ACCELERATING DIFFUSION-BASED TEXT-TO-AUDIO GENERATION WITH CONSISTENCY DISTILLATION

Yatong Bai[*1,2]   *Trung Dang*[2]   *Dung Tran*[2]   *Kazuhito Koishida*[2]   *Somayeh Sojoudi*[1]

[1]University of California, Berkeley   [2]Microsoft

ABSTRACT

Diffusion models power a vast majority of the text-to-audio (TTA) generation methods. Unfortunately, such models suffer from a slow inference speed due to iteratively querying the underlying denoising network, thus unsuitable for applications with time or computational constraints. This work modifies the recently proposed "consistency distillation" framework to train TTA models that only require a single neural network query, accelerating the generation hundreds of times. By incorporating classifier-free guidance into the distillation framework, our models retain diffusion models' impressive generation quality and diversity. Furthermore, the non-recurrent differentiable structure resulting from the distillation allows fine-tuning with novel loss functions. We use the CLAP loss as an example, confirming that end-to-end fine-tuning further boosts the generation quality.

Index Terms— Diffusion models, Consistency models, Audio generation, Generative AI, Neural networks

1. INTRODUCTION

Text-to-audio (TTA) generation is a recently popularized task where a model produces realistic audio based on the user's textual prompt. TTA models have been rapidly improving, demonstrating impressive capabilities for generating diverse, precise, and realistic audio. Most TTA methods are based on latent diffusion models (LDM). While LDMs have gained popularity in various generative fields due to their superior generation quality, they suffer from their slow inference speed resulting from the need to iteratively query the underlying neural network. Such a drawback is unacceptable in applications with limited inference time or computational resources.

This work proposes to significantly accelerate diffusion-based TTA models by distilling them into consistency models (CM) that only require a single neural network query during inference. The distillation procedure is crafted and modified to take full advantage of classifier-free guidance (CFG), an operation that vastly boosts the performance of text-conditioned generative models. Our experiments show that hundreds-fold acceleration can be achieved while incurring little objective and subjective quality reduction.

Unlike diffusion models, our consistency TTA models are not recursive and can be optimized end-to-end by back-propagating audio quality loss functions. Since the CLAP score considers both ground-truth audio and the textual prompt, we select it as an example loss function, and demonstrate that end-to-end fine-tuning can enhance human-evaluated text correspondence of the generated audio clips.

Throughout this paper, vectors and tensors are denoted as bold symbols whereas scalars use regular symbols. Some additional discussions and details are deferred to the technical report at `bai-yt.github.io/consistency_tta/report.pdf`.

2. BACKGROUND AND RELATED WORK

2.1. Diffusion models

Diffusion models [1, 2] have recently rapidly gained popularity among conditional and unconditional generation tasks in vision and audio fields [3, 4, 5, 6]. Diffusion models are preferable for various practical applications due to their diverse, high-quality generations.

In the vision domain, pixel-level diffusion methods such as EDM [4] mainly focus on smaller datasets. To generate larger images, the diffusion process is usually carried out in a latent space to relieve the computation requirement, with an encoder-decoder pair responsible for the conversion to and from the latent space. The resulting model architecture is referred to as the latent diffusion model [7]. On the audio side, audio generation can be further categorized into speech, music, and in-the-wild audio. This paper considers the in-the-wild audio setting, where the data emphasizes diversity and covers a variety of real-world sound clips. While early audio generation efforts considered autoregressive models [8] and Mel-space diffusion [9], LDMs have dominated recent advancements in in-the-wild audio generation [10, 5, 11, 12, 13, 14, 15].

The intuition of diffusion models is to gradually recover a clean sample (an image, a spectrogram, or a latent representation) from a noisy sample. During training, Gaussian noise is progressively added to a ground-truth dataset sample $\mathbf{z}_0$, forming a continuous diffusion trajectory. At the end of the trajectory, the noisy sample becomes indistinguishable from Gaussian noise. This trajectory is then discretized into N sections, and the noisy sample at each discretization time step is denoted as $\mathbf{z}_n$ for $n = 1, \dots, N$. Each training step selects a random time step and injects the appropriate amount of noise (which depends on the time step [2]) into the randomly drawn clean data to produce $\mathbf{z}_n$. A denoising neural network, often a U-Net [16], is then optimized to recover the noise information from the noisy sample. During inference, Gaussian noise is used as a surrogate as the noisy sample $\mathbf{z}_N$ at the final time step. The diffusion model gradually generates a clean sample by iteratively querying denoising network step by step, producing the recovery sequence $\hat{\mathbf{z}}_{N-1}, \dots, \hat{\mathbf{z}}_0$. The final $\hat{\mathbf{z}}_0$ is then used as the generated sample.

2.2. Accelerating diffusion model inference

Diffusion models often suffer from high generation latency and limited throughput due to the need for iterative denoising queries, preventing them from real-time applications. To this end, the literature has proposed several methods to reduce the number of model queries. Such methods are mostly presented and evaluated for image generation tasks and can be grouped into two main categories: improved differential equation solvers and distillation methods.

Improved differential equation solvers reduce the required number of discretization steps N during the inference of existing diffusion models without tuning the denoising network weights. Exam-

[*] Work done during internship at Microsoft. Correspondance email `yatong_bai@berkeley.edu`.

ples of such solvers include DDIM [17], Euler [18], Heun, DPM [19, 20], and PNDM [21]. The best solvers can reduce the number of steps to 10-50 from hundreds required by vanilla inference (which uses DDPM [2]). Some solvers, such as Heun, are higher-order, requiring more than one neural network query per step, inducing a trade-off between the number of steps and per-step computation.

Distillation methods have been considered to bring the number of denoising steps below ten. These methods use a pre-trained diffusion model as the teacher and train a student model to replicate several teacher passes within one neural network query. One of the most representative methods is progressive distillation (PD) [22], which iteratively halves the number of diffusion steps. PD reduces the required steps to only a few, but the single-step capability is unideal, and the iterative distillation procedure can be cumbersome. To this end, consistency distillation (CD) [23] has been proposed. The training goal of CD is to directly reconstruct the noiseless image within a single step from an arbitrary step on the teacher model's diffusion trajectory. Note that both PD and CD were initially proposed for *unconditional* image generation. For text-conditioned generation, there are additional considerations, which we discuss below.

2.3. Classifier-free guidance

CFG [24] is a simple yet effective method for adjusting the text conditioning strength for guided generation problems, significantly improving the performance of existing diffusion-based TTA generative models. CFG queries the denoising network in the diffusion model twice – once with text conditioning and once without (by masking the text embeddings). We use $\mathbf{v}_{\text{cond}}$ and $\mathbf{v}_{\text{uncond}}$ to denote the conditional and unconditional noise estimations from the denoising network. The guided estimation, denoted by $\mathbf{v}_{\text{CFG}}$, is calculated via

$$\mathbf{v}_{\text{CFG}} = w \cdot \mathbf{v}_{\text{cond}} + (1 - w) \cdot \mathbf{v}_{\text{uncond}}, \tag{1}$$

where $w \geq 0$ is the guidance strength. When w is between 0 and 1, CFG interpolates the conditioned and unconditioned estimations. When w is greater than 1, CFG becomes an extrapolation. For example, for TANGO, $w = 3$ produces the best overall result [15].

Since CFG is external to the denoising network in diffusion models, it makes distillating guided models more complex than their unguided counterparts. The authors of [25] outlined a two-stage pipeline for performing PD on a CFG classifier. The first stage absorbs CFG into the denoising network by letting the student network take w as an additional input. The second stage then performs conventional PD on top of the stage-1 student. During both training stages, w is randomized, and the resulting distilled network allows for selecting the CFG strength w during inference. However, we are unaware of existing extensions of CD to CFG models.

3. CONSISTENCY DISTILLATION FOR TTA

Among existing diffusion-based TTA generation methods, we select TANGO [15] as the baseline for our work due to its high generation quality and elegance of implementation. However, we highlight that most of the innovations of this paper can also be incorporated into other diffusion-based TTA methods.

3.1. Overall setup

Similar to TANGO, our model has four components, and the only component to be trained is a U-Net module. The three other components, pre-trained and frozen, are the following.

- A FLAN-T5-Large language model [26] that processes the textual prompt. We use the same checkpoint as [15].
- A VAE encoder-decoder pair that maps between the Mel spectrogram space and the U-Net latent space (in both directions).
- A HiFi-GAN vocoder [27] that produces high-quality time-domain audio waveform from the Mel spectrogram.

For the VAE and the HiFI-GAN, we use the checkpoint pre-trained on AudioSet released by the authors of [5] as in [15]. During inference, the FLAN-T5 encodes the prompt into a text embedding, which then guides the U-Net module to reconstruct a latent audio representation. Next, the VAE decoder recovers the Mel spectrogram from the latent representation. Finally, the HiFi-GAN produces the generated waveform. The VAE encoder is not used.

During all training stages, a short-time Fourier transform converts the ground-truth audio into a spectrogram and then a Mel spectrogram. The VAE encoder produces a latent representation of the Mel spectrogram, which then supervises the U-Net. The FLAN-T5 operates the same as during inference, producing textual conditioning for the U-Net. The VAE decoder and the HiFi-GAN are not used.

3.2. Consistency distillation

The goal of CD is to learn a student U-Net $f_{\text{stu}}(\cdot, \cdot, \cdot)$ from the diffusion U-Net module in the teacher TTA model $f_{\text{tea}}(\cdot, \cdot, \cdot)$. The architecture of f_{stu} is the same as the f_{tea}, both taking three inputs: the noise latent representation $\mathbf{z}_n$, the corresponding time step n, and the text embedding $\mathbf{e}_{\text{tex}}$. Furthermore, the parameters in f_{stu} are initialized using f_{tea} information.

The ultimate task for the student U-Net is to generate latent representations of realistic audio clips within a single forward pass. It should directly produce an estimated clean example $\hat{\mathbf{z}}_0$ from $\mathbf{z}_n$, where $n \in \{0, \ldots, N\}$ is an arbitrary step along the diffusion trajectory [23, Algorithm 2]. The risk function to be minimized for achieving this goal is

$$\mathbb{E}_{\substack{(\mathbf{z}_0, \mathbf{e}_{\text{tex}}) \sim \mathcal{D} \\ n \sim \text{Unif}(1,N)}} \Big[d\Big(f_{\text{stu}}(\mathbf{z}_n, n, \mathbf{e}_{\text{tex}}), f_{\text{stu}}(\hat{\mathbf{z}}_{n-1}, n-1, \mathbf{e}_{\text{tex}}) \Big) \Big], \tag{2}$$

where $d(\cdot, \cdot)$ is a distance measurement, $\mathcal{D}$ is the training dataset, and $\hat{\mathbf{z}}_{n-1} = \text{solve} \circ f_{\text{tea}}(\mathbf{z}_n, n, \mathbf{e}_{\text{tex}})$ is the teacher diffusion model's estimation for $\mathbf{z}_{n-1}$. Here, solve $\circ f_{\text{tea}}$ denotes the composite function of the teacher denoising U-Net and the solver that converts the U-Net raw output to the estimation of the previous time step. We use the ℓ_2 distance in this latent space as $d(\cdot, \cdot)$, with additional discussions in Appendix A.2 in the supplemental materials. Intuitively, this risk measures the expected distance between the student's reconstructions from two adjacent time steps on the diffusion trajectory.

The authors of [23] used the Heun solver for querying the teacher diffusion model during distillation. In addition, they adopted the "Karras noise schedule", a discretization scheme that unevenly selects the time steps on the diffusion trajectory. We investigate multiple solvers and noise schedules, with detailed discussions presented in Section 4.

3.3. Consistency distillation with classifier-free guidance

Since CFG is crucial to the conditional generation quality, we ablate three methods for incorporating it into the distilled model.

The first method is "direct guidance", which directly performs CFG on the consistency model output by applying (1). Since this method naïvely extrapolates or interpolates on the consistency model $\mathbf{z}_0$ prediction, the CFG operation will likely move the prediction outside the manifold of realistic latent representations.

The second method is "fixed guidance distillation", which means distilling from the diffusion model coupled with CFG using a fixed guidance strength w. Specifically, the training risk function is again (2), but $\hat{\mathbf{z}}_{n-1}$ is replaced with the estimation after CFG. Now, $\hat{\mathbf{z}}_{n-1}$ becomes solve $\circ\, f_{\text{tea}}^{\text{CFG}}(\mathbf{z}_n, n, \mathbf{e}_{\text{tex}}, w)$, where

$$f_{\text{tea}}^{\text{CFG}}(\mathbf{z}_n, n, \mathbf{e}_{\text{tex}}, w) := w \cdot f_{\text{tea}}(\mathbf{z}_n, n, \varnothing) + (1 - w) \cdot f_{\text{tea}}(\mathbf{z}_n, n, \mathbf{e}_{\text{tex}}),$$

with $\varnothing$ denoting the masked language token. Here, w should be fixed to the value corresponding to the best teacher generation quality.

The third method is "variable guidance distillation". It is largely similar to fixed guidance distillation, but with randomized guidance strength w during distillation, so that w can be adjusted during inference. To make the student network compatible with adjustable w, we add an additional w-encoding condition branch to f_{stu} (which now has four inputs). We use Fourier encoding for w following [25], and merge the embedding into f_{stu} similarly to the time step embedding. Each training iteration samples a random guidance strength w via the uniform distribution supported on $[0, 6)$.

The latter two methods are closely related to the two-stage distillation procedure outlined in [25], with the details described in Appendix A.3 in the supplemental materials.

3.4. Min-SNR training loss weighing strategy

The authors of [28] have proposed to use the truncated signal-noise ratio (SNR) to weigh the training loss at each time step n for diffusion models. The specific calculation of this "Min-SNR" weight strategy depends on the parameterization of the diffusion model. Specifically, diffusion models can be trained to predict the clean example $\mathbf{z}_0$, the additive noise ϵ, or the noise velocity v. While the three parameterizations have equal representation power, the Min-SNR weighting formulation is different to produce similar results.

This work investigates whether the Min-SNR strategy also improves CD. Since consistency models predict the clean sample $\mathbf{z}_0$, we use the Min-SNR formulation for $\mathbf{z}_0$-predicting diffusion models, which is $\omega(n) = \min\{\text{SNR}(t_n), \gamma\}$, where $\omega(n)$ is the loss weight for the n^{th} time step, $\text{SNR}(t)$ is the SNR at time t, t_n is the time corresponding to the n^{th} time step, and γ is a constant defaulted to 5. For the Heun solver used in most of our experiments, $\text{SNR}(t)$ is the inverse of the additive Gaussian noise variance at time t.

3.5. End-to-end fine-tuning with CLAP

Since our consistency TTA model only requires one neural network query, it is straightforwardly end-to-end differentiable. Thus, we can back-propagate loss functions rewarding high-quality generation and fine-tune the U-Net generation module. On the contrary, diffusion models have recurrent inference procedures, making end-to-end fine-tuning prohibitively time-consuming and potentially unstable.

In this work, we use the CLAP score [29] as an example of fine-tuning loss function. The CLAP score, denoted by CS, is defined as:

$$\text{CS}(\mathbf{Gen}, \mathbf{Ref}) = \max\left\{100 \times \frac{\mathbf{E}_{\text{Gen}} \cdot \mathbf{E}_{\text{Ref}}}{\|\mathbf{E}_{\text{Gen}}\| \cdot \|\mathbf{E}_{\text{Ref}}\|}, 0\right\}, \quad (3)$$

where $\mathbf{Gen}$ is the generated audio waveform, $\mathbf{Ref}$ is the reference (ground-truth waveform or textual prompt), and $\mathbf{E}_{\text{Gen}}$ and $\mathbf{E}_{\text{Ref}}$ are the corresponding embeddings extracted by the CLAP model. We select the CLAP score mainly due to the following reasons:

- Superior embedding quality. While the feature extraction models used in many other metrics are for classification, the CLAP encoders are trained on more diverse tasks and datasets.
- CLAP considers both the ground-truth audio and the textual prompts. Since the CD training loss (2) does not use the ground truth information, incorporating the CLAP score provides important feedback to the model.

4. EXPERIMENTS

4.1. Experiment settings and dataset

This section discusses the empirical results of the proposed consistency TTA model. While we explicitly use TANGO [15] as the baseline, our methods apply to diffusion-based TTA models in general.

The most popular benchmark dataset for in-the-wild audio generation is the AudioCaps test set [30], a set of human-captioned YouTube audio. Our copy contains 882 audio clips with a length of at most ten seconds. Like many existing works, the core generative U-Net of our models is trained on the AudioCaps training set (our copy has 45260 instances), leaving larger datasets for future work.

To accelerate the training, we shrink the generative U-Net size from 866M parameters used in [15] to 557M. Appendix A.2 in the supplemental materials describes the details of this smaller U-Net. As shown in Table 2, we observe that our TANGO model with this smaller U-Net performs similarly to the checkpoint shared in [15], with slightly higher CLAP_{T} and KLD but slightly lower CLAP_{A} and FAD. All consistency models are distilled from this smaller TANGO model.

We consider the following objective metrics:

- FD: Fréchet Distance between generated and ground-truth audio. It uses VGGish [31] as the feature extractor.
- FAD: Fréchet Audio Distance between the waveforms. It uses PANN [32] as the feature extractor.
- KLD: Kullback-Leibler Divergence between the waveforms. It also uses PANN [32] as the feature extractor.
- CLAP_{A}: CLAP score with respect to the ground-truth audio waveform. We use the CLAP model checkpoint from [33] trained on LAION-Audio-630k [33], AudioSet [34], and music data. The CLAP scores calculation follows (3).
- CLAP_{T}: CLAP score with respect to the textual prompt.

4.2. Objective results

We first ablate the performance of the consistency TTA generation model under different training settings, with the results presented in Table 1. Note that "guided initialization" refers to initializing the consistency model with a guidance-aware diffusion model, whereas "TANGO initialization" refers to initializing with the unmodified diffusion teacher weights. Table 1 demonstrates that:

- Distilling CFG models boosts performance over direct guidance.
- Using the more accurate Heun solver for querying the diffusion teacher model is advantageous over the simpler DDIM solver.
- The uniform noise schedule is preferred over the Karras schedule. Note that all results are based on $N = 18$ discretization steps as in [23]. See Appendix A.1 for a detailed discussion.
- The Min-SNR weights and the guided initialization improve the FD and FAD but slightly sacrifice the KLD.

The best settings are then used in Table 2, which compares consistency TTA models with the diffusion baseline models. We also lengthen the distillation to 60 epochs, improving all metrics. Table 2 proves that the gap between the single-step consistency models and the 400-step diffusion models (we use 200 steps following [15]; each

Table 1. Compare various guidance weights, distillation techniques, teacher solvers, noise schedules, training lengths, loss weights, and initialization. "CFG w" represents the guidance weight; "# queries" indicates the number of neural network queries during inference. U-Net modules have 557M parameters, except in variable guidance models (559M). Distillation runs are 40 epochs; inference uses FP32 precision.

# queries (↓)	Teacher solver	Noise schedule	CFG w	Guidance method	Min-SNR	Initialization	FAD (↓)	FD (↓)	KLD (↓)
1	DDIM	Uniform	1	-	✗	TANGO	13.48	45.75	2.409
	Heun	Karras					10.97	50.19	2.425
2	DDIM	Uniform	3	Direct guidance	✗	TANGO	8.565	38.67	2.015
	Heun	Karras					7.421	39.36	1.976
1	Heun	Karras	3 (best for teacher)	Fixed guidance distillation	✗	TANGO	5.702	33.18	1.494
		Uniform			✗	TANGO	4.168	28.54	1.384
		Uniform			✓	TANGO	3.766	27.74	1.443
		Uniform			✗	Guided	3.859	27.79	1.421
1	Heun	Uniform	3	Variable guidance distillation	✓	Guided	3.956	28.27	1.442
			4				3.180	27.92	1.394
			6				2.975	28.63	1.378

Table 2. Compare consistency models to the diffusion baselines. Distillation runs are 60 epochs; CLAP-fine-tuning uses 10 additional epochs. All CD runs use the Heun teacher solver, uniform noise schedule, variable guidance distillation, guided initialization, Min-SNR loss weights, and BF16 inference precision. Bold numbers indicate the best results from CM; blue numbers are where CM achieves best overall.

Setting	U-Net params	CFG w	# queries (↓)	CLAP_T (↑)	CLAP_A (↑)	FAD (↓)	FD (↓)	KLD (↓)
AudioLDM-L reported in [5]	739M	2	400	-	-	2.08	27.12	1.86
TANGO reported in [15]	866M	3		-	-	1.59	24.53	1.37
TANGO [15] tested by us	866M	3		23.99	72.83	1.631	25.84	1.359
Our TANGO model	557M	3	400	24.57	72.65	1.908	24.63	1.326
Consistency model without CLAP fine-tuning	559M	3	1	20.44	71.25	3.203	25.02	1.387
		4		21.54	71.97	2.611	24.61	1.342
		5		22.04	72.18	2.575	24.92	**1.321**
Consistency model with CLAP fine-tuning	559M	3	1	23.94	72.27	**2.182**	**23.62**	1.350
		4		**24.18**	72.42	2.406	24.10	1.339
		5		24.14	**72.43**	2.626	24.51	1.339

Table 3. Compare the human evaluation results of consistency and diffusion models. Bold and blue numbers are defined same as Table 2.

U-Net	# queries	Model	CLAP	CFG	Human	Human	CLAP_T	CLAP_A	FAD	FD	KLD
params	(↓)	type	fine-tuning	w	Quality (↑)	Corresp (↑)	(↑)	(↑)	(↓)	(↓)	(↓)
557M	400	Diffusion	✗	3	4.300	4.164	24.49	72.76	1.908	24.63	1.326
559M	1	Consistency	✗	5	**3.936**	3.944	22.11	72.05	2.575	24.92	**1.321**
			✓	4	3.868	**4.080**	**24.18**	**72.42**	**2.406**	**24.10**	1.339
Ground-truth audio					4.536	4.364	26.14	100.0	0.000		

step requires two neural noise estimations due to CFG) is small for all quality metrics, with the best consistency model even surpassing the diffusion baseline in FD and KLD.

On top of the best-trained consistency model, we then perform end-to-end CLAP fine-tuning, co-optimizing three loss components: the consistency loss (2), CLAP_A, and CLAP_T. Table 2 suggests that fine-tuning further improves all objective metrics except KLD.

4.3. Subjective results

Finally, we let human evaluators rate the generated audio from the consistency and diffusion models on a scale of 1 to 5 in two aspects: overall audio quality and audio-text correspondence, with the details presented in Appendix A.4 in the supplemental materials. The comparison, presented in Table 3, reveals that the subjective qualities of the consistency and diffusion models are comparable, but the difference could be larger than suggested by the objective metrics. Moreover, optimizing the CLAP scores improves the human-rated text-audio correspondence score, which makes sense since CLAP_T provides closed-loop feedback to help understand prompt information. In summary, the proposed consistency model reduces the computation to generate an audio clip by a factor of 400 (potentially even higher due to BF16 inference and smaller model size) with minimal performance drop. We share 50 generated audio examples from the AudioCaps test set at `bai-yt.github.io/consistency_tta/demo.html`.

We also observe that the Gaussian inputs to the consistency model from different random seeds generate noticeably different audio, confirming that our consistency TTA model produces diverse generations like diffusion models. For illustration, an example can be found in Appendix A.5 in the supplemental materials.

5. CONCLUSION

This work proposes accelerating diffusion-based TTA generation models hundreds of folds via consistency distillation. The delicate design of the distillation procedure emphasizing CFG achieves this vast acceleration with minimal generation quality reduction, enabling diverse and realistic in-the-wild audio generation within one neural network query. The differentiability of the resulting model allows for end-to-end fine-tuning, unlocking possibilities for improving the training method of such models.

6. ACKNOWLEDGMENTS

We sincerely appreciate Chih-Yu Lai, Mo Zhou, Afrina Tabassum, You Zhang, Sara Abdali, Uros Batricevic, Yinheng Li, Asing Chang, Rogerio Bonatti, and Sam Pfrommer for helping with human evaluation.

7. REFERENCES

[1] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International Conference on Machine Learning*, 2015.

[2] Jonathan Ho, Ajay Jain, and Pieter Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, 2020.

[3] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer, "High-resolution image synthesis with latent diffusion models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[4] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine, "Elucidating the design space of diffusion-based generative models," in *Advances in Neural Information Processing Systems*, 2022.

[5] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley, "Audioldm: Text-to-audio generation with latent diffusion models," *arXiv preprint arXiv:2301.12503*, 2023.

[6] Qingqing Huang, Daniel S Park, Tao Wang, Timo I Denk, Andy Ly, Nanxin Chen, Zhengdong Zhang, Zhishuai Zhang, Jiahui Yu, Christian Frank, et al., "Noise2music: Text-conditioned music generation with diffusion models," *arXiv preprint arXiv:2302.03917*, 2023.

[7] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer, "High-resolution image synthesis with latent diffusion models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[8] Felix Kreuk, Gabriel Synnaeve, Adam Polyak, Uriel Singer, Alexandre Défossez, Jade Copet, Devi Parikh, Yaniv Taigman, and Yossi Adi, "Audiogen: Textually guided audio generation," in *International Conference on Learning Representations*, 2023.

[9] Seth Forsgren and Hayk Martiros, "Riffusion-stable diffusion for real-time music generation," *URL https://riffusion.com*, 2022.

[10] Dongchao Yang, Jianwei Yu, Helin Wang, Wen Wang, Chao Weng, Yuexian Zou, and Dong Yu, "Diffsound: Discrete diffusion model for text-to-sound generation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.

[11] Haohe Liu, Qiao Tian, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D Plumbley, "Audioldm 2: Learning holistic audio generation with self-supervised pretraining," *arXiv preprint arXiv:2308.05734*, 2023.

[12] Rongjie Huang, Jiawei Huang, Dongchao Yang, Yi Ren, Luping Liu, Mingze Li, Zhenhui Ye, Jinglin Liu, Xiang Yin, and Zhou Zhao, "Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models," *arXiv preprint arXiv:2301.12661*, 2023.

[13] Jiawei Huang, Yi Ren, Rongjie Huang, Dongchao Yang, Zhenhui Ye, Chen Zhang, Jinglin Liu, Xiang Yin, Zejun Ma, and Zhou Zhao, "Make-an-audio 2: Temporal-enhanced text-to-audio generation," *arXiv preprint arXiv:2305.18474*, 2023.

[14] Zineng Tang, Ziyi Yang, Chenguang Zhu, Michael Zeng, and Mohit Bansal, "Any-to-any generation via composable diffusion," *arXiv preprint arXiv:2305.11846*, 2023.

[15] Deepanway Ghosal, Navonil Majumder, Ambuj Mehrish, and Soujanya Poria, "Text-to-audio generation using instruction-tuned llm and latent diffusion model," *arXiv preprint arXiv:2304.13731*, 2023.

[16] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, 2015.

[17] Jiaming Song, Chenlin Meng, and Stefano Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[18] Leonhard Euler, *Institutionum calculi integralis*, vol. 1, impensis Academiae imperialis scientiarum, 1824.

[19] 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," in *Advances in Neural Information Processing Systems*, 2022.

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

[21] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao, "Pseudo numerical methods for diffusion models on manifolds," in *International Conference on Learning Representations*, 2022.

[22] Tim Salimans and Jonathan Ho, "Progressive distillation for fast sampling of diffusion models," in *International Conference on Learning Representations*, 2021.

[23] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever, "Consistency models," in *International Conference on Machine Learning*, 2023.

[24] Jonathan Ho and Tim Salimans, "Classifier-free diffusion guidance," in *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

[25] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans, "On distillation of guided diffusion models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[26] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al., "Scaling instruction-finetuned language models," *arXiv preprint arXiv:2210.11416*, 2022.

[27] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," in *Advances in Neural Information Processing Systems*, 2020.

[28] Tiankai Hang, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo, "Efficient diffusion training via min-snr weighting strategy," *arXiv preprint arXiv:2303.09556*, 2023.

[29] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang, "Clap learning audio concepts from natural language supervision," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023.

[30] Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim, "Audiocaps: Generating captions for audios in the wild," in *Conference of the North American Chapter of the Association for Computational Linguistics*, 2019.

[31] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al., "Cnn architectures for large-scale audio classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

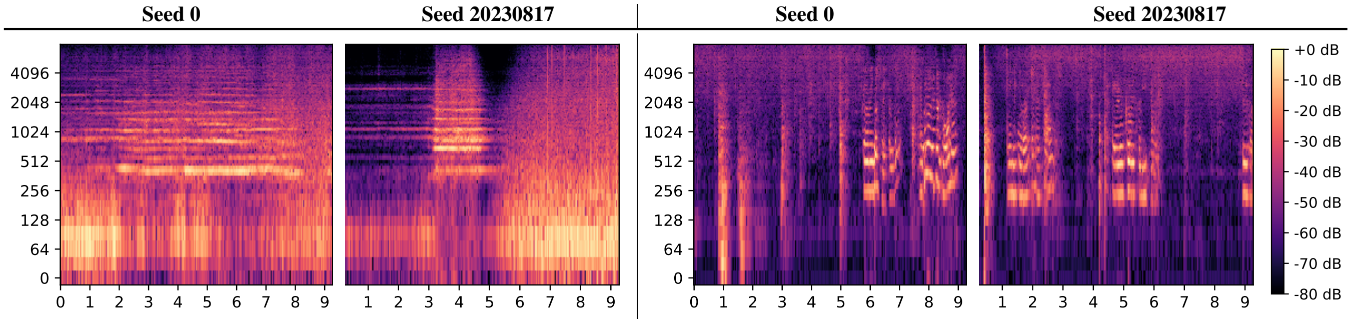
[32] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[33] Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov, "Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023.

[34] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

[35] Brian McFee, "ResamPy: efficient sample rate conversion in python," *Journal of Open Source Software*, vol. 1, no. 8, pp. 125, 2016.

[36] Yao-Yuan Yang, Moto Hira, Zhaoheng Ni, Anjali Chourdia, Artyom Astafurov, Caroline Chen, Ching-Feng Yeh, Christian Puhrsch, David Pollack, Dmitriy Genzel, Donny Greenberg, Edward Z. Yang, Jason Lian, Jay Mahadeokar, Jeff Hwang, Ji Chen, Peter Goldsborough, Prabhat Roy, Sean Narenthiran, Shinji Watanabe, Soumith Chintala, Vincent Quenneville-Bélair, and Yangyang Shi, "Torchaudio: Building blocks for audio and speech processing," *arXiv preprint arXiv:2110.15018*, 2021.



A train sounds horn and travels.

Food sizzling with some knocking and banging followed by a woman speaking.

Table 4. The generated audio is noticeably different with different random seeds. The vertical axis of each figure is the frequency in Hz and the horizontal axis is time in seconds. We use our CLAP-finetuned model for this illustration.

A. ADDITIONAL DISCUSSIONS AND DETAILS

A.1. Additional discussions regarding teacher solver

Table 1 presents the generation quality of the consistency model f_{stu} distilled with various solver settings, confirming our selection of the Heun solver and the uniform schedule. While [23] agreed that the Heun solver achieved better results, the authors suggested using the Karras schedule. Our explanation of this discrepancy is that TANGO was trained using the uniform schedule, whereas the teacher models considered in [23] were trained with the Karras schedule. It is likely beneficial to use the same scheduler as the training procedure of the diffusion model.

We also compared the TANGO inference performance using multiple solvers (fixing $N = 18$ inference steps), including DDPM, DDIM, Euler, Heun, and DPM++(2S), and confirmed that the Heun solver with a uniform schedule returned the best FAD and FD metrics. DPM++(2S) achieved a better KLD but the difference is very small.

A.2. Model and training details

We noticed that the implementation of the audio resampling operation has a major influence on some metrics, with FAD being especially sensitive. To ensure high training quality and fair evaluation, we use ResamPy [35] for all resampling procedures unless the resampling step needs to be differentiable. Specifically, CLAP fine-tuning requires differentiable resampling, and we use TorchAudio [36] instead.

The structure of our 557M-parameter U-Net is similar to the 866M U-Net used in [15], with the only modification being reducing the "block out channels" from $(320, 640, 1280, 1280)$ to $(256, 512, 1024, 1024)$. All CD runs use two 48GB-VRAM GPUs, with a total batch size of 12 and five gradient accumulation steps. The optimizer is AdamW with a 10^{-4} weight decay, and the learning rate is 10^{-5} for CD and 10^{-6} for CLAP fine-tuning. The "CD target network" (see [23] for details) is an exponential model average (EMA) copy with a 0.95 decay rate. We also maintain an EMA copy with a 0.999 decay rate for the reported experiment results. All training uses BF16 numerical precision.

Regarding the distance measure $d(\cdot, \cdot)$ introduced in (2), the authors of [23] considered several options for $d(\cdot, \cdot)$ for image generation tasks and concluded that using LPIPS (an evaluation metric that embeds the generated image and calculates the weighted distance in several feature spaces) as the optimization objective produced higher generation quality than using the pixel-level ℓ_2 or ℓ_1 distance. However, since our latent diffusion model already operates in a latent feature space, we simply use the ℓ_2 distance in this latent space.

A.3. Relationship to two-stage progressive distillation

Unlike PD considered in [25], which requires iteratively halving the number of diffusion steps, CD used in our method reduces the required inference step to one within a single training process. As a result, the two distillation stages proposed in [25] can be merged. Specifically, stage-2 distillation can be performed without stage 1, provided that the step of querying the stage-1 model is replaced by querying the original teacher model with CFG. Merging stage 1 and stage 2 then results in our "variable guidance distillation" method discussed in Section 3.3. Subsequently, stage 1 becomes optional since it only serves to provide a guidance-aware initialization to stage 2.

A.4. Human evaluation details

The human evaluation results reported in Table 3 are based on 10 evaluators each rating 25 audio clips for each of the three models as well as the ground truth (the models use the same set of prompts), forming a total sample size of 250 per model. The type of model that generated each waveform is not disclosed to the evaluator, and the generations of the models are shuffled. We find it impossible to distinguish the generations from the three generative models, with many ground truth waveforms also indistinguishable, ensuring the fairness of the evaluation. An example evaluation form is available at `bai-yt.github.io/consistency_tta/evaluation.html`.

A.5. Generation diversity

To demonstrate that the consistency TTA model can generate diverse audio as do diffusion-based models, we select two example prompts to show that the generations noticeably differ with different random seeds (i.e., different initial Gaussian latent) in Table 4.