Cross-Resolution Distribution Matching for Diffusion Distillation

Feiyang Chen[1,3*,‡], Hongpeng Pan[1*], Haonan Xu[1,2‡], Xinyu Duan[1], Zhefeng Wang[1†], and Yang Yang[2†]

[1] Huawei, Shanghai & Hangzhou, China
[2] Nanjing University of Science and Technology, Nanjing, China
[3] HiThink Research, Hangzhou, China

Abstract. Diffusion distillation is central to accelerating image and video generation, yet existing methods are fundamentally limited by the denoising process, where step reduction has largely saturated. Partial-timestep, low-resolution generation can further accelerate inference, but it suffers from noticeable quality degradation due to cross-resolution distribution gaps. We propose Cross-**R**esolution Distribution **M**atching **D**istillation (RMD), a novel distillation framework that bridges cross-resolution distribution gaps for high-fidelity, few-step multi-resolution cascaded inference. Specifically, RMD divides the timestep intervals for each resolution using logarithmic signal-to-noise ratio (logSNR) curves, and introduces logSNR-based mapping to compensate for resolution-induced shifts. Distribution matching is conducted along resolution trajectories to reduce the gap between low-resolution generator distributions and the teacher's high-resolution distribution. In addition, a predicted-noise re-injection mechanism is incorporated during upsampling to stabilize training and improve synthesis quality. Quantitative and qualitative results show that RMD preserves high-fidelity generation while accelerating inference across various backbones. Notably, RMD achieves up to 33.4× speedup on SDXL and 25.6× on Wan2.1-14B, while preserving high visual fidelity.

Keywords: Cross-Resolution · Distribution Matching · Diffusion Distillation

1 Introduction

In recent years, diffusion models [8, 14] have demonstrated impressive performance in high-fidelity visual generation [28, 38, 39]. Despite their success, diffusion models require hundreds of iterative denoising steps, each involving a full forward pass through a large-scale neural network. Moreover, in modern Diffusion Transformer [26] (DiT) architectures, the computational cost of each

*Equal contribution.
†Corresponding authors. Email: yyang@njust.edu.cn, wangzhefeng@huawei.com
‡Work done while Feiyang Chen and Haonan Xu were with Huawei.

Fig. 1: Under the same text prompt and random seed, the SDXL model produces distinct images at 512×512 and 1024×1024 resolutions, revealing a clear resolution-dependent distribution shift. The corresponding prompts are provided in Appendix D.

forward pass is further amplified at high resolutions [49], as the number of tokens scales with spatial dimensions, leading to a quadratic increase in attention complexity. As a result, visual synthesis incurs substantial computational overhead and latency, limiting the feasibility of real-time or resource-constrained applications.

Up to now, many efforts have been made in pursuing efficient sampling techniques [32]. Among them, training-free methods [15, 18, 33] accelerate sampling but still need dozens of steps. In contrast, step distillation methods, such as trajectory-based distillation [12, 29, 35] and distribution-matching distillation [47, 48, 53], can significantly improve sampling efficiency, reducing the process to only a few steps (e.g., to 4–8 steps) while maintaining high-fidelity visual generation. However, empirical evidence [22] indicates that excessively aggressive reduction of generation steps (e.g., to 1–3 steps) can result in a catastrophic decrease in performance. Consequently, relying solely on step reduction via distillation imposes inherent limitations on improving sampling efficiency.

To address the efficiency bottleneck in step distillation, we instead focus on lowering the resolution at selected timesteps during the sampling process. Prior work has shown that the role of each denoising step depends on the noise level: early steps under high-noise conditions primarily reconstruct the global structural layout, whereas later steps under low-noise conditions refine fine-grained visual details [24,42,50]. However, performing high-resolution reconstruction during the high-noise stage leads to redundant computation of fine-grained details that are not yet required. Therefore, to further improve the efficiency of step distillation, we adopt a multi-resolution cascaded generation strategy: global structures are first recovered via coarse denoising at a lower resolution, and high-resolution processing is deferred to later stages to refine fine-grained details.

However, as illustrated in Fig. 1, the same model yields distinct data distributions across resolutions; thus, directly reducing the generation resolution at selected timesteps introduces distributional inconsistencies and degrades syn-

thesis quality [23, 41]. This issue stems from the training paradigm of existing state-of-the-art diffusion models [38, 39, 51], which adopt a multi-stage curriculum schedule: models are first trained on large-scale, low-resolution data of varying quality, and subsequently fine-tuned on carefully curated high-resolution data. Consequently, under a coarse-to-fine multi-resolution cascaded generation scheme, the global structure is effectively sampled from a low-resolution distribution with heterogeneous quality, rather than from the inherently higher-fidelity high-resolution distribution, resulting in a fundamental mismatch.

In this paper, we introduce RMD, a novel distillation framework (see Fig. 2) for few-step multi-resolution cascaded generation that explicitly bridges cross-resolution distribution inconsistencies. Concretely, RMD segments the denoising trajectory according to logSNR curves and progressively increases the resolution as noise decreases, enabling coarse-grained semantic construction at early stages and fine-grained detail refinement at later stages. Within each segment, noisy samples are upsampled and aligned at the final high resolution for distribution matching distillation, ensuring global distribution consistency across resolutions. In addition, a predicted-noise re-injection mechanism is employed during upsampling to provide prior trajectory guidance, stabilizing training and enhancing synthesis quality. As a result, RMD achieves few-step generation with progressive resolution scaling, breaking through the sampling efficiency bottleneck of conventional step-distillation methods, while preserving high-fidelity visual synthesis. Extensive experiments on image synthesis and its natural extension to video synthesis validate the effectiveness of the proposed RMD method, consistently achieving improved sampling speed while preserving high visual fidelity.

2 Related Work

2.1 Step Distillation

Trajectory-based Distillation trains a few-step student model by aligning its denoising trajectory with that of a multi-step teacher model [17,25,35]. Progressive distillation [29] iteratively trains a sequence of student models, each halving the number of sampling steps of its predecessor by matching two-step trajectories in a single step. Instaflow [17] straightens ODE trajectories to reduce discretization error and enable generation with fewer steps. Consistency models [35] and their subsequent extensions [12,34,43,43] train student models so that their outputs remain self-consistent at any timestep along the ODE trajectory. However, these methods suffer from difficulties in instance-level trajectory matching and numerical errors when solving the teacher's probability flow ODE [22].

Distribution Matching Distillation (DMD) [48] trains the student by aligning its output distribution with the multi-step teacher, avoiding the difficulty of instance-level trajectory matching. To date, many follow-up works [1,11,19,46,53] have proposed improvements. For example, DMD2 [47] introduces a discriminator to align the student model's distribution with a specified target distribution in a GAN-like manner [7]. TDM [22] enforces distribution-level alignment between the student's trajectory and that of the teacher. DP-DMD [44]

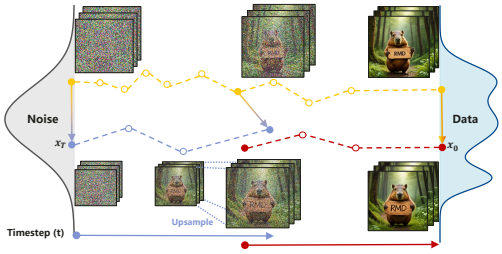

Fig. 2: Overview of the RMD Framework, which compresses the denoising trajectory of a pretrained diffusion model into a multi-resolution, few-step cascaded denoising process.

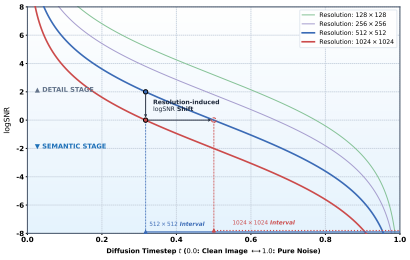

Fig. 3: Illustration of Cross-Resolution Timestep Interval Alignment. logSNR curves at different resolutions show resolution-dependent variations in noising dynamics.

exempts the first step from DMD, preserving sample diversity while enabling efficient, high-quality generation. Our work also starts with DMD, where we incorporate a coarse-to-fine generation paradigm into end-to-end distillation, going beyond the efficiency limits of step-only compression.

2.2 Multi-Resolution Cascaded Methods

Multi-resolution cascaded methods accelerate diffusion-based generation by decomposing the denoising process into a multi-resolution hierarchy, proceeding in a coarse-to-fine manner [23, 40, 50, 52]. For instance, PixelFlow [4] is trained from scratch to model cascaded flows in the pixel space, allowing images to be progressively synthesized from low to high resolution. Bottleneck Sampling [41] employs a high-low-high denoising workflow to maintain both visual fidelity and sampling efficiency. Moreover, cascade strategy has been incorporated into large-scale state-of-the-art diffusion models [2, 51], employing cascade refiners to facilitate efficient high-resolution visual synthesis. In this work, we unify multi-resolution cascading into a step distillation framework and address the distribution inconsistency between low- and high-resolution cascaded denoising [41].

3 Methodology

Cross-Resolution Distribution Matching Distillation (RMD) is introduced as a novel distillation framework that aligns the student's low-resolution distribution with the teacher's high-resolution distribution at the distribution level. We first partition the diffusion timestep into resolution-specific intervals based on their log signal-to-noise ratio (logSNR) profiles. A cross-resolution distribution matching objective is then formulated to learn distributional matching across resolutions. Finally, an optimized upsampling strategy is adopted to reduce the difficulty of cross-resolution alignment and improve training effectiveness.

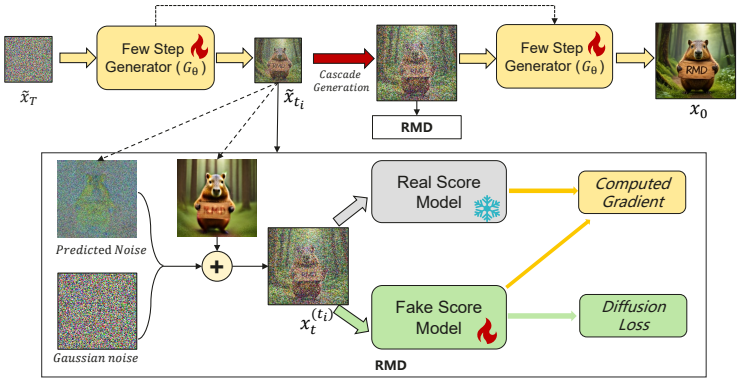

Fig. 4: The overall pipeline for distilling a pretrained diffusion model into a cascaded generator G_θ that performs generation across two resolution distribution spaces.

3.1 Resolution Trajectory Division

Resolution scaling intrinsically modifies the effective noise schedule. For identical independent noise magnitude, higher resolutions suffer relatively weaker pixel corruption [5, 9, 40]. This effect can be characterized via the logSNR. As shown in Fig. 3, the growth rate of logSNR differs across resolutions: in the low-logSNR regime, low-resolution trajectories denoise more rapidly, whereas in the high-logSNR regime, higher resolutions achieve a higher denoising rate. This observation motivates partitioning the diffusion process according to the logSNR trajectory and assigning resolution-specific timestep intervals. Note that Fig. 3 plots the *compensated* logSNR schedules after applying Eq. (2), where low-resolution curves sit above high-resolution curves, indicating higher compensated logSNR (less effective noise) at each timestep.

To ensure distributional alignment with the teacher model, we partition the teacher's logSNR curve at predefined thresholds $\{\text{logSNR}_1, \ldots, \text{logSNR}_{K-1}\}$, yielding K non-overlapping diffusion segments. Each breakpoint is converted to a diffusion boundary via the logSNR–timestep mapping. Under the Rectified Flow parameterization [16], the conversion is

$$\sigma_{T_i} = \frac{1}{1 + \exp\left(\frac{\text{logSNR}_i}{2}\right)}, \tag{1}$$

where σ_{T_i} denotes the noise schedule value at the boundary timestep T_i, with $\sigma_{T_0} = 1$ and $\sigma_{T_K} = 0$ [16], corresponding to the diffusion timestep partition defined by the logSNR curve. Each segment $i \in \{1, 2, \ldots, K\}$ corresponds to a timestep interval $[T_{i-1}, T_i]$, and is assigned a predefined resolution r_i, satisfying $r_1 < r_2 < \cdots < r_K$, where r_K equals the teacher model's generation resolution.

To maintain consistency with the teacher's timestep intervals during RMD, we compute the timestep shifts induced by resolution variations. Following the

resolution-dependent scaling in SD3 [6], the adjusted logSNR at resolution r_i is formulated as:

$$\text{logSNR}_i^{(r_i)} = \text{logSNR}_i - 2\log\left(\frac{r_i}{r_K}\right), \tag{2}$$

where the second term compensates for the resolution-induced SNR shift. By substituting the adjusted logSNR into Eq. (1), we derive the student's timestep interval $[T_{i-1}^{(r_i)}, T_i^{(r_i)}]$ that corresponds to the teacher's segment $[T_{i-1}, T_i]$. As illustrated in Fig. 3, we exemplify this for a two-stage ($K = 2$) configuration: by mapping the high-resolution interval onto the low-resolution temporal axis through logSNR invariance, we can bridge the gap between varying scales. This temporal synchronization ensures that distillation occurs over identical denoising states, regardless of the spatial resolution.

3.2 Cross-Resolution Distribution Matching

The objective of RMD is to compress a pretrained diffusion model into a few-step, multi-resolution cascaded generator G_θ. For each resolution interval, the generator is trained to match the teacher distribution under aligned logSNRs. Formally, we minimize the joint KL divergence [47, 48, 53]:

$$\min_\theta \text{KL}\Big(p_\theta(\tilde{x}_{t_i})\, p_\theta(x_0 \mid \tilde{x}_{t_i}) \parallel p_\varphi(x_t)\, p_\varphi(x_0 \mid x_t)\Big), \tag{3}$$

where p_φ and p_θ denote the pretrained teacher and generator, respectively. The index $t_i := (r_i, t)$ represents the resolution-aware timestep corresponding to teacher timestep t with resolution r_i under matched logSNR, as in Eq. (2). $\tilde{x}_{t_i}$ denotes the generator state aligned with the teacher distribution x_t. The conditionals $p_\varphi(x_0 \mid x_t)$ and $p_\theta(x_0 \mid \tilde{x}_{t_i})$ correspond to teacher ODE denoising and generator cross-resolution denoising.

Optimizing the above objective introduces two challenges. First, point-wise distribution matching along the generation trajectory is insufficient for effective distillation [22]. Second, as x_0 and $\tilde{x}_{t_i}$ reside at different resolutions, directly denoising $\tilde{x}_{t_i}$ to recover x_0 is ill-posed. Moreover, $\tilde{x}_{t_i}$ contains stochastic diffusion noise, and naive upsampling may distort low-resolution structural priors.

To overcome these issues, we project the generator state $\tilde{x}_{t_i}$ into the teacher space via a differentiable upsampling transformation [41,52], and perform distribution level matching along the inference trajectory. Specifically, the teacher scheduler determines N inference steps $\{t_j\}_{j=0}^{N-1}$, uniformly defined as $t_j = T_0 - \frac{T_0}{N} j$. These steps are mapped to low-resolution timestep as $t_{i,j} := (r_i, t_j)$. Multi-resolution cascaded inference under G_θ produces intermediate states $\tilde{x}_{t_{i,j}}$, which are subsequently transformed to the high-resolution space via the following upsampling transformation:

$$\begin{aligned} \tilde{x}_0^{(t_{i,j})} &= \tilde{x}_{t_{i,j}} - \sigma_{t_{i,j}}\, \epsilon_\theta(\tilde{x}_{t_{i,j}}, t_{i,j}), \\ x_{t_j}^{(t_{i,j})} &= (1 - \sigma_{t_j})\, U_{r_K}(\tilde{x}_0^{(t_{i,j})}) + \sigma_{t_j}\, \epsilon. \end{aligned} \tag{4}$$

Here, ϵ_θ denotes the noise prediction of G_θ, and $U_{r_K}(\cdot)$ denotes an interpolation upsampling operator to resolution r_K. The time-dependent standard deviation σ_t controls the noise scale, and $\epsilon \sim \mathcal{N}(0, I)$ represents the injected stochastic noise. $\tilde{x}_0^{(t_{i,j})}$ denotes the clean latent from the low-resolution distribution at timestep $t_{i,j}$. For simplicity, we adopt a rectified flow formulation [16] for noise injection, though other strategies [8, 14] are readily applicable.

Substituting the induced distribution of $x_{t_j}^{(t_{i,j})}$ into the distillation objective, we minimize the marginal reverse KL divergence under score distillation [20, 48]:

$$L(\theta) = \sum_{j=1}^{N} \mathrm{KL}\Big(p_\theta(x_{t_j}^{(t_{i,j})}) \parallel p_\varphi(x_{t_j})\Big). \tag{5}$$

However, multi-step alignment across resolutions is computationally demanding. Furthermore, because different resolutions require varying numbers of sampling steps, the resulting supervision becomes imbalanced, leading to slow convergence. Consequently, we perform distribution matching exclusively along resolution-wise stages. For each resolution stage r_i, we uniformly sample timesteps:

$$t_i \sim \mathcal{U}([T_{i-1}^{(r_i)}, T_i^{(r_i)}]).$$

We denote the corresponding expectation simply as $\mathbb{E}_{t_i}$ for brevity, and perform cross-resolution distribution alignment at the sampled steps. The overall objective is formulated as:

$$L(\theta) = \sum_{i=1}^{K} \lambda_{r_i} \, \mathbb{E}_{t_i} \Big[\mathrm{KL}\big(p_\theta(x_t^{(t_i)}) \parallel p_\varphi(x_t)\big)\Big]. \tag{6}$$

Here $p_\theta(x_t^{(t_i)})$ denotes the marginal diffusion distribution at timestep t_i within resolution stage r_i, projected to the high-resolution space at timestep t. The coefficient λ_{r_i} is a resolution-aware weight that balances the contribution of each resolution interval in the optimization objective. In practice, the expectation is approximated via Monte Carlo sampling during training.

Using score approximation, the gradient can be written as:

$$\nabla_\theta L(\theta) \approx \sum_{i=1}^{K} \lambda_{r_i} \mathbb{E}_{t_i} \Big[\big(s_\theta(x_t^{(t_i)}, t) - s_\varphi(x_t, t)\big) \frac{\partial x_t^{(t_i)}}{\partial \theta}\Big]. \tag{7}$$

Here, $x_t^{(t_i)}$ denotes the upsampled distribution obtained from the generator outputs via the upsampling transformation, which remains differentiable with respect to the model parameters θ.

The generator score s_θ is intractable. Following DMD [48] and TDM [22] score approximation, we introduce a fake diffusion model s_ϕ to estimate it, termed the fake score, while s_φ is treated as the true score. With the introduced fake diffusion model s_ϕ, the final gradient update formulation for RMD is:

$$L(\theta) = \sum_{i=1}^{K} \lambda_{r_i} \mathbb{E}_{t_i} \Big\| x_t^{(t_i)} - \mathrm{sg}\left(x_t^{(t_i)} + s_\phi(x_t^{(t_i)}, t) - s_\varphi(x_t, t)\right) \Big\|_2^2. \tag{8}$$

Here, $\mathrm{sg}(\cdot)$ denotes the stop-gradient operator, which can be interpreted as the upsampled generator distribution after being refined by one step of gradient descent, while blocking gradient back-propagation. A concise overview of the cross-resolution matching distillation training procedure is shown in Fig. 4.

The fake diffusion model is initialized from a pretrained DM and optimized with a standard denoising objective to track score variations along generated trajectories:

$$L(\phi) = \sum_{i=1}^{K} \lambda_{\mathrm{snr}} \mathbb{E}_{t_i} \left\| f_\phi(x_t^{(t_i)}, t) - U_{r_K}(\tilde{x}_0^{(t_i)}) \right\|_2^2, \tag{9}$$

where $U_{r_K}(\tilde{x}_0^{(t_i)})$ is the clean target. f_ϕ denotes the denoiser network parameterized to predict the clean data. λ_{snr} is the resolution-aware denoising weight, defined as the SNR at timestep t_i and resolution r_i.

3.3 Upsampling Transformation Optimization

In the upsampling transformation, the low-resolution distribution is upsampled and perturbed with Gaussian noise, enabling approximate alignment with the high-resolution distribution during training. However, such stochastic perturbation requires re-solving SDE trajectories across resolutions and does not inherit the teacher's ODE trajectory, which increases the difficulty of RMD. In contrast, injecting purely predicted noise to mimic the teacher ODE trajectory leads to severe degradation when the resolution gap is large.

To address this issue, we introduce a noise re-injection strategy for RMD, where Gaussian noise is replaced by a resolution-aware combination of predicted and Gaussian noise. The revised formulation is:

$$\begin{aligned} \epsilon_{t_i} &= \alpha\, U_{r_K}(\epsilon_\theta(\tilde{x}_{t_i}, t_i)) + \beta\, \epsilon \\ x_t^{(t_i)} &= (1 - \sigma_t)\, U_{r_K}(\tilde{x}_0^{(t_i)}) + \sigma_t\, \epsilon_{t_i} \end{aligned} \tag{10}$$

Here, ϵ_{t_i} comprises predicted noise and Gaussian noise. The predicted component is obtained by upsampling the generator's noise prediction ϵ_θ, while the stochastic term is a Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$. The two components are balanced by α and β with $\alpha^2 + \beta^2 = 1$. As the resolution gap enlarges, stronger stochasticity is required to bridge the distribution mismatch. Accordingly, α is reduced, assigning greater weight to the Gaussian noise.

3.4 Training and Inference

Warm-up Training with Low logSNR The distribution induced by low-resolution generative trajectories largely determines the overall layout of the high-resolution distribution during distillation [24, 50, 52]. In RMD, significant misalignment between the generated low-resolution distribution and the target high-resolution distribution can hinder the convergence of cascaded generative

trajectories. To enable efficient optimization, the generative process is divided into semantic and detail generation intervals based on the logSNR partitioning scheme from Wan2.2. Specifically, a threshold $\text{logSNR}_{K/2}$ is selected to separate these intervals, and the low-logSNR (semantic) interval is first distilled in a warm-up phase to provide a stable initialization. The full trajectory is then trained end-to-end, accelerating the overall distillation process. The pseudocode for RMD training is provided in Algorithm 1 in Appendix A.

Multi-Resolution Cascaded Inference. During inference, a multi-resolution cascaded scheme is adopted to synthesize the target-resolution distribution. The procedure dynamically determines whether upsampling is required at each subsequent timestep and, if so, re-injects noise to maintain consistency with the diffusion trajectory. Denoising begins from Gaussian noise at the lowest resolution r_1 and progressively increases the resolution of the generated distribution until the target resolution is reached. Within each stage, if the resolution remains unchanged at the next timestep, standard conditional diffusion denoising is performed. When the resolution changes, the current-stage distribution is first projected to the target stage, and then upsampling and noise are re-injected to match the higher-resolution distribution corresponding to the next timestep. This multi-resolution cascaded inference ensures seamless resolution transitions across stages while preserving the temporal consistency of the generation process. The complete pseudocode is provided in Algorithm 2 in Appendix A.

4 Experiments

4.1 Experiments Setup

Baseline. To assess the generalizability of RMD, we conduct extensive experiments across both image and video generation tasks. For text-to-image synthesis, we evaluate our method on various representative architectures, including the UNet-based Stable Diffusion XL (SDXL) [27] as well as the Transformer-based PixArt-α [3] and Stable Diffusion 3.5 medium (SD3.5) [6]. Specifically, we evaluate our method against two categories of publicly available distillation frameworks: (1) adversarial distillation, including SDXL-Turbo [30] and SDXL-Lighting [13] (for SDXL), YOSO [21] (for PixArt-α), and SD3.5-Turbo [1] (for SD3.5); and (2) distribution matching methods, such as DMD2 [47] and TDM [22]. Furthermore, to demonstrate scalability to large-scale video models, we extend our evaluation to Wan2.1-T2V-14B [39], where we conduct a comparison with distribution matching approaches.

Metrics. For image generation, following TDM [22], we use Human Preference Score v2.1 (HPS) [45] for human-centric alignment, Aesthetic Score (AeS) [31] for perceptual beauty, and CLIP Score for text-visual semantic consistency. These metrics are cross-validated on images generated from the HPS v2.1 prompt set. For video generation, following Bottleneck Sampling [41], we use VBench [10] to evaluate comprehensive video quality (e.g., temporal consistency and motion smoothness) and T2V-Compbench [36] to assess compositional capabilities. For both benchmarks, we use their provided original prompts for video generation.

Furthermore, an extensive user study is conducted, with details provided in Appendix C.

Implementation Details. As our approach is image-free, training relies exclusively on prompts from JourneyDB [37]. For both base and distilled models, CFG scales are set to 7.5 (SDXL), 3.5 (PixArt-α), 4.5 (SD3.5), and 5.0 (Wan2.1). We employ bilinear interpolation for upsampling. All benchmarks and speed measurements are performed on a single NVIDIA A100 GPU. In our primary experiments, we scale the resolution from 512px to 1024px for image synthesis and from 480p (834×480) to 720p (1280×720) for video generation. This is implemented via a two-stage ($K = 2$) multi-resolution strategy, using $\text{logSNR}_1 = -2.5$ as the transition threshold. For instance, in a 4-step SDXL inference, the trajectory is partitioned into a high-noise interval ($t \in [502, 1000]$) processed at low resolution, followed by a refinement stage ($t \in [0, 502]$) at high resolution. Model-specific schedules and further details are provided in Appendix B. The distillation training cost of RMD is comparable to TDM and lower than DMD2 (see Appendix B for detailed comparisons).

Table 1: Quantitative comparison of RMD with state-of-the-art methods in text-to-image generation. The best results among distillation methods are highlighted in **bold**.

Method	Backbone	NFE	Speedup	HPS↑					Aes↑	CLIP Score↑
				Animation	Art	Painting	Photo	Average		
Base Model-1024	SDXL	40×2	1.0×	27.73	31.16	28.93	28.73	29.14	6.48	35.64
SDXL-Turbo-512	SDXL	4	20.0×	31.20	29.72	29.56	27.36	29.46	6.41	34.36
SDXL-Lighting	SDXL	4	20.0×	32.46	31.07	31.23	28.57	30.83	6.54	34.62
DMD2	SDXL	4	20.0×	32.75	31.79	31.79	28.87	31.30	6.71	35.10
TDM	SDXL	4	20.0×	33.64	**32.37**	**32.18**	29.51	31.93	6.70	35.03
RMD (Ours)	SDXL	2 + 2	**33.4**×	**33.71**	32.14	31.94	**30.16**	**31.99**	**6.73**	**35.13**
Base Model-1024	PixArt-α	25×2	1.0×	32.22	30.71	30.45	29.58	30.74	6.73	34.12
YOSO-512	PixArt-α	4	12.5×	31.40	31.18	31.26	28.15	30.60	6.23	31.83
DMD2	PixArt-α	4	12.5×	32.56	31.57	31.57	29.76	31.36	6.71	33.53
TDM	PixArt-α	4	12.5×	33.07	32.25	32.23	30.64	32.05	6.82	33.63
RMD (Ours)	PixArt-α	2 + 2	**21.0**×	**33.24**	**32.42**	**32.55**	**30.71**	**32.23**	**6.89**	**33.89**
Base Model-1024	SD3.5	40×2	1.0×	31.86	30.99	30.75	27.58	30.29	6.49	34.76
SD3.5-Turbo	SD3.5	4	20.0×	30.07	28.25	28.82	25.78	28.23	6.35	**32.97**
DMD2	SD3.5	4	20.0×	31.38	30.47	31.20	28.65	30.42	6.32	32.16
TDM	SD3.5	4	20.0×	31.24	30.20	30.94	27.74	30.03	6.37	32.25
RMD (Ours)	SD3.5	2 + 2	**32.0**×	**31.62**	**30.50**	**31.22**	**28.89**	**30.56**	**6.38**	32.51

4.2 Main Results

Results on Text-to-Image Generation We evaluate the text-to-image generation performance of RMD across multiple representative backbones. We adopt a two-stage inference strategy consisting of two low-resolution steps followed by two high-resolution steps, denoted as an NFE of 2+2. The results are presented in Tab. 1. From the results, we draw the following observations: (1)

Table 2: Quantitative Results on Text-to-Video Generation. All methods use Wan2.1-T2V-14B as the backbone and are evaluated via VBench and T2V-CompBench (average) using original prompts. The best results among distillation methods are highlighted in **bold**.

Method	NFE	Speedup	Total score↑	Quality score↑	Semantic score↑	T2V-Compbench↑
Base Model-720p	50×2	1.0×	83.75	85.44	76.95	54.17
DMD2	6	16.7×	80.30	81.98	73.60	52.81
TDM	6	16.7×	80.48	81.85	75.00	52.27
RMD (Ours)	3 + 3	**25.6**×	**82.51**	**84.37**	**75.05**	**54.00**

Fig. 5: Visual comparison of our model with DMD2, TDM, and the base model. All distilled models are evaluated with 6 sampling steps, while the teacher model uses 50 steps with classifier-free guidance. For fair comparison, all results are generated using identical noise seeds and text prompts. Left prompt: A basketball player soaring through the air for a thunderous slam dunk. Right prompt: A frog jumping into a bowl of water, splashing droplets onto the surrounding grass.

Distribution matching methods frequently outperform the base models in certain metrics. This superiority is primarily attributable to the transition from error-prone multi-step trajectories to direct mappings, thereby eliminating cumulative sampling drift. (2) Our method consistently outperforms baselines via cross-resolution distillation. By anchoring core semantic layouts at lower resolutions during early denoising, the model expands its effective receptive field and suppresses high-frequency artifacts. This prioritizes global structural coherence before transitioning to fine-grained local refinement. (3) RMD maintains high fidelity while delivering superior inference efficiency—achieving a 33.4× acceleration compared to base SDXL and outperforming current step-distillation baselines. Relative to distilled baselines, RMD achieves 1.67× (SDXL), 1.68× (PixArt-α), 1.60× (SD3.5), and 1.53× (Wan2.1) speedup. Qualitative examples are provided in Appendix E.

Results on Text-to-Video. We extend RMD to the Wan2.1-T2V-14B as the base model. While SOTA baselines like DMD2 and TDM are constrained to a fixed 6-step NFE, RMD employs a 3+3 multi-stage strategy. Quantitative results in Tab. 2 show RMD outperforming competitors across all metrics, achieving a 25.6× speedup—over 50% more efficient than 6-step baselines. Qualitative com-

Table 3: Ablation study of the proposed components. Evaluations are conducted on SDXL/Wan2.1 backbone. The best results are highlighted in **bold**.

Target Resolution	Components		Evaluation Metrics			
	RM	UP	HPS↑	Aes↑	CLIP Score↑	Vbench↑
512px/480p	×	×	22.64	6.15	28.76	80.01
	✓	×	**30.28**	**6.63**	**34.81**	**80.81**
1024px/720p	×	✓	29.57	6.58	33.68	81.11
	✓	✓	**31.99**	**6.73**	**35.13**	**82.51**

Fig. 6: Visual ablation comparison of models with and without RM on SDXL at a 1024px target resolution. The corresponding prompts are provided in Appendix D.

parisons (Fig. 5) reveal that RMD preserves superior motion details and semantic coherence, whereas DMD2 exhibits limited temporal dynamics. These findings confirm RMD's ability to substantially accelerate video synthesis without compromising quality.

4.3 Ablation Study

Component Ablation Analysis. Tab. 3 isolates the effects of the Cross-Resolution Distribution Matching (RM) and Upsampling (UP) modules under a unified evaluation protocol. During training, all models are initialized at a base resolution of 512px/480p, while the target resolution is determined by the UP module, resulting in outputs at 512px/480p or 1024px/720p. Applying RM alone (i.e., without UP) yields substantial improvements over the baseline; these gains primarily arise from RM's ability to eliminate the distribution gap between low- and high-resolution representations. In contrast, a naive cascaded upsampling strategy without RM performs poorly (see Fig. 6), as the low-resolution stage fails to establish robust semantic structures that can be preserved and refined during subsequent upsampling. When cascaded upsampling is combined with RM, the two components exhibit clear complementarity: UP alleviates the high

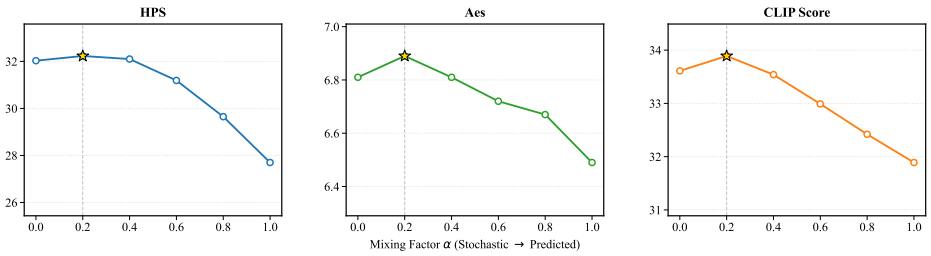

Fig. 7: Ablation study of noise re-injection strategies. Effect of varying the mixing factor α, which controls the balance between predicted and stochastic noise. The backbone is PixArt-α (note that α here is part of the model name).

Table 4: Hyperparameter analysis on logSNR-driven step allocation. The logSNR_1 governs the assignment of sampling steps (N_{512} and N_{1024}) across 512px and 1024px resolutions. The backbone is Pixart-α.

$N_{512} + N_{1024}$	logSNR_1	Speedup	HPS↑	Aes↑	CLIP Score↑
$0 + 4$	-	$1\times$	32.05	6.82	33.63
$1 + 3$	-6.00	$1.25\times$	32.12	**6.87**	33.64
$2 + 2$	-2.50	$1.68\times$	**32.13**	6.83	**33.86**
$3 + 1$	0.00	**2.54**$\times$	24.79	6.03	32.22

inference cost at early high-resolution stages, while RM ensures cross-resolution distribution alignment. Together, they achieve the best overall performance.

Ablation on Noise Re-Injection Strategy. We analyze the mixing factor α in Eq. (10) under an NFE = 2+2 multi-resolution schedule (Fig. 7). Pure Gaussian noise ($\alpha = 0$) enables cross-resolution alignment but fails to inherit the teacher's ODE trajectory, leading to weak structural consistency. Pure predicted noise ($\alpha = 1.0$) strictly follows the teacher trajectory but degrades under large resolution gaps, as upsampling artifacts are treated as hard guidance. The optimal setting $\alpha = 0.2$ balances trajectory inheritance and stochastic flexibility, effectively bridging the multi-resolution gap and achieving the best generation fidelity.

4.4 Analysis of Resolution Trajectory Division

We analyze the impact of the threshold logSNR_1 on the distribution of sampling steps across scales. As shown in Tab. 4, decreasing logSNR_1 shifts the computational budget toward the lower resolution, yielding a higher speedup by mitigating the quadratic attention complexity at 1024px. The configuration with $N_{512} = 2$ and $N_{1024} = 2$ achieves the best balance between efficiency and quality, delivering a $1.68\times$ speedup while attaining the highest HPS (32.13) and CLIP (33.86) scores. While the $1 + 3$ configuration offers a minor improvement

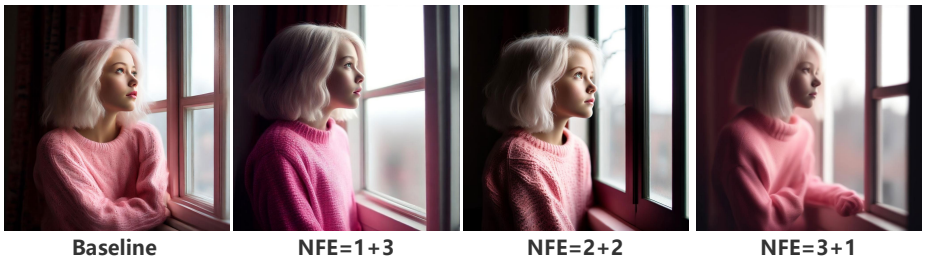

Fig. 8: Qualitative comparison of different step allocations. Prompt: A white-haired girl in a pink sweater looks out a window in her bedroom.

Table 5: Effect of multi-stage resolution training with RMD on SDXL. We evaluate progressive resolution schedules (256px → 512px → 1024px), corresponding to logSNR_1 and logSNR_2, which induce different NFE allocations across stages. The backbone model is SDXL. ‡ NFE per stage; omitted if 1.

Method	Resolution (NFE)‡	$\text{logSNR}_1, \text{logSNR}_2$	Speedup	HPS↑	Aes↑	CLIP↑
Baseline	1024(4)	–	1.00×	31.93	6.70	35.03
	512(2)→1024(2)	-2.5	1.56×	29.57	6.58	33.68
RMD	512(2)→1024(2)	-2.5	1.56×	**31.99**	**6.73**	**35.13**
	256→512→1024(2)	-6.0,-2.5	1.73×	31.59	6.70	34.84
	256→512(2)→1024	-6.0,-0.0	2.54×	31.48	6.68	34.77
	256(2)→512→1024	-2.5,-0.0	2.90×	31.03	6.67	34.57
	128→256→512→1024	-6.0, -2.5, -0.0	**3.08**×	29.48	6.59	33.41

in aesthetic score (6.87), this marginal gain is achieved at the expense of a significant reduction in inference speed. Conversely, compressing the high-resolution refinement to a single step (3+1) leads to a sharp degradation in quality. This observation is visually substantiated in Fig. 8, where the 2+2 assignment preserves intricate details and structural sharpness, whereas the $3 + 1$ scheme manifests as significant blurriness and a total loss of fine textures. Consequently, we adopt $\text{logSNR}_1 = -2.5$ as the default setting for our model to achieve an optimal balance between efficiency and generative fidelity.

4.5 Effectiveness of Multi-stage Resolution Training

The evaluation of our resolution progressive strategy on SDXL, summarized in Tab. 5, demonstrates that RMD effectively optimizes the trade-off between generative quality and inference speed. Specifically, the 2-stage RMD configuration ($2 + 2$ NFE) outperforms the single-resolution baseline across all metrics—achieving an HPS of 31.99 and an Aesthetic score of 6.73—while providing a 1.56× speedup. This performance significantly exceeds a naive 2-stage baseline (HPS 29.57), confirming that our RM module successfully bridges the distribu-

tion gap during resolution transitions. Furthermore, extending this to a 3-stage 256px $\rightarrow$ 512px $\rightarrow$ 1024px strategy enables even greater efficiency, reaching a 2.90$\times$ speedup with the $2 + 1 + 1$ NFE allocation while maintaining favorable visual fidelity. These results validate the scalability of our coarse-to-fine distillation logic, proving it can achieve high-quality synthesis with substantially reduced computational overhead.

5 Conclusions

We propose a cross-resolution distribution matching distillation (RMD) framework for few-step, coarse-to-fine multi-resolution cascaded generation. RMD explicitly resolves distribution mismatch across resolutions and enables progressive resolution expansion within a limited number of inference steps, overcoming the efficiency bottleneck of conventional step-reduction distillation while preserving high-fidelity synthesis. Extensive experiments on image and video generation validate its effectiveness. RMD consistently achieves state-of-the-art sampling speed without sacrificing visual quality, providing a scalable solution for accelerating future generative models.

6 Acknowledgements

This work was supported by TokensEngine 1.0 under Grant No. 9473277, the National Natural Science Foundation of China under Grant No. 62276131, the Natural Science Foundation of Jiangsu Province under Grant No. BK20240081, and the Special Research Project on Teaching Reform of General Artificial Intelligence Courses in Jiangsu Undergraduate Universities under Grant No. ZNT-10.

References

1. Bandyopadhyay, H., Entezari, R., Scott, J., Adithyan, R., Song, Y., Jampani, V.: Sd3.5-flash: Distribution-guided distillation of generative flows. CoRR (2025)
2. Cai, X., Huang, Q., Kang, Z., Li, H., Liang, S., Ma, L., Ren, S., Wei, X., Xie, R., Zhang, T.: Longcat-video technical report. CoRR (2025)
3. Chen, J., Yu, J., Ge, C., Yao, L., Xie, E., Wang, Z., Kwok, J.T., Luo, P., Lu, H., Li, Z.: Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In: ICLR (2024)
4. Chen, S., Ge, C., Zhang, S., Sun, P., Luo, P.: Pixelflow: Pixel-space generative models with flow. arXiv preprint arXiv:2504.07963 (2025)
5. Chen, T.: On the importance of noise scheduling for diffusion models. In: ICML (2023)
6. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., et al.: Scaling rectified flow transformers for high-resolution image synthesis. arXiv preprint arXiv:2403.03206 (2024)
7. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Communications of the ACM **63**(11), 139–144 (2020)

8. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: NeurIPS (2020)
9. Hoogeboom, E., Heek, J., Salimans, T.: Simple diffusion: End-to-end diffusion for high resolution images. In: ICLR (2023)
10. Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., Zhang, Y., Wu, T., Jin, Q., Chanpaisit, N., Wang, Y., Chen, X., Wang, L., Lin, D., Qiao, Y., Liu, Z.: Vbench: Comprehensive benchmark suite for video generative models. In: CVPR (2024)
11. Jiang, D., Liu, D., Wang, Z., Wu, Q., Li, L., Li, H., Jin, X., Liu, D., Li, Z., Zhang, B., Wang, M., Hoi, S.C.H., Gao, P., Yang, H.: Distribution matching distillation meets reinforcement learning. CoRR, abs/2511.13649 (2025)
12. Kim, D., Lai, C., Liao, W., Murata, N., Takida, Y., Uesaka, T., He, Y., Mitsufuji, Y., Ermon, S.: Consistency trajectory models: Learning probability flow ODE trajectory of diffusion. In: ICLR (2024)
13. Lin, S., Wang, A., Yang, X.: Sdxl-lightning: Progressive adversarial diffusion distillation. CoRR (2024)
14. Lipman, Y., Chen, R.T.Q., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: ICLR (2023)
15. Liu, F., Zhang, S., Wang, X., Wei, Y., Qiu, H., Zhao, Y., Zhang, Y., Ye, Q., Wan, F.: Timestep embedding tells: It's time to cache for video diffusion model. In: CVPR. pp. 7353–7363 (2025)
16. Liu, X., Huang, C., Lin, X., Wang, Y.: Flow straight and fast: Learning to generate and transfer data with rectified flow. In: ICLR (2023)
17. Liu, X., Zhang, X., Ma, J., Peng, J., Liu, Q.: Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In: ICLR (2024)
18. Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., Zhu, J.: Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. Mach. Intell. Res. (2025)
19. Lu, Y., Ren, Y., Xia, X., Lin, S., Wang, X., Xiao, X., Ma, A.J., Xie, X., Lai, J.: Adversarial distribution matching for diffusion distillation towards efficient image and video synthesis. CoRR, abs/2507.18569 (2025)
20. Luo, W., Hu, T., Zhang, S., Sun, J., Li, Z., Zhang, Z.: Diff-instruct: A universal approach for transferring knowledge from pre-trained diffusion models. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S. (eds.) NeurIPS (2023)
21. Luo, Y., Chen, X., Qu, X., Hu, T., Tang, J.: You only sample once: Taming one-step text-to-image synthesis by self-cooperative diffusion gans. In: ICLR (2025)
22. Luo, Y., Hu, T., Sun, J., Cai, Y., Tang, J.: Learning few-step diffusion models by trajectory distribution matching. arXiv preprint arXiv:2503.06674 (2025)
23. Ma, Y., Cheng, B., Liu, S., Zhou, H., Wu, X., Wu, L., Leng, D., Yin, Y.: Nami: Efficient image generation via bridged progressive rectified flow transformers. arXiv preprint arXiv:2503.09242 (2025)
24. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J., Ermon, S.: Sdedit: Guided image synthesis and editing with stochastic differential equations. In: ICLR (2022)
25. Meng, C., Rombach, R., Gao, R., Kingma, D.P., Ermon, S., Ho, J., Salimans, T.: On distillation of guided diffusion models. In: CVPR. pp. 14297–14306 (2023)
26. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: ICCV. pp. 4172–4182 (2023)
27. Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., Rombach, R.: SDXL: improving latent diffusion models for high-resolution image synthesis. In: ICLR (2024)
28. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR. pp. 10674–10685 (2022)

29. Salimans, T., Ho, J.: Progressive distillation for fast sampling of diffusion models. In: ICLR (2022)
30. Sauer, A., Lorenz, D., Blattmann, A., Rombach, R.: Adversarial diffusion distillation. In: ECCV (2024)
31. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., Schramowski, P., Kundurthy, S., Crowson, K., Schmidt, L., Kaczmarczyk, R., Jitsev, J.: Laion-5b: An open large-scale dataset for training next generation image-text models (2022)
32. Shen, H., Zhang, J., Xiong, B., Hu, R., Chen, S., Wan, Z., Wang, X., Zhang, Y., Gong, Z., Bao, G., Tao, C., Huang, Y., Yuan, Y., Zhang, M.: Efficient diffusion models: A survey. Trans. Mach. Learn. Res. **2025** (2025)
33. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. In: ICLR (2021)
34. Song, Y., Dhariwal, P.: Improved techniques for training consistency models. In: ICLR (2024)
35. Song, Y., Dhariwal, P., Chen, M., Sutskever, I.: Consistency models. In: ICML (2023)
36. Sun, K., Huang, K., Liu, X., Wu, Y., Xu, Z., Li, Z., Liu, X.: T2v-compbench: A comprehensive benchmark for compositional text-to-video generation. In: CVPR (2025)
37. Sun, K., Pan, J., Ge, Y., Li, H., Duan, H., Wu, X., Zhang, R., Zhou, A., Qin, Z., Wang, Y., Dai, J., Qiao, Y., Wang, L., Li, H.: Journeydb: A benchmark for generative image understanding. In: NeurIPS (2023)
38. Team, T.H.: Hunyuanvideo: A systematic framework for large video generative models. arXiv preprint (2025)
39. Team, W.: Wan: Open and advanced large-scale video generative models. arXiv preprint (2025)
40. Teng, J., Zheng, W., Ding, M., Hong, W., Wangni, J., Yang, Z., Tang, J.: Relay diffusion: Unifying diffusion process across resolutions for image synthesis. In: ICLR (2024)
41. Tian, Y., Xia, X., Ren, Y., Lin, S., Wang, X., Xiao, X., Tong, Y., Yang, L., Cui, B.: Training-free diffusion acceleration with bottleneck sampling. arXiv preprint arXiv:2503.18940 (2025)
42. Wang, B., Vastola, J.J.: Diffusion models generate images like painters: an analytical theory of outline first, details later. CoRR, abs/2303.02490 (2023)
43. Wang, F., Huang, Z., Bergman, A.W., Shen, D., Gao, P., Lingelbach, M., Sun, K., Bian, W., Song, G., Liu, Y., Wang, X., Li, H.: Phased consistency models. In: NeurIPS (2024)
44. Wu, T., Li, R., Zhang, L., Ma, K.: Diversity-preserved distribution matching distillation for fast visual synthesis. CoRR, abs/2602.03139 (2026)
45. Wu, X., Hao, Y., Sun, K., Chen, Y., Zhu, F., Zhao, R., Li, H.: Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. CoRR (2023)
46. Xu, Y., Nie, W., Vahdat, A.: One-step diffusion models with f-divergence distribution matching. CoRR, abs/2502.15681 (2025)
47. Yin, T., Gharbi, M., Park, T., Zhang, R., Shechtman, E., Durand, F., Freeman, B.: Improved distribution matching distillation for fast image synthesis. In: NeurIPS (2024)
48. Yin, T., Gharbi, M., Zhang, R., Shechtman, E., Durand, F., Freeman, W.T., Park, T.: One-step diffusion with distribution matching distillation. In: CVPR (2024)

49. Yuan, Z., Zhang, H., Pu, L., Ning, X., Zhang, L., Zhao, T., Yan, S., Dai, G., Wang, Y.: Ditfastattn: Attention compression for diffusion transformer models. In: NeurIPS (2024)
50. Zhang, S., Li, W., Chen, S., Ge, C., Sun, P., Zhang, Y., Jiang, Y., Yuan, Z., Peng, B., Luo, P.: Flashvideo: Flowing fidelity to detail for efficient high-resolution video generation. CoRR, abs/2502.05179 (2025)
51. Zhang, Y., Yang, H., Zhang, Y., Hu, Y., Zhu, F., Lin, C., Mei, X., Jiang, Y., Peng, B., Yuan, Z.: Waver: Wave your way to lifelike video generation. CoRR, abs/2508.15761 (2025)
52. Zhang, Y., Xing, J., Xia, B., Liu, S., Peng, B., Tao, X., Wan, P., Lo, E., Jia, J.: Training-free efficient video generation via dynamic token carving. CoRR, abs/2505.16864 (2025)
53. Zhou, M., Zheng, H., Wang, Z., Yin, M., Huang, H.: Score identity distillation: Exponentially fast distillation of pretrained diffusion models for one-step generation. In: ICML. vol. 235, pp. 62307–62331 (2024)