Scalable Diffusion Models with Transformers

Lebedyuk Eva

May 2024

Table of contents

Introduction + Related work

- Machine learning renaissance powered by ***transformers***:
  - NLP
  - Vision
  - … other domains
- ***Diffusion models*** don’t follow the trend:
  - U-Net architecture – de-facto choice of backbone

- Introduction of ***DiTs***:
  - U-Net ***is not*** crucial to the performance of diffusion models
  - It can be replaced by transformers

Diffusion transformers

- Diffusion problem statement:
  - Forward noising process: $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t$, where $\epsilon_t \sim \mathcal{N}(0, \mathbf{I})$
  - Diffusion models are trained to learn ***the reverse process***: $p_\theta(x_{t-1}|x_t) = \mathcal{N}(\mu_\theta(x_t), \Sigma_\theta(x_t))$

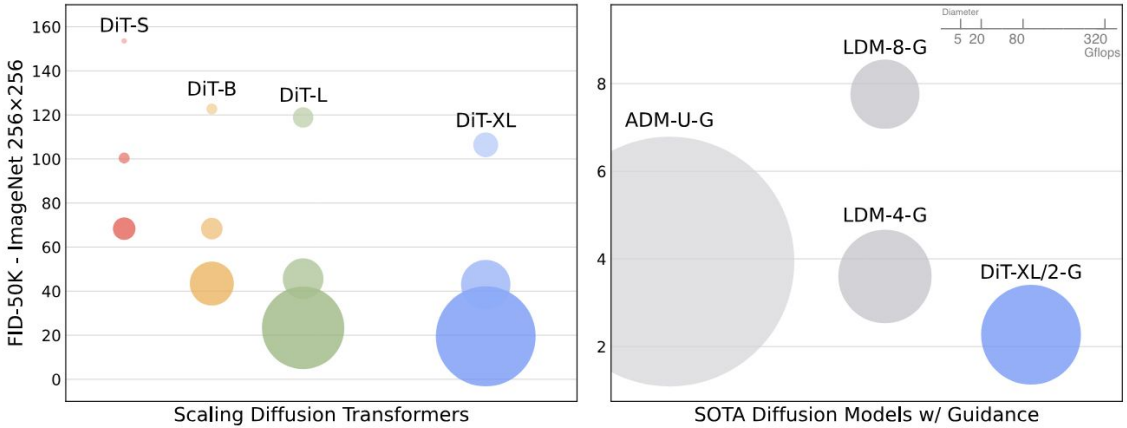

Figure 2. **ImageNet generation with Diffusion Transformers (DiTs).** Bubble area indicates the flops of the diffusion model. *Left:* FID-50K (lower is better) of our DiT models at 400K training iterations. Performance steadily improves in FID as model flops increase. *Right:* Our best model, DiT-XL/2, is compute-efficient and outperforms all prior U-Net-based diffusion models, like ADM and LDM.

Architecture: patchify

- Converts the spatial input into a sequence of ***T*** tokens, each of dimension ***d***
- ***T*** is determined by the patch size hyperparameter ***p***

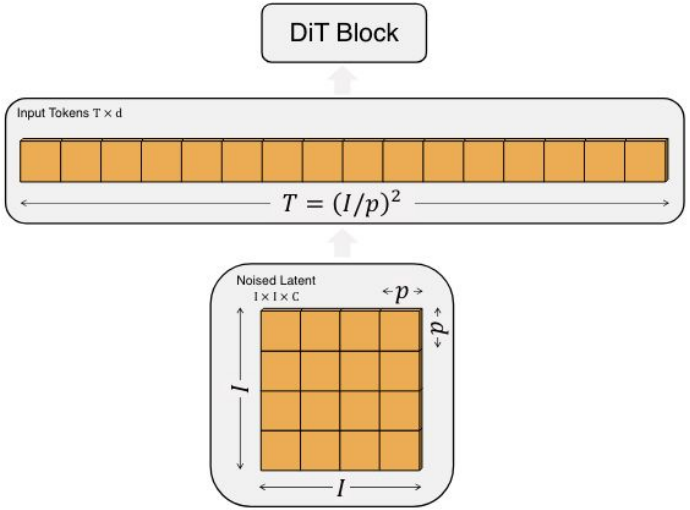

Figure 4. **Input specifications for DiT.** Given patch size $p \times p$, a spatial representation (the noised latent from the VAE) of shape $I \times I \times C$ is "patchified" into a sequence of length $T = (I/p)^2$ with hidden dimension d. A smaller patch size p results in a longer sequence length and thus more Gflops.

Architecture: DiT blocks

- In-context conditioning
- Cross-attention block
- Adaptive layer norm (adaLN) block
- adaLN-Zero block

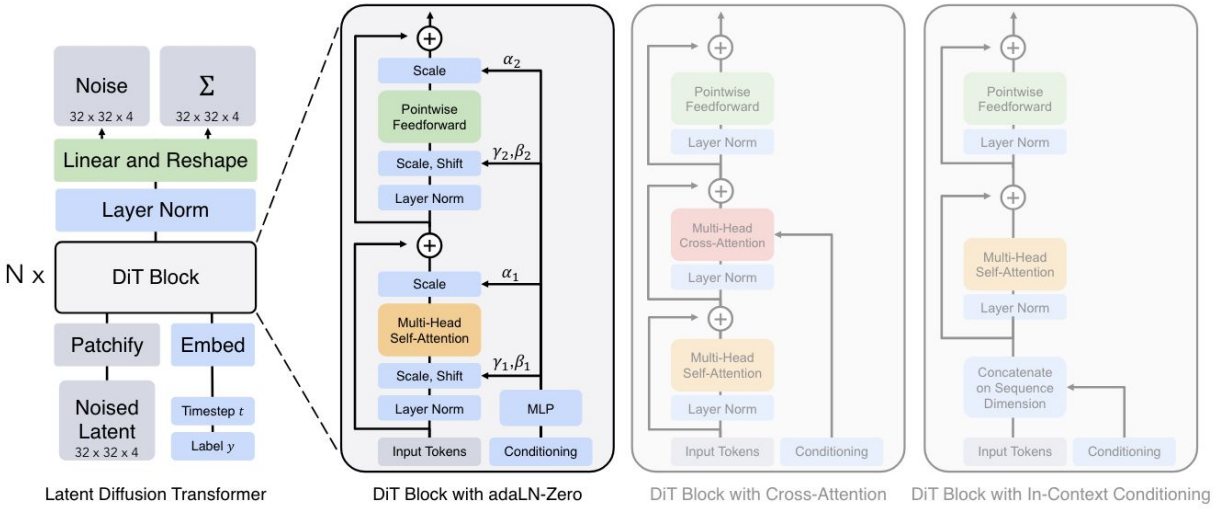

Figure 3. **The Diffusion Transformer (DiT) architecture.** *Left:* We train conditional latent DiT models. The input latent is decomposed into patches and processed by several DiT blocks. *Right:* Details of our DiT blocks. We experiment with variants of standard transformer blocks that incorporate conditioning via adaptive layer norm, cross-attention and extra input tokens. Adaptive layer norm works best.

Architecture: model size

- DiT-S,
- DiT-B
- DiT-L
- DiT-XL

Model	Layers N	Hidden size d	Heads	Gflops (I=32, p=4)
DiT-S	12	384	6	1.4
DiT-B	12	768	12	5.6
DiT-L	24	1024	16	19.7
DiT-XL	28	1152	16	29.1

Table 1. **Details of DiT models.** We follow ViT [10] model configurations for the Small (S), Base (B) and Large (L) variants; we also introduce an XLarge (XL) config as our largest model.

Architecture: transformer decoder

- Needs to decode sequence of image tokens into:
  - output noise prediction
  - output diagonal covariance prediction.
- Standard linear decoder is used
- Rearrange the decoded tokens into their original spatial layout

The complete DiT design space:
patch size, transformer block architecture and model size.

Experiments: model & patch size scaling

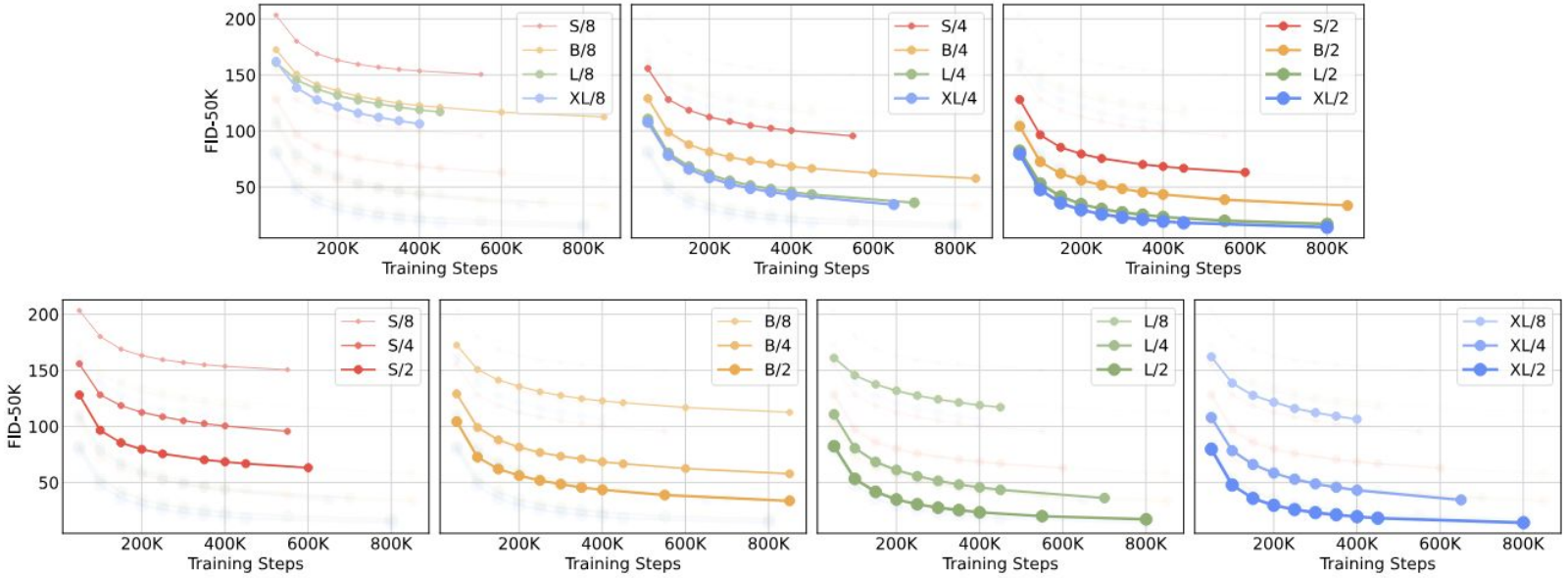

Figure 6. **Scaling the DiT model improves FID at all stages of training.** We show FID-50K over training iterations for 12 of our DiT models. *Top row:* We compare FID holding patch size constant. *Bottom row:* We compare FID holding model size constant. Scaling the transformer backbone yields better generative models across all model sizes and patch sizes.

Experiments: model & patch size scaling

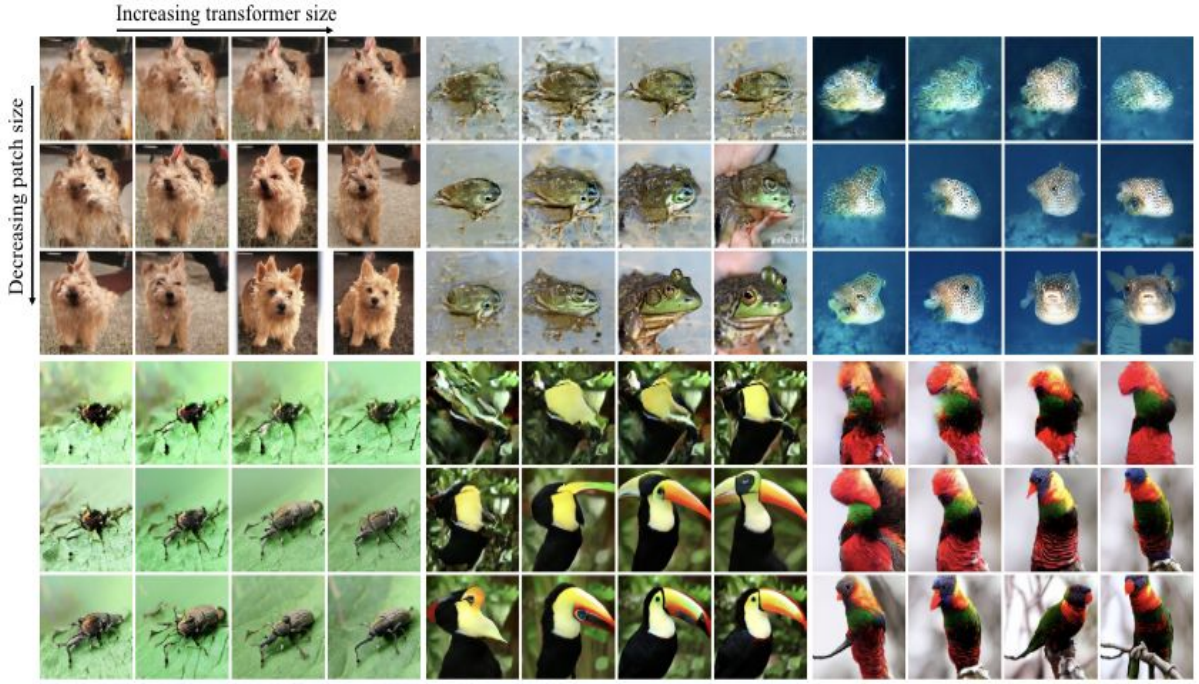

Figure 7. **Increasing transformer forward pass Gflops increases sample quality.** *Best viewed zoomed-in.* We sample from all 12 of our DiT models after 400K training steps using the same input latent noise and class label. Increasing the Gflops in the model—either by increasing transformer depth/width or increasing the number of input tokens—yields significant improvements in visual fidelity.

Scaling model vs Sampling compute

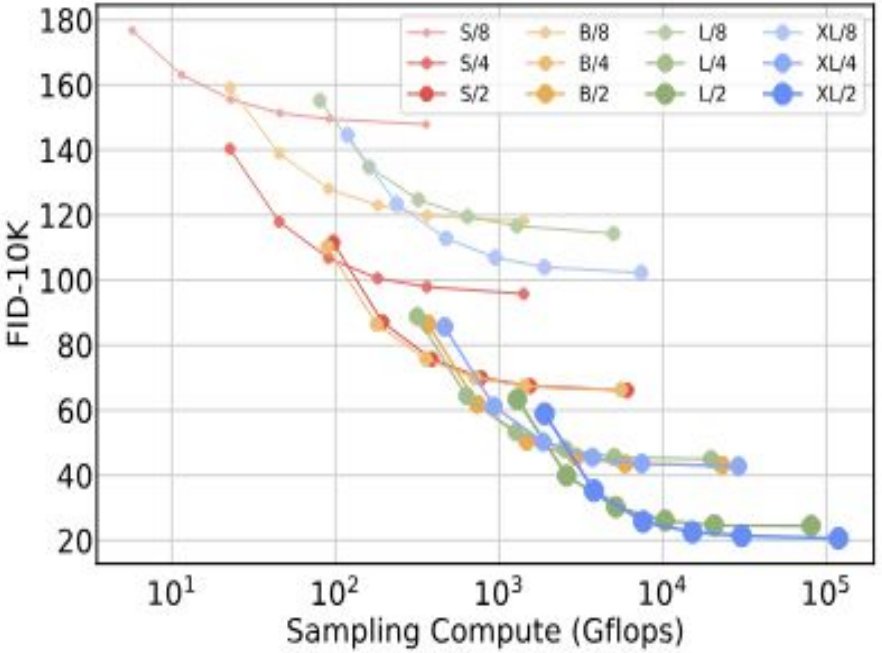

Figure 10. **Scaling-up *sampling* compute does not compensate for a lack of *model* compute.** For each of our DiT models trained for 400K iterations, we compute FID-10K using [16, 32, 64, 128, 256, 1000] sampling steps. For each number of steps, we plot the FID as well as the Gflops used to sample each image. Small models cannot close the performance gap with our large models, even if they sample with more test-time Gflops than the large models.

Conclusion

- Diffusion Transformers (DiTs) are introduced
  - Simple transformer-based backbone
  - Inherits the ***excellent scaling properties*** of the transformer model class
  - Future work possibilities:
    - continue to scaling
    - DiT as drop-in backbone for text-to-image models

Source:

- https://arxiv.org/pdf/2212.09748