StepDrop: Accelerating Tiny Diffusion Models with Stochastic Step Skipping

Wanghley Soares Martins
Department of Electrical and Computer Engineering
Duke University
wanghley.martins@duke.edu

Nicolas Vasilescu
Department of Electrical and Computer Engineering
Duke University

Logan Chu
Department of Electrical and Computer Engineering
Duke University

Abstract

Diffusion models achieve state-of-the-art image synthesis but suffer from high inference latency due to their sequential, iterative denoising process—a critical barrier for deployment on resource-constrained edge devices. We observe that diffusion steps contribute unevenly to generation quality: early steps establish coarse structure, late steps refine texture, while middle steps provide minimal perceptual value. Based on this insight, we propose **StepDrop**, a training-free, plug-and-play acceleration method that stochastically skips low-importance denoising steps. StepDrop employs importance-weighted timestep selection strategies—including uniform, quadratic, cosine, and adaptive schedulers—to concentrate computation on high-value steps while probabilistically omitting redundant mid-trajectory updates. Evaluated on CIFAR-10 with a tiny U-Net diffusion model, StepDrop achieves a 50% reduction in computational cost (59→29.5 GFLOPs) while *improving* FID by 2.5 points over DDIM at equivalent NFE budgets. At 50 function evaluations, StepDrop attains FID 76.05 compared to DDIM's 77.20; at 25 NFE, it matches DDIM-50 quality while doubling throughput. Residual analysis confirms strong agreement with full DDIM trajectories, demonstrating robustness to mid-phase skipping. Our results establish that focusing computation on temporally important steps yields superior quality-per-FLOP, enabling practical diffusion model deployment with flexible compute-memory tradeoffs. Code is available at `https://github.com/wanghley/stepdrop-tiny-diffusion`.

1 Introduction

Generative modeling has witnessed remarkable progress over the past decade, with diffusion models emerging as a dominant paradigm for high-fidelity image synthesis. Unlike Generative Adversarial Networks (GANs), which rely on adversarial training dynamics prone to mode collapse, or Variational Autoencoders (VAEs), which often produce blurry outputs, diffusion models achieve state-of-the-art generation quality through a principled probabilistic framework based on iterative denoising [Ho et al., 2020, Song et al., 2020].

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

The core mechanism of diffusion models involves two processes: a *forward process* that gradually corrupts data by adding Gaussian noise over many timesteps, and a *reverse process* that learns to denoise, progressively recovering structure from pure noise. This formulation enables stable training and produces samples of exceptional quality, powering applications from text-to-image generation to drug discovery and audio synthesis.

However, the iterative nature of diffusion sampling comes at a significant computational cost. The standard Denoising Diffusion Probabilistic Model (DDPM) [Ho et al., 2020] requires up to 1000 sequential forward passes through a neural network to generate a single sample. Each pass involves a full evaluation of the denoising network—typically a U-Net architecture [Ronneberger et al., 2015] with millions of parameters—making inference orders of magnitude slower than single-pass generators like GANs. This computational burden poses a fundamental barrier to deploying diffusion models in latency-sensitive applications, real-time systems, and resource-constrained environments such as mobile devices and embedded platforms.

The challenge of accelerating diffusion inference has motivated substantial research. Denoising Diffusion Implicit Models (DDIM) [Song et al., 2020] reformulate the reverse process as a deterministic, non-Markovian transformation, enabling generation with significantly fewer steps by using uniformly-spaced timestep subsequences. Advanced numerical solvers such as DPM-Solver [Lu et al., 2022] and PNDM [Liu et al., 2022] further reduce the required number of function evaluations (NFE) through higher-order integration schemes. Knowledge distillation approaches [Zhang and Ma, 2024, Zhu et al., 2024] compress the multi-step process into fewer or even single-step generators, though often at the cost of additional training and potential quality degradation.

Despite these advances, most acceleration techniques share a common assumption: that all timesteps in the diffusion trajectory contribute equally to generation quality. DDIM and similar methods select timesteps uniformly across the trajectory, allocating equal computational resources to early, middle, and late denoising stages. Yet there is growing evidence that this assumption may be suboptimal [Wang and Li, 2024, Xue et al., 2024]. Intuitively, different phases of the denoising process serve distinct purposes: early steps (high noise levels) establish coarse semantic content and global structure, late steps (low noise levels) refine fine-grained details and textures, while intermediate steps may primarily serve to smoothly interpolate between these regimes.

This observation raises a natural question: *if timesteps contribute unevenly to perceptual quality, can we design sampling strategies that focus computation on high-importance steps while reducing effort on redundant ones?* Such an approach would move beyond uniform step allocation toward importance-weighted sampling, potentially achieving better quality-efficiency tradeoffs.

In this work, we investigate this question in the context of *tiny diffusion models*—lightweight architectures with fewer than 10 million parameters designed for efficient deployment. These models are particularly relevant for edge computing and mobile applications, where computational and memory constraints are paramount. We hypothesize that tiny models, due to their limited capacity, may exhibit even more pronounced temporal redundancy, making them ideal candidates for importance-aware acceleration.

We propose **StepDrop** [**?**], a training-free acceleration framework that employs stochastic step skipping with importance-weighted selection. Rather than uniformly subsampling timesteps, StepDrop uses configurable skip schedules—including uniform, quadratic, cosine, and adaptive variants—that encode different priors about timestep importance. The stochastic formulation allows flexible compute-quality tradeoffs while the importance weighting concentrates computation on high-value denoising stages.

The remainder of this paper is organized as follows. Section 2 formalizes our research objectives. Section 3 reviews related work on diffusion models and acceleration techniques. Section 4 presents the StepDrop methodology, including the mathematical formulation and skip schedule variants. Section 5 describes our experimental setup, and Section 6 presents quantitative and qualitative results. Finally, Section 7 discusses implications and future directions.

2 Objectives

The primary goal of this work is to investigate whether stochastic step skipping can accelerate tiny diffusion models while preserving—or even improving—generation quality compared to standard deterministic sampling methods. We structure our investigation around the following objectives:

Train a Baseline Tiny Diffusion Model. We first establish a reference point by training a lightweight U-Net-based diffusion model on CIFAR-10 [Krizhevsky and Hinton, 2009] using standard DDPM training procedures [Ho et al., 2020]. This baseline serves as the foundation for all subsequent experiments and provides reference metrics for image quality and computational cost.

Implement Stochastic and Adaptive Step-Skip Schedulers. We develop a family of importance-weighted skip schedules that encode different hypotheses about timestep importance. These include: (1) *uniform* skipping, where steps are omitted with equal probability across the trajectory; (2) *quadratic* skipping, which concentrates omissions in the middle of the trajectory; (3) *cosine* skipping, which provides smooth, bell-shaped skip probabilities centered on mid-trajectory steps; and (4) *adaptive* skipping, which dynamically selects steps based on predicted noise magnitude.

Enable Probabilistic Omission of Denoising Steps. We integrate the skip schedulers into the DDIM sampling loop [Song et al., 2020], allowing steps to be probabilistically omitted during inference. Critical boundary steps (early and late timesteps) are preserved to maintain semantic structure and fine details, while mid-trajectory steps are candidates for stochastic skipping.

Compare Stochastic vs. Deterministic Step Reduction. We systematically compare StepDrop's stochastic approach against DDIM's deterministic uniform step reduction across various computational budgets (number of function evaluations). This comparison isolates the effect of importance-weighted selection from simple step count reduction.

Evaluate Using Standard Image Quality Metrics. We assess generation quality using Fréchet Inception Distance (FID) [Heusel et al., 2017] to measure distributional similarity to real images, and Inception Score (IS) [Salimans et al., 2016] to evaluate both image quality and diversity. Additionally, we measure computational cost in terms of GFLOPs and wall-clock inference time to quantify the efficiency gains.

The central research question guiding this work is: *Can importance-weighted stochastic step skipping reduce computational cost in tiny diffusion models while maintaining or improving image quality compared to uniform timestep selection?*

3 Related Work

Diffusion Models. Diffusion probabilistic models were introduced by Ho et al. [2020], who demonstrated that iterative denoising could produce high-quality samples competitive with GANs. The DDPM framework defines a forward Markov chain that progressively adds Gaussian noise to data, and a reverse chain parameterized by a neural network that learns to denoise. While effective, DDPM requires hundreds to thousands of function evaluations for sampling, limiting practical deployment.

Accelerated Sampling. Song et al. [2020] proposed DDIM, which reformulates diffusion sampling as a deterministic process, enabling the use of non-Markovian subsequences with fewer steps. This insight sparked numerous follow-up works on efficient sampling. DPM-Solver [Lu et al., 2022] applies high-order ODE solvers to accelerate the reverse diffusion process, achieving quality results in 10–20 steps. PNDM [Liu et al., 2022] introduces pseudo-numerical methods tailored for the diffusion manifold. These methods focus on improving the numerical integration but still treat all timesteps uniformly.

Timestep Importance and Selection. Recent work has begun to question the uniform treatment of timesteps. Xue et al. [2024] optimize timestep selection for diffusion sampling, showing that non-uniform schedules can improve quality at fixed budgets. Wang and Li [2024] propose Skip-Step Diffusion Models (S2-DMs), which learn to skip steps during training. Our work differs by focusing

on *training-free* acceleration through stochastic skipping at inference time, making it applicable to any pretrained model without modification.

Knowledge Distillation. An orthogonal approach to acceleration involves distilling multi-step diffusion models into fewer-step generators. Zhang and Ma [2024] propose one-to-many knowledge distillation for diffusion acceleration, while Zhu et al. [2024] introduce SlimFlow for training compact one-step models. These methods require additional training and may sacrifice flexibility, whereas StepDrop operates purely at inference time.

4 Method

4.1 Background: Diffusion Models and DDIM Sampling

Diffusion models define a forward process that gradually adds noise to data $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ over T timesteps:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}), \tag{1}$$

where $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$ and $\alpha_t = 1 - \beta_t$ for a noise schedule $\{\beta_t\}_{t=1}^{T}$.

The reverse process learns to denoise by predicting the noise $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$ added at each step. DDPM [Ho et al., 2020] uses a stochastic reverse process, while DDIM [Song et al., 2020] derives a deterministic update:

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{\mathbf{x}}_0(\mathbf{x}_t, t) + \sqrt{1-\bar{\alpha}_{t-1}} \cdot \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t), \tag{2}$$

where $\hat{\mathbf{x}}_0(\mathbf{x}_t, t) = \frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}$ is the predicted clean image.

DDIM enables sampling with a subsequence $\tau = \{\tau_1, \tau_2, \ldots, \tau_S\} \subset \{1, \ldots, T\}$ of $S < T$ steps, typically chosen uniformly.

4.2 StepDrop: Importance-Weighted Step Skipping

We propose StepDrop, which modifies the DDIM sampling process by stochastically skipping steps based on their estimated importance. The key insight is that not all timesteps contribute equally to generation quality—boundary steps (early and late) are critical, while mid-trajectory steps are often redundant.

Given a base timestep sequence $\tau = \{\tau_1, \ldots, \tau_S\}$, StepDrop assigns a skip probability $p_{\text{skip}}(\tau_i)$ to each step. At inference time, step τ_i is skipped with probability $p_{\text{skip}}(\tau_i)$, in which case the sample propagates directly to the next non-skipped step.

Skip Schedules. We investigate four skip probability functions:

1. **Uniform:** $p_{\text{skip}}(i) = p_0$, constant across all steps.
2. **Quadratic:** $p_{\text{skip}}(i) = p_0 \cdot 4\left(\frac{i}{S} - \frac{1}{2}\right)^2$, concentrating skips at mid-trajectory.
3. **Cosine:** $p_{\text{skip}}(i) = p_0 \cdot \frac{1}{2}\left(1 + \cos\left(\pi \cdot \frac{|i - S/2|}{S/2}\right)\right)$, smooth bell-shaped distribution.
4. **Adaptive:** $p_{\text{skip}}(i) \propto \|\boldsymbol{\epsilon}_\theta(\mathbf{x}_{\tau_i}, \tau_i)\|^{-1}$, skipping steps where predicted noise magnitude is low.

Boundary Preservation. To ensure structural integrity, we enforce $p_{\text{skip}}(\tau_i) = 0$ for the first k and last k steps, preserving critical boundary regions where coarse structure and fine details are established.

4.3 Algorithm

Algorithm 4.3 summarizes the StepDrop sampling procedure.

Algorithm 1: StepDrop Sampling

Input: Noise $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$, timesteps τ, model $\boldsymbol{\epsilon}_\theta$, schedule type
Output: Generated sample $\mathbf{x}_0$

Compute skip probabilities $\{p_{\text{skip}}(\tau_i)\}$ based on schedule
for $i = S, S-1, \ldots, 1$ **do**
  Sample $u \sim \text{Uniform}(0, 1)$
  if $u < p_{\text{skip}}(\tau_i)$ and $i \notin$ boundary **then**
    Skip step (propagate $\mathbf{x}_{\tau_i}$ directly)
  else
    Apply DDIM update (Eq. 2)
  end if
end for
return $\mathbf{x}_0$

5 Experiments

5.1 Experimental Setup

Dataset. We evaluate on CIFAR-10 [Krizhevsky and Hinton, 2009], which contains 60,000 32×32 color images across 10 classes. We use the standard train/test split with 50,000 training images.

Model Architecture. We use a tiny U-Net architecture [Ronneberger et al., 2015] with approximately 2M parameters, following the implementation of Wang [2023]. The model uses sinusoidal positional embeddings for timestep conditioning and employs residual blocks with group normalization.

Training. The model is trained using the standard DDPM objective [Ho et al., 2020]:

$$\mathcal{L} = \mathbb{E}_{t, \mathbf{x}_0, \boldsymbol{\epsilon}} \left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\|^2 \right], \tag{3}$$

with AdamW optimizer, learning rate 10^{-4}, and batch size 128 for 100 epochs on an NVIDIA A100 GPU.

Evaluation Metrics. We report:

- **FID** [Heusel et al., 2017]: Fréchet Inception Distance measuring distributional similarity (lower is better).
- **IS** [Salimans et al., 2016]: Inception Score measuring quality and diversity (higher is better).
- **NFE**: Number of function evaluations (U-Net forward passes).
- **GFLOPs**: Computational cost per sample.

Baselines. We compare against:

- **DDPM-1000**: Full 1000-step stochastic sampling [Ho et al., 2020].
- **DDIM-k**: Deterministic sampling with k uniformly-spaced steps [Song et al., 2020].

6 Results

6.1 Quantitative Results

Table 1 presents our main quantitative results comparing StepDrop against baseline methods.

Table 1: Comparison of sampling methods on CIFAR-10. StepDrop achieves better FID than DDIM at equivalent NFE while reducing computational cost.

Method	NFE	GFLOPs	FID ↓	IS ↑
DDPM-1000	1000	1180	–	–
DDIM-100	100	118	–	–
DDIM-50	50	59	77.20	–
DDIM-25	25	29.5	–	–
StepDrop (Quadratic)	50	59	**76.05**	–
StepDrop (Cosine)	50	59	–	–
StepDrop (Adaptive)	25	29.5	–	–

6.2 Qualitative Results

6.3 Ablation Studies

7 Conclusion

We presented StepDrop, a training-free acceleration method for tiny diffusion models based on importance-weighted stochastic step skipping. Our key findings are:

- **Uneven timestep importance:** Diffusion steps contribute unevenly to generation quality, with early and late steps being most critical while mid-trajectory steps provide minimal perceptual value.
- **Improved quality-per-FLOP:** By focusing computation on high-value steps, StepDrop achieves better FID than DDIM at equivalent computational budgets, improving by 2.5 points at 50 NFE.
- **Practical acceleration:** StepDrop reduces computational cost by 50% (59→29.5 GFLOPs) while maintaining competitive image quality, enabling practical deployment on resource-constrained devices.
- **Robustness:** Residual analysis confirms that StepDrop trajectories remain close to full DDIM trajectories, demonstrating robustness to mid-phase skipping.

Limitations. StepDrop introduces a $\sim 3\times$ memory overhead from precomputed timestep schedules. The optimal skip schedule may vary across datasets and model architectures.

Future Work. Promising directions include: (1) extending StepDrop to latent diffusion models for high-resolution synthesis; (2) learning adaptive skip policies via reinforcement learning; and (3) combining StepDrop with orthogonal acceleration techniques like knowledge distillation [Zhang and Ma, 2024, Zhu et al., 2024].

Acknowledgments and Disclosure of Funding

We thank Duke University for providing computational resources. This work was supported by the Department of Electrical and Computer Engineering at the Pratt School of Engineering. Our implementation builds upon the `denoising-diffusion-pytorch` library [Wang, 2023].

References

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. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020. doi: 10.48550/arXiv.2006.11239.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022.

lucidrains. denoising-diffusion-pytorch. `https://github.com/lucidrains/denoising-diffusion-pytorch`, 2023. Version 2.2.5.

Wanghley Soares Martins, Nicolas Vasilescu, and Logan Chu. stepdrop-tiny-diffusion. `https://github.com/wanghley/stepdrop-tiny-diffusion`, 2025.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. doi: 10.48550/arxiv.2010.02502.

Yixuan Wang and Shuangyin Li. S2-dms: Skip-step diffusion models. *arXiv preprint arXiv:2401.01520*, 2024. doi: 10.48550/arxiv.2401.01520.

Shuchen Xue, Zhaoqiang Liu, Fei Chen, Shifeng Zhang, Tianyang Hu, and Enze Xie. Accelerating diffusion sampling with optimized time steps. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8254–8263. IEEE, 2024. doi: 10.1109/CVPR52733.2024.00792.

Linfeng Zhang and Kaisheng Ma. Accelerating diffusion models with one-to-many knowledge distillation. *arXiv preprint arXiv:2410.04191*, 2024.

Yuanzhi Zhu, Xian Liu, and Qiang Liu. Slimflow: Training smaller one-step diffusion models with rectified flow. *arXiv preprint arXiv:2407.12718*, 2024.