This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

SnapGen: Taming High-Resolution Text-to-Image Models for Mobile Devices with Efficient Architectures and Training

Jierun Chen[1,2,*] Dongting Hu[1,3,*] Xijie Huang[1,2,*] Huseyin Coskun[1] Arpit Sahni[1]
Aarush Gupta[1] Anujraaj Goyal[1] Dishani Lahiri[1] Rajesh Singh[1] Yerlan Idelbayev[1]
Junli Cao[1] Yanyu Li[1] Kwang-Ting Cheng[2] S.-H. Gary Chan[2] Mingming Gong[3, 4]
Sergey Tulyakov[1] Anil Kag[1,†] Yanwu Xu[1,†] Jian Ren[1,†]

[1] Snap Inc. [2] HKUST [3] The University of Melbourne [4] MBZUAI
[*] Equal contribution [†] Equal advising

Project Page: https://snap-research.github.io/snapgen

Figure 1. Comparison of various text-to-image models in terms of model size, mobile device compatibility, and visual output quality. Our model, with only *379M* parameters, demonstrates competitive visual quality while being mobile-compatible. Input text prompts are shown above each image grid; all images are generated at 1024^2 resolution—zoom in for details. More examples are shown in webpage.

Abstract

Existing text-to-image (T2I) diffusion models face several limitations, including large model sizes, slow runtime, and low-quality generation on mobile devices. This paper aims to address all of these challenges by developing ***an extremely small and fast T2I model that generates high-resolution and high-quality images on mobile platforms****. We propose several techniques to achieve this goal. First, we systematically examine the design choices of the network architecture to reduce model parameters and latency, while ensuring high-quality generation. Second, to further improve generation quality, we employ cross-architecture knowledge distillation from a much larger model, using a multi-level approach to guide the training of our model from scratch. Third, we enable a few-step generation by integrating adversarial guidance with knowledge distillation. For the first time, our model SnapGen, demonstrates the generation of* 1024^2 *px images on a mobile device around* 1.4 *seconds. On ImageNet-1K, our model, with only* 372*M parameters, achieves an FID of* 2.06 *for* 256^2 *px generation. On T2I benchmarks (*i.e.*, GenEval and DPG-Bench), our model with merely* 379*M parameters, surpasses large-scale models with billions of parameters at a significantly smaller size (*e.g.*,* $7\times$ *smaller than SDXL,* $14\times$ *smaller than IF-XL).*

1. Introduction

Large-scale text-to-image (T2I) diffusion models [12, 13, 15, 51, 53, 57–59] have achieved remarkable success in content generation, powering numerous applications like image editing [48, 64, 72, 85] and video creation [50, 54, 78]. However, T2I models often come with substantial model sizes and slow runtime, and deploying them on the cloud raises concerns related to data security and high costs [65].

To address these challenges, there is huge growing interest in developing smaller and faster T2I models through techniques like model compression (*e.g.*, pruning and quantization) [40, 67, 86], step reduction by distillation [77, 80], and efficient attention mechanisms that mitigate the quadratic complexity [46, 74]. Nevertheless, current works still encounter limitations, *e.g.*, low-resolution generation on mobile devices, that constrain their broader application.

Most importantly, a critical question remains unexplored: *how can we train a T2I model from scratch to generate high-quality, high-resolution images on mobile?* Such a model would offer substantial advantages in speed, compactness, cost-effectiveness, and secure deployment. To build this model, we introduce several innovations:

- **Efficient Network Architectures:** We conduct an in-depth examination of network architectures, including the denoising *UNet* and *Autoencoder* (AE), to obtain optimal trade-off between resource usage and performance. Unlike prior works that optimize and compress pre-trained diffusion models [10, 32, 83], we directly focus on macro- and micro-level design choices to achieve a novel architecture that greatly reduces model size and computational complexity, while preserving high-quality generation.
- **Improved Training Techniques:** We introduce several improvements to train a *compact* T2I model from *scratch*. We utilize flow matching [44, 47] as objective, aligning with larger models like SD3 [18] and SD3.5 [3]. This design enables effective knowledge and step distillation, transferring rich representations from large-scale diffusion models to our much smaller one. Further, we propose a multi-level knowledge distillation with a timestep-aware scaling that combines multiple training objectives. Instead of weighting objectives through a linear combination as in prior works [32, 46], we consider *target prediction difficulty* (i.e., student-teacher difference) across various timesteps in flow matching.
- **Advanced Step Distillation:** We perform step distillation on our model by combining the adversarial training along with the knowledge distillation using a few-step teacher model (*i.e.*, SD3.5-Large-Turbo [5]), enabling ultra-fast high-quality generation with only 4 or 8 steps.

We demonstrate the superior advantages of our approach and model through extensive experiments:

- On ImageNet-1K [16] class-conditional image generation task, our model achieves the FID comparable to existing works with significantly reduced model size and computation, *i.e.*, half the model size and one-third of the compute resources compared with SiT-XL[49], as in Tab. 1.
- For large-scale T2I generation, our UNet model, with only 379M parameters, demonstrates superior generation quality compared to billion-parameter models [42, 53, 87], *e.g.*, improved metrics on benchmark datasets (Tab. 3) and human evaluation (Fig. 8).
- Our compressed decoder, trained from scratch, has competitive reconstruction quality compared to commonly used models [18, 53], with more than $36\times$ smaller size, enabling the mobile deployment.
- Notably, for the *first* time, we show a T2I model achieving high-resolution generation (*e.g.*, 1024^2 px) on mobile devices (*e.g.*, iPhone 16 Pro-Max) around 1.4 seconds.

2. Related Work

High-Resolution Text-to-Image Models have emerged with advanced architectures and multi-stage approaches designed to enhance visual fidelity and user customization. SDXL [53] is a pioneering work in this field, employing a refined cascading approach with UNet backbone to generate high-detailed images, resulting in photorealistic outputs that maintain sharpness and clarity. The following studies explore different techniques like more advanced text encoders, better image refinement, or improved dataset preparation,

to obtain better text-image alignment or higher-quality generation [6, 7, 15, 20, 30, 36–38, 41, 45, 48, 69]. However, most of these models contain billions of parameters, making them extremely slow, and not being able to run on resource-contained hardware like mobile devices. In this work, we aim to build a small and fast model that can perform high-resolution generation even on mobile platforms.

Efficient Diffusion Models address the challenges of bulky model size and long runtime. There have been efforts exploring the architecture optimization to remove redundancy from large models, demonstrating the on-device generation within seconds [11, 40, 67, 86]. However, these models are constrained to low-resolution output, *i.e.*, 512^2 px. To enable efficient high-resolution generation, SANA [74] and LinFusion [46] incorporate linear attention [9, 14, 31] to achieve 1K generation on laptop GPUs. In contrast, we target a broader range of platforms, supporting high-resolution generation (*e.g.*, 1K) directly on mobile devices.

Knowledge Distillation in Diffusion Models. In the context of diffusion models, previous works focus on distilling large, high-capacity teacher models into more compact, efficient student models within a *homogeneous* architecture [32, 46]. They reduce the complexity of the model by removing certain components like attention [70] or residual blocks [23], while maintaining the architectural structure. However, our approach diverges from this trend by utilizing a *heterogeneous* architecture for more aggressively efficient yet challenging distillation.

Adversarial Step Distillation uses the techniques from adversarial training [22] to reduce the number of diffusion steps, while maintaining high image quality [60, 61]. For example, UFOGen [77] employs a diffusion-GAN formulation [71, 73, 76] to significantly reduce inference time while maintaining competitive performance. DMD2 [79] builds on prior distillation methods by using distribution matching with adversarial loss. Different from exiting works, we conduct step-distillation on a very compact model and train the model along with the knowledge distillation.

3. Method

In this section, we present how to craft and train a highly efficient T2I model for high-resolution generation. Specifically, starting from the architecture designed in the latent diffusion model [58], we optimize both denoising backbone (Sec. 3.1, Fig. 2, and Fig. 3) and autoencoder (Sec. 3.2 and Fig. 4) to make them compact and fast, even on mobile devices. We then propose the improved training recipe and knowledge distillation (Sec. 3.3 and Fig. 5), empowering a high-performance T2I model. Lastly, we introduce our step distillation to significantly reduce the number of denoising steps for a faster T2I model (Sec. 3.4 and Fig. 7).

3.1. Efficient UNet Architecture

Here we describe the design choices for the denoising UNet.

Baseline Architecture. We choose UNet from SDXL [53] as the baseline (Fig. 2(a)) for our backbone since it has superior efficiency and faster convergence [39] than pure transformer-based models [12, 13]. We adjust the UNet into a thinner and shorter model (*i.e.*, reducing the number of transformer blocks from $[0, 2, 10]$ in three stages to $[0, 2, 4]$ and their channel dimensions from $[320, 640, 1280]$ to $[256, 512, 896]$), and iterate design choices on top of it.

Evaluation Metrics. We train models on ImageNet-1K [16] under class-conditional generation for 120 epochs, unless specified otherwise, and report the FID score [25] for 256^2 px generation. Similarly to existing work [30], we inject the class conditions through a text template "a photo of <class name>". We then encode it with a light text encoder to align the pipeline for the T2I generation. We also calculate the number of parameters for different models, floating point operations (FLOPs) (measured on a latent size of 128×128, equivalent to a 1024×1024 image after decoding), and the runtime on mobile device (tested on iPhone 15 Pro). Detailed training and evaluation settings can be found in Supplemental Materials. In the following, we introduce the key architectural changes that improve the model.

Remove Self-Attention from High-Resolution Stages. Self-attention (SA) layer is restricted by its quadratic computational complexity, incurring a heavy computational cost and high memory consumption for high-resolution input. As such, we keep the SA layer only in the lowest-resolution stage while removing it from other higher-resolution stages, *i.e.*, Fig. 2 (b). This leads to 17% fewer FLOPs and 24% latency reduction, as shown in Fig. 3. Interestingly, we even observe a performance improvement, *i.e.*, FID decreased to 3.12 from 3.76. We hypothesize that models with SA in high-resolution stages converge more slowly.

Replace Conv with Expanded Separable Conv. Regular convolution (Conv) is redundant in both parameters and computation. To address this, we replace all Conv layers with the separable convolution (SepConv) [28], composed of a depthwise convolution (DW) followed by a pointwise convolution (PW), as shown in Fig. 2 (c). This replacement reduces parameters by 24% and latency by 62%, but also leads to a performance drop (FID increases from 3.12 to 3.38). To address the issue, we expand the intermediate channels. Specifically, the number of channels after the first PW layer is increased with an expansion ratio, and reduced back to the original number after the second PW layer. The expansion ratio is set to 2 to balance the trade-off between performance, latency, and model parameters. Such a design aligns our residual block with the recently proposed Universal Inverted Bottleneck (UIB) block [55]. As a result, our model achieves 15% fewer parameters, 27% less computation, and 2.4× speedup, while obtaining a lower FID.

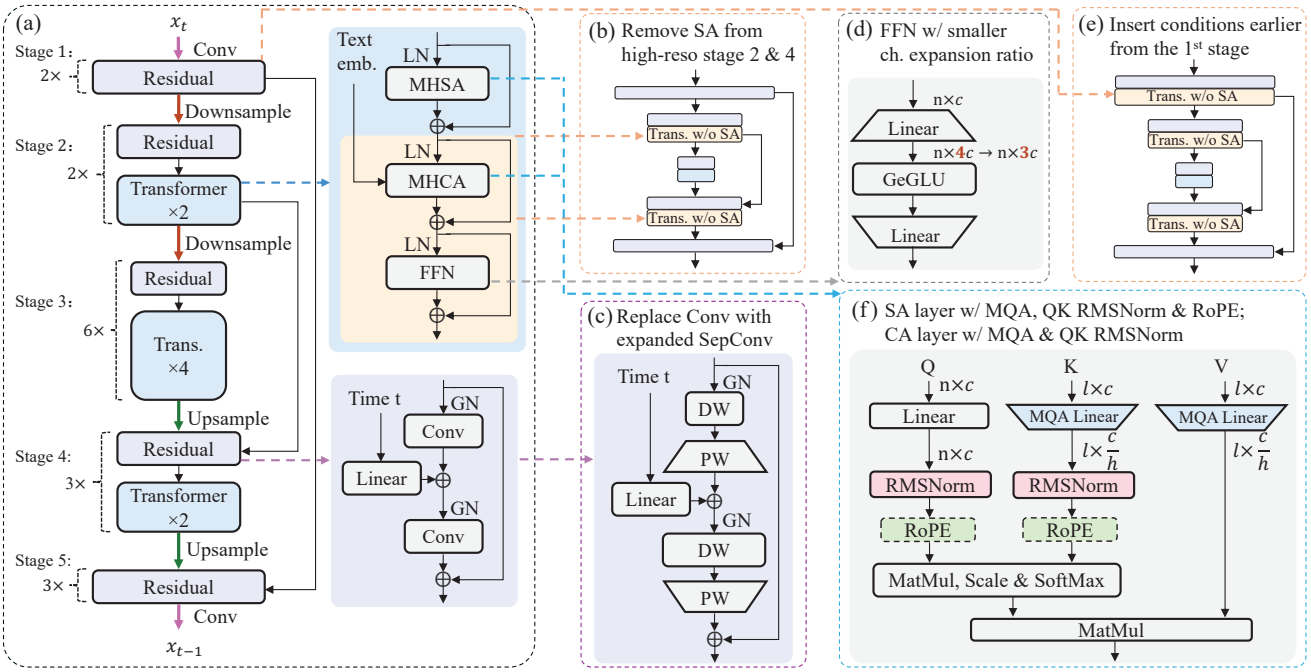

Figure 2. **Efficient UNet.** Starting from a thinner and shorter version of the UNet from SDXL (as in (a)), we explore a series of architectural changes, *i.e.*, (b)–(f), to develop a smaller and faster model while retaining high-quality generation performance, as evaluated in Fig. 3.

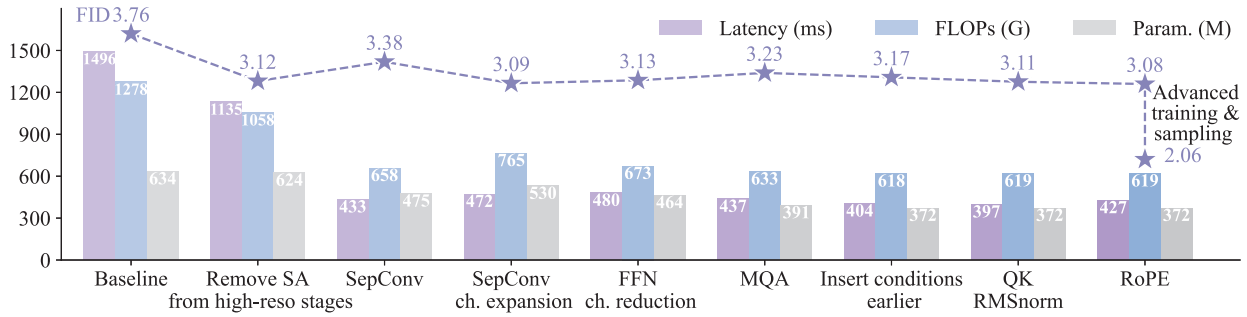

Figure 3. **Comparisons of Performance and Efficiency for various Design Choices of Efficient UNet.** The generation quality is evaluated using FID calculated on ImageNet-1K for 256^2 px generation. The efficiency metrics include model parameters, latency, and FLOPs. FLOPs and latency (on iPhone 15 Pro) are measured with a 128×128 latent, equivalent to a 1024×1024 decoded image, for one forward pass. We show the architecture enhancements that improve any of the metrics without hurting others.

Trim FFN Layers. For the layers in the feed-forward network (FFN), the hidden channel expansion ratio is set to 4 by default and further doubled by using the gated unit. This substantially inflates the model parameters, computation, and memory usage. Following MobileDiffusion [86], we examine the efficacy of simply reducing the expansion ratio, as shown in Fig. 2 (d). We show that reducing the expansion ratio to 3 preserves comparable FID performance while reducing both parameters and FLOPs by 12%.

Replace MHSA with MQA. Multi-Head Self-Attention (MHSA) requires multiple sets of keys and values for each attention head. In contrast, Multi-Query Attention (MQA) [63] is more efficient by sharing a single set of keys and values across all heads. Replacing MHSA with MQA reduces parameters by 16% and latency by 9%, with minimal impact on performance. Interestingly, the 9% saving in latency exceeds the 6% decrease in FLOPs, as the reduced memory access enables higher computational intensity.

Inject Conditions to the First Stage. Cross-attention (CA) blends the conditional information (*e.g.*, textural description) along with the spatial features to generate images that align with the condition. However, the UNet of SDXL only applies CA in transformer blocks starting from the second stage, resulting in the missing conditional guidance for the first stage. In response, we propose to introduce the conditional embeddings from the very first stage, as in Fig. 2(e). Specifically, we replace the residual blocks with transformer blocks that include CA and FFN while without SA layers. This adjustment makes the model smaller, faster, and more efficient while improving FID.

Employ QK RMSNorm and RoPE Positional Embeddings. We extend two advanced techniques developed originally for language models, Query-Key (QK) Normalization [24] with RMSNorm [82] and Rotary Position Embedding (RoPE) [66], to enhance the model (Fig. 2 (f)). RMSNorm, applied after the Query-Key projection in the attention mechanism, reduces the risk of softmax saturation without sacrificing model expressiveness while stabilizing training for faster convergence. In addition, we adapt RoPE from one dimension to two dimensions for better supporting higher resolution since it significantly mitigates artifacts like repeated objects. Together, RMSNorm and RoPE introduce negligible computational and parameter overhead, while offering measurable gains in FID performance.

Discussion. The above optimization results in an efficient and powerful diffusion backbone capable of generating high-resolution images on mobile devices. Before proceeding with large-scale T2I training, we compare the capacity of our model against existing works on ImageNet-1K. We train the model for $1,000$ epochs by following the setting from prior work [58]. We evaluate the model using varied CFG [26, 34] across different inference timesteps. As shown in Tab. 1, our efficient UNet achieves comparable FID to SiT-XL [49], while being almost 45% smaller.

Table 1. **Class-conditional image generation on ImageNet** 256×256 with CFG. FLOPs are calculated for one forward pass.

Model	Param (M)	FLOPs (G)	FID↓
LDM-4 [58]	400	104	3.60
UViT-L [8]	287	77	3.40
UViT-H [8]	501	133	2.29
DiT-XL [52]	675	119	2.27
SiT-XL [49]	675	119	2.06
Ours	372	38	2.06

3.2. Tiny and Fast Decoder

Besides the denoising model, the decoder also takes a significant ratio of total runtime, especially for on-device deployment [40, 86]. Here we introduce a new architecture of decoder (Fig. 4) for efficient high-resolution generation.

Baseline Decoder. We use the autoencoder (AE) from SD3 [18] as our baseline model (*i.e.*, the same encoder from SD3 AE), due to its superior reconstruction quality. The AE maps an image $X \in \mathbb{R}^{H \times W \times 3}$ into a lower-dimensional latent $x \in \mathbb{R}^{\frac{H}{f} \times \frac{W}{f} \times c}$ (f, c as $8, 16$ in SD3). The encoded latent x is then decoded back to an image through a decoder. For high-resolution generation, we observe that the decoder in SD3 is very slow on mobile devices. Specifically, it encounters out-of-memory (OOM) errors when generating a 1024^2 px image on both the ANE processor of the iPhone 15 Pro and the mobile GPU (Tab. 2). To overcome the latency issue, we propose a much smaller and faster decoder.

Efficient Decoder. We conduct a series of experiments to decide the efficient decoder with the following key changes compared with the baseline architecture:

1. We remove attention layers to greatly reduce peak memory without a noticeable impact on decoding quality.
2. We keep a minimal amount of GroupNorm (GN) to find a trade-off between latency and performance (*i.e.*, mitigating the color shifting).
3. We make the decoder thinner (*i.e.*, fewer channels or narrower width) and replace Conv with SepConvs.
4. We use fewer residual blocks in high-resolution stages.
5. We remove the Conv shortcut in residual blocks and use the upsampling layer for channel transition.

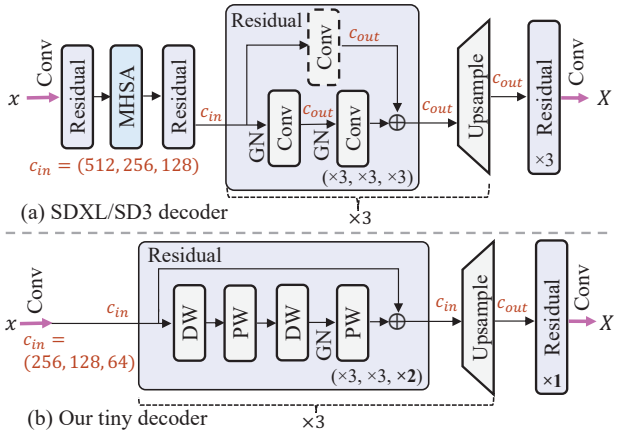

Figure 4. **Comparisons of Decoder Architecture** between (a) SDXL/SD3 decoder and (b) our tiny decoder.

Table 2. **Performance Comparison of Decoder.** PSNR is calculated on COCO 2017 validation set [43]. FLOPs and latency (on iPhone 15 Pro) are measured for decoding a 128×128 latent into a 1024×1024 image. The decoder from SDXL and SD3 fail to run on the neural engine of mobile, resulting in a huge runtime.

Decoder	Ch	PSNR	Param (M)	FLOPs (G)	Latency (ms) on ANE	Latency (ms) on GPU
SDXL [53]	4	24.89	49.49	4970	OOM	9469
SD3 [18]	16	27.92	49.55	4970	OOM	OOM
Ours	16	27.85	1.38	224	174	-

Training of the Decoder. We train our decoder with the mean squared error (MSE) loss, lpips loss [84], adversarial loss [22], and discard the KL term [33] as the encoder is fixed. The decoder is trained on 256^2 image patches with a batch size of 256 and for 1M iterations. As in Tab. 2, our tiny decoder achieves a competitive PSNR score for reconstruction, while being $35.9\times$ *smaller* and $54.4\times$ *faster* for high-resolution generation on mobile devices compared to conventional ones (*e.g.*, the decoder from SDXL and SD3).

Discussion of Total On-Device Latency. We finally measure T2I model latency for a 1024^2 px generation on iPhone 16 Pro-Max. The decoder takes 119ms, and the per-step latency for the UNet is 274ms. This results in a $1.2 \sim 2.3$s runtime for 4 to 8 step generation. Note that text encoder runtime is negligible compared to other components [40].

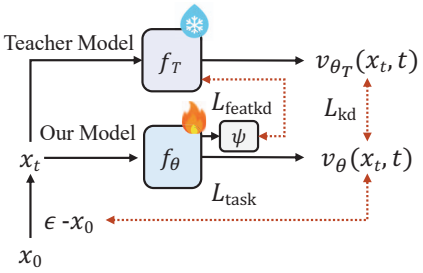

Figure 5. Overview of Multi-level Knowledge Distillation, where we perform output distillation and feature distillation.

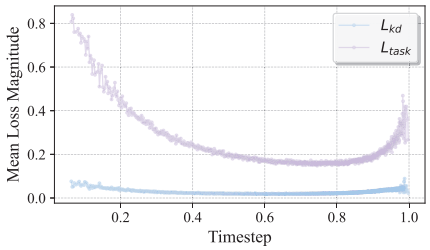

Figure 6. Mean loss magnitude for task loss $\mathcal{L}_{\text{task}}$ and output distillation loss $\mathcal{L}_{\text{kd}}$.

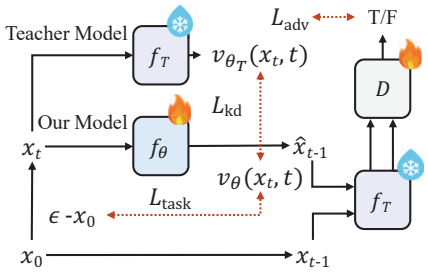

Figure 7. Overview of Adversarial Step Distillation. Output distillation and distribution-matching distillation are performed.

3.3. Training Recipe and Multi-Level Distillation

To improve the generation quality of our efficient diffusion model, we propose a series of training techniques.

Flow-based Training and Inference. Rectified Flows (RFs) [44, 47] define the forward process as straight paths connecting the data distribution to a standard normal distribution, *i.e.*,

$$x_t = (1 - \sigma_t)x_0 + \sigma_t \epsilon, \tag{1}$$

where x_0 is the clean (latent) image, t is the timestep, σ_t is a timestep-dependent factor, and ϵ is random noise sampled from $\mathcal{N}(0, I)$. The denoising UNet is formulated to predict a velocity field with the objective as

$$\mathcal{L}_{\text{task}} = \mathbb{E}_{\epsilon \sim \mathcal{N}(0,I), t}\Big[||(\epsilon - x_0) - v_\theta(x_t, t)||_2^2\Big], \tag{2}$$

where $v_\theta(x_t, t)$ is the predicted velocity from UNet parameterized by θ. To further enhance training stability, we apply logit-normal sampling [18] for the timestep during training, which assigns more samples to the intermediate steps. In the inference stage, we use the Flow-Euler sampler [19], which predicts the next sample based on the velocity, *i.e.*,

$$x_{t-1} = x_t + (\sigma_{t-1} - \sigma_t) \cdot v_\theta(x_t, t). \tag{3}$$

To achieve a lower signal-to-noise ratio on high-resolution (*i.e.*, 1024^2 px) images, we apply a timestep shift similar to SD3 [18] to adjust the scheduling factor σ_t during both training and inference.

Multi-Level Knowledge Distillation. To improve the generation quality for compact models, one common practice for previous works is applying knowledge distillation to mimic the prediction of scaled-up teacher models [32]. Benefiting from the aligned flow matching objective and (AE) latent space, powerful SD3.5-Large model [4] can be used as the teacher for output distillation. However, we still face challenges due to 1) the heterogeneous architecture between the U-Net and DiT, 2) the scale difference between the distillation loss and task loss, and 3) varying prediction difficulty across different timesteps. To tackle these, we propose a novel multi-level distillation loss and apply timestep-aware scaling to stabilize and accelerate the convergence of distillation. The overview of our knowledge-distillation scheme is shown in Fig. 5 and detailed techniques are elaborated as follows.

Aside from the task loss defined in Eq. 2, the major objective for knowledge distillation is to supervise our model θ directly with the output of teacher model θ_T, which can be indicated as

$$\mathcal{L}_{\text{kd}} = \mathbb{E}\Big[||v_{\theta_T}(x_t, t) - v_\theta(x_t, t)||_2^2\Big]. \tag{4}$$

Given the capacity gap between the teacher and our model, applying output-level supervision alone leads to instability and slow convergence. Therefore, we further incorporate a cross-architecture feature-level distillation loss as

$$\mathcal{L}_{\text{featkd}} = \mathbb{E}\Big[\sum_{(l_T, l)} ||f_{\theta_T}^{l_T}(x_t, t) - \psi(f_\theta^l(x_t, t))||_2^2\Big], \tag{5}$$

where $f_{\theta_T}^{l_T}(\cdot)$ and $f^l(\cdot)$ indicate the feature output from the l_T-th layer and l-th layer in teacher model and student model, respectively. Different from previous work [32, 46], we consider cross-architecture distillation from a DiT to UNet. Since the richest information in transformers sits around the last layer, we set the distillation target to this layer in both models, and use a lightweight trainable projector $\psi(\cdot)$ with only two Conv layers to map the student feature to match the dimension of teacher feature. The proposed feature-level distillation loss provides additional supervision to the student model, leading to faster alignment to the generation quality of the teacher model.

Timestep-Aware Scaling. Weighting multiple objectives has been a major challenge in knowledge distillation, especially in diffusion models. The overall training objectives from previous works [32, 46, 67] are simple linear combination of multiple loss term, *i.e.*,

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda_1 \mathcal{L}_{\text{kd}} + \lambda_2 \mathcal{L}_{\text{featkd}}, \tag{6}$$

where the weighting coefficient λ_1 and λ_2 are empirically set to constant. However, this baseline setting fails to consider the *prediction difficulty* in various time steps. We investigate the distribution of empirical risk magnitude of $\mathcal{L}_{\text{task}}$ and $\mathcal{L}_{\text{kd}}$ across different timestep t during model training. Fig. 6 illustrates that, in intermediate steps, *prediction difficulty* are lower compared to t closer to 0 or 1.

Building on this important observation, we propose a timestep-aware scaling of the objective to close the gap in loss magnitude across different values of t and to account for prediction difficulties at each timestep, as follows:

$$\mathcal{S}(\mathcal{L}_{\text{task}}, \mathcal{L}_{\text{kd}}) = \mathbb{E}_t\Big[\lambda(t)\cdot\mathcal{L}^t_{\text{task}} + \big(1-\lambda(t)\big)\frac{|\mathcal{L}^t_{\text{task}}|}{|\mathcal{L}^t_{\text{kd}}|}\cdot\mathcal{L}^t_{\text{kd}}\Big], \tag{7}$$

where $\lambda(t)$ is the normalized standard (location 0, scale 1) logit-norm density function and $|\cdot|$ indicates the magnitude. In $\mathcal{S}$, we first ensure the same scale between task loss and distillation loss across different t, then apply more teacher supervision where *prediction difficulty* is higher (*i.e.*, t closer to 0 or 1), and more real data supervision where *prediction difficulty* is lower (*i.e.*, intermediate timesteps). The proposed scheme considers the variation of timestep t and helps accelerate the distillation training. The final multi-level distillation objective $\mathcal{L}_{\text{md}}$ is defined as

$$\mathcal{L}_{\text{md}} = \mathcal{S}(\mathcal{L}_{\text{task}}, \mathcal{L}_{\text{kd}}) + \mathcal{S}(\mathcal{L}_{\text{task}}, \mathcal{L}_{\text{featkd}}). \tag{8}$$

3.4. Step Distillation

We take one step further to enhance the sampling efficiency of our model following a distribution-matching-based step distillation scheme. Following Latent Adversarial Diffusion Distillation (LADD) [60], we use a diffusion-GAN hybrid structure to distill our model into fewer steps with the optimization objective as

$$\begin{aligned}\min_{D_{\theta_T}}\max_{G_\theta}\mathbb{E}\Big[&[\log(D_{\theta_T}(x_{t-1}, t))] \\ &+ [\log(1 - D_{\theta_T}(x'_{t-1}, t))] - \mathcal{S}(\mathcal{L}_{\text{task}}, \mathcal{L}_{\text{kd}})\Big],\end{aligned} \tag{9}$$

where D_{θ_T} is the discriminator model partially initialized with pretrained fewer-step teacher model θ_T (SD3.5-Large-Turbo [5]). The large-scale teacher model are only used as the feature extractor and are frozen during distillation. We only train a few linear layers in the discriminator after feature extraction. We sample $x_{t-1} \sim q(x_{t-1}|x_0)$ and $x'_{t-1} \sim q(x_{t-1}|x'_0)$, where x'_0 is the prediction of our denoising generator[1] $G_\theta(x_t, t)$ as our student model, and $q(x)$ is the forward process of diffusion model defined in Eq. 1. The objective consists of an adversarial loss to match noisy samples at time step $t-1$ and the output-level distillation loss $\mathcal{S}(\mathcal{L}_{\text{task}}, \mathcal{L}_{\text{kd}})$ after applying timestep-aware scaling. The proposed step distillation, visualized in Fig. 7, can be interpreted as training a diffusion model with adversarial refinement and knowledge distillation, where teacher guidance serves as an additional inductive bias. This advanced step distillation empowers our compact model for high-quality generation, with only a few denoising steps.

[1]For simplicity, we use the x_0 prediction in our derivation, and v prediction of rectified flow-based training would not break the formulation.

4. Experiments

Model Details. Our T2I pipeline consists of the efficient UNet (Sec. 3.1) and the efficient encoder-decoder model (Sec. 3.2). To obtain text embeddings from the input prompt, we leverage multiple text encoders, namely lightweight CLIP-L [56], CLIP-G [56], and the large Gemma-2-2b language model [68]. We follow SD3 [18] strategy to combine these three text-encoders into a unified rich textual embedding. To enable classifier-free guidance [27], we employ these embeddings with an individual drop-out probability such that we can use an arbitrary subset of the encoders during inference. This allows us to deploy one or more encoders based on the resource constraints.

Training Recipe. Similar as prior work [30], we use a multi-stage strategy to train our UNet model from scratch. First, we pre-train the model using the ImageNet-1K [16] at 256 resolution as described in Sec. 3.1. Second, we fine-tune this model in a progressive manner from $256 \rightarrow 512 \rightarrow 1024$ resolutions. Third, we employ knowledge distillation with our timestep-aware scaling (Sec. 3.3) to improve the finer details in our models using a much larger teacher model (SD3.5-Large [4]) and all three text-encoders. Before KD, we initialize the student model by pre-training on ImageNet and T2I datasets, a cost-effective approach that helps the model learn pixel distributions. Finally, we obtain a few-step model through step distillation using SD3.5-Large-Turbo [5] model as the teacher. We optimize the rectified-flow [18] objective using AdamW optimizer [17] to train our UNet backbone.

Hyper-parameters. We sample the timesteps using the logit-normal distribution with $(0, 1)$ as the location and scale parameters. We use a time shift of 3 for both training and inference. See the Supplemental Materials for details.

4.1. Evaluation

Quantitative Benchmarks. We use GenEval [21] and DPG-Bench [29] benchmarks to evaluate the text-to-image alignment of our model on short and long prompts, respectively. We report CLIP score on a 6K subset of MS-COCO validation data [43]. In addition, to measure the aesthetic quality of our model, we compute the Image Reward [75] score on selected PixArt prompts [12]. Tab. 3 lists our performance alongside existing state-of-the-art T2I baselines. We provide additional details in Supplemental Materials. We highlight the salient observations below:

- Our 0.38B parameter model achieves even better performance than significantly larger models such as SDXL (2.6B), Playground (2.6B), and IF-XL (5.5B).
- KD non-trivially improves the prompt following ability of the base model as illustrated by an absolute five-point increase in DPG-Bench and GenEval scores.
- In terms of aesthetic performance, our model has similar Image Reward scores as Playground models [37, 38].

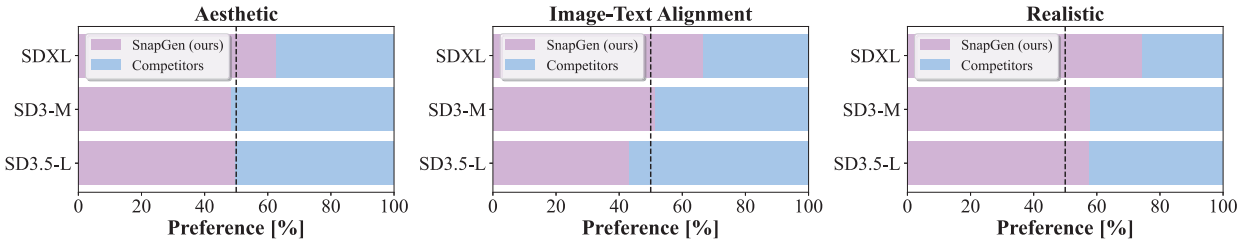

Figure 8. **Human Evaluation.** We conduct a user study to compare images generated by our model against baselines on three attributes: aesthetic quality, text-image alignment, and realistic generations. Our model surpasses the quality of SDXL and SD3 models, while performing competitively against the teacher SD3.5-Large model.

Table 3. **Evaluation on Quantitative Benchmarks.** We list the scores on GenEval, DPG-Bench, CLIP score on COCO, and Image Reward on aesthetic prompts. We report the parameters for the UNet/DiT backbone in the Param column. Throughput (samples/s) is measured on a single 80GB A100 GPU using the largest batch size supported for each model in a practical scenario to generate 1024^2 px images. Here our sampling step is set to be 28.

Model	Param	Throughput	GenEval ↑	DPG ↑	CLIP ↑	Image Reward ↑
PixArt-α [12]	0.6B	0.42	0.48	71.1	0.316	1.15
PixArt-Σ [13]	0.6B	0.46	0.53	80.5	0.317	1.13
SD-1.5 [1]	0.9B	-	0.43	63.2	0.287	0.19
SD-2.1 [2]	0.9B	-	0.50	64.2	0.281	0.29
KOALA [35]	1.0B	0.66	0.52	74.3	0.317	1.05
MicroDiT [62]	1.2B	-	0.52	75.1	0.318	1.25
Sana [74]	1.6B	1.00	**0.66**	**84.8**	0.327	1.25
LUMINA-Next [87]	2.0B	0.06	0.46	74.6	0.309	0.88
SDXL [53]	2.6B	0.18	0.55	74.7	0.301	0.99
Playgroundv2 [37]	2.6B	0.18	0.59	74.5	0.317	1.25
Playgroundv2.5 [38]	2.6B	0.18	0.56	75.5	0.319	**1.34**
IF-XL [15]	5.5B	0.06	0.61	75.6	0.311	0.65
Ours w/o KD	0.38B	1.04	0.61	76.3	0.321	1.20
SnapGen (ours)	0.38B	1.04	**0.66**	81.1	**0.332**	1.32

Qualitative Comparison. To visually evaluate the image-text alignment and aesthetics, we compare the generated images from different T2I models in Fig. 1. We observe that many existing models fail to fully capture the full prompt and miss important elements. Further, human generations often result in smoothened-out faces, leading to the loss of details. In contrast, our model generates much more photo-realistic images with better image-text alignment.

Human Evaluation. For a thorough comparison between baselines, we perform a user study with the widely used Parti prompts [81]. We use SDXL, SD3-M, and SD3.5-Large models as the baselines and generate images using this prompt set. We ask the users to select images with better attributes between the baselines and our model. These attributes include image-text alignment, aesthetic quality, and realistic images. Fig. 8 shows that our model convincingly outperforms SDXL on all three attributes. Our model beats SD3 on image-text alignment and realistic generations, with a tie on aesthetic quality. Compared to the teacher (SD3.5-Large), we lag the teacher a bit behind on the image-text alignment, yet our model still has better realistic generations, and yields a toss-up on aesthetic quality. With this study, we can conclude our efficient T2I pipeline achieves generation quality that is quite comparable to the SD3.5-Large teacher that has 8.1B parameters.

Few-Step Generation. After step-distillation (Sec. 3.4), our model can generate high-quality images within a few steps. Fig. 9 compares our model's performance before and after step-distillation, with respective GenEval scores. The results demonstrate that our model, after step distillation, achieves comparable performance to the baseline model with 28 steps, even when using only 4 or 8 steps. While the few-step generation shows slight qualitative degradation compared to the 28-step baseline, it still outperforms most existing T2I models with significantly more inference steps, such as SDXL (50 steps) and PixArt-α (100 steps).

Figure 9. Performance comparison of few-step generation for our model before (top) and after step distillation (bottom).

5. Conclusion

In this work, we propose a novel and efficient T2I model for high-resolution generation on mobile phones. We systematically detail the process to obtain a tiny 379M parameter UNet architecture along with an efficient latent decoder. We devise a novel training method consisting of multi-stage pre-training followed by knowledge distillation from a large teacher and adversarial step distillation. With these, we achieve an extremely efficient T2I model that comprehensively outperforms many existing multi-billion parameter models such as SDXL, Lumina-Next, and Playgroundv2.

References

[1] Stability AI. Stable diffusion 1.5. *https://huggingface.co/stable-diffusion-v1-5/stable-diffusion-v1-5*, 2022. 8

[2] Stability AI. Stable diffusion 2.1. *https://huggingface.co/stabilityai/stable-diffusion-2-1*, 2022. 8

[3] Stability AI. Stable diffusion 3.5. *https://github.com/Stability-AI/sd3.5*, 2024. 2

[4] Stability AI. Stable diffusion 3.5 large. *https://huggingface.co/stabilityai/stable-diffusion-3.5-large*, 2024. 6, 7

[5] Stability AI. Stable diffusion 3.5 large turbo. *https://huggingface.co/stabilityai/stable-diffusion-3.5-large-turbo*, 2024. 2, 7

[6] Vladimir Arkhipkin, Viacheslav Vasilev, Andrei Filatov, Igor Pavlov, Julia Agafonova, Nikolai Gerasimenko, Anna Averchenkova, Evelina Mironova, Anton Bukashkin, Konstantin Kulikov, et al. Kandinsky 3: Text-to-image synthesis for multifunctional generative framework. *arXiv preprint arXiv:2410.21061*, 2024. 3

[7] Shakhmatov Arseni, Razzhigaev Anton, Nikolich Aleksandr, Arkhipkin Vladimir, Pavlov Igor, Kuznetsov Andrey, and Denis Dimitrov. Kandinsky 2.1, 2023. 3

[8] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22669–22679, 2023. 5

[9] Han Cai, Junyan Li, Muyan Hu, Chuang Gan, and Song Han. Efficientvit: Lightweight multi-scale attention for high-resolution dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17302–17313, 2023. 3

[10] Thibault Castells, Hyoung-Kyu Song, Bo-Kyeong Kim, and Shinkook Choi. LD-Pruner: Efficient Pruning of Latent Diffusion Models using Task-Agnostic Insights. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 821–830, 2024. 2

[11] Thibault Castells, Hyoung-Kyu Song, Tairen Piao, Shinkook Choi, Bo-Kyeong Kim, Hanyoung Yim, Changgwun Lee, Jae Gon Kim, and Tae-Ho Kim. EdgeFusion: On-Device Text-to-Image Generation. *arXiv preprint arXiv:2404.11925*, 2024. 3

[12] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023. 2, 3, 7, 8

[13] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-\sigma: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. *arXiv preprint arXiv:2403.04692*, 2024. 2, 3, 8

[14] Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060*, 2024. 3

[15] DeepFloyd. Deepfloyd. *https://github.com/deep-floyd/IF*, 2023. 2, 3, 8

[16] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2, 3, 7

[17] P Kingma Diederik. Adam: A method for stochastic optimization. *(No Title)*, 2014. 7

[18] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024. 2, 5, 6, 7

[19] Leonhard Euler. *Institutionum calculi integralis*. imp. Acad. imp. Sa'ent., 1768. 6

[20] Peng Gao, Le Zhuo, Ziyi Lin, Chris Liu, Junsong Chen, Ruoyi Du, Enze Xie, Xu Luo, Longtian Qiu, Yuhang Zhang, et al. Lumina-T2X: Transforming Text into Any Modality, Resolution, and Duration via Flow-based Large Diffusion Transformers. *arXiv preprint arXiv:2405.05945*, 2024. 3

[21] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36, 2024. 7

[22] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 3, 5

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[24] Alex Henry, Prudhvi Raj Dachapally, Shubham Pawar, and Yuxuan Chen. Query-key normalization for transformers. *arXiv preprint arXiv:2010.04245*, 2020. 5

[25] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 3

[26] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 5

[27] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 7

[28] Andrew G Howard. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 3

[29] Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment. *arXiv preprint arXiv:2403.05135*, 2024. 7

[30] Anil Kag, Huseyin Coskun, Jierun Chen, Junli Cao, Willi Menapace, Aliaksandr Siarohin, Sergey Tulyakov, and Jian Ren. Ascan: Asymmetric convolution-attention networks for efficient recognition and generation. *arXiv preprint arXiv:2411.04967*, 2024. 3, 7

[31] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pages 5156–5165. PMLR, 2020. 3

[32] Bo-Kyeong Kim, Hyoung-Kyu Song, Thibault Castells, and Shinkook Choi. Bk-sdm: Architecturally Compressed Stable Diffusion for Efficient Text-to-Image Generation. In *Workshop on Efficient Systems for Foundation Models@ ICML2023*, 2023. 2, 3, 6

[33] Diederik P Kingma. Auto-encoding variational bayes. *2nd International Conference on Learning Representations*, 2014. 5

[34] Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. *arXiv preprint arXiv:2404.07724*, 2024. 5

[35] Youngwan Lee, Kwanyong Park, Yoorhim Cho, Yong-Ju Lee, and Sung Ju Hwang. Koala: Empirical lessons toward memory-efficient and fast diffusion models for text-to-image synthesis, 2023. 8

[36] Daiqing Li, Aleks Kamko, Ali Sabet, Ehsan Akhgari, Linmiao Xu, and Suhail Doshi. Playground v1, . 3

[37] Daiqing Li, Aleks Kamko, Ali Sabet, Ehsan Akhgari, Linmiao Xu, and Suhail Doshi. Playground v2, . 7, 8

[38] 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. 3, 7, 8

[39] Hao Li, Yang Zou, Ying Wang, Orchid Majumder, Yusheng Xie, R Manmatha, Ashwin Swaminathan, Zhuowen Tu, Stefano Ermon, and Stefano Soatto. On the scalability of diffusion-based text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9400–9409, 2024. 3

[40] Yanyu Li, Huan Wang, Qing Jin, Ju Hu, Pavlo Chemerys, Yun Fu, Yanzhi Wang, Sergey Tulyakov, and Jian Ren. Snapfusion: Text-to-Image Diffusion Model on Mobile Devices within Two Seconds. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3, 5

[41] 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. 3

[42] Shanchuan Lin, Anran Wang, and Xiao Yang. SDXL-Lightning: Progressive Adversarial Diffusion Distillation, 2024. 2

[43] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 5, 7

[44] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. 2, 6

[45] Bingchen Liu, Ehsan Akhgari, Alexander Visheratin, Aleks Kamko, Linmiao Xu, Shivam Shrirao, Joao Souza, Suhail Doshi, and Daiqing Li. Playground v3: Improving text-to-image alignment with deep-fusion large language models. *arXiv preprint arXiv:2409.10695*, 2024. 3

[46] Songhua Liu, Weihao Yu, Zhenxiong Tan, and Xinchao Wang. Linfusion: 1 gpu, 1 minute, 16k image. *arXiv preprint arXiv:2409.02097*, 2024. 2, 3, 6

[47] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022. 2, 6

[48] Xian Liu, Jian Ren, Aliaksandr Siarohin, Ivan Skorokhodov, Yanyu Li, Dahua Lin, Xihui Liu, Ziwei Liu, and Sergey Tulyakov. Hyperhuman: Hyper-realistic human generation with latent structural diffusion. *arXiv preprint arXiv:2310.08579*, 2023. 2, 3

[49] 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. *arXiv preprint arXiv:2401.08740*, 2024. 2, 5

[50] Willi Menapace, Aliaksandr Siarohin, Ivan Skorokhodov, Ekaterina Deyneka, Tsai-Shien Chen, Anil Kag, Yuwei Fang, Aleksei Stoliar, Elisa Ricci, Jian Ren, et al. Snap video: Scaled spatiotemporal transformers for text-to-video synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7038–7048, 2024. 2

[51] OpenAI. 2

[52] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 5

[53] 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. 2, 3, 5, 8

[54] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 2

[55] Danfeng Qin, Chas Leichner, Manolis Delakis, Marco Fornoni, Shixin Luo, Fan Yang, Weijun Wang, Colby Banbury, Chengxi Ye, Berkin Akin, et al. Mobilenetv4-universal models for the mobile ecosystem. *arXiv preprint arXiv:2404.10518*, 2024. 3

[56] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 7

[57] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 2

[58] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 3, 5

[59] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. 2

[60] Axel Sauer, Frederic Boesel, Tim Dockhorn, Andreas Blattmann, Patrick Esser, and Robin Rombach. Fast high-resolution image synthesis with latent adversarial diffusion distillation. *arXiv preprint arXiv:2403.12015*, 2024. 3, 7

[61] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pages 87–103. Springer, 2024. 3

[62] Vikash Sehwag, Xianghao Kong, Jingtao Li, Michael Spranger, and Lingjuan Lyu. Stretching each dollar: Diffusion training from scratch on a micro-budget. *arXiv preprint arXiv:2407.15811*, 2024. 8

[63] Noam Shazeer. Fast transformer decoding: One write-head is all you need. *arXiv preprint arXiv:1911.02150*, 2019. 4

[64] Shelly Sheynin, Adam Polyak, Uriel Singer, Yuval Kirstain, Amit Zohar, Oron Ashual, Devi Parikh, and Yaniv Taigman. Emu edit: Precise image editing via recognition and generation tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8871–8879, 2024. 2

[65] Yuda Song, Zehao Sun, and Xuanwu Yin. Sdxs: Real-time one-step latent diffusion models with image conditions. *arXiv preprint arXiv:2403.16627*, 2024. 2

[66] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. 5

[67] Yang Sui, Yanyu Li, Anil Kag, Yerlan Idelbayev, Junli Cao, Ju Hu, Dhritiman Sagar, Bo Yuan, Sergey Tulyakov, and Jian Ren. Bitsfusion: 1.99 bits weight quantization of diffusion model. *arXiv preprint arXiv:2406.04333*, 2024. 2, 3, 6

[68] Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024. 7

[69] Kolors Team. Kolors: Effective Training of Diffusion Model for Photorealistic Text-to-Image Synthesis. *arXiv preprint*, 2024. 3

[70] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 3

[71] Zhendong Wang, Huangjie Zheng, Pengcheng He, Weizhu Chen, and Mingyuan Zhou. Diffusion-gan: Training gans with diffusion. *arXiv preprint arXiv:2206.02262*, 2022. 3

[72] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. *arXiv preprint arXiv:2409.11340*, 2024. 2

[73] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion gans. *arXiv preprint arXiv:2112.07804*, 2021. 3

[74] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Yujun Lin, Zhekai Zhang, Muyang Li, Yao Lu, and Song Han. Sana: Efficient high-resolution image synthesis with linear diffusion transformers. *arXiv preprint arXiv:2410.10629*, 2024. 2, 3, 8

[75] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36, 2024. 7

[76] Yanwu Xu, Mingming Gong, Shaoan Xie, Wei Wei, Matthias Grundmann, Kayhan Batmanghelich, and Tingbo Hou. Semi-implicit denoising diffusion models (siddms). *Advances in neural information processing systems*, 36:17383, 2023. 3

[77] Yanwu Xu, Yang Zhao, Zhisheng Xiao, and Tingbo Hou. Ufogen: You forward once large scale text-to-image generation via diffusion gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8196–8206, 2024. 2, 3

[78] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 2

[79] Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *arXiv preprint arXiv:2405.14867*, 2024. 3

[80] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6613–6623, 2024. 2

[81] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *Transactions on Machine Learning Research*. 8

[82] Biao Zhang and Rico Sennrich. Root mean square layer normalization. *Advances in Neural Information Processing Systems*, 32, 2019. 5

[83] Dingkun Zhang, Sijia Li, Chen Chen, Qingsong Xie, and Haonan Lu. Laptop-Diff: Layer Pruning and Normalized Distillation for Compressing Diffusion Models. *arXiv preprint arXiv:2404.11098*, 2024. 2

[84] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 5

[85] Zhixing Zhang, Ligong Han, Arnab Ghosh, Dimitris N Metaxas, and Jian Ren. Sine: Single image editing with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6027–6037, 2023. 2

[86] Yang Zhao, Yanwu Xu, Zhisheng Xiao, and Tingbo Hou. Mobilediffusion: Subsecond text-to-image generation on mobile devices. *arXiv preprint arXiv:2311.16567*, 2023. 2, 3, 4, 5

[87] 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. 2, 8