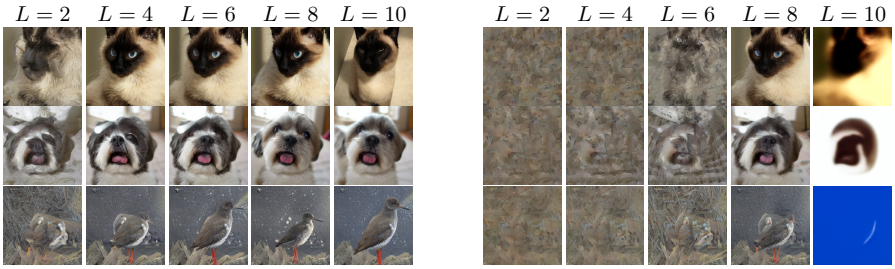
ELT: Elastic Looped Transformers for Visual Generation

Sahil Goyal*, Swayam Agrawal*, Gautham Govind Anil, Prateek Jain, Sujoy Paul†, and Aditya Kusupati†

Google DeepMind



(a) **ELT**'s **Any-Time inference** from **single training run**

(b) Vanilla looped transformers

Fig. 1: Class-conditional Image Generation on ImageNet 256 × 256. Comparing our proposed Elastic Looped Transformers (left) against vanilla looped transformers (right), which simply repeat transformer layers during a forward pass of diffusion. As shown, standard looping only produces a coherent image when the inference loop count exactly matches its training setup ($L = 8$), with quality severely degrading at all other values. In contrast, our approach maintains high-fidelity generation across several evaluated compute budgets.

Abstract. We introduce Elastic Looped Transformers (ELT), a highly parameter-efficient class of visual generative models based on a recurrent transformer architecture. While conventional generative models rely on deep stacks of unique transformer layers, our approach employs iterative, weight-shared transformer blocks to drastically reduce parameter counts while maintaining high synthesis quality. To effectively train these models for image and video generation, we propose the idea of Intra-Loop Self Distillation (ILSD), where student configurations (intermediate loops) are distilled from the teacher configuration (maximum training loops) to ensure consistency across the model's depth in a single training step. Our framework yields a family of elastic models from a single training run, enabling Any-Time inference capability with dynamic trade-offs between computational cost and generation quality, with the same parameter count. ELT significantly shifts the efficiency frontier for visual synthesis. With *4*× reduction in parameter count under iso-inference-compute settings, ELT achieves a competitive FID of *2.0* on class-conditional ImageNet 256 × 256 and FVD of *72.8* on class-conditional UCF-101.

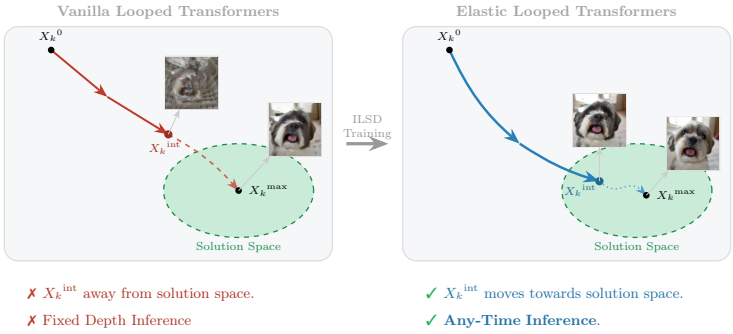


Fig. 2: Latent Trajectories of Standard vs. Elastic Looped Transformers. ${X_k}^{\text{int}}$ & ${X_k}^{\text{max}}$ represent output of intermediate (L_{int}) & final loops (L_{max}) respectively for k^{th} generation sampling step. Unlike standard recurrent models (left) where only the final iteration ${X_k}^{\text{max}}$ reaches the solution space, our ILSD training (right) guides intermediate states ${X_k}^{\text{int}}$ also toward the target space. This transformation shifts the model from a fixed-depth architecture to an Any-Time inference framework, supporting flexible computational budgets through early exits within a sampling step.

1 Introduction

Traditional techniques to increase compute capacity in deep learning models, such as stacking deeper layers or increasing network width, inevitably lead to a proportionally larger memory footprint. Recurrence offers a powerful alternative paradigm, enabling models to utilize large amounts of compute while maintaining a minimal memory footprint by leveraging the same set of parameters repeatedly. This architectural efficiency parallels the biological visual systems [35, 36], where recurrent processing, rather than strictly feedforward pathways, is essential for resolving complex visual inputs.

While looping of transformers was popularized by Universal Transformers [9] and has recently empowered language models with stronger reasoning capabilities [57, 71], its potential for high-fidelity visual generation remains largely untapped. From a practical standpoint, compared to the traditional models, Looped Transformers (a) are extremely parameter efficient and can perform significantly more compute (FLOPs) per parameter, (b) can have higher throughput by minimizing the "memory wall" bottleneck. They use a compact set of shared parameters and maintain its major parameter footprint on-chip or adjacent to the chip. This avoids the cost of repeated transfers between different units of the accelerator (GPUs/TPUs) required in large transformers, and (c) can exhibit robustness against overfitting in data-constrained settings.

Despite the parameter efficiency of looped architectures, their training remains challenging because intermediate representations often remain uninterpretable until the final loop (see Figure 2). We address this by introducing Elastic Looped Transformers (ELT), a class of generative models designed for progressive refinement. Unlike traditional recurrent transformers, ELT is optimized to

provide meaningful, high-quality synthesis even at intermediate repeats. This enables Any-Time (elastic) inference capability - where a single model can scale its compute based on available resources without sacrificing much on generation quality. We present a pictorial representation of our proposed method in Figure 3 and generation results for Any-Time inference capability in Figure 1.

To achieve this functional flexibility across loops, we propose an Intra-Loop Self Distillation (ILSD) algorithm for training looped transformers. Rather than treating the loop as a fixed-depth process, our framework operates as a dual-path system: a teacher path executes the full loop count to provide a high-fidelity target, while a student path, defined strictly as a subset of the teacher's trajectory, learns to produce comparable results with fewer iterations. Note that both the paths have the same parameter count. By framing the student process as a literal subset of the teacher's forward propagation, we ensure there is no additional overhead during training. This approach forces the shared parameters to compress complex transformations into earlier loops. Consequently, the model does not just fix the output at the end. It learns an efficient, progressive refinement process that maintains the generation quality regardless of the exit point as motivated in Figure 2. Our contributions are summarized as follows:

State-of-the-Art Parameter Efficiency: Through the reuse of block of transformer layers across loops, ELT achieves a competitive FID of *2.0* on class-conditional ImageNet 256×256 and FVD of *72.8* on class-conditional UCF-101. This represents a *4* $\times$ reduction in parameters compared to baselines MaskGIT [7] (image generation) and MAGVIT [72] (video generation), while matching or improving their performance under iso-inference-compute settings.

Elastic/Any-Time Inference: By treating the looped block as an iterative refiner, our models enable Any-Time inference [79], allowing to traverse the pareto frontier of quality versus compute at test-time without retraining. This allows to serve diverse deployment tiers: from latency-critical on-device generation (few loops) to high-fidelity cloud rendering (more loops).

Scalability: While model size remains a primary driver of quality, recursive looping provides a unique test-time compute lever that scales predictably across both Masked Generative Transformers [7, 72, 73] and Diffusion Transformers [49].

2 Related Work

Recursive Architectures: The principle of recursively applying a shared block of parameters to enhance model efficiency is a well-established concept. This approach, often referred to as *looping* has been shaped by Universal Transformers [9], which introduced the idea of iterating over a single transformer layer. Notably, there has been a resurgence of interest in this area. Saunshi et al. [57] demonstrated the power of looping for sophisticated reasoning tasks. Mixture-of-Recursions [3] further explored input-dependent dynamic and adaptive depth

in looped transformers. Mixture-of-Recursions-VIT [42] extends it for image understanding. Geiping et al. [19] demonstrated that scaling test-time compute via recurrent depth allows language models to perform complex latent reasoning.

Deep Equilibrium Models (DEQs) [1, 4, 16, 20, 43, 50], instead of unrolling a weight-tied layer for a fixed number of iterations, define the output as the fixed point of a non-linear transformation. Unlike DEQs that rely on black-box solvers for an analytical fixed point, our ELT framework explicitly optimizes unrolled intermediate states via Intra-Loop Self Distillation (ILSD), retaining the flexibility of Any-Time inference without requiring the network to reach a strict analytical fixed point.

Parameter-Efficient Visual Generation: Standard efficiency techniques [44] that work across deep learning models also help in speeding up visual generation models. MobileDiffusion [76] prunes redundant residual blocks and replaces standard layers with separable convolutions in UNet to get an optimized architecture ($\sim$400M) for on-device visual generation. EdgeFusion [6] employs BK-SDM, a lightweight Stable Diffusion variant, and refines the step distillation process of Latent Consistency Model. MaGNeTS [22] trains a family of nested transformers [38, 39] with schedules of model sizes over the generation process, without increasing the parameter count.

Elastic Visual Generation: The paradigm of Any-Time or elastic generation focuses on decoupling model's parameter count from its computational depth. E-DiT [66] introduces adaptive block skipping and MLP width reduction, allowing a single model to traverse varying computational budgets without retraining. In the context of visual reasoning, LoopViT [59] uses a weight-tied recursive architecture, employing a parameter-free dynamic exit mechanism to halt inference based on uncertainty of prediction. EvoSearch [24] proposes a search-based strategy that optimizes sampling trajectories at inference time. Unlike these methods that rely on architectural skipping or external search, our ELT framework equipped with Intra-Loop Self Distillation (ILSD) algorithm directly regularizes the recursive process, ensuring stability across the entire loop spectrum. Other methods achieve elasticity via adaptive token sequence lengths [2, 14, 45, 58, 70].

Relation to Few-Step and Consistency Models: Recent approaches such as Consistency Models [62] and progressive distillation [55] address inference efficiency by reducing the number of sampling steps (inter-step acceleration). ELT is fundamentally orthogonal: it reduces compute *within* each sampling step by varying the loop count L (intra-step acceleration). These two axes are complementary, we can combine ELT with few-step methods to achieve further efficiency gains. Moreover, unlike consistency models which require specialized training objectives and architectural constraints to enable variable-step inference, ELT's elastic capability emerges naturally from ILSD applied to standard training objectives. We note that ELT is particularly compelling for one-step generative paradigms, where the loop count L becomes the *sole* lever for controlling the compute-quality trade-off at inference time.

3 Preliminaries

Masked Generative Transformers: MaskGIT [7] introduced iterative parallel decoding for image generation using discrete tokens [72]. Unlike autoregressive models, MaskGIT generates and refines all tokens simultaneously over K sampling steps:

$$\mathbf{X}_k \leftarrow \text{Mask} \circ \text{Sample}(M(\mathbf{X}_{k-1}, c), k) \tag{1}$$

where M is a fixed-depth transformer predicting tokens conditioned on class c. MAGVIT [72] extends this framework to video generation.

Diffusion Transformers: Diffusion models [26, 63] generate data by reversing a noising process. Diffusion Transformers (DiTs) [49] shift away from traditionally used U-Nets [53] by treating image latents as sequences of tokens, and using transformer blocks for processing these tokens. Both paradigms share a common structure: recursive refinement over multiple sampling steps through a model M. ELT naturally aligns with this progressive refinement by implementing M as a recurrent, weight-shared transformer block. This introduces recursive computation *within* each sampling step, providing a test-time compute lever to dynamically trade-off inference speed and generation quality.

4 Method

Looping Mechanism: Let the number of transformer layers to be looped be N and number of loops per sampling step be L. The total effective depth for a single sampling or denoising step of the generation process is then $N \times L$. Let $f_{\theta_i}(\mathbf{x})$ denote a single transformer layer with parameters θ_i. We define a composite block $g_{\Theta}(x)$ consisting of N unique transformer layers with parameters $\Theta = \{\theta_1, \theta_2, \ldots, \theta_N\}$ as follows:

$$g_{\Theta}(\mathbf{x}) = f_{\theta_N}(f_{\theta_{N-1}}(\cdots f_{\theta_1}(\mathbf{x})))$$

In a standard transformer model with total depth $\mathcal{D} = N \times L$, the effective transformation $F_{\mathcal{D}}(x)$ would require $\mathcal{D}$ sets of unique parameters. In *looping*, we reuse the same block of parameters Θ for L successive applications, resulting in only N unique layers of parameters. The effective transformation for a $N \times L$ configuration is given by:

$$F_{(N,L)}(\mathbf{x}) = \underbrace{g_{\Theta}\left(g_{\Theta}(\cdots g_{\Theta}(\mathbf{x}))\right)}_{L \text{ loops}} \equiv g_{\Theta}^{L}(\mathbf{x})$$

This looping architecture decouples physical model size from computational depth (see Figure 3 for a visual overview). The parameter count (Θ) is constrained by the number of unique blocks (N), while representational capacity and depth ($\mathcal{D}$) scale with loop count (L). This setup provides two primary advantages: extreme parameter efficiency and high throughput (refer Section 5.2).

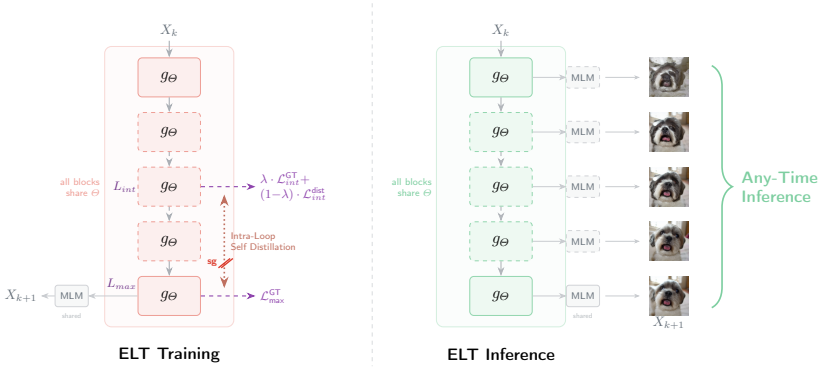


Fig. 3: Overview of ELT framework. Training (left) utilizes shared parameters Θ with recurrent loops and Intra-Loop Self Distillation (ILSD) to improve intermediate representations. This enables Any-Time inference (right), allowing the model to exit early and predict X_{k+1} from any intermediate block via a shared MLM head. Note that X_k represents the input for k^{th} sampling iteration and output images in ELT Any-Time Inference (right) assume $k = K$ (last sampling step). Refer Section 3 for details of annotations.

Intra-Loop Self Distillation (ILSD): In a standard weight-tied (looped) transformer, the model is typically optimized only for its final output after fixed $L_{\max}$ iterations i.e. the loop count for which it is trained. However, this creates a "black box" internal trajectory where intermediate loops ($L < L_{\max}$) may produce suboptimal representations until the final projection layer. By treating the looped block g_Θ as an iterative refiner, we can motivate a training objective that ensures the model remains useful at multiple depths. This not only encourages the model to learn a more robust, incremental transformation but also allows for elastic inference, where the model can be exited early with minimal performance drop. Towards this end, we propose Intra-Loop Self Distillation (ILSD).

From a distillation perspective, ILSD leverages the fact that a model with more loops ($L_{\max}$) naturally possesses a more mature and refined representational space than its shallower version (L_{int}), even though they have the same unique parameters. By treating the full-depth model as an internal teacher, we provide a high-fidelity, low-variance signal for the shallower student to follow (see Figure 3). This forces the shared parameters Θ to compress complex transformations into fewer steps, effectively distilling the knowledge of the deep model into the early stages of the computation.

Stochastic Student Sampling (S^3): We train with a fixed number of loops, $L_{\max}$, for a block of N layers with unique parameters. During each training step, we treat the model as a dual-path system sharing a single set of parameters Θ. We define a teacher path that executes the full $L_{\max}$ loops and a stochastic student path that exits early at an intermediate loop L_{int}. The intermediate path receives supervision from the ground truth labels as well as the online teacher with maximum training loop count $L_{\max}$.

We denote the training loss for the output of a configuration $N \times L$ as $\mathcal{L}_\Theta(F_{(N,L)}(\mathbf{x}), \mathbf{y})$ where $\mathbf{y}$ denote the ground-truth for a certain input masked / denoise level. At every training iteration, we randomly sample an intermediate loop count L_{int} from uniform distribution such that $L_{\min} \leq L_{\text{int}} < L_{\max}$. Note that $L_{\min}$ is just used for constraining the student sampling distribution. The effective joint loss, $\mathcal{L}_\Theta^{\text{ILSD}}$, is computed as:

$$
\begin{aligned}
\mathcal{L}_\Theta^{\text{ILSD}} =& \mathcal{L}^{\text{GT}}(F_{(N,L_{\max})}(\mathbf{x}), \mathbf{y}) && \text{(1) Ground-truth for teacher} \\
&+\lambda \mathcal{L}^{\text{GT}}(F_{(N,L_{\text{int}})}(\mathbf{x}), \mathbf{y}) && \text{(2) Ground-truth for student} \\
&+(1-\lambda)\mathcal{L}^{\text{dist}}(F_{(N,L_{\text{int}})}(\mathbf{x}), \text{sg}(F_{(N,L_{\max})}(\mathbf{x}))) && \text{(3) Intra-Loop Self Distillation} \\
\text{with} \quad & L_{\text{int}} \sim \mathcal{U}(L_{\min}, L_{\max}) && \text{Stochastic Student Sampling}
\end{aligned}
$$

where sg is stop-grad for teacher ($L_{\max}$) in ILSD, λ is hyperparameter controlling the weight between the ground truth and distillation losses. We introduce a curriculum for λ and linearly decay it from 1 to 0 as training progresses. This initially anchors the student to reliable ground-truth labels while the teacher is still untrained and gradually shift to mimicking the teacher's predictions once they have matured.

Loss Formulation: The exact forms of $\mathcal{L}^{\text{GT}}$ and $\mathcal{L}^{\text{dist}}$ depend on the algorithm used for training. For masked generative models with discrete tokens, we use the cross-entropy loss for both ground-truth and distillation:

$$
\begin{aligned}
\mathcal{L}^{\text{GT}} &= -\sum_{i \in Mask} \log P_{(N,L_{\text{int}})}(y_i \mid \mathbf{x}_{mask}) \\
\mathcal{L}^{\text{dist}} &= -\sum_{i \in Mask} \sum_{v \in \mathcal{V}} P_{(N,L_{\max})}(v \mid \mathbf{x}_{mask}) \log P_{(N,L_{\text{int}})}(v \mid \mathbf{x}_{mask})
\end{aligned}
$$

where $\mathbf{x}_{mask}$ is the masked input, $y = \{y_i\}_{i \in \text{Mask}}$ represents the ground-truth tokens for the masked positions, and $\mathcal{V}$ is the full vocabulary of the tokenizer. For diffusion, we use the sigmoid-weighted Mean Squared Error (MSE) for both ground-truth and distillation losses. Let $\mathbf{x}_t$ be the noised version of ground truth latent $\mathbf{x}_0$, the ground-truth loss is:

$$
\mathcal{L}^{\text{GT}} = w(t) \left\| F_{(N,L)}(\mathbf{x}_t) - \mathbf{x}_0 \right\|_2^2
$$

where L is L_{int} for student and $L_{\max}$ for teacher. $w(t)$ is a time-dependent sigmoid weighting term [32]. The distillation loss is:

$$
\mathcal{L}^{\text{dist}} = w(t) \left\| F_{(N,L_{\max})}(\mathbf{x}_t) - F_{(N,L_{\text{int}})}(\mathbf{x}_t) \right\|_2^2
$$

Note that gradients from both computational paths, corresponding to L_{int} and $L_{\max}$, update the single, shared set of block parameters Θ. This joint optimization provides a significantly richer training signal; the shared block g_Θ is forced to learn a transformation that is not only effective at L_{int} loops but

remains incrementally useful for the subsequent iterations up to $L_{\max}$ loops. This constraint prevents the model from learning a *shortcut* that might minimize loss at a specific depth but fail when composed further. Consequently, the shared block generalizes better to lower depths, leading to higher performance even with fewer loops. It is interesting to note that unlike traditional distillation, where we have to forward propagate through both the student and teacher models separately, in the proposed way of Intra-Loop Self Distillation (ILSD), the training overhead is minimal, as the computation of $F_{(N,L_{\text{int}})}(\mathbf{x})$ is a strictly required intermediate step for computing $F_{(N,L_{\max})}(\mathbf{x})$ i.e. the student trajectory (L_{int}) is a strict prefix of the teacher trajectory ($L_{\max}$).

5 Experiments and Results

We evaluate ELT on class-conditional image generation using both masked generative and diffusion transformers, and class-conditional video generation using masked generative transformers.

5.1 Experimental Setup

Datasets: We experiment on ImageNet 256×256 [10] for image generation and UCF-101 [64] for class-conditional video generation.

Implementation Details: **(i) Masked Generative Transformers**: We use pretrained tokenizers from MaskGIT [7] (images) and MAGVIT [72] (videos) with a codebook size of 1024 tokens. Image models are trained at 256×256 resolution, compressed to 16×16 discrete tokens with an embedding dimension of 1024. Video models are trained for $16 \times 128 \times 128$ sequences, compressed to $4 \times 16 \times 16$ tokens. Following MaskGIT, we adopt the BERT [12] architecture as the transformer backbone and perform experiments at several model scales to understand the scaling behaviors of ELT. We train all models for 270 epochs unless otherwise specified. We employ a cosine schedule for unmasking tokens during inference. For image generation, we use classifier-free guidance by dropping class condition labels for 10% of the training batches. **(ii) Diffusion Transformers:** We use a pretrained Stable Diffusion v1.4 VAE [52] model to map 256×256 images into a continuous $32 \times 32 \times 4$ latent space ($8\times$ spatial downsampling). We train a DDPM-style diffusion model which operates on these latents using a DiT architecture. We employ a shifted cosine noise schedule and sigmoid-weighted MSE loss for training [32]. Models are trained for 500K steps with a batch size of 512 using Adam. Unless mentioned otherwise, sampling uses 512-step DDPM with classifier-free guidance scale 3.0.

Evaluation Metrics and Efficiency: We evaluate synthesis quality using Fréchet Inception Distance (FID) [13, 25] and Inception Score (IS) [54] for images, and Fréchet Video Distance (FVD) [65] for video. Model efficiency is measured via inference-time GFLOPs and throughput (samples/sec). In our $N \times L$ design space, computational complexity and latency scale linearly with the loop count L, allowing to navigate the quality-speed trade-off at inference time.

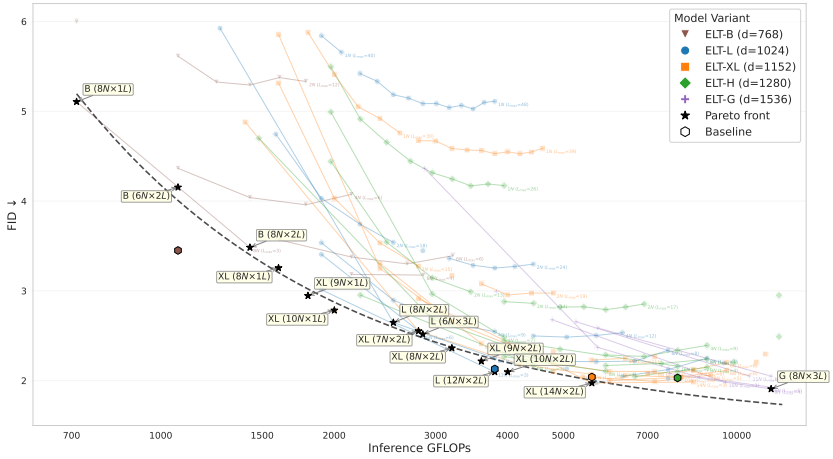


Fig. 4: Pareto front of FID vs. Inference GFLOPs. The black curve denotes the fit (FID $= 1922.5 \cdot G^{-0.95} + 1.48$) over pareto-optimal configurations, representing the best achievable FID for a given computational budget. Points are labeled as N layers $\times$ L loops. Results demonstrate that ELT scales as effectively as the baseline while remaining significantly more parameter-efficient. Scaling both the model dimension (d) and the number of loops (L) follows a predictable efficiency frontier, where larger models with fewer loops often compete with smaller models with more loops at specific target GFLOPs. Trends across faded points show scaling along number of loops L in inference, from a single run trained with a certain L_{max} loops. Refer supp. material for the exact baseline configuration, including the specific number of layers for each comparison point.

5.2 Image Generation

Comparison with Baselines: We present the results for 256×256 image generation on ImageNet-1k in Table 1. Despite using **4**$\times$ less parameters, ELT-XL achieves same FID of 2.0 as MaskGIT-XL, which is the base setup for ELT. As shown in recent literature, using a superior tokenizer [69, 73, 75] or optimized training & inference configurations [46, 47] can further boost ELT's performance.

We also present comparisons of ELT using diffusion models in Table 2. Notably, ELT outperforms the baseline model with 32 layers (FID 3.43) using iso-inference-compute $8N \times 4L$ (FID 3.16) and $16N \times 2L$ (FID 2.83) settings, achieving **4**$\times$ and **2**$\times$ reduction in parameter count respectively. The $1N \times 32L$ configuration (FID 10.30) reveals that a single unique transformer layer, despite 32 effective passes, lacks the representational capacity for high-fidelity generation, highlighting that a minimum block size N is necessary to provide sufficient architectural expressiveness within each iteration. While vanilla looping (without ILSD) gives competitive performance when running inference with same loops as training ($L = L_{max}$), performance degrades drastically for lower number of loops which is mitigated by ILSD as show in Figure 8b.

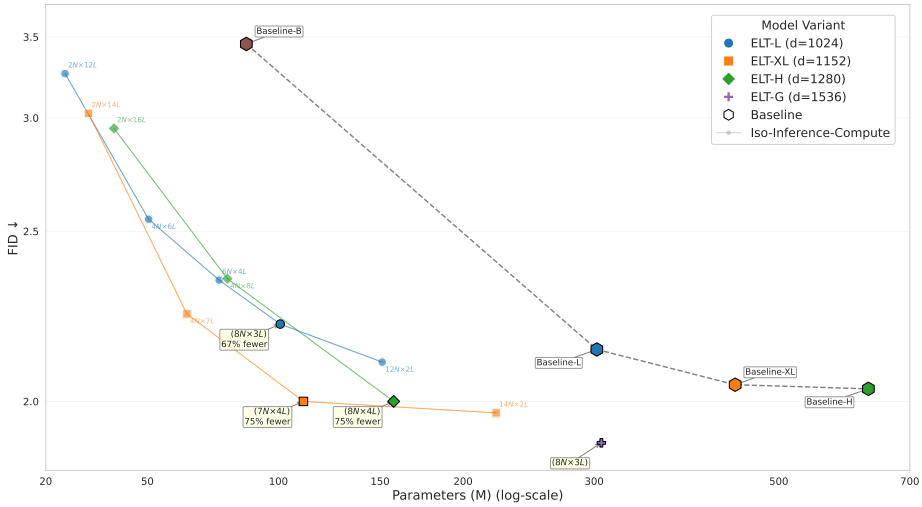


Fig. 5: Scaling with Parameters. We show the best-achievable FID as a function of model parameters (log scale). Each series corresponds to iso-inference-compute configuration for a specific model width (d). Points are labeled as N layers $\times$ L loops. Results show that increasing parameter count via model width yields superior FID. Proposed ELT mechanism shows that even parameter-constrained models can achieve similar performance through recursive depth. Each ELT point in the figure is best inference configuration ($N \times L$) chosen from its corresponding training run ($N \times L_{\max}$). Refer supp. material for the exact baseline configuration, including the specific number of layers for each comparison point.

Qualitative Results: Figure 1 compare ELT against vanilla looped transformers within a diffusion framework. It is clear that ELT unlocks Any-Time inference capability providing dynamic trade-off between generation quality and inference speed. Figure 9 and Figure 10 visualize the generation results of ELT in diffusion and masked generative framework respectively.

Scaling Inference GFLOPs and Pareto Front: Figure 4 illustrates the trade-off between generation quality (FID) and inference compute (GFLOPs). The pareto front (black curve) reveals that while increasing the loop count (L) consistently improves FID (faded points) for a fixed unique layers (N), the gains eventually diminish, where transitioning to the next architecture scale becomes more performant than over-looping smaller models. Crucially, ELT allows for Any-Time inference, we can traverse the pareto curve at test-time by simply adjusting L to meet specific hardware constraints without retraining.

Scaling Parameters: We further investigate the relationship between model capacity and performance in Figure 5. By plotting the best FID achieved at each parameter budget, we observe a clear power-law scaling trend across all model widths. While increasing the number of unique layers (N) reduces FID, the gains are most pronounced when accompanied by an increase in model width

Table 1: Class-conditional Image Generation on ImageNet 256×256. "# steps" refers to the number of neural network runs. $^{\Delta}$ denotes values taken from prior publications. * indicates usage of extra training data. ^{g} denotes use of classifier-free guidance [28]. Note that L in $(N \times L)$ notation is inference loop count per sampling step.

Model	AR	FID ↓	IS ↑	# params	# steps	# Gflops
BigGAN-deep$^{\Delta}$ [5]		7.0	171.4	160M	1	-
StyleGAN-XL$^{\Delta g}$ [56]		2.3	265.1	166M	1	-
Improved DDPM$^{\Delta}$ [48]		12.3	-	280M	250	>150k
ADM + Upsample$^{\Delta g}$ [13]		3.9	215.8	554M	250	371k
LDM-4$^{\Delta g*}$ [52]		3.6	247.7	400M	250	51.5k
DiT-XL/2$^{\Delta g*}$ [49]		2.3	278.2	675M	250	59.5k
MDT$^{\Delta g*}$ [17]		1.8	283.0	676M	250	>59k
MaskDiT$^{\Delta g*}$ [78]		2.3	276.6	736M	250	>28k
CDM$^{\Delta}$ [27]		4.9	158.7	-	8100	-
RIN$^{\Delta}$ [34]		3.4	182.0	410M	1000	334k
Simple Diffusion$^{\Delta g}$ [31]		2.4	256.3	2B	512	-
VDM++$^{\Delta g}$ [37]		2.1	267.7	2B	512	-
EDiff$^{\Delta g}$ [23]		2.1	-	450M	50	119k
LPDM-ADM$^{\Delta g}$ [68]		2.7	-	-	50	7.8k
MAR$^{\Delta g}$ [41]	✓	1.8	296.0	479M	128	-
VQVAE-2$^{\Delta}$ [51]	✓	31.1	∼45	13.5B	5120	-
VQGAN$^{\Delta}$ [15]	✓	15.8	78.3	1.4B	256	-
MaskGIT$^{\Delta}$[7]		6.2	182.1	227M	8	647
Mo-VQGAN$^{\Delta}$ [77]		7.2	130.1	389M	12	∼1k
MaskBit$^{\Delta g}$ [69]		1.7	341.8	305M	64	10.3k
PAR-4×$^{\Delta}$ [67]	✓	3.8	218.9	343M	147	-
PAR-16×$^{\Delta}$ [67]	✓	2.9	262.5	3.1B	51	-
MaskGIT-L^{g}		2.1	270.1	303M	24	3.7k
MaskGIT-XL^{g}		2.0	294.8	446M	24	3.9k
ELT-L ($\boldsymbol{8N \times 3L}$)		2.2	254.3	**101M**	24	3.7k
ELT-L ($\boldsymbol{12N \times 2L}$)		2.1	281.8	**152M**	24	3.7k
ELT-XL ($\boldsymbol{7N \times 4L}$)		2.0	266.1	**111M**	24	3.9k

(d). Specifically, the $d = 1536$ configuration (G) achieves the lowest overall FID with only 8 unique layers, while the full G model has 48 unique layers. However, for on-device budgets, smaller looped variants (L and XL) remain highly efficient, providing a flexible scale-to-performance ratio that is critical for resource-constrained visual generation.

Faster Training Convergence: Our method demonstrates significantly faster convergence than standard DiT architectures [49]. As illustrated in Figure 6, ELT-based diffusion models achieve 2× and 1.4× speedups when using configurations of $16N \times 2L_{\max}$ and $8N \times 4L_{\max}$, respectively, in iso-inference-compute settings with $N = 32$ DiT baseline. Note that effective depth $\mathcal{D}$ is same for both ELT and DiT baseline (32).

Table 2: Class-conditional Image Generation on ImageNet 256×256 using DiT. We report $N \times L$ used for inference in parenthesis in Model column.

Model	FID ↓	# params	$\mathcal{D}$
DiT - 16 layers	3.87	1.1B	16
DiT - 32 layers	3.43	2.1B	32
ELT ($1N \times 32L$)	10.30	69M	32
ELT ($4N \times 8L$)	3.96	271M	32
ELT ($8N \times 4L$)	**3.16**	**539M**	32
ELT ($16N \times 2L$)	**2.83**	**1.1B**	32

Table 3: Throughput gains for ELT. We report throughput (images/sec) ratio (ELT / Baseline) on a Google Cloud TPU v6e [21] with 1×1 topology and inference batch size of 8. The speedup arises from ELT's compact shared parameters fitting on-chip, reducing repeated HBM-to-SRAM transfers. ELT achieves a peak speedup of *3.5*× for model scale H.

ELT	d_{model}	Throughput Ratio
$6N \times 2L$ (B)	768	1.0
$8N \times 3L$ (L)	1024	2.9
$7N \times 4L$ (XL)	1152	3.3
$8N \times 4L$ (H)	1280	**3.5**

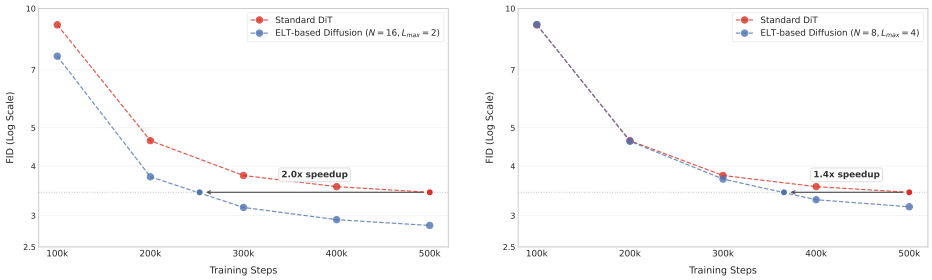


Fig. 6: Faster Training Convergence with ELT. While using 2× less parameters, ELT improves the training efficiency in diffusion framework.

ELT has high Throughput: ELT utilizes a compact set of shared parameters and maintain its major weight footprint closer to the accelerator computation unit. This avoids the cost of repeated memory transfers typically required in standard models. We evaluate the efficiency of ELT by measuring the throughput ratio relative to the baseline across various model scales with an inference batch size of 8, on 1 TPU slice. We choose the best performing ELT inference configuration from each model scale (refer Figure 5) and report their throughput gains in Table 3. Our method delivers significant performance gains across model scales L, XL and H. Note that these gains are contingent on the model size: as long as the shared parameters remain within memory capacity, the architecture eliminates redundant memory movement across iterations. For model scale B, the baseline (MaskGIT [7]) is small enough to fit entirely within device memory capacity, eliminating the transfer penalty. Generally speaking, ELT offers extreme parameter efficiency, which would reduce significant memory transfers even for bigger models which need to be sharded along multiple devices.

Method	Class	FVD ↓	IS↑	# params	# steps	# GFlops
RaMViD$^{\Delta *}$ [33]		-	21.71 ± 0.21	308M	500	-
StyleGAN-V$^{\Delta *}$[61]		-	23.94 ± 0.73	-	1	-
DIGAN$^{\Delta}$ [74]		577±21	32.70± 0.35	-	1	∼148
DVD-GAN$^{\Delta}$ [8]	✓	-	32.97± 1.70	-	1	-
Video Diffusion$^{\Delta *}$ [29]			57.00± 0.62	1.1B	256	-
TATS$^{\Delta}$ [18]		420± 18	57.63± 0.24	321M	1024	-
CCVS+StyleGAN$^{\Delta}$ [40]		386± 15	24.47± 0.13	-	-	-
Make-A-Video$^{\Delta *}$ [60]	✓	367	33.00	-	-	-
TATS$^{\Delta}$ [18]	✓	332± 18	79.28± 0.38	321M	1024	-
CogVideo$^{\Delta *}$ [30]	✓	626	50.46	9.4B	-	-
Make-A-Video$^{\Delta *}$ [60]	✓	81	82.55	≫3.5B	≫250	-
PAR-4×$^{\Delta}$ [67]	✓	99.5	-	792M	323	-
PAR-16×$^{\Delta}$ [67]	✓	103.4	-	792M	95	-
MaGNeTS$^{\Delta}$ [22]	✓	96.4±2	88.53±0.20	306M	12	∼1.7k
MAGVIT-L$^{\Delta}$ [72]	✓	76± 2	89.27± 0.15	306M	12	∼4.3k
ELT ($6N \times 4L$)	✓	72.8±2.5	88.27±0.33	**76M**	12	∼4.3k
ELT ($6N \times 6L$)	✓	60.8±2.7	87.88±0.39	**76M**	24	∼13k

Table 4: Video Generation on UCF-101. Methods in gray are pretrained on additional large video data. ✓ denotes class-conditional. * indicates custom resolutions. $^{\Delta}$ denotes values from prior publications. No guidance is used.

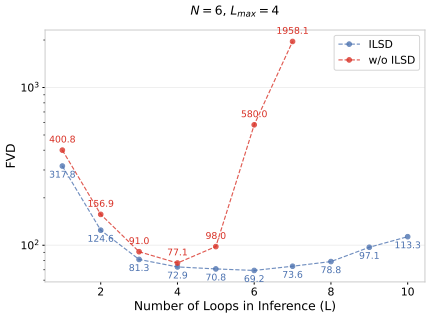


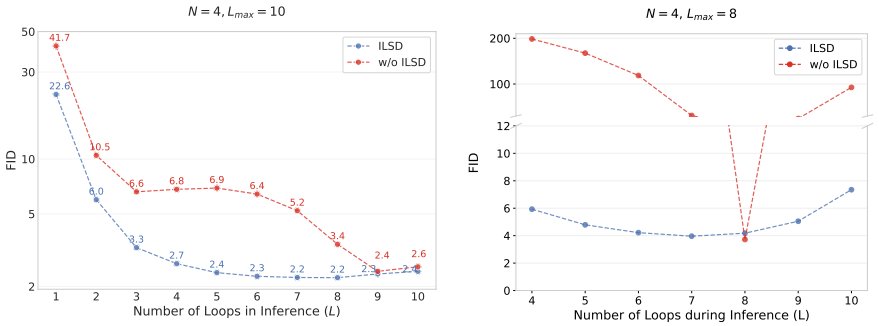
Fig. 7: Impact of ILSD on class-conditional UCF-101 generation. ILSD significantly improves performance for all values of L in inference especially when $L \neq L_{max}$.

5.3 Video Generation

We use the MAGVIT [72] framework to train parallel decoding based video generation models. We summarize the results for class-conditional video generation on UCF-101 in Table 4. Our compact 76M ELT model outperforms MAGVIT baseline (FVD 72.8 vs 76) on data-constrained settings of UCF-101 (∼13.7M training tokens) in iso-inference-compute settings. Scaling the compute further with number of loops and sampling steps gives a boost in performance, reaching FVD of 60.8. This suggests that looped transformers can exhibit robustness against overfitting in data-constrained regimes like UCF-101, effectively regularizing the learning process while maintaining the expressive capacity for high-quality generation.

5.4 Intra-Loop Self Distillation (ILSD) drives Elasticity

We analyze the impact of ILSD for image generation across masked generative models (refer to Figure 8a) and diffusion models (refer to Figure 8b). Models trained without ILSD exhibit significant divergence from their fixed training depth (L_{max}). In contrast, ELT maintains stable, high-quality generation across the entire inference loop spectrum (see Figure 1). We further analyse class-conditional video generation on UCF-101 in Figure 7. Interestingly, ILSD enables the model to maintain reasonable quality even at unseen depths ($L > L_{\max}$). On UCF-101, the model achieves a peak FVD of *69.20* at $L = 6$ despite being trained with $L_{\max} = 4$, suggesting that ILSD regularizes the iterative process sufficiently for modest extrapolation beyond training depth. We note that this extrapolation behavior warrants further investigation across datasets and scales.



(a) Masked Generative Models **(b)** Diffusion Transformers

Fig. 8: Impact of ILSD across generation processes. ILSD improves performance for all L in inference especially when $L \neq L_{max}$. For w/o ILSD, L_{max} here refers to the number of loops the model is trained with. Results are on ImageNet 256×256.

6 Conclusion

We proposed a novel parameter-efficient approach to visual generation using recurrent transformers called Elastic Looped Transformers (ELT). Our approach achieves a strong empirical performance, similar to baselines, with $4\times$ fewer parameters in iso-inference-compute setting in both image and video generation tasks. Beyond significantly improved performance per parameter, we identified fundamental scaling properties of looped transformers: while increasing model width remains a primary driver of quality, recursive looping provides a unique "test-time" compute lever. Through our proposed Intra-Loop Self Distillation (ILSD) strategy, we train a single model that is performant across a variable number of iterations. This strategy effectively yields a continuous family of models from a single training run, enabling Any-Time inference where practitioners can traverse the pareto front to balance image quality and GFLOPs dynamically.

Looking forward, we note that ELTs can potentially unlock more efficient inference for diffusion models. While existing approaches rely on the same network (and hence allocate same compute) across denoising steps, ELT can dynamically allocate compute across denoising steps, spending more compute where it matters the most. Additionally, in the context of recent one-step generative modeling paradigms such as consistency models [62] and drifting models [11], ELT can enable true elasticity: since there is only one sampling step, one can dynamically control the quality of the model at inference by varying the number of loops, without having to pre-determine the number of sampling steps as is the case with traditional multi-step diffusion models. We believe this paradigm of flexible, weight-efficient scaling offers a promising direction for deploying high-fidelity generative models on resource-constrained hardware.

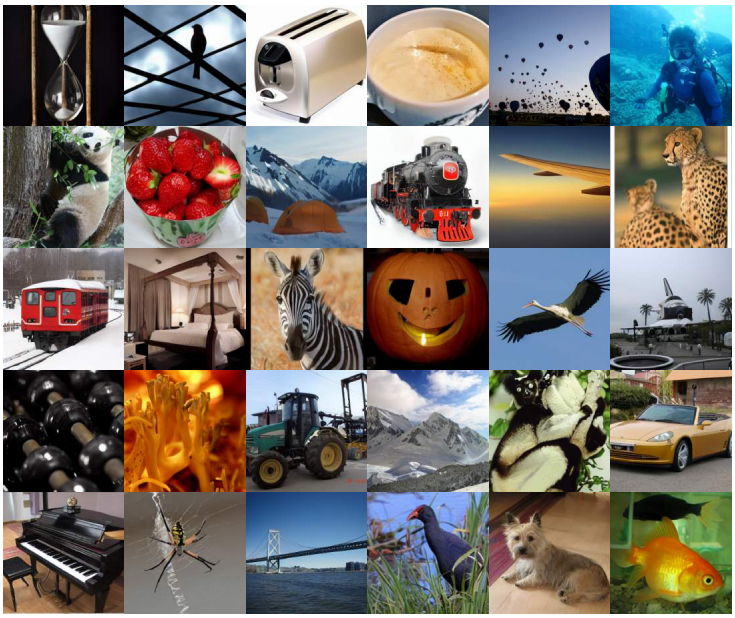


Fig. 9: Class-conditional Image Generation on ImageNet 256 × 256. Qualitative results for ELT in diffusion framework, $16N \times 2L$ inference configuration & trained with $L_{\max} = 2$.

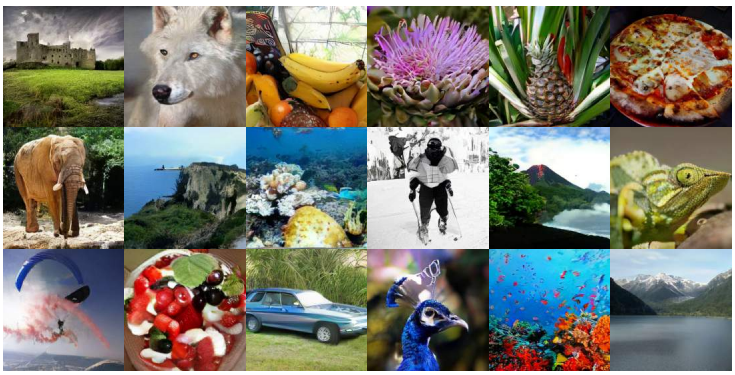


Fig. 10: Class-conditional Image Generation on ImageNet 256 × 256. Qualitative results for ELT in masked generative framework. Model scale: ELT-G, $8N \times 3L$ inference config & trained with $L_{\max} = 4$ (FID=1.9).

Bibliography

[1] Anil, C., Pokle, A., Liang, K., Treutlein, J., Wu, Y., Bai, S., Kolter, Z., Grosse, R.: Path independent equilibrium models can better exploit test-time computation (2022), URL https://arxiv.org/abs/2211.09961

[2] Bachmann, R., Allardice, J., Mizrahi, D., Fini, E., Kar, O.F., Amirloo, E., El-Nouby, A., Zamir, A., Dehghan, A.: Flextok: Resampling images into 1d token sequences of flexible length (2025), URL https://arxiv.org/abs/2502.13967

[3] Bae, S., Kim, Y., Bayat, R., Kim, S., Ha, J., Schuster, T., Fisch, A., Harutyunyan, H., Ji, Z., Courville, A., Yun, S.Y.: Mixture-of-recursions: Learning dynamic recursive depths for adaptive token-level computation (2025), URL https://arxiv.org/abs/2507.10524

[4] Bai, S., Kolter, J.Z., Koltun, V.: Deep equilibrium models (2019), URL https://arxiv.org/abs/1909.01377

[5] Brock, A.: Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096 (2018)

[6] Castells, T., Song, H.K., Piao, T., Choi, S., Kim, B.K., Yim, H., Lee, C., Kim, J.G., Kim, T.H.: Edgefusion: On-device text-to-image generation (2024), URL https://arxiv.org/abs/2404.11925

[7] Chang, H., Zhang, H., Jiang, L., Liu, C., Freeman, W.T.: Maskgit: Masked generative image transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11315–11325 (2022)

[8] Clark, A., Donahue, J., Simonyan, K.: Adversarial video generation on complex datasets. arXiv preprint arXiv:1907.06571 (2019)

[9] Dehghani, M., Gouws, S., Vinyals, O., Uszkoreit, J., Kaiser, Ł.: Universal transformers. arXiv preprint arXiv:1807.03819 (2018)

[10] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition, pp. 248–255, Ieee (2009)

[11] Deng, M., Li, H., Li, T., Du, Y., He, K.: Generative modeling via drifting. arXiv preprint arXiv:2602.04770 (2026)

[12] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2019), URL https://arxiv.org/abs/1810.04805

[13] Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Advances in neural information processing systems **34**, 8780–8794 (2021)

[14] Duggal, S., Isola, P., Torralba, A., Freeman, W.T.: Adaptive length image tokenization via recurrent allocation (2024), URL https://arxiv.org/abs/2411.02393

[15] Esser, P., Rombach, R., Ommer, B.: Taming transformers for high-resolution image synthesis. In: CVPR, pp. 12873–12883 (2021)

[16] Gabor, M., Piotrowski, T., Cavalcante, R.L.G.: Positive concave deep equilibrium models (2024), URL https://arxiv.org/abs/2402.04029

[17] Gao, S., Zhou, P., Cheng, M.M., Yan, S.: Masked diffusion transformer is a strong image synthesizer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 23164–23173 (2023)
[18] Ge, S., Hayes, T., Yang, H., Yin, X., Pang, G., Jacobs, D., Huang, J.B., Parikh, D.: Long video generation with time-agnostic vqgan and time-sensitive transformer (2022), URL https://arxiv.org/abs/2204.03638
[19] Geiping, J., McLeish, S., Jain, N., Kirchenbauer, J., Singh, S., Bartoldson, B.R., Kailkhura, B., Bhatele, A., Goldstein, T.: Scaling up test-time compute with latent reasoning: A recurrent depth approach (2025), URL https://arxiv.org/abs/2502.05171
[20] Geng, Z., Pokle, A., Kolter, J.Z.: One-step diffusion distillation via deep equilibrium models (2023), URL https://arxiv.org/abs/2401.08639
[21] Google Cloud: TPU v6e (Trillium) Documentation. https://cloud.google.com/tpu/docs/v6e (2024), accessed: 2024-05-22
[22] Goyal, S., Tula, D., Jain, G., Shenoy, P., Jain, P., Paul, S.: Masked generative nested transformers with decode time scaling (2025), URL https://arxiv.org/abs/2502.00382
[23] Hang, T., Gu, S., Li, C., Bao, J., Chen, D., Hu, H., Geng, X., Guo, B.: Efficient diffusion training via min-snr weighting strategy (2024), URL https://arxiv.org/abs/2303.09556
[24] He, H., Liang, J., Wang, X., Wan, P., Zhang, D., Gai, K., Pan, L.: Scaling image and video generation via test-time evolutionary search (2025), URL https://arxiv.org/abs/2505.17618
[25] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems **30** (2017)
[26] Ho, J., Jain, A.P., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
[27] Ho, J., Saharia, C., Chan, W., Fleet, D.J., Norouzi, M., Salimans, T.: Cascaded diffusion models for high fidelity image generation. Journal of Machine Learning Research **23**(47), 1–33 (2022)
[28] Ho, J., Salimans, T.: Classifier-free diffusion guidance (2022), URL https://arxiv.org/abs/2207.12598
[29] Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., Fleet, D.J.: Video diffusion models. Advances in Neural Information Processing Systems **35**, 8633–8646 (2022)
[30] Hong, W., Ding, M., Zheng, W., Liu, X., Tang, J.: Cogvideo: Large-scale pretraining for text-to-video generation via transformers. arXiv preprint arXiv:2205.15868 (2022)
[31] Hoogeboom, E., Heek, J., Salimans, T.: simple diffusion: End-to-end diffusion for high resolution images. In: International Conference on Machine Learning, pp. 13213–13232, PMLR (2023)
[32] Hoogeboom, E., Mensink, T., Heek, J., Lamerigts, K., Gao, R., Salimans, T.: Simpler diffusion: 1.5 fid on imagenet512 with pixel-space diffusion. In: Proceedings of the Computer Vision and Pattern Recognition Conference, pp. 18062–18071 (2025)

[33] Höppe, T., Mehrjou, A., Bauer, S., Nielsen, D., Dittadi, A.: Diffusion models for video prediction and infilling. arXiv preprint arXiv:2206.07696 (2022)
[34] Jabri, A., Fleet, D., Chen, T.: Scalable adaptive computation for iterative generation. arXiv preprint arXiv:2212.11972 (2022)
[35] Kar, K., Kubilius, J., Schmidt, K., Issa, E.B., DiCarlo, J.J.: Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. Nature neuroscience **22**(6), 974–983 (2019)
[36] Kietzmann, T.C., Spoerer, C.J., Sörensen, L.K., Cichy, R.M., Hauk, O., Kriegeskorte, N.: Recurrence is required to capture the representational dynamics of the human visual system. Proceedings of the National Academy of Sciences **116**(43), 21854–21863 (2019)
[37] Kingma, D.P., Gao, R.: Understanding the diffusion objective as a weighted integral of elbos. arXiv preprint arXiv:2303.00848 **2** (2023)
[38] Kudugunta, S., Kusupati, A., Dettmers, T., Chen, K., Dhillon, I., Tsvetkov, Y., Hajishirzi, H., Kakade, S., Farhadi, A., Jain, P., et al.: Matformer: Nested transformer for elastic inference. arXiv preprint arXiv:2310.07707 (2023)
[39] Kusupati, A., Bhatt, G., Rege, A., Wallingford, M., Sinha, A., Ramanujan, V., Howard-Snyder, W., Chen, K., Kakade, S., Jain, P., et al.: Matryoshka representation learning. Advances in Neural Information Processing Systems **35**, 30233–30249 (2022)
[40] Le Moing, G., Ponce, J., Schmid, C.: Ccvs: Context-aware controllable video synthesis. Advances in Neural Information Processing Systems **34**, 14042–14055 (2021)
[41] Li, T., Tian, Y., Li, H., Deng, M., He, K.: Autoregressive image generation without vector quantization (2024), URL https://arxiv.org/abs/2406.11838
[42] Li, Y.: Mor-vit: Efficient vision transformer with mixture-of-recursions (2025), URL https://arxiv.org/abs/2507.21761
[43] McCallum, S., Arora, K., Foster, J.: Reversible deep equilibrium models (2025), URL https://arxiv.org/abs/2509.12917
[44] Menghani, G.: Efficient deep learning: A survey on making deep learning models smaller, faster, and better. ACM Computing Surveys **55**(12), 1–37 (2023)
[45] Miwa, K., Sasaki, K., Arai, H., Takahashi, T., Yamaguchi, Y.: One-d-piece: Image tokenizer meets quality-controllable compression (2025), URL https://arxiv.org/abs/2501.10064
[46] Ni, Z., Wang, Y., Zhou, R., Guo, J., Hu, J., Liu, Z., Song, S., Yao, Y., Huang, G.: Revisiting non-autoregressive transformers for efficient image synthesis (2024), URL https://arxiv.org/abs/2406.05478
[47] Ni, Z., Wang, Y., Zhou, R., Han, Y., Guo, J., Liu, Z., Yao, Y., Huang, G.: Enat: Rethinking spatial-temporal interactions in token-based image synthesis (2024), URL https://arxiv.org/abs/2411.06959
[48] Nichol, A.Q., Dhariwal, P.: Improved denoising diffusion probabilistic models. In: International conference on machine learning, pp. 8162–8171, PMLR (2021)

[49] Peebles, W., Xie, S.: Scalable diffusion models with transformers (2023), URL https://arxiv.org/abs/2212.09748
[50] Pokle, A., Geng, Z., Kolter, Z.: Deep equilibrium approaches to diffusion models (2022), URL https://arxiv.org/abs/2210.12867
[51] Razavi, A., Van den Oord, A., Vinyals, O.: Generating diverse high-fidelity images with vq-vae-2. Advances in neural information processing systems **32** (2019)
[52] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models (2022), URL https://arxiv.org/abs/2112.10752
[53] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation (2015), URL https://arxiv.org/abs/1505.04597
[54] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans (2016), URL https://arxiv.org/abs/1606.03498
[55] Salimans, T., Ho, J.: Progressive distillation for fast sampling of diffusion models (2022), URL https://arxiv.org/abs/2202.00512
[56] Sauer, A., Schwarz, K., Geiger, A.: Stylegan-xl: Scaling stylegan to large diverse datasets (2022), URL https://arxiv.org/abs/2202.00273
[57] Saunshi, N., Dikkala, N., Li, Z., Kumar, S., Reddi, S.J.: Reasoning with latent thoughts: On the power of looped transformers (2025), URL https://arxiv.org/abs/2502.17416
[58] Shen, J., Tirumala, K., Yasunaga, M., Misra, I., Zettlemoyer, L., Yu, L., Zhou, C.: Cat: Content-adaptive image tokenization (2025), URL https://arxiv.org/abs/2501.03120
[59] Shu, W.J., Qiu, X., Zhu, R.J., Chen, H.H., Liu, Y., Yang, H.: Loopvit: Scaling visual arc with looped transformers (2026), URL https://arxiv.org/abs/2602.02156
[60] Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., et al.: Make-a-video: Text-to-video generation without text-video data. arXiv preprint arXiv:2209.14792 (2022)
[61] Skorokhodov, I., Tulyakov, S., Elhoseiny, M.: Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3626–3636 (2022)
[62] Song, Y., Dhariwal, P., Chen, M., Sutskever, I.: Consistency models (2023), URL https://arxiv.org/abs/2303.01469
[63] Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456 (2020)
[64] Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild (2012), URL https://arxiv.org/abs/1212.0402
[65] Unterthiner, T., Van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., Gelly, S.: Towards accurate generative models of video: A new metric & challenges. arXiv preprint arXiv:1812.01717 (2018)

[66] Wang, J., Lai, Z., Chen, J., Guo, J., Guo, H., Li, X., Yue, X., Guo, C.: Elastic diffusion transformer (2026), URL https://arxiv.org/abs/2602.13993
[67] Wang, Y., Ren, S., Lin, Z., Han, Y., Guo, H., Yang, Z., Zou, D., Feng, J., Liu, X.: Parallelized autoregressive visual generation (2024), URL https://arxiv.org/abs/2412.15119
[68] Wang, Z., Jiang, Y., Zheng, H., Wang, P., He, P., Wang, Z., Chen, W., Zhou, M.: Patch diffusion: Faster and more data-efficient training of diffusion models (2023), URL https://arxiv.org/abs/2304.12526
[69] Weber, M., Yu, L., Yu, Q., Deng, X., Shen, X., Cremers, D., Chen, L.C.: Maskbit: Embedding-free image generation via bit tokens (2024), URL https://arxiv.org/abs/2409.16211
[70] Yan, W., Mnih, V., Faust, A., Zaharia, M., Abbeel, P., Liu, H.: Elastictok: Adaptive tokenization for image and video (2025), URL https://arxiv.org/abs/2410.08368
[71] Yang, L., Lee, K., Nowak, R., Papailiopoulos, D.: Looped transformers are better at learning learning algorithms. arXiv preprint arXiv:2311.12424 (2023)
[72] Yu, L., Cheng, Y., Sohn, K., Lezama, J., Zhang, H., Chang, H., Hauptmann, A.G., Yang, M.H., Hao, Y.K., Essa, I., et al.: Magvit: Masked generative video transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10459–10469 (2023)
[73] Yu, L., Lezama, J., Gundavarapu, N.B., Versari, L., Sohn, K., Minnen, D., Cheng, Y., Gupta, A., Gu, X., Hauptmann, A.G., et al.: Language model beats diffusion–tokenizer is key to visual generation. arXiv preprint arXiv:2310.05737 (2023)
[74] Yu, S., Tack, J., Mo, S., Kim, H., Kim, J., Ha, J.W., Shin, J.: Generating videos with dynamics-aware implicit generative adversarial networks. arXiv preprint arXiv:2202.10571 (2022)
[75] Zhao, Y., Xiong, Y., Krähenbühl, P.: Image and video tokenization with binary spherical quantization (2024), URL https://arxiv.org/abs/2406.07548
[76] Zhao, Y., Xu, Y., Xiao, Z., Jia, H., Hou, T.: Mobilediffusion: Instant text-to-image generation on mobile devices (2024), URL https://arxiv.org/abs/2311.16567
[77] Zheng, C., Vuong, L.T., Cai, J., Phung, D.: Movq: Modulating quantized vectors for high-fidelity image generation (2022), URL https://arxiv.org/abs/2209.09002
[78] Zheng, H., Nie, W., Vahdat, A., Anandkumar, A.: Fast training of diffusion models with masked transformers. arXiv preprint arXiv:2306.09305 (2023)
[79] Zilberstein, S.: Using anytime algorithms in intelligent systems. AI magazine **17**(3), 73–73 (1996)