidance scale of 3.25 applied over the entire denoising trajectory from t=0.10 to t=1.00. Increasing the scale to 3.50 or decreasing it to 3.00 slightly worsens gFID while mainly trading off IS and recall, and restricting the active interval to $[0.10, 0.95]$ or $[0.10, 0.90]$ does not lead to better performance. For the 320-epoch checkpoint, the optimum shifts toward milder guidance: a scale of 2.75 active on the interval $[0.10, 0.90]$ achieves the best gFID of 1.61 together with strong recall of 0.64, whereas both larger and smaller scales yield at most marginal IS gains at the cost of higher gFID. In the main paper we therefore adopt a guidance scale of 3.25 with interval $[0.10, 1.00]$ for the 80-epoch ImageNet-256 results, and a scale of 2.75 with interval $[0.10, 0.90]$ for the 320-epoch checkpoint that underpins our best reported scores.

3. Model Architecture Design

3.1. Ablations on Model Size and Patch Size

We investigate the impact of the patch size p on the performance of models with different scales: PixelDiT-B (184M), PixelDiT-L (569M), and PixelDiT-XL (797M). We evaluate patch sizes of 4, 8, 16, and 32 on ImageNet-256; Figure 4(a) visualizes the resulting convergence behavior. For PixelDiT-B, moving from p=32 to p=16 and p=4 substantially accelerates convergence: at 200K iterations gFID

Model	Epochs	Training Steps	CFG	Interval	gFID↓	sFID↓	IS↑	Prec.↑	Rec.↑
PixelDiT-XL	80	400K	3.25	[0.10, 1.00]	2.36	5.11	282.3	0.80	0.57
PixelDiT-XL	80	400K	3.50	[0.10, 1.00]	2.60	5.07	305.2	0.82	0.57
PixelDiT-XL	80	400K	3.00	[0.10, 1.00]	2.60	5.06	277.6	0.80	0.58
PixelDiT-XL	80	400K	2.75	[0.10, 1.00]	2.76	5.17	259.2	0.79	0.59
PixelDiT-XL	80	400K	3.25	[0.10, 0.95]	2.73	5.24	285.8	0.80	0.58
PixelDiT-XL	80	400K	3.25	[0.10, 0.90]	2.75	5.32	285.4	0.80	0.58
PixelDiT-XL	320	1600K	2.75	[0.10, 0.90]	1.61	4.68	292.7	0.78	0.64
PixelDiT-XL	320	1600K	2.75	[0.10, 0.95]	1.65	4.64	293.8	0.77	0.64
PixelDiT-XL	320	1600K	2.75	[0.10, 1.00]	1.66	4.60	294.2	0.78	0.64
PixelDiT-XL	320	1600K	2.50	[0.10, 0.95]	1.69	4.68	276.9	0.77	0.65
PixelDiT-XL	320	1600K	2.50	[0.10, 0.90]	1.71	4.60	275.2	0.77	0.65
PixelDiT-XL	320	1600K	2.50	[0.10, 1.00]	1.71	4.62	277.9	0.77	0.65

Table 3. CFG settings and results for PixelDiT-XL on ImageNet-256.

Patch	80 epochs		120 epochs	
	FID↓	IS↑	FID↓	IS↑
p=16	**2.97**	**270.29**	**2.23**	**287.62**
p=32	4.66	235.52	3.78	256.06

Table 4. **Patch size ablation on ImageNet-512.** p=16 consistently outperforms p=32 despite the latter's lower compute cost.

drops from 48.5 (B/32) to 15.1 (B/16) and 6.7 (B/4), and B/4 ultimately reaches 3.4 gFID at 500K iterations. Larger models follow a similar trend, but the benefit of very small patches diminishes with scale. For PixelDiT-L, using p=8 rather than p=16 improves gFID only modestly (from 2.72 to 2.15 at 300K iterations), and for PixelDiT-XL the gap between p=8 and p=16, both configurations converge to gFID near 2.0. These results highlight a trade-off: smaller patches yield better or faster convergence but incur a quadratic cost in sequence length, and the relative gain shrinks as model capacity increases. In practice we therefore use p=16 as the default patch size for PixelDiT-XL, which offers near-optimal quality at substantially lower compute.

We further extend this analysis to ImageNet-512 to examine whether a larger patch size (p=32) could provide a favorable compute–quality trade-off at higher resolution. As shown in Table 4, while p=32 reduces computational cost (see Table 10), it consistently underperforms p=16 at both 80 and 120 epochs. At 120 epochs, p=16 achieves an FID of 2.23 compared to 3.78 for p=32. These results suggest that the finer-grained patch tokenization remains important at higher resolutions, and that the quality benefit of smaller patches outweighs the compute savings from larger patches.

To analyze the effect of model size, Figure 4(b) compares B, L, and XL variants at a fixed patch size. Scaling the model yields consistent gains across the entire training trajectory: at 200K iterations, gFID improves from 15.1 (B/16) to 4.95 (L/16) and 2.95 (XL/16), and at 1M iterations XL/16 reaches 1.94 gFID compared to roughly 2.1 for L/16. Thus, increasing capacity improves both image quality and the speed at which a given quality level is reached, *supporting the scalability claims made in the main paper*.

3.2. Ablation on Depth N and M

We analyze how to allocate depth between the patch-level pathway (N layers) and the pixel-level pathway (M layers) under a fixed total budget of roughly $N+M\approx30$ layers. Figure 5 shows convergence curves for several (N, M) configurations evaluated on ImageNet-256. All evaluations use the same CFG guidance scale 3.25 with interval $[0.10, 1.00]$. Introducing even a shallow pixel pathway (e.g., N=28, M=2) dramatically improves convergence and reduces final gFID to around 2.1. Our default configuration (N=26, M=4) provides the best overall behavior: it reaches gFID 2.34 by 300K iterations and continues to improve to 1.94 at 1M iterations, outperforming both the shallower pixel pathway (N=28, M=2) and the deeper one (N=22, M=8). The latter attains similar final gFID but converges more slowly in early epochs. These trends indicate that dedicating a moderate but not excessive number of layers to the pixel-level pathway is crucial for efficient pixel modeling and underpins the strong ImageNet results reported in the main paper.

3.3. Ablation on Representation Alignment (REPA)

We ablate the representation alignment (REPA) loss that encourages patch-level features to stay aligned with frozen DINOv2 encoder features. Table 7 compares PixelDiT-XL with and without REPA on ImageNet-256 at 80 and 160 epochs. Removing REPA leads to substantially worse FID and IS at both checkpoints: at 80 epochs, FID degrades from 2.36 to 6.58 and IS drops from 282.3 to 165.96. While longer training partially closes the gap (FID improves from 6.58 to 4.33 at 160 epochs without REPA), the model with REPA consistently outperforms by a large margin. These results confirm that representation alignment is an essen-

Model	Params (B)	Overall ↑	Objects		Counting	Colors	Position	Color Attribution
			Single	Two				
512 ×512 resolution								
PixArt-α	0.6	**0.48**	0.98	0.50	0.44	0.80	0.08	0.07
PixArt-Σ	0.6	**0.52**	0.98	0.59	0.50	0.80	0.10	0.15
PixelFlow [3]	0.9	**0.60**	-	-	-	-	-	-
PixNerd [23]	1.2	**0.73**	0.97	0.86	0.44	0.83	0.71	0.53
PixelDiT-1.3B	1.3	**0.78**	1.00	0.94	0.70	0.90	0.53	0.65
1024 ×1024 resolution								
LUMINA-Next [29]	2.0	**0.46**	0.92	0.46	0.48	0.70	0.09	0.13
SDXL [22]	2.6	**0.55**	0.98	0.74	0.39	0.85	0.15	0.23
PlayGroundv2.5 [13]	2.6	**0.56**	0.98	0.77	0.52	0.84	0.11	0.17
Hunyuan-DiT [15]	1.5	**0.63**	0.97	0.77	0.71	0.88	0.13	0.30
DALLE3 [19]	-	**0.67**	0.96	0.87	0.47	0.83	0.43	0.45
FLUX-dev [12]	12.0	**0.67**	0.99	0.81	0.79	0.74	0.20	0.47
PixelDiT-1.3B	1.3	**0.74**	1.00	0.95	0.55	0.88	0.41	0.68

Table 5. **GenEval category-wise results** at 512×512 and 1024×1024 for text-to-image generation. *Overall* is the unweighted mean over Single Object, Two Objects, Counting, Colors, Position, and Color Attribution.

Model	Params (B)	Overall ↑	Global	Entity	Attribute	Relation	Other
512 ×512 resolution							
PixArt-α [2]	0.6	**71.6**	81.7	80.1	80.4	81.7	76.5
PixArt-Σ [1]	0.6	**79.5**	87.5	87.1	86.5	84.0	86.1
PixelFlow [3]	0.9	**77.9**	-	-	-	-	-
PixNerd [23]	1.2	**80.9**	80.5	87.9	87.2	91.3	72.8
PixelDiT-1.3B	1.3	**83.7**	88.0	90.9	87.6	89.8	88.5
1024 ×1024 resolution							
LUMINA-Next [29]	2.0	**74.6**	82.8	88.7	86.4	80.5	81.8
SDXL [22]	2.6	**74.7**	83.3	82.4	80.9	86.8	80.4
PlayGroundv2.5 [13]	2.6	**75.5**	83.1	82.6	81.2	84.1	83.5
Hunyuan-DiT [15]	1.5	**78.9**	84.6	80.6	88.0	74.4	86.4
PixArt-Σ [1]	0.6	**80.5**	86.9	82.9	88.9	86.6	87.7
DALLE3 [19]	-	**83.5**	91.0	89.6	88.4	90.6	89.8
FLUX-dev [12]	12.0	**84.0**	82.1	89.5	88.7	91.1	89.4
PixelDiT-1.3B	1.3	**83.5**	83.0	88.6	87.8	91.2	89.6

Table 6. **DPG-Bench category-wise results** at 512×512 and 1024×1024 resolutions for text-to-image generation.

tial component of our pixel-space diffusion framework, providing critical regularization for the patch-level pathway to maintain semantically meaningful representations throughout training.

Configuration	80 epochs		160 epochs	
	FID↓	IS↑	FID↓	IS↑
PixelDiT-XL w/ REPA	**2.36**	**282.3**	**1.97**	**299.4**
PixelDiT-XL w/o REPA	6.58	165.9	4.33	242.4

Table 7. **Ablation of representation alignment (REPA)** on ImageNet-256. Removing REPA leads to substantially degraded FID and IS at both training stages.

3.4. Study on Pixel Token Compaction (PTC) Rate

We investigate the effectiveness of Pixel Token Compaction (PTC) by varying the compaction rate. Recall that our default patch size is p. Without compaction, the pixel-level pathway would process a sequence of length $H \times W$. With standard compaction (denoted as "Seq Len L ($1\times$)" or Base), the $p \times p$ pixels in a patch are compressed into a single token, reducing the sequence length to $L = (H/p) \times (W/p)$. We explore relaxing this compression by allowing the compacted sequence length to be multiples of the base length L. Specifically:

- **Seq Len** L ($1\times$): The default setting. Compresses p^2 pixels to 1 token. Compression rate: p^2.
- **Seq Len** $2L$ ($2\times$): Compresses p^2 pixels to 2 tokens.

Compression rate: $p^2/2$.

- **Seq Len** $4L$ $(4\times)$: Compresses p^2 pixels to 4 tokens. Compression rate: $p^2/4$.

Figure 6 presents the ablation results. Across training, all three settings converge to strong gFID values around 2.0, while the model with the most aggressive compression (Seq Len $1\times$) obtains slightly better results. For example, at 300K iterations the three configurations obtain gFID of roughly 2.34 ($1\times$), 2.38 ($2\times$), and 2.43 ($4\times$), respectively. At 1M iterations the $1\times$ variant is further improved to 1.94 while the longer sequences plateau slightly higher. The result suggests that, for the pixel-level pathway, a compact representation is sufficient to capture the residual information needed for texture refinement. This could be due to the redundancy nature of the pixel-space tokens. Interestingly, the results indicate that lower compression rates do not necessarily lead to better image quality. We suspect that a longer, redundant token sequence and a larger attention space can be more challenging to optimize and slower to converge under the same training setting. *Using more tokens in the pixel-level pathway may require carefully adjusted training settings to unlock its full potential*. Since the attention cost grows almost quadratically with the compressed token length, the model with $1\times$ compaction performs similarly to other configurations, thus we adopt it as the default setting throughout the paper for efficiency.

4. Benchmark Details

4.1. DPG-Bench and GenEval Category Breakdown

Tables 5 and 6 report category-wise results for GenEval and DPG-Bench at 512^2 and 1024^2 resolutions. On GenEval at 512^2, PixelDiT-1.3B outperforms prior pixel-space models. At 1024^2 resolution, PixelDiT-1.3B matches or surpasses several widely used latent diffusion systems despite using fewer parameters, and maintains competitive performance across all individual categories. On DPG-Bench, PixelDiT-1.3B ranks among the top-performing models while maintaining balanced scores across categories. These detailed breakdowns corroborate that PixelDiT delivers strong text–image alignment and compositional reasoning, closing much of the gap to heavily engineered latent diffusion models.

4.2. Editing Background Preservation

A key advantage of pixel-space diffusion models over latent-space counterparts is the ability to preserve fine-grained details in unedited regions during image editing, since pixel-space models avoid the information loss introduced by VAE encoding and decoding. To quantify this advantage, we evaluate background preservation using the FlowEdit [11] dataset, which consists of 281 source–target editing pairs. For each sample, we obtain the editing bounding box using SAM3 and compute MSE and SSIM on the background region outside the bounding box. As shown in Table 8, our pixel-space PixelDiT achieves substantially better background consistency than the latent-space models FLUX and SD3, with $6.0\times$ lower MSE and higher SSIM compared to FLUX. This confirms that pixel-space diffusion models naturally preserve unedited regions more faithfully, which is a practically important property for editing applications.

	Background Consistency	
Method	**MSE↓**	**SSIM↑**
FLUX	0.009105	0.8254
SD3	0.004349	0.8400
PixelDiT	**0.001522**	**0.8628**

Table 8. **Background preservation in image editing.** We evaluate background consistency (MSE and SSIM outside the editing bounding box) on 281 FlowEdit samples. PixelDiT preserves unedited regions significantly better than latent-space models.

5. FLOPs Estimation and Comparison

We estimate GFLOPs for a single forward pass at 256^2 input resolution. For the GFLOPs of prior work, we reuse the numbers reported in their papers [8, 14] and convert them to a **unified convention where one multiply-add counts as two FLOPs**. Table 9 compares the compute cost and FID of PixelDiT-XL with representative latent-space and pixel-space generative models on ImageNet-256. Latent models achieve very strong FIDs with around 240–290 GFLOPs, whereas many pixel models require several hundred to several thousand GFLOPs to close this quality gap. In contrast, PixelDiT-XL obtains a **1.61** FID with only **311** GFLOPs, offering superior image quality over the state-of-the-art pixel generators and closing much of the gap between the best latent models, while using a compute that is close to latent models and substantially less than most prior pixel-space models.

We further provide a detailed GFLOPs breakdown of PixelDiT-XL across different input resolutions and patch sizes in Table 10. The compute cost scales roughly quadratically with resolution at a fixed patch size, reflecting the quadratic cost of self-attention over the patch token sequence. Increasing the patch size dramatically reduces the cost: at 1024^2, moving from $p{=}8$ to $p{=}16$ reduces GFLOPs by $7.4\times$, and $p{=}32$ further reduces it by $3.1\times$. These results provide practical guidance for resolution–compute trade-offs and motivate our default choice of $p{=}16$, which balances quality (see Table 4) and efficiency.

Methods	Params	GFLOPs (multi-add = 2 FLOPs)	FID↓
Latent Generative Models			
DiT-XL/2 [21]	675+49M	238	2.27
SiT-XL/2 [18]	675+49M	238	2.06
REPA, SiT-XL/2 [27]	675+49M	238	1.42
LightningDiT-XL/2 [26]	675+49M	238	1.35
DDT-XL/2 [24]	675+49M	238	1.26
RAE, DiT^{DH}-XL/2 [28]	839+415M	292	1.13
Pixel Generative Models			
ADM-G [4]	559M	2240	7.72
RIN [9]	320M	668	3.95
SiD, UViT/2 [7]	2B	1110	2.44
VDM++, UViT/2 [10]	2B	1110	2.12
SiD2, UViT/2 [8]	N/A	274	1.73
SiD2, UViT/1 [8]	N/A	1306	1.38
PixelFlow-XL/4 [3]	677M	5818	1.98
PixNerd-XL/16 [23]	700M	268	2.15
JiT-G/16 [14]	2B	766	1.82
PixelDiT-XL (ours)	797M	311	1.61

Table 9. **Compute comparison on ImageNet 256×256.** We report model parameters, GFLOPs per forward pass, and FID under our convention that one multiply-add equals two FLOPs.

Patch Size	256^2	512^2	1024^2
8×8	1115.69	6200.99	52634.10
16×16	311.19	1352.24	7147.17
32×32	135.54	547.74	2298.42

Table 10. **GFLOPs of PixelDiT-XL** across input resolutions and patch sizes. Compute scales roughly quadratically with resolution at a fixed patch size.

6. Extended Training Iterations

We extend the training of PixelDiT-XL on ImageNet-256 beyond the 320-epoch setting used in the main paper to an 800-epoch schedule. Table 11 reports gFID, sFID, IS, precision, and recall at 80, 320, and 800 epochs. Performance steadily improves from 80 to 320 epochs: gFID decreases from 2.36 to 1.61 and recall rises from 0.57 to 0.64, while sFID and IS also become better. Extending training further to 800 epochs yields additional but smaller gains, with gFID reaching 1.54 and recall improving to 0.65.

Model	Epochs	gFID↓	sFID↓	IS↑	Prec.↑	Rec.↑
PixelDiT-XL	80	2.36	5.11	282.3	0.80	0.57
PixelDiT-XL	320	1.61	4.68	292.7	0.78	0.64
PixelDiT-XL	800	1.54	4.49	297.0	0.78	0.65

Table 11. Quantitative results of PixelDiT-XL on ImageNet-256 across extended training epochs.

In addition to extended pre-training at 256×256, we evaluate the effect of longer fine-tuning at 512×512 resolution. Starting from the 320-epoch ImageNet-256 checkpoint, we fine-tune PixelDiT-XL for 40 and 530 epochs. Table 12 presents the results. Increasing the fine-tuning duration improves generation quality.

Method	Epochs	Generation@512				
		gFID↓	sFID↓	IS↑	Precision↑	Recall↑
PixelDiT-XL	320 + 40	2.21	5.84	271.1	0.78	0.65
PixelDiT-XL	320 + 530	1.81	5.61	278.6	0.78	0.67

Table 12. **Quantitative comparison** on ImageNet 512×512 with different fine-tuning durations. "320 + N" denotes fine-tuning for N epochs from a 320-epoch ImageNet-256 checkpoint.

7. Empirical Insights and Failed Attempts – (*Things We Tried But Did Not Work*)

During the development of PixelDiT, we explored a large number of architectural variants and training strategies beyond what has been reported in the main paper. Many of these explorations involved not only our final PielDiT design but also alternative pixel-space modeling paradigms. In this section, we share empirical observations and negative results that may be informative for future research on pixel-space diffusion models. We note that the observations below reflect trends across our experimental campaigns; since the experiments were not all conducted under perfectly controlled settings, we report qualitative findings rather than specific numbers.

7.1. Pixel-Level Modeling Paradigms

Factorized Spatial Modulation. As an alternative to our pixel-wise AdaLN, we explored *factorized AdaLN*, where instead of producing distinct modulation parameters for every pixel (P^2 sets of parameters per patch), we generate modulation coefficients in a low-frequency DCT basis. Concretely, the conditioning vector produces K spectral coefficients per modulation channel, and the per-pixel parameters are reconstructed via $\mathbf{m}(i) = \sum_k c_k \cdot \phi_k(i)$, where ϕ_k is a 2D separable cosine basis function. This reduces the parameter count from $O(P^2 \cdot C)$ to $O(K \cdot C)$ where $K \ll P^2$. In our experiments, this factorization did not match the performance of the full pixel-wise AdaLN. This is perhaps not surprising given that the low-frequency basis inherently limits the spatial resolution of the modulation signal.

Haar Wavelet Pretransform. We investigated applying a reversible Haar wavelet decomposition as a preprocessing step, transforming each patch from spatial domain into frequency subbands before the diffusion process. The forward transform uses pixel unshuffle followed by the 4×4 Haar matrix, decomposing the image into low-frequency and three high-frequency bands. We also ex-

plored band-specific loss weighting, downweighting high-frequency components. This approach did not yield improvements over direct pixel-space modeling in our setting.

7.2. Conditioning and Feature Interaction

Conditioning Mechanisms for Pixel-Level Modulation. We extensively investigated how patch-level semantic context should modulate the pixel-level transformer blocks. Beyond the pixel-wise AdaLN presented in the paper, we experimented with: (i) *additive conditioning*, where patch context is projected and directly added to the compressed pixel tokens before cross-patch attention; (ii) *cross-attention conditioning*, where pixel tokens attend to patch context tokens with separate query/key/value projections; (iii) *self-attention concatenation* (concatenating patch context tokens with pixel tokens along the sequence dimension and processing them jointly in a single self-attention layer); and (iv) *per-pixel additive conditioning*, where patch context is projected to the full pixel spatial resolution and added before token compaction. Among these alternatives, both cross-attention and self-attention concatenation led to worse generation quality compared to AdaLN. Additive conditioning does not noticeably harm performance. While it is slightly worse, it removes the P^2 parameters introduced by pixel-wise AdaLN, making it more scalable to increase the number of PiT blocks.

In-Context Prefix Tokens. Inspired by prefix tuning in language models, we experimented with prepending learnable context tokens to the patch sequence at a configurable depth. These tokens, initialized from the class embedding with added learnable positional embeddings, participate in self-attention alongside patch tokens and are stripped before the pixel pathway. This did not lead to measurable improvements in the generation quality on ImageNet class-conditional generation. This mechanism may be more relevant in settings with more complex conditioning (e.g., text-to-image).

7.3. Training Stability and Optimization

Representation Alignment Loss is Essential. We experimented with removing the feature alignment loss that encourages the patch-level representations to be aligned with the frozen DINOv2 encoder features. Without this auxiliary objective, the training became unstable and eventually diverged.

Prediction Target. In addition to velocity prediction, we experimented with $\mathbf{x}$-prediction, where the network directly predicts the clean image $\hat{\mathbf{x}}$ and the loss is computed in velocity space via $\hat{\mathbf{v}} = (\hat{\mathbf{x}} - \mathbf{x}_t)/\max(\sigma(t), \epsilon)$ with $\epsilon = 0.05$ for numerical stability. In practice, we found that $\mathbf{x}$-prediction can effectively mitigate loss spikes during training. However, it did not outperform direct velocity prediction in our setting and required additional tuning of both the ϵ threshold and the logit-normal timestep sampling hyperparameters (p_{mean}, p_{std}), since the optimal timestep distribution differs between velocity and $\mathbf{x}$-prediction parameterizations.

7.4. Architectural Design

Encoder–Decoder Bottleneck for Token Compression. We explored adding a hierarchical encoder–decoder (ED) bottleneck around the patch-level transformer. The encoder progressively merges tokens via 2×2 patch merging [5] stages, processes them with transformer blocks at reduced resolution, and the decoder mirrors this with patch expanding [5] and skip connections. While the ED bottleneck architecture reduced computational cost, the optimal skip connection pattern (we tested many configurations, e.g., varying which encoder stages connect to which decoder stages) was highly sensitive to the overall architecture. The token compaction mechanism in PiT blocks ultimately proved to be a simpler and more robust alternative.

Multi-Scale Pixel Hidden Dimensions. In the pixel pathway, we explored varying the per-pixel hidden dimension across a wide range (from 4 to 128). Smaller dimensions were insufficient for good performance, while larger dimensions increased memory consumption in the compaction and expansion projections (which scale as $P^2 \times d_{\text{pixel}} \times d_{\text{attn}}$) without proportional quality gains. We found that a relatively compact (e.g., 16) pixel representation was sufficient when paired with adequate attention dimension in the cross-patch attention.

Shared vs. Separate Modulation. We tested sharing the AdaLN modulation network across all conditioning blocks, which reduces parameter count. This did not degrade quality substantially for shallower models but became limiting for deeper architectures where different layers may benefit from distinct modulation patterns.

Learnable vs. Fixed Positional Embeddings. We compared fixed sinusoidal 2D positional embeddings with fully learnable positional embeddings for the pixel-level tokens. Both approaches yielded comparable generation quality. Given the simplicity and resolution-generalizability of sinusoidal embeddings, we opted for fixed embeddings in our final model.

8. Qualitative Examples

We include additional qualitative results for ImageNet-256 and ImageNet-512 single-class-conditioned image generations, as shown in Figures 11–22, together with high-

resolution (approximately 1024^2) text-to-image generation results in Figures 7–10, illustrating the visual quality, diversity, and prompt alignment achieved by PixelDiT.

9. Limitations

Due to the limited model capacity and insufficient high-quality training data, our 1.3B-parameter PixelDiT text-to-image model sometimes struggles to generate objects that are both geometrically and texturally complex, such as human hands and intricate architectural scenes. Additionally, we observe that training pixel-space diffusion models with velocity prediction is prone to loss spikes, particularly for deeper architectures and during long training runs. Although we have identified several stabilization techniques (see Section 7), fully eliminating loss spikes without sacrificing training efficiency remains an open challenge. In future work, we plan to address these limitations by scaling up the model capacity, curating larger and higher-quality training data, and conducting more foundational research into the training dynamics of pixel-space diffusion to better understand and mitigate loss instabilities.

References

[1] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. In *ECCV*, 2024. 5

[2] Junsong Chen, YU Jincheng, GE Chongjian, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *ICLR*, 2024. 5

[3] Shoufa Chen, Chongjian Ge, Shilong Zhang, Peize Sun, and Ping Luo. Pixelflow: Pixel-space generative models with flow. *arXiv preprint arXiv:2504.07963*, 2025. 5, 7

[4] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. In *NeurIPS*, 2021. 7

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 8

[6] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024. 1

[7] Emiel Hoogeboom, Jonathan Heek, and Tim Salimans. Simple diffusion: End-to-end diffusion for high resolution images. In *ICML*, 2023. 7

[8] 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. 6, 7

[9] Allan Jabri, David Fleet, and Ting Chen. Scalable adaptive computation for iterative generation. In *ICML*, 2023. 7

[10] Diederik Kingma and Ruiqi Gao. Understanding diffusion objectives as the elbo with simple data augmentation. *NeurIPS*, 36, 2024. 7

[11] Vladimir Kulikov, Matan Kleiner, Inbar Huberman-Spiegelglas, and Tomer Michaeli. Flowedit: Inversion-free text-based editing using pre-trained flow models. In *ICCV*, pages 19721–19730, 2025. 6

[12] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. 5

[13] 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. *arXiv preprint arXiv:2402.17245*, 2024. 5

[14] Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise, 2025. 6, 7

[15] Zhimin Li, Jianwei Zhang, Qin Lin, Jiangfeng Xiong, Yanxin Long, Xinchi Deng, Yingfang Zhang, Xingchao Liu, Minbin Huang, Zedong Xiao, et al. Hunyuan-dit: A powerful multi-resolution diffusion transformer with fine-grained chinese understanding. *arXiv preprint arXiv:2405.08748*, 2024. 5

[16] I Loshchilov. Decoupled weight decay regularization. In *ICLR*, 2019. 1

[17] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*, 2022. 1

[18] 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 *ECCV*, 2024. 7

[19] OpenAI. Dalle-3, 2023. 5

[20] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. In *TMLR*, 2023. 1

[21] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 7

[22] 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. *arXiv preprint arXiv:2307.01952*, 2023. 5

[23] Shuai Wang, Ziteng Gao, Chenhui Zhu, Weilin Huang, and Limin Wang. Pixnerd: Pixel neural field diffusion. *arXiv preprint arXiv:2507.23268*, 2025. 5, 7

[24] Shuai Wang, Zhi Tian, Weilin Huang, and Limin Wang. Ddt: Decoupled diffusion transformer, 2025. 7

[25] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, et al. Sana: Efficient high-resolution text-to-image synthesis with linear diffusion transformers. In *ICLR*, 2025. 1

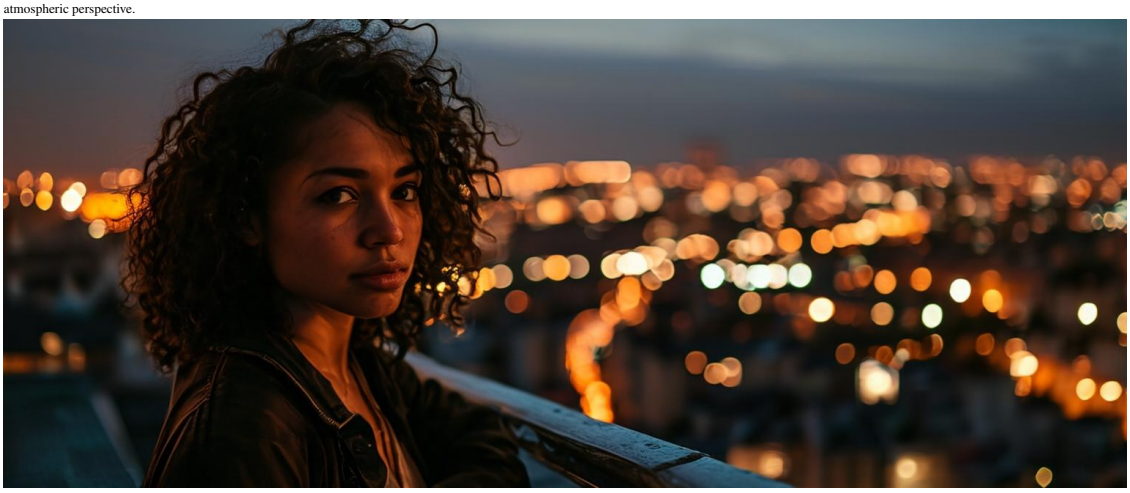
[26] Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *CVPR*, 2025. 7

[27] 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 *ICLR*, 2025. 1, 7

[28] Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. *arXiv preprint arXiv:2510.11690*, 2025. 7

[29] Le Zhuo, Ruoyi Du, Han Xiao, Yangguang Li, Dongyang Liu, Rongjie Huang, Wenze Liu, Lirui Zhao, Fu-Yun Wang, Zhanyu Ma, et al. Lumina-next: Making lumina-t2x stronger and faster with next-dit. *arXiv preprint arXiv:2406.18583*, 2024. 5

A group of figures are gathered at a table near a trellised terrace, overlooking a river with rowers and a boat. The scene, rendered in an Impressionistic style, employs loose brushstrokes and soft, diffused light. The man in the foreground, dressed in a blue and white striped shirt, gestures casually, holding what appears to be a cigarette. The table is set with wine bottles and glasses, indicating a leisurely gathering. Through the trellis, a glimpse of the river reveals rowers in action, with a lone figure in a boat further out. The greenery of the trellis and the river landscape blend, contributing to the painting's overall sense of depth and atmospheric perspective.



A woman stands on a rooftop at dusk, overlooking a cityscape illuminated by twinkling lights. Her curly hair frames a serious expression, and she leans casually against the rooftop railing. The soft lighting of the sunset blends with the artificial glow of the city, creating a warm yet muted atmosphere. The out-of-focus background emphasizes the vastness of the urban landscape, dotted with skyscrapers and distant roads. Her dark jacket adds a layer of contrast, focusing the viewer's attention on her face. The overall style evokes a sense of urban solitude and reflection against a backdrop of a vibrant cityscape.

A golden-hued lioness rests serenely in a sunlit grassy field. The lioness, bathed in the warm glow of the setting or rising sun, is positioned in the foreground, lying comfortably on a small mound of earth covered with dry grass. Her paws are outstretched, relaxed and slightly crossed. The background is a soft, blurred mix of green and golden foliage, hinting at a savanna-like landscape. The golden light emphasizes the texture of her fur and highlights the contours of her face, adding a sense of calm and natural grandeur to the scene.

Figure 7. Additional Text-to-Image Generation Results. The caption below each image corresponds to the text prompt used to generate it.

Photo of a person moving with motion blur, shot with a Leica M6 and VISION3 500T Color Negative Film, reminiscent of a Wong Kar Tai film set.

A young man wearing 18th century noble clothing in blues and pinks and standing in front of the green grass with white flowers.

Portrait shot of a pretty woman, latex suit fashion, contrasting background, fashion magazine cover, 35mm kodachrome.

A portrait of a human growing colorful flowers from her hair. Hyperrealistic oilpainting.

knitted cat-whale plush toy on rug in warm sunlit living room, cozy decor.

Behold the Joymonger, photorealistic, 1990s, hyper realism, extremely detailed.

Figure 8. Additional Text-to-Image Generation Results. The caption below each image corresponds to the text prompt used to generate it.

A young woman with striking eyes gazes directly at the viewer, set against a soft, blurred background. Her long, auburn hair falls loosely around her shoulders, partially obscuring one side of her face, while a vertically striped, collared shirt completes her casual yet elegant look. The lighting is warm and natural, emphasizing the subtle contours of her face and the slight flush in her cheeks. The photograph captures a blend of relaxed confidence and introspective beauty, rendered with a soft focus that lends a dreamlike quality to the image. The surrounding environment is intentionally muted, ensuring that the woman remains the primary focal point. The color palette leans towards earthy tones, enhancing the overall warmth and approachability of the portrait.

Close-up portrait of a beautiful Baltic model wearing white flower-shaped earrings, emphasis on the earrings, everything is in full focus, pores and skin imperfections are visible, neutral lighting from a large studio softbox, natural beauty, professional studio shooting.

funny Candid photo, cat sleeping across aman's eyes as he sleeps on his back,blocking his face, man is 25 years old,neck length frizzy dark black-brown hair,very unruly hair, stubble, wearing blackdress pants, long sleeve white button upshirt, socks, laying on a four post bed, aesthetic.

Ultra-realistic photo of an anthropomorphic chili pepper with a glossy red surface, smiling with human teeth and wearing black sunglasses. Surrounded by realistic flames, the pepper is sharply in focus, while the fiery background is slightly blurred.

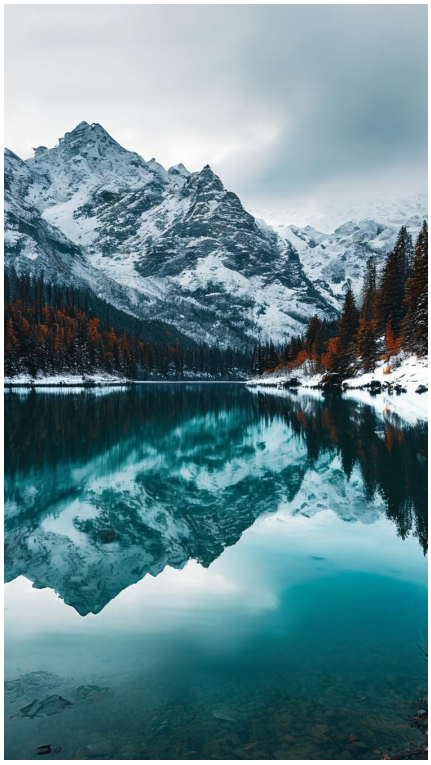
Tiny fluffy lamb standing on a fingertip, ultra-detailed wool texture, soft natural light.

Colorful woolen parrot plush wearing ornate psychic-style vest, detailed fabric textures.

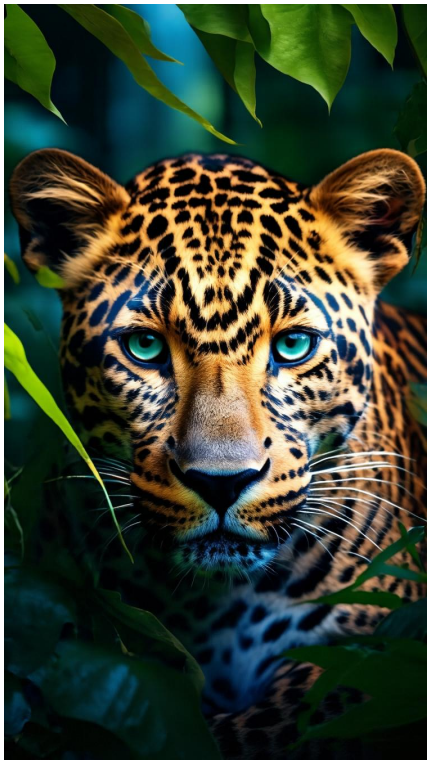
young woman in warm sunset light, soft shadows, long blonde hair, pink pajamas, calm expression.

Figure 9. Additional Text-to-Image Generation Results. The caption below each image corresponds to the text prompt used to generate it.

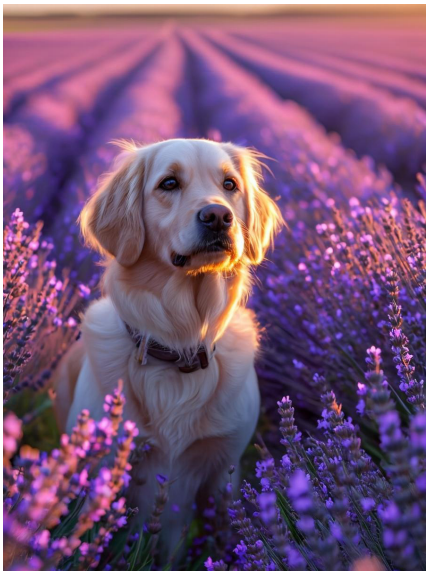
An image of a very tired old man, fisherman, long beard, black background settings. The facial expression reflects wisdoms and test of time.



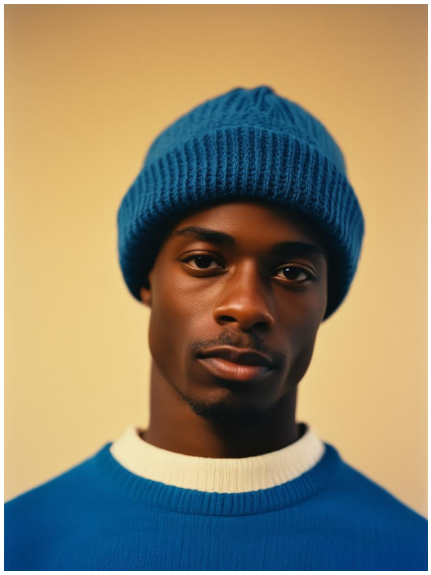
snow-covered mountains rising above a calm turquoise lake, their peaks perfectly mirrored in the water, framed by dense autumn-tinged pines.



A leopard hiding in the jungle, a photo-realistic portrait, captured with a Canon EOS R5 camera and a macro lens for detailed wildlife portraits.



A serene moment unfolds as a dog leisurely navigates a tranquil lavender field at sunset. The golden light enhances the beauty of the swaying purple blooms.



Close-up shot of an African American man wearing a blue beanie, against a beige background, with a vintage aesthetic, in the style of Kodak film photography.

Two conjoined strawberry. The strawberrys are resting on undulating red-pink holographic icy slush. Vaporwave. Intense color. Ice textures. Solar aesthetic.

Figure 10. Additional Text-to-Image Generation Results. The caption below each image corresponds to the text prompt used to generate it.

Figure 11. **Uncurated 512 × 512 PixelDiT-XL samples.** Class 8. CFG scale = 4.0.

Figure 13. **Uncurated 512 × 512 PixelDiT-XL samples.** Class 22. CFG scale = 4.0.

Figure 12. **Uncurated 512 × 512 PixelDiT-XL samples.** Class 9. CFG scale = 4.0.

Figure 14. **Uncurated 512 × 512 PixelDiT-XL samples.** Class 949. CFG scale = 4.0.

Figure 15. **Uncurated 512 × 512 PixelDiT-XL samples.** Class 173. CFG scale = 4.0.

Figure 17. **Uncurated 512 × 512 PixelDiT-XL samples.** Class 285. CFG scale = 4.0.

Figure 16. **Uncurated 512 × 512 PixelDiT-XL samples.** Class 933. CFG scale = 4.0.

Figure 18. **Uncurated 512 × 512 PixelDiT-XL samples.** Class 850. CFG scale = 4.0.

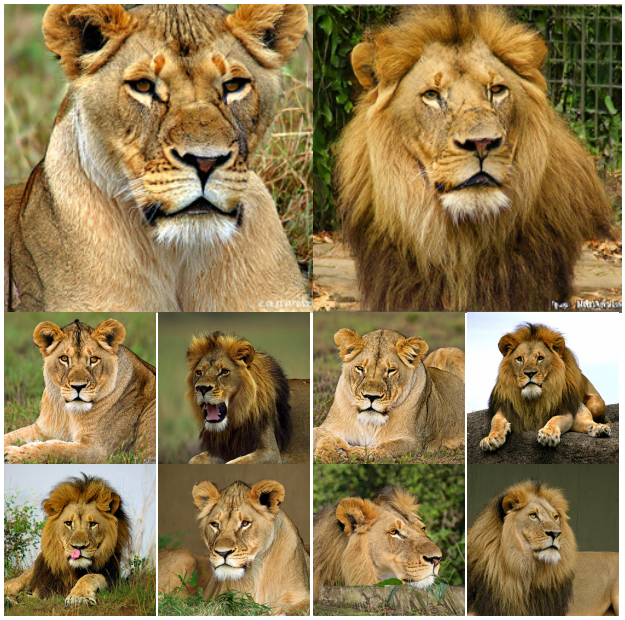


Figure 19. **Uncurated 256 × 256 PixelDiT-XL samples.** Class 291. CFG scale = 4.0.

Figure 21. **Uncurated 256 × 256 PixelDiT-XL samples.** Class 24. CFG scale = 4.0.

Figure 20. **Uncurated 256 × 256 PixelDiT-XL samples.** Class 259. CFG scale = 4.0.

Figure 22. **Uncurated 256 × 256 PixelDiT-XL samples.** Class 970. CFG scale = 4.0.