SCALING IMAGE AND VIDEO GENERATION VIA TEST-TIME EVOLUTIONARY SEARCH

Anonymous authors
Paper under double-blind review

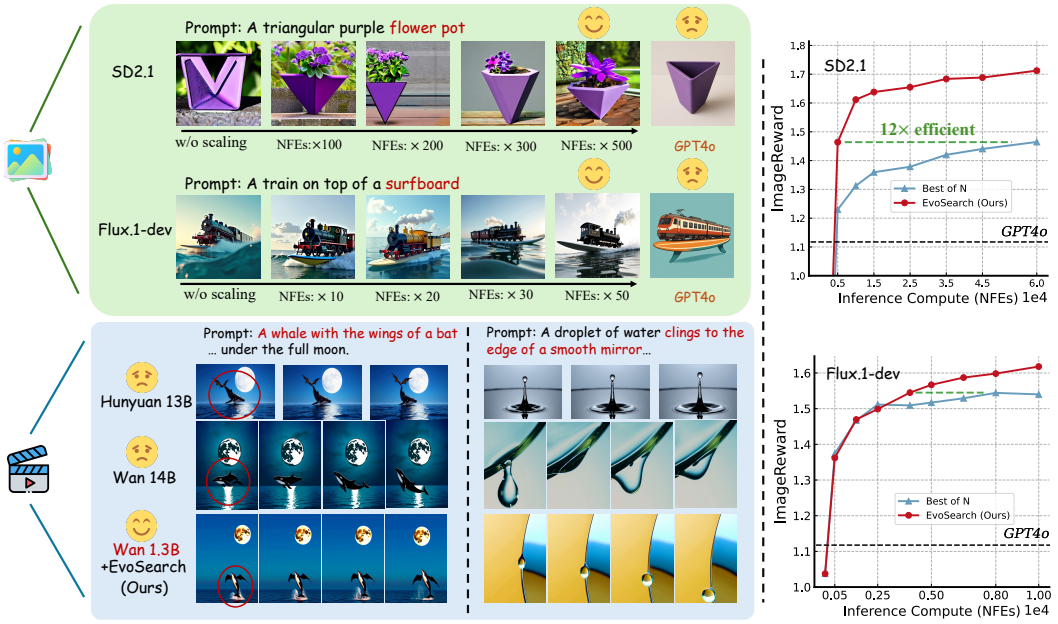


Figure 1: We propose **Evo**lutionary **Search** (EvoSearch), a novel, generalist, and compute-optimal test-time scaling framework applicable to both image and video generation tasks. EvoSearch significantly enhances sample quality through strategic computation allocation during inference, **enabling Stable Diffusion 2.1 to be comparable to GPT4o, and Wan 1.3B to outperform Wan 14B model and Hunyuan 13B model with** $10\times$ **fewer parameters**.

ABSTRACT

As the marginal cost of scaling computation (data and parameters) during model pre-training continues to increase substantially, test-time scaling (TTS) has emerged as a promising direction for improving generative model performance by allocating additional computation at inference time. While TTS has demonstrated significant success across multiple language tasks, there remains a notable gap in understanding the test-time scaling behaviors of image and video generative models (diffusion-based or flow-based models). Although recent works have initiated exploration into inference-time strategies for vision tasks, these approaches face critical limitations: being constrained to task-specific domains, exhibiting poor scalability, or falling into reward over-optimization that sacrifices sample diversity. In this paper, we propose **Evo**lutionary **Search** (EvoSearch), a novel, generalist, and efficient TTS method that effectively enhances the scalability of both image and video generation across diffusion and flow models, without requiring additional training or model expansion. EvoSearch reformulates test-time scaling for diffusion and flow models as an evolutionary search problem, leveraging principles from biological evolution to efficiently explore and refine the denoising trajectory. By incorporating carefully designed selection and mutation mechanisms tailored to the stochastic differential equation denoising process, EvoSearch iteratively generates higher-quality offspring while preserving population diversity. Through extensive evaluation across both diffusion and flow architectures for image and video generation tasks, we demonstrate that our method consistently outperforms existing approaches, achieves higher diversity, and shows strong generalizability to unseen evaluation metrics.

1 INTRODUCTION

Generative models have witnessed remarkable progress across various fields, including language (Achiam et al., 2023; Jaech et al., 2024; Guo et al., 2025), image (Esser et al., 2024; Labs, 2024), and video generation (Brooks et al., 2024; Kong et al., 2024; Wang et al., 2025), demonstrating powerful capabilities to capture complex data distributions. The central driver of this success is their ability to scale up during training by increasing data volumes, computational resources, and model sizes. This scaling behavior during the training process is commonly described as Scaling Laws (Hoffmann et al., 2022; Kaplan et al., 2020). Despite these advancements, further scaling at training time is increasingly reaching its limits due to the rapid depletion of available internet data and increasing computational costs. Post-training alignment (Tie et al., 2025) has been proven to be effective in addressing this challenge. For diffusion and flow models, these approaches typically include parameter tuning via reinforcement learning (Black et al., 2023; Fan et al., 2023) or direct reward gradient backpropagation (Clark et al., 2024; Prabhudesai et al., 2024). However, they suffer from reward over-optimization due to their mode-seeking behavior, high computational costs, and requirement of direct model weight access. Alternative methods (Ahn et al., 2024; Zhou et al., 2024) propose directly optimizing initial noise, as some lead to better generations than others, but demand specialized training and struggle with cross-model generalization.

Recent advances in large language models (LLMs) have expanded to test-time scaling (TTS) (Brown et al., 2024; Wu et al., 2025), showing promising results to complement traditional training-time scaling law. TTS (Zhang et al., 2025) allocates additional computation budget during inference, offering a novel paradigm for improving generation quality without additional training. However, diffusion and flow models present unique challenges for test-time scaling, since they must navigate the complex, high-dimensional state space along the denoising trajectory, where existing methods in LLMs struggle to transfer effectively. Current approaches of test-time scaling for diffusion and flow models include (i) best-of-N sampling (Ma et al., 2025; Liu et al., 2025a), which, despite its simplicity, suffers from severe search inefficiency in high-dimensional noise spaces; and (ii) particle sampling (Kim et al., 2025b; Singhal et al., 2025a), which, while enabling search across the entire denoising trajectory, compromises both exploration capability and generation diversity due to its reliance on initial candidate pools. These simple heuristic designs lack fundamental adaptability to the complex generation pathways, leading to sample diversity collapse and inefficient computation.

In this paper, we aim to address the above critical challenges and develop a general and efficient test-time scaling method that is versatile for both image and video generation across diffusion and flow models without parameter tuning or gradient backpropagation. To enable test-time scaling of flow models, we transform their deterministic sampling process (ODE) into a stochastic process (SDE), thereby broadening the generation space, which paves the way for a unified framework for inference-time optimization. Through systematic analysis of latent spaces along the denoising trajectory, including both starting Gaussian noises and intermediate states, we find that neighboring states in the latent space exhibit similar generation qualities, suggesting that high-quality samples are not solely isolated. Based on this insight, we propose **Evo**lutionary **Search** (EvoSearch), a novel test-time scaling method inspired by biological evolution. EvoSearch reframes test-time scaling of image and video generation as an evolutionary search problem, incorporating selection and mutation mechanisms specifically designed for the denoising process in both diffusion and flow models. At each generation, EvoSearch first selects high-reward parents while preserving population diversity, and then generates new offspring through our designed denoising-aware mutation mechanisms to explore new states, enabling iterative improvement in sample quality. The key insight of EvoSearch is to actively explore high-reward particles through evolutionary mechanisms, overcoming the limitations of previous search methods that are confined to a fixed candidate space. To optimize computational efficiency, we dynamically search along the denoising trajectory, progressing from Gaussian noises to states at larger denoising steps, thereby continuously reducing computational costs as we approach the terminal states. Through extensive experiments on both text-conditioned image generation and video generation tasks, we find that EvoSearch achieves substantial improvements in sample quality and human-preference alignment as test-time compute increases.

We summarize our key contributions as follows: (i) We propose EvoSearch, a novel, generalist, and efficient TTS framework which enhances generation quality by allocating more compute during inference, unifying optimization for both diffusion and flow generative models. (ii) Based on our observations of latent space structure, we design specialized selection and mutation mechanisms tailored to the denoising process, effectively enhancing exploration while maintaining diversity. (iii) Extensive experiments show that EvoSearch effectively improves generative model performance by scaling up inference-time compute, outperforming competitive baselines across both image and video

generation tasks. Notably, EvoSearch enables SD2.1 (Dhariwal & Nichol, 2021) to be comparable to GPT4o, and allows the Wan 1.3B model (Wang et al., 2025) to achieve competitive performance with the $10\times$ larger Wan 14B model. Our project is available at `evosearch.github.io`.

2 PRELIMINARY

Diffusion Models and ODE-to-SDE Transformation of Flow Models. Both diffusion models and flow models map the source distribution, often a standard Gaussian distribution, to a true data distribution p_0. A forward diffusion process progressively perturbing data to noise, defined as $\boldsymbol{x}_t = \alpha_t \boldsymbol{x}_0 + \sigma_t \varepsilon$, where $\varepsilon \in \mathcal{N}(0, I)$ is the added noise at timestep $t \in [0, T]$, and (α_t, σ_t) denote the noise schedule. To restore from diffused data, diffusion models naturally utilize an SDE-based sampler during inference (Song et al., 2020b;a), which introduces stochasticity at each denoising step as follows: $\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}}\left((\boldsymbol{x}_t - \sqrt{1-\alpha_t}\varepsilon_\theta(\boldsymbol{x}_t, t))/\sqrt{\alpha_t}\right) + \sqrt{1-\alpha_{t-1}-\sigma_t^2}\varepsilon_\theta(\boldsymbol{x}_t, t) + \sigma_t \varepsilon_t$. In contrast, flow models learn the velocity $u_t \in \mathbb{R}^d$, which enables sampling of $\boldsymbol{x}_0$ by solving the flow ODE (Song et al., 2020b) backward from $t = T$ to $t = 0$: $\boldsymbol{x}_{t-1} = \boldsymbol{x}_t + u_t(\boldsymbol{x}_t)dt$, leading all $\boldsymbol{x}_{t-1}$ drawn from $\boldsymbol{x}_t$ identical. This restricts the applicability of test-time scaling search methods like particle sampling and our proposed EvoSearch in flow models (Kim et al., 2025a), since the sampling process lacks stochasticity beyond initial noise. To address this limitation, we transform the deterministic Flow-ODE into an equivalent SDE process. Following previous works (Albergo et al., 2023; Ma et al., 2024; Patel et al., 2024; Kim et al., 2025a; Singh & Fischer, 2024), we rewrite the ODE sampling process by $d\boldsymbol{x}_t = \left(u_t(\boldsymbol{x}_t) - \frac{\sigma_t^2}{2}\nabla \log p_t(\boldsymbol{x}_t)\right) dt + \sigma_t d\boldsymbol{w}$, where the score $\log p_t(\boldsymbol{x}_t)$ can be computed by velocity u_t (see Eq. (13) in (Singh & Fischer, 2024)), and $d\boldsymbol{w}$ injects stochasticity at each sampling step.

Evolutionary Algorithms. Evolutionary algorithms (EAs) (Koza, 1992; Bäck, 1996) are biologically inspired, gradient-free methods that found effective in optimization (Goldberg, 1989; Grefenstette, 1993; Vikhar, 2016), algorithm search (Co-Reyes et al., 2021; Real et al., 2020), and neural architecture search (Real et al., 2019; Yang et al., 2020; So et al., 2019). The key idea of EAs is mimicking the process of natural evolution (Ao, 2005), by maintaining a population of solutions that evolve over generations. EAs involve initializing random solutions, evaluating fitness, selecting parents, and applying genetic operators (crossover and mutation) to create offspring that constitute the next generation. Due to the diversity within populations and the mutation operations, EAs excel at global optimization and solving multimodal problems compared to traditional local search methods.

3 RELATED WORK

Alignment for Diffusion and Flow Models. Aligning pre-trained diffusion and flow generative models can be achieved by guidance (Dhariwal & Nichol, 2021; Song et al., 2020b) or fine-tuning (Lee et al., 2023; Fan & Lee, 2023), which aim to enhance sample quality by steering outputs towards a desired target distribution. Guidance methods (Ho et al., 2022; Song et al., 2023a; Chung et al., 2023; Bansal et al., 2023; Song et al., 2023b; Guo et al., 2024b) rely on predicting clean samples from noisy data and differentiable reward functions to calculate guidance. Typical fine-tuning methods involve supervised fine-tuning (Lee et al., 2023; **?**; Fan & Lee, 2023; Wu et al., 2023c), RL fine-tuning (Black et al., 2023; Fan et al., 2023; Liu et al., 2025b; Miao et al., 2024), DPO-based policy optimization (Wallace et al., 2024; Yang et al., 2024; Liang et al., 2024; Liu et al., 2024; Zhang et al., 2024), direct reward backpropagation (Clark et al., 2024; Xu et al., 2023; Prabhudesai et al., 2024), stochastic optimization (Domingo-Enrich et al., 2024; Yeh et al., 2024), and noise optimization (Ahn et al., 2024; Zhou et al., 2024; Tang et al., 2024; Guo et al., 2024a; Eyring et al., 2024). Previous works (Miao et al., 2024) calculate reward signals on fully denoised samples for ensuring trustworthy feedback. However, these methods require additional dataset curation and parameter tuning, and can distort alignment or reduce sample diversity due to their mode-seeking behavior and reward over-optimization. In contrast, our proposed EvoSearch method offers significant advantages through its universal applicability across any reward function and model architecture (including flow-based, diffusion-based, image, and video models) without requiring additional training. Moreover, EvoSearch complements existing fine-tuning methods, as it can be applied to any fine-tuned model to further enhance reward alignment. While a related work (Domingo-Enrich et al., 2024) demonstrates that SDE dynamics are critical for fine-tuning diffusion models, our work is distinctive since we focus on the test-time inference phase instead of the training phase with parameter updates.

Test-Time Scaling in Vision. Several test-time scaling (TTS) methods have been proposed to extend the performance boundaries of image and video generative models. These methods fundamentally

operate as search, with reward models providing judgments and algorithms selecting better candidates. Best-of-N generates N batches of samples and selects the one with the highest reward, which has been validated effective for both image and video generation (Ma et al., 2025; Liu et al., 2025a). More advanced search method for diffusion models is particle sampling (Singhal et al., 2025a; Li et al., 2024; 2025; Singh et al., 2025; Kim et al., 2025b), which resamples particles over the full denoising trajectory based on their importance weights, demonstrating superior results than naive BoN. Video-T1 (Liu et al., 2025a) and other recent works (Yang et al., 2025; Xie et al., 2025; Oshima et al., 2025; Liu et al., 2025a) propose leveraging beam search (Snell et al., 2024) for scaling video generation. However, in the context of diffusion and flow models, we remark that beam search represents a specialized case of particle sampling with a predetermined beam size, as both methodologies iteratively propagate high-reward samples while discarding lower-reward ones in practice. Furthermore, Video-T1 is constrained to autoregressive video models, limiting its applicability to more advanced diffusion and flow generative models. All existing search methods rely heavily on passive filtering, failing to explore new particles actively, while our proposed method, EvoSearch, leverages the idea of natural selection and evolution, enabling the generation of new, higher-quality offspring iteratively. EvoSearch is also a generalist framework with superior scalability and extensive applicability across both diffusion and flow models for image and video generation, contrary to previous methods that are constrained to specific models or tasks.

4 Proposed Method

4.1 Problem Formulation

In this work, we investigate how to efficiently harness additional test-time compute to enhance the sample quality of image and video generative models. Given a pre-trained flow-based or diffusion-based model and a reward function, our objective is to generate samples from the following target distribution (Uehara et al., 2024a; Li et al., 2024; Wu et al., 2023a; Uehara et al., 2024b):

$$p^{\mathrm{tar}} = \frac{1}{\mathcal{Z}} p_0^{\mathrm{pre}}(\boldsymbol{x}_0) \exp\big(\frac{r(\boldsymbol{x}_0)}{\alpha}\big), \tag{1}$$

where $\mathcal{Z}$ denotes a normalization constant (Rafailov et al., 2023; Uehara et al., 2024a) and p_0^{pre} is the pre-trained distribution. Notably, directly sampling from the target distribution is infeasible: the normalization factor $\mathcal{Z}$ requires integrating over the entire sample space, making it computationally intractable for high-dimensional spaces in diffusion and flow models.

4.2 Limitations of Existing Approaches

Test-time approaches to sampling from the target distribution p^{tar} in Eq. equation 1 employ importance sampling (Owen & and, 2000), which generates k particles $\boldsymbol{x}_0^i \sim p_0^{\mathrm{pre}}(\boldsymbol{x}_0)$ and then resamples the particles based on the scores $\exp(r(\boldsymbol{x}_0)/\alpha)$. A straightforward implementation of this concept is best-of-N sampling, which simply generates multiple samples and selects the one with the highest reward. A more sophisticated approach, called particle sampling (Singhal et al., 2025b; Kim et al., 2025b), searches across the entire denoising path $\tau = \{\boldsymbol{x}_T, \cdots, \boldsymbol{x}_k, \cdots, \boldsymbol{x}_0\}$, guiding samples toward trajectories that yield higher rewards. However, both of these methods suffer from fundamental limitations in their efficiency and exploration capabilities. Best-of-N only resamples at the final step ($t = 0$), taking the entire distribution $p_0^{\mathrm{pre}}(\boldsymbol{x}_0) = \int \prod_t \{p_t^{\mathrm{pre}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\} d\boldsymbol{x}_{1:T}$ as its proposal distribution. This passive filtering approach is computationally wasteful, as it expends a large amount of computation generating complete trajectories for samples that ultimately yield low rewards. In contrast, particle sampling can search and resample at each intermediate step along the denoising path, using $p_t^{\mathrm{pre}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ as its proposal distribution at each step t. However, it is still constrained by the fixed initial candidate pool, struggling to actively explore and generate novel states beyond those proposed by p_0^{pre} during the search process. This limitation becomes increasingly restrictive as the search progresses, which leads to restricted performance due to limited exploration and reduced diversity.

To better understand these inherent limitations more concretely, we visualize the behavior of different approaches in Fig. 2. As shown, re-training methods, including RL (DDPO (Black et al., 2023)) and reward backpropagation (Clark et al., 2024), struggle to generalize to the unseen target distribution, largely due to their heavy reliance on pre-trained models and mode-seeking behavior. While test-time search methods (best-of-N and particle sampling) achieve higher rewards than re-training methods, they still fail to capture all modes of the multimodal target distribution, converging to limited regions of the solution space. These findings highlight the need for a novel test-time scaling framework capable of effectively balancing between exploitation and exploration while maintaining computational

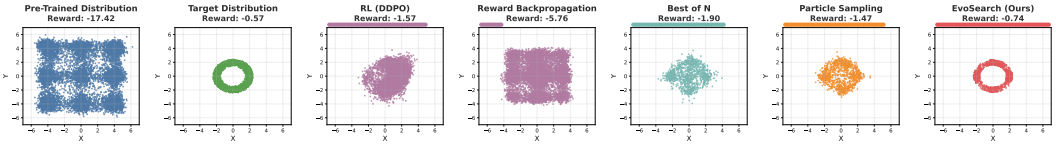


Figure 2: Visualization of a test-time alignment experiment. We train a diffusion model with 3-layer MLP on Gaussian mixtures (pre-trained distribution), with the goal to capture multimodal unseen target distribution, where reward $r(X, Y) = -|X^2 + Y^2 - 4|$. EvoSearch achieves superior performance, capturing all the modes with the highest reward (-0.74).

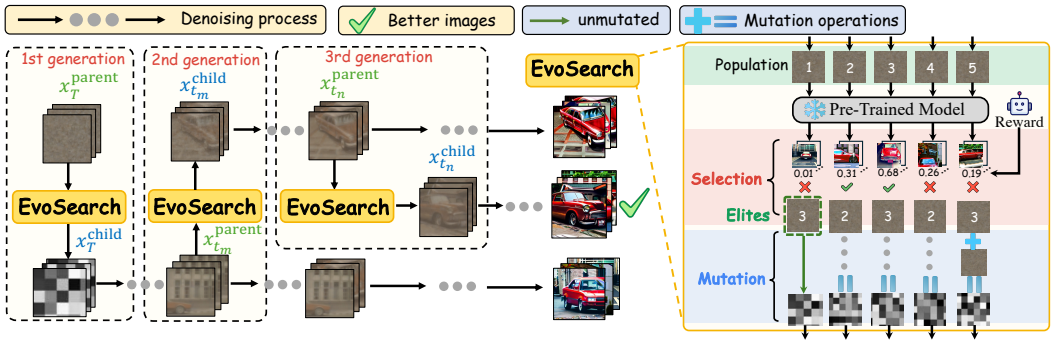


Figure 3: Overview of our method. EvoSearch progressively moves forward along the denoising trajectory to refine and explore new states. At each evolution step defined by $\mathcal{T}$, our proposed *EvoSearch* process generates novel, high-quality offspring x_T^{child} based on the parent population x_t^{parent}. The population size across each generation is defined by schedule $\mathcal{K}$. *EvoSearch* contains evaluation, selection, and mutation operations to ensure the effectiveness of the evolutionary process. The generation quality consistently improves with the progression of the evolutionary search.

efficiency for scaling up. In the following sections, we introduce how our EvoSearch method overcomes these fundamental limitations, which achieves the highest reward with comprehensive mode coverage as shown in Fig. 2.

4.3 Evolutionary Search

We propose **Evo**lutionary **Search** (EvoSearch), a novel evolutionary framework that reformulates the sampling from the target distribution p^{tar} in Eq. 1 at test time as an active evolutionary optimization problem rather than passive filtering. EvoSearch introduces a unified way for achieving efficient and effective test-time scaling across both diffusion and flow models for image and video generation tasks. The overview of our method is provided in Fig. 3. Our algorithm is summarized in Alg 1&2. EvoSearch introduces a novel perspective that reinterprets the denoising trajectory as an evolutionary path, where both the initial noise $\boldsymbol{x}_T$ and the intermediate state $\boldsymbol{x}_t$ can be evolved towards higher-quality generation, actively expanding the exploration space beyond the constraints of the pre-trained model's distribution. Below, we introduce the core components of EvoSearch.

Evolution Schedule. For a typical sampling process in diffusion and flow models, the change between $\boldsymbol{x}_{t-1}$ and $\boldsymbol{x}_t$ is not substantial. Therefore, performing EvoSearch at every sampling step would be computationally wasteful. To address this efficiency problem, EvoSearch defines an evolution schedule $\mathcal{T} = \{T, \cdots, t_m, \cdots, t_n\}$ that specifies the timesteps at which EvoSearch should be conducted. Concretely, EvoSearch first thoroughly optimizes the starting noise $\boldsymbol{x}_T$ to identify high-reward regions in the Gaussian noise space, establishing a strong initialization for the subsequent denoising process. After a high-quality $\boldsymbol{x}_T$ is obtained, EvoSearch progressively applies our proposed evolutionary operations to intermediate states $\boldsymbol{x}_{t_i}$ at predetermined timesteps $t_i \in \mathcal{T}$. This cascading way enables each subsequent generation beginning directly from the cached intermediate state $\boldsymbol{x}_{t_i}$ obtained from the previous generation, instead of repeatedly denoising from $\boldsymbol{x}_T$, eliminating the redundant denoising computations from $\boldsymbol{x}_T \to \boldsymbol{x}_{t_i}$. In practice, we implement this evolution schedule using uniform intervals between timesteps, which significantly reduces computational overhead.

Population Initialization. Following the evolution schedule $\mathcal{T}$, we introduce a corresponding population size schedule $\mathcal{K} = \{k_T, \cdots, k_m, \cdots, k_n\}$, where each k_i specifies the population size for the generation at timestep t_i. This adaptive approach enables flexible trade-offs between computational cost and exploration of the state space (Appendix B.1 for further analysis on ablation of $\mathcal{K}$ and $\mathcal{T}$). The initial generation of EvoSearch begins with k_T randomly sampled Gaussian noises $\{x_T^i\}_{i=1}^{k_T}$ at timestep $t = T$, which serve as the first-generation parents for the subsequent evolutionary process.

Fitness Evaluation. To guide the evolutionary process, EvoSearch evaluates the quality of each parent using an off-the-shelf reward model at each evolution timestep t_i:

$$R(\boldsymbol{x}_{t_i}) = \mathbb{E}_{\boldsymbol{x}_0 \sim p_0(\boldsymbol{x}_0|\boldsymbol{x}_{t_i})}\left[r(\boldsymbol{x}_0)|\boldsymbol{x}_{t_i}\right], \tag{2}$$

where the reward model r can correspond to various objectives, including human preference scores (Xu et al., 2023; Wu et al., 2023b; Hessel et al., 2021) and vision-language models (Liu et al., 2025a; He et al., 2024). Note that previous methods typically rely on either lookahead estimators (Oshima et al., 2025; Li et al., 2025) or Tweedie's formula (Efron, 2011; Kim & Ye, 2021) to predict $\boldsymbol{x}_0$ from noisy data for reward calculation in Eq. equation 2, which can induce significant prediction inaccuracies and approximation errors. In contrast, we evaluate the reward directly on fully denoised $\boldsymbol{x}_0$ (e.g., clean image or video), thereby obtaining high-fidelity reward signals. Considering computing the exact expectation is computationally prohibitive, as it would require generating multiple full trajectories for every candidate at each intermediate denoising step. Therefore, in our implementation, we use a single-sample Monte Carlo approximation, estimating the expectation based on a single sample x_0. Although this is an estimate of the true expectation, the evolutionary population dynamics allow EvoSearch to robustly optimize the objective despite the variance in individual estimates.

Selection. To propagate high-quality candidates across generations while maintaining population diversity, EvoSearch employs tournament selection (Goldberg & Deb, 1991) to sample parents from the population of size k_i through cycles. Specifically, each cycle picks a tournament of $b < k_i$ candidates at random and selects the best candidate in the tournament as a parent.

Mutation. Recent works (Zhou et al., 2024; Ahn et al., 2024) have shown that different initial noises yield varying generation quality. Intuitively, this property extends naturally to intermediate denoising states. While this phenomenon serves as a basis for making best-of-N and particle sampling useful, it raises a more fundamental question: *do these noises and intermediate states possess other exploitable patterns or structural regularities that can be leveraged to enhance inference-time generation quality?*

To investigate this critical question, we visualize the latent states at different denoising steps using t-SNE (Van der Maaten & Hinton, 2008). Our findings, as shown in Fig. 4, reveal that neighboring states in the latent space exhibit similar generation qualities, suggesting that high-quality samples are not solely isolated. Building upon this discovery, we develop a specialized mutation strategy that leverages this exploitable structure in the reward landscape of diffusion and flow models. Specifically, we preserve m elite parents (those with top fitness scores) at each generation to ensure convergence, where $m \ll k_i$. For the remaining $k_i - m$ parents, we mutate them to explore the neighborhoods around selected parents to discover higher-quality samples. This approach avoids premature convergence to a narrow region of the denoising state space, facilitating effective exploration of novel regions while maintaining population diversity. To align with the characteristics of the underlying SDE sampling process, we develop different mutation operations for initial noises and intermediate denoising states.

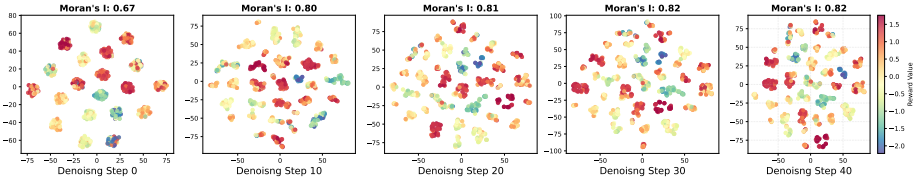


Figure 4: t-SNE Visualization of latent $\boldsymbol{x}_t$ from SD2.1 model at different steps, colored by their corresponding ImageReward scores. At denoising step 0, $\boldsymbol{x}_t$ is Gaussian noise. High-reward states exhibit significant spatial correlations, as measured by Moran's I (Moran, 1950).

• **Initial noise mutation.** For the initial noise $\boldsymbol{x}_T$, which is sampled from a Gaussian distribution, the corresponding mutation operation is designed to preserve the Gaussian nature of the noise based on

$$\boldsymbol{x}_T^{\text{child}} = \sqrt{1-\beta^2}\boldsymbol{x}_T^{\text{parent}} + \beta\varepsilon_T, \quad \varepsilon_T \sim \mathcal{N}(0, I), \tag{3}$$

where β controls the added stochasticity to the parents. The first term ensures that the mutated children preserve the high-reward region density, while the second term encourages exploration.

• **Intermediate denoising state mutation.** For intermediate states $\boldsymbol{x}_t$, the mutation operation defined in Eq. 3 is not applicable since $\boldsymbol{x}_t$ is no longer Gaussian due to the denoising process. To synthesize meaningful variations while preserving the intrinsic structure of the latent state $\boldsymbol{x}_t$, we propose an alternative mutation operator inspired by the reverse-time SDE:

$$\boldsymbol{x}_{t-1}^{\text{child}} = \boldsymbol{x}_{t-1}^{\text{parent}} + \sigma_t\varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, I), \tag{4}$$

where σ_t is the diffusion coefficient defined in reverse-time SDE, controlling the level of injected stochasticity. This mutation operation effectively generates novel $\boldsymbol{x}_{t-1}$, enabling exploration of an expanded state space while preserving the inherent distribution established during the denoising process. The theoretical validation of the proposed mutation strategies is provided in the Appendix A.2. In the next generation of EvoSearch, we sample $\boldsymbol{x}_0 \sim p_0(\boldsymbol{x}_0|\boldsymbol{x}_t^{\text{child}})$ based on the new offspring $\boldsymbol{x}_t^{\text{child}}$, and repeat the above evolutionary search process, including evaluation, selection, and mutation. We highlight that EvoSearch provides a unified framework that encompasses both best-of-N and particle sampling as special cases.

5 Experiments

In this section, we evaluate the efficacy of EvoSearch through extensive experiments on large-scale text-conditioned generation tasks, encompassing both image and video domains.

5.1 Experiment Setup

Image Generation. (i) **Tasks and Metrics.** We adopt DrawBench (Saharia et al., 2022) for evaluation, which consists of 200 prompts spanning 11 different categories. We utilize multiple metrics to evaluate generation quality, including ImageReward (Xu et al., 2023), HPSv2 (Wu et al., 2023b), Aesthetic score (Schuhmann et al., 2022), and ClipScore (Hessel et al., 2021). ImageReward and ClipScore are employed as guidance rewards during search. Please refer to evaluation details in Appendix A.3. (ii) **Models.** We employ two different text-to-image models to evaluate EvoSearch and baselines, which are Stable Diffusion 2.1 (Rombach et al., 2022) and Flux.1-dev (Labs, 2024), respectively. SD2.1 is a diffusion-based text-to-image model with 865M parameters, while Flux-dev is a rectified flow-based model with 12B parameters. For both models, we use 50 denoising steps with a guidance scale of 5.5, with other hyperparameters remaining as the default.

Video Generation. (i) **Tasks and Metrics.** We take the recently released VideoReward (Liu et al., 2025b) as the guidance reward to provide feedback during search. VideoReward, built on Qwen2-VL-2B (Wang et al., 2024), evaluates generated videos on multiple dimensions: visual quality, motion quality, and text alignment. To measure the generalization performance to unseen rewards, we utilize both automatic metrics and human assessment for comprehensive evaluation. For automatic evaluation, we employ multiple metrics from VBench (Huang et al., 2024) and VBench2 (Zheng et al., 2025), which encompass 625 distinct prompts distributed across six fundamental dimensions, including *dynamic*, *semantic*, *human fidelity*, *composition*, *physics*, and *aesthetic*. For human evaluation, we hire annotators to evaluate videos on 200 prompts sampled from VideoGen-Eval (Zeng et al., 2024). Evaluation details are in Appendix A.3. (ii) **Models.** To evaluate the scalability and performance of baselines, we utilize two widely adopted video generative models: HunyuanVideo (Kong et al., 2024) and Wan (Wang et al., 2025). Given the computational intensity of video generation compared to image generation, we specifically use the 1.3B parameter variant of Wan for practical evaluation. Each video comprises 33 frames, with other hyperparameters following default configurations.

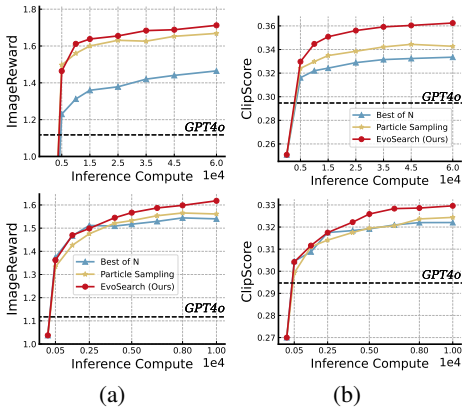


Figure 5: Scaling behavior of EvoSearch and baselines as inference-time computation increases on DrawBench. *Top*: SD2.1. *Bottom*: Flux.1-dev. (a) and (b) use ImageReward and ClipScore as guidance rewards, respectively.

Baselines. As we evaluate the scalability of both diffusion and flow models across image and video generation tasks, we benchmark EvoSearch against two widely-used search methods that are applicable to our experimental settings: (i) **Best of N** samples multiple random noises at beginning, assign reward values to them via denoising and evaluation, and choose the candidate yielding the highest reward. (ii) **Particle Sampling** follows the implementation of FK-Steering (Singhal et al., 2025a), which maintains a set of candidates along the denoising process, called particles, and iteratively propagates high-reward samples while discarding lower-reward ones. Implementation details of EvoSearch and baselines are provided in Appendix A.1. To ensure fair comparison, we employ the same random seeds to generate videos/images for each method.

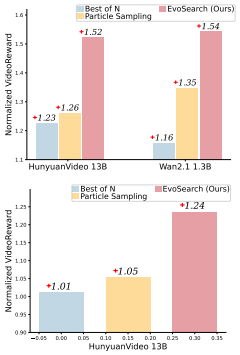


Figure 6: VideoRewards on VBench & VBench2.0 (*top*) and VideoGen-Eval (Zeng et al., 2024) (*bottom*).

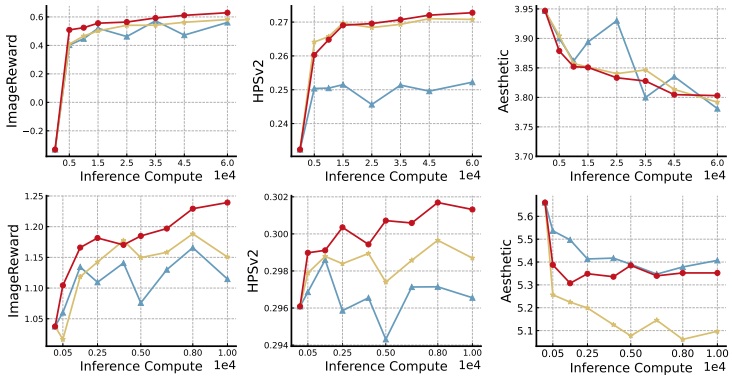


Figure 7: EvoSearch can generalize to unseen metrics. *Top row*: DrawBench results on SD2.1. *Bottom row*: DrawBench results on Flux.1-dev.

5.2 Results Analysis

To evaluate EvoSearch's versatility and practical performance, we include image generation on diffusion model (SD2.1) and flow model (Flux.1-dev), video generation on flow models (HunyuanVideo and Wan) for comprehensive empirical analysis. In addition to comparing the number of function evaluations (NFEs), we provide a wall-clock time comparison in the Appendix B.2 to further demonstrate the computational efficiency of our method, wherein the advantages of EvoSearch become even more evident.

Question 1. *Can EvoSearch consistently yield performance improvement with scaled inference-time computation?*

As shown in Fig. 5, where we evaluate performance using both ImageReward and ClipScore, EvoSearch exhibits monotonic performance improvements with increasing inference-time computation. Notably, for the Flux.1-dev model (12B parameters), EvoSearch continues to demonstrate performance gains as NFEs increase, whereas baseline methods plateau after approximately $1e4$ NFEs. Qualitative results in Fig. 1 show that both SD2.1 and Flux.1-dev generate images with progressively improved prompt alignment as inference computation (i.e., NFEs) increases.

Question 2. *How does EvoSearch compare to baselines for scaling image and video generation at inference time?*

For image generation tasks, as evidenced in Fig. 5 and Fig. 7, EvoSearch demonstrates consistent superior performance over all baseline methods across varying computational budgets, for both diffusion-based SD2.1 and flow-based Flux.1-dev models. The results on other benchmarks like GenEval (Ghosh et al., 2023) and DPGBench (Hu et al., 2024) are provided in Appendix B.3. For video generation tasks where VideoReward serves as the guidance reward, EvoSearch continues to obtain the highest score across different generative models compared to the baselines. Quantitative results in Fig. 6 (top row) show that for the Wan 1.3B model, EvoSearch outperforms best-of-N and particle sampling by 32.8% and 14.1%, respectively. When applied to the larger HunyuanVideo 13B model, EvoSearch demonstrates improvements of 23.6% and 20.6% over best-of-N and particle sampling, respectively. Results on the prompts sample from Videogen-Eval (Zeng et al., 2024), as illustrated in Fig. 6 (bottom row), further corroborate these findings, with EvoSearch showing improvements of 22.8% and 18.1% compared to best-of-N and particle sampling, respectively. Qualitative assessment in Fig. 8 reveals that only EvoSearch successfully generates images with both background consistency and accurate text prompt alignment. In contrast, particle sampling fails to comprehend the complex text prompt, while best-of-N produces results of inferior visual quality. More qualitative results are provided in Appendix B.4. The superior performance of EvoSearch can be attributed to its active exploration and refinement within the denoising state space, whereas best-of-N and particle sampling are limited to a local candidate pool.

Question 3. *How does EvoSearch generalize to unseen reward functions (metrics)?*

As demonstrated in a recent work (Ma et al., 2025), reward hacking (Skalse et al., 2022) can significantly impair test-time scaling performance, where the model exploits flaws or ambiguities in the reward function to obtain high rewards. However, our method, EvoSearch, can mitigate

Table 1: Evaluation results across multiple metrics from both Vbench and VBench2.0.

Methods	Dynamic	Semantic	Human Fidelity	Composition	Physics	Aesthetic	Average
Wan 1.3B	13.18	16.83	82.98	38.08	64.44	64.01	46.59
+Best of N	15.38 ↑ +2.2	13.67 ↓ −3.16	**87.58** ↑ +4.6	44.71 ↑ +6.63	56.10 ↓ −8.34	**64.84** ↑ +0.83	47.04 ↑ +0.45
+Particle Sampling	13.18 ↑ +0.0	12.67 ↓ −4.16	86.13 ↑ +3.15	39.43 ↑ +1.35	56.41 ↓ −8.03	64.54 ↑ +0.53	45.39 ↓ −1.2
+EvoSearch (Ours)	**16.48**↑ +3.3	**15.51** ↓ −1.32	86.84 ↑ +3.86	**51.57** ↑ +13.49	**57.5** ↓ −6.9	64.35 ↑ +0.34	**48.71** ↑ +2.12
HunyuanVideo 13B	8.79	16.11	90.28	47.89	56.10	66.31	47.58
+Best of N	6.59 ↓ −2.2	12.84 ↓ −3.27	91.31 ↑ +1.03	**50.53** ↑ +2.64	47.62 ↓ −8.48	66.28 ↓ −0.03	45.86 ↓ −1.72
+Particle Sampling	6.59 ↓ −2.2	11.00 ↓ −5.11	93.17 ↑ +2.89	36.67 ↓ −11.22	54.29 ↓ −1.81	65.55 ↓ −0.76	44.55 ↓ −3.03
+EvoSearch (Ours)	**7.69** ↓ −1.1	**14.92** ↓ −1.19	**94.63** ↑ +4.35	51.37 ↑ +3.48	**61.54** ↑ +5.44	**66.75** ↑ +0.44	**49.48** ↑ +1.90

Figure 8: A qualitative example showing that EvoSearch generates videos with superior visual quality, enhanced background consistency, and improved semantic alignment with the input text prompts.

the reward hacking problem to some extent since it maintains higher diversity through the search process, effectively capturing multimodal modes from target distributions. We evaluate the generation performance on unseen (out-of-distribution) metrics in Fig. 7, where ClipScore is used as the guidance reward. EvoSearch still showcases superior scalability and performance across different models and metrics. For o.o.d. metric Aesthetic, which is not aligned with ClipScore (as demonstrated in Fig. 8 of (Ma et al., 2025)), EvoSearch shows less performance degradation compared to particle sampling.

For video generation tasks, we include 9 different unseen metrics spanning 6 main categories to evaluate EvoSearch's generalizability to unseen rewards. From the results shown in Table 1, we observe that EvoSearch consistently gains more stable performance improvements compared with baselines. Notably, even for metrics that are not aligned with VideoReward (e.g., Semantic), EvoSearch maintains robust performance with minimal degradation. For the physics metric on HunyuanVideo, EvoSearch even achieves distinctive performance improvements while both best-of-N and particle sampling exhibit significant degradation.

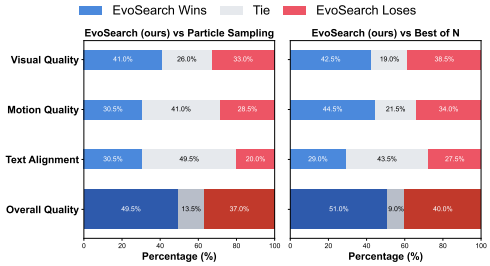


Figure 9: Human evaluation results.

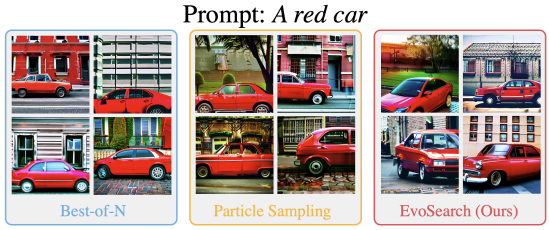


Figure 10: For the same prompt, EvoSearch generates more visually diverse images.

Question 4. *How does EvoSearch perform under human evaluation?*

To validate EvoSearch's alignment with human preferences, we conduct a comprehensive human evaluation study employing professional annotators. The assessment focused on four key dimensions: Visual Quality, Motion Quality, Text Alignment, and Overall Quality. As illustrated in Fig. 9, EvoSearch achieves higher win rates compared to baseline methods across all evaluation dimensions.

Question 5. *Can EvoSearch remains high diversity when maximizing guidance rewards?*

EvoSearch demonstrates superior capability in sampling diverse solutions through its continuous exploration of novel states during the search process. We randomly select 10 prompts from DrawBench, and generate 10 images per prompt using EvoSearch and baselines under 100× scaled inference-time compute. After generation, we evaluate the quality of the generated images by ImageReward, and evaluate the diversity of these images by the L_2 distance between their corresponding hidden features extracted from the CLIP encoder. We observe in Table 2 that EvoSearch obtains the highest reward while achieving the highest diversity. Qualitative results in Fig. 10 further support this finding, revealing that EvoSearch generates text-aligned images with notably greater diversity in backgrounds and poses compared to baseline methods.

Table 2: Results of reward and diversity.

Method	Reward	Diversity
Best of N	0.16	0.62
Particle Sampling	0.13	0.94
EvoSearch (Ours)	**0.18**	**1.34**

Question 6. *Can EvoSearch enable smaller-scale model outperform larger-scale model?*

In image generation tasks, as illustrated in Fig. 5, SD2.1 achieves competitive performance compared to GPT4o with fewer than $5e3$ NFEs (≈ 30 seconds inference time). Qualitative results presented in Fig. 1 further demonstrate how EvoSearch enables smaller models to reach GPT4o's level through strategic inference-time scaling. For video generation tasks, we allocate $5\times$ inference computation to Wan 1.3B, ensuring equivalent inference time with Wan 14B on identical GPUs. Results documented in Table 3 show that the Wan 1.3B model with EvoSearch achieves competitive performance to its $10\times$ larger counterpart, the Wan 14B model. These findings highlight the significant potential of test-time scaling as a complement to traditional training-time scaling laws for visual generative models, opening new avenues for future research.

Table 3: EvoSearch scales Wan 1.3B to have the same inference time as Wan 14B. Results are evaluated on 625 prompts from VBench and VBench2.0.

Methods	VideoReward
Wan 14B	-1.24
Wan 1.3B **+ EvoSearch (ours)**	**-0.15**

6 Discussions

In this work, we propose Evolutionary Search (EvoSearch), a novel, generalist and efficient test-time scaling framework for diffusion and flow models across image and video generation tasks. Through our proposed specialized evolutionary mechanisms, EvoSearch enables the generation of higher-quality samples iteratively by actively exploring new states along the denoising trajectory. Limitations and future work are discussed in Appendix C.

Reproducibility Statement

To ensure reproducibility, we provide a detailed experimental setup and hyperparameters used during training and evaluation in Appendix A.1. Moreover, we provide our codebase at `https://anonymous.4open.science/r/EvoSearch`.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Donghoon Ahn, Jiwon Kang, Sanghyun Lee, Jaewon Min, Minjae Kim, Wooseok Jang, Hyoungwon Cho, Sayak Paul, SeonHwa Kim, Eunju Cha, et al. A noise is worth diffusion guidance. *arXiv preprint arXiv:2412.03895*, 2024.

Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, 2023.

Ping Ao. Laws in darwinian evolutionary theory. *Physics of life Reviews*, 2(2):117–156, 2005.

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 843–852, 2023.

Heli Ben-Hamu, Omri Puny, Itai Gat, Brian Karrer, Uriel Singer, and Yaron Lipman. D-flow: Differentiating through flows for controlled generation. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=SE20BFqj6J`.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. URL `https://openai.com/research/video-generation-models-as-world-simulators`.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024.

Thomas Bäck. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press, 02 1996. ISBN 9780195099713. doi: 10.1093/oso/9780195099713.001.0001. URL https://doi.org/10.1093/oso/9780195099713.001.0001.

Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=OnD9zGAGT0k.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J. Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=1vmSEVL19f.

John D Co-Reyes, Yingjie Miao, Daiyi Peng, Esteban Real, Quoc V Le, Sergey Levine, Honglak Lee, and Aleksandra Faust. Evolving reinforcement learning algorithms. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=0XXpJ4OtjW.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Carles Domingo-Enrich, Michal Drozdzal, Brian Karrer, and Ricky TQ Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. *arXiv preprint arXiv:2409.08861*, 2024.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Luca Eyring, Shyamgopal Karthik, Karsten Roth, Alexey Dosovitskiy, and Zeynep Akata. ReNO: Enhancing one-step text-to-image models through reward-based noise optimization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=MXY0qsGgeO.

Ying Fan and Kangwook Lee. Optimizing ddpm sampling with shortcut fine-tuning. In *International Conference on Machine Learning*, 2023. URL https://api.semanticscholar.org/CorpusID:256415971.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 36:79858–79885, 2023.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36: 52132–52152, 2023.

David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., USA, 1st edition, 1989. ISBN 0201157675.

David E. Goldberg and Kalyanmoy Deb. A comparative analysis of selection schemes used in genetic algorithms. volume 1 of *Foundations of Genetic Algorithms*, pp. 69–93. Elsevier, 1991. doi: https://doi.org/10.1016/B978-0-08-050684-5.50008-2. URL https://www.sciencedirect.com/science/article/pii/B9780080506845500082.

John J Grefenstette. Genetic algorithms and machine learning. In *Proceedings of the sixth annual conference on Computational learning theory*, pp. 3–4, 1993.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Xiefan Guo, Jinlin Liu, Miaomiao Cui, Jiankai Li, Hongyu Yang, and Di Huang. Initno: Boosting text-to-image diffusion models via initial noise optimization. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9380–9389, 2024a. URL https://api.semanticscholar.org/CorpusID:269005550.

Yingqing Guo, Hui Yuan, Yukang Yang, Minshuo Chen, and Mengdi Wang. Gradient guidance for diffusion models: An optimization perspective. *arXiv preprint arXiv:2404.14743*, 2024b.

Xuan He, Dongfu Jiang, Ge Zhang, Max Ku, Achint Soni, Sherman Siu, Haonan Chen, Abhranil Chandra, Ziyan Jiang, Aaran Arulraj, et al. Videoscore: Building automatic metrics to simulate fine-grained human feedback for video generation. *arXiv preprint arXiv:2406.15252*, 2024.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment, 2024.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Jaihoon Kim, Taehoon Yoon, Jisung Hwang, and Minhyuk Sung. Inference-time scaling for flow models via stochastic generation and rollover budget forcing. *arXiv preprint arXiv:2503.19385*, 2025a.

Kwanyoung Kim and Jong Chul Ye. Noise2score: tweedie's approach to self-supervised image denoising without clean images. *Advances in Neural Information Processing Systems*, 34:864–874, 2021.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=vi3DjUhFVm.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

John R. Koza. *Genetic programming: on the programming of computers by means of natural selection*. MIT Press, Cambridge, MA, USA, 1992. ISBN 0262111705.

Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback. *arXiv preprint arXiv:2302.12192*, 2023.

Xiner Li, Yulai Zhao, Chenyu Wang, Gabriele Scalia, Gokcen Eraslan, Surag Nair, Tommaso Biancalani, Shuiwang Ji, Aviv Regev, Sergey Levine, et al. Derivative-free guidance in continuous and discrete diffusion models with soft value-based decoding. *arXiv preprint arXiv:2408.08252*, 2024.

Xiner Li, Masatoshi Uehara, Xingyu Su, Gabriele Scalia, Tommaso Biancalani, Aviv Regev, Sergey Levine, and Shuiwang Ji. Dynamic search for inference-time alignment in diffusion models. *ArXiv*, abs/2503.02039, 2025. URL https://api.semanticscholar.org/CorpusID:276768450.

Zhanhao Liang, Yuhui Yuan, Shuyang Gu, Bohan Chen, Tiankai Hang, Ji Li, and Liang Zheng. Step-aware preference optimization: Aligning preference with denoising performance at each step. *arXiv preprint arXiv:2406.04314*, 2(3), 2024.

Fangfu Liu, Hanyang Wang, Yimo Cai, Kaiyan Zhang, Xiaohang Zhan, and Yueqi Duan. Video-t1: Test-time scaling for video generation. *arXiv preprint arXiv:2503.18942*, 2025a.

Jie Liu, Gongye Liu, Jiajun Liang, Ziyang Yuan, Xiaokun Liu, Mingwu Zheng, Xiele Wu, Qiulin Wang, Wenyu Qin, Menghan Xia, Xintao Wang, Xiaohong Liu, Fei Yang, Pengfei Wan, Di Zhang, Kun Gai, Yujiu Yang, and Wanli Ouyang. Improving video generation with human feedback. *arXiv preprint arXiv:2501.13918*, 2025b.

Runtao Liu, Haoyu Wu, Zheng Ziqiang, Chen Wei, Yingqing He, Renjie Pi, and Qifeng Chen. Videodpo: Omni-preference alignment for video diffusion generation. *arXiv preprint arXiv:2412.14167*, 2024.

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pp. 23–40. Springer, 2024.

Nanye Ma, Shangyuan Tong, Haolin Jia, Hexiang Hu, Yu-Chuan Su, Mingda Zhang, Xuan Yang, Yandong Li, Tommi Jaakkola, Xuhui Jia, et al. Inference-time scaling for diffusion models beyond scaling denoising steps. *arXiv preprint arXiv:2501.09732*, 2025.

Zichen Miao, Jiang Wang, Ze Wang, Zhengyuan Yang, Lijuan Wang, Qiang Qiu, and Zicheng Liu. Training diffusion models towards diverse image generation with reinforcement learning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10844–10853, 2024. doi: 10.1109/CVPR52733.2024.01031.

Patrick AP Moran. Notes on continuous stochastic phenomena. *Biometrika*, 37(1/2):17–23, 1950.

Yuta Oshima, Masahiro Suzuki, Yutaka Matsuo, and Hiroki Furuta. Inference-time text-to-video alignment with diffusion latent beam search. *arXiv preprint arXiv:2501.19252*, 2025.

Art Owen and Yi Zhou Associate and. Safe and effective importance sampling. *Journal of the American Statistical Association*, 95(449):135–143, 2000. doi: 10.1080/01621459.2000.10473909. URL https://www.tandfonline.com/doi/abs/10.1080/01621459.2000.10473909.

Zeeshan Patel, James DeLoye, and Lance Mathias. Exploring diffusion and flow matching under generator matching. *arXiv preprint arXiv:2412.11024*, 2024.

Mihir Prabhudesai, Russell Mendonca, Zheyang Qin, Katerina Fragkiadaki, and Deepak Pathak. Video diffusion alignment via reward gradients, 2024. URL https://arxiv.org/abs/2407.08737.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V Le. Regularized evolution for image classifier architecture search. In *Proceedings of the aaai conference on artificial intelligence*, volume 33, pp. 4780–4789, 2019.

Esteban Real, Chen Liang, David So, and Quoc Le. Automl-zero: Evolving machine learning algorithms from scratch. In *International conference on machine learning*, pp. 8007–8019. PMLR, 2020.

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

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022.

Anuj Singh, Sayak Mukherjee, Ahmad Beirami, and Hadi Jamali Rad. Code: Blockwise control for denoising diffusion models. *ArXiv*, abs/2502.00968, 2025. URL https://api.semanticscholar.org/CorpusID:276094284.

Saurabh Singh and Ian Fischer. Stochastic sampling from deterministic flow models. *arXiv preprint arXiv:2410.02217*, 2024.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models, 2025a. URL https://arxiv.org/abs/2501.06848.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models. *arXiv preprint arXiv:2501.06848*, 2025b.

Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward gaming. *Advances in Neural Information Processing Systems*, 35:9460–9471, 2022.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

David So, Quoc Le, and Chen Liang. The evolved transformer. In *International conference on machine learning*, pp. 5877–5886. PMLR, 2019.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023a.

Jiaming Song, Qinsheng Zhang, Hongxu Yin, Morteza Mardani, Ming-Yu Liu, Jan Kautz, Yongxin Chen, and Arash Vahdat. Loss-guided diffusion models for plug-and-play controllable generation. In *International Conference on Machine Learning*, pp. 32483–32498. PMLR, 2023b.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Zhiwei Tang, Jiangweizhi Peng, Jiasheng Tang, Mingyi Hong, Fan Wang, and Tsung-Hui Chang. Inference-time alignment of diffusion models with direct noise optimization. *arXiv preprint arXiv:2405.18881*, 2024.

Guiyao Tie, Zeli Zhao, Dingjie Song, Fuyang Wei, Rong Zhou, Yurou Dai, Wen Yin, Zhejian Yang, Jiangyue Yan, Yao Su, et al. A survey on post-training of large language models. *arXiv preprint arXiv:2503.06072*, 2025.

Masatoshi Uehara, Yulai Zhao, Tommaso Biancalani, and Sergey Levine. Understanding reinforcement learning-based fine-tuning of diffusion models: A tutorial and review. *arXiv preprint arXiv:2407.13734*, 2024a.

Masatoshi Uehara, Yulai Zhao, Kevin Black, Ehsan Hajiramezanali, Gabriele Scalia, Nathaniel Lee Diamant, Alex M Tseng, Tommaso Biancalani, and Sergey Levine. Fine-tuning of continuous-time diffusion models as entropy-regularized control. *arXiv preprint arXiv:2402.15194*, 2024b.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Pradnya A. Vikhar. Evolutionary algorithms: A critical review and its future prospects. In *2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC)*, pp. 261–265, 2016. doi: 10.1109/ICGTSPICC.2016.7955308.

Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8228–8238, 2024.

Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Luhuan Wu, Brian Trippe, Christian Naesseth, David Blei, and John P Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. *Advances in Neural Information Processing Systems*, 36:31372–31403, 2023a.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023b.

Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score: Better aligning text-to-image models with human preference. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2096–2105, 2023c.

Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. Inference scaling laws: An empirical analysis of compute-optimal inference for LLM problem-solving. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=VNckp7JEHn`.

Enze Xie, Junsong Chen, Yuyang Zhao, Jincheng Yu, Ligeng Zhu, Chengyue Wu, Yujun Lin, Zhekai Zhang, Muyang Li, Junyu Chen, et al. Sana 1.5: Efficient scaling of training-time and inference-time compute in linear diffusion transformer. *arXiv preprint arXiv:2501.18427*, 2025.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Haolin Yang, Feilong Tang, Ming Hu, Yulong Li, Yexin Liu, Zelin Peng, Junjun He, Zongyuan Ge, and Imran Razzak. Scalingnoise: Scaling inference-time search for generating infinite videos. *arXiv preprint arXiv:2503.16400*, 2025.

Kai Yang, Jian Tao, Jiafei Lyu, Chunjiang Ge, Jiaxin Chen, Weihan Shen, Xiaolong Zhu, and Xiu Li. Using human feedback to fine-tune diffusion models without any reward model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8941–8951, 2024.

Zhaohui Yang, Yunhe Wang, Xinghao Chen, Boxin Shi, Chao Xu, Chunjing Xu, Qi Tian, and Chang Xu. Cars: Continuous evolution for efficient neural architecture search. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1829–1838, 2020.

Po-Hung Yeh, Kuang-Huei Lee, and Jun-Cheng Chen. Training-free diffusion model alignment with sampling demons. *arXiv preprint arXiv:2410.05760*, 2024.

Ailing Zeng, Yuhang Yang, Weidong Chen, and Wei Liu. The dawn of video generation: Preliminary explorations with sora-like models. *arXiv preprint arXiv:2410.05227*, 2024.

Jiacheng Zhang, Jie Wu, Weifeng Chen, Yatai Ji, Xuefeng Xiao, Weilin Huang, and Kai Han. Onlinevpo: Align video diffusion model with online video-centric preference optimization. *arXiv preprint arXiv:2412.15159*, 2024.

Qiyuan Zhang, Fuyuan Lyu, Zexu Sun, Lei Wang, Weixu Zhang, Zhihan Guo, Yufei Wang, Irwin King, Xue Liu, and Chen Ma. What, how, where, and how well? a survey on test-time scaling in large language models. *arXiv preprint arXiv:2503.24235*, 2025.

Dian Zheng, Ziqi Huang, Hongbo Liu, Kai Zou, Yinan He, Fan Zhang, Yuanhan Zhang, Jingwen He, Wei-Shi Zheng, Yu Qiao, and Ziwei Liu. VBench-2.0: Advancing video generation benchmark suite for intrinsic faithfulness. *arXiv preprint arXiv:2503.21755*, 2025.

Zikai Zhou, Shitong Shao, Lichen Bai, Zhiqiang Xu, Bo Han, and Zeke Xie. Golden noise for diffusion models: A learning framework. *arXiv preprint arXiv:2411.09502*, 2024.

A Experimental Details

A.1 Implementation Details

A.1.1 Implementation Details of EvoSearch

Evolution schedule $\mathcal{T}$. Evolution schedule $\mathcal{T}$ can be flexibly defined based on the available amount of inference-time compute. If the inference-time computation budget is sufficient, we can perform EvoSearch at more timesteps; otherwise, we can deploy EvoSearch at several timesteps. In our implementation, we set $\mathcal{T}$ to have uniform intervals.

Population size schedule $\mathcal{K}$. Population size schedule is defined as $\mathcal{K} = \{k_{\text{start}}, k_T, \cdots, k_j, \cdots, k_n\}$. $\mathcal{K}$ can be flexibly defined based on the available amount of inference-time compute. We can increase population size as inference-time computation increases. In our implementation, we assign $2\times$ larger population size at the first generation of EvoSearch, while keeping the population size at the remaining generations the same. This means that k_{start} is twice as large as the other population sizes.

Stable Diffusion 2.1. We set the guidance scale as 5.5, and set the resolution size as 512×512. We employ the DDIM scheduler from the diffusers library (von Platen et al., 2022) for inference. We set the mutation rate $\beta = 0.3$, with σ_t following the default DDIM configurations.

Flux.1-dev. We set the guidance scale as 5.5, and set the resolution size as 512×512. We employ the *sde-dpmsolver++* sampler in *FlowDPMSolverMultistepScheduler* (von Platen et al., 2022) for inference in SDE process. We set the mutation rate $\beta = 0.3$, with σ_t following the default sde-dpmsolver configurations.

Wan. Following the official codes (Wang et al., 2025), we set the resolution size as 832×480, with a video consists of 33 frames. We set the guidance scale as 5.0. For transforming the ODE denoising process in Wan to SDE process, we leverage the *sde-dpmsolver++* sampler in *FlowDPMSolverMultistepScheduler* (von Platen et al., 2022) for inference.

Hunyuan. Following the official implementation (Kong et al., 2024), we set the resolution size as 544×960 to ensure the generation quality, with a video consisting of 33 frames. The guidance scale is set at 1.0 as suggested, and the embedded guidance scale is 6.0. For transforming the ODE denoising process in Wan to SDE process, we leverage the *sde-dpmsolver++* sampler in *FlowDPMSolverMultistepScheduler* (von Platen et al., 2022) for inference. To save computation for a large number of experiments conducted in this paper, we set the inference steps to 30.

We refer to the pseudocodes of EvoSearch in Alg. 1 and Alg. 2. At the beginning of EvoSearch, we denote the size of randomly sampled Gaussian noises as k_{start}. The implementation of EvoSearch is provided in the supplementary material, ensuring reproducibility.

Algorithm 1 Overview of EvoSearch

Input: Pre-trained model p_θ, population size schedule $\mathcal{K} = \{k_{\text{start}}, k_T, \cdots, k_j, \cdots, k_n\}$, evolution schedule $\mathcal{T} = \{T, \cdots, t_j, \cdots, t_n\}$
Initialize population list $\mathcal{P} = [\phi \text{ for } \_ \text{ in } \mathcal{T}]$.
Initialize reward list $\mathcal{R} = [\phi \text{ for } \_ \text{ in } \mathcal{T}]$
Sample initial Gaussian noises $\boldsymbol{x}_T$ with population size k_{start}
Initialize generation $g = 0$
for $t = T, T-1, \cdots, 1$ **do**
  if t in $\mathcal{T}$ **then**
    $\boldsymbol{x}_t, \mathcal{P}, \mathcal{R} = \textit{evosearch\_at\_denoising\_states}(p_\theta, \boldsymbol{x}_t, \mathcal{P}, \mathcal{R}, \mathcal{T}, \mathcal{K}, g)$ // Alg 2
    $g \leftarrow g + 1$
    Evolutionary generation process
  end if
  $\boldsymbol{x}_{t-1} = \text{denoise}(p_\theta, \boldsymbol{x}_t, t)$
  Standard denoising process
end for

Algorithm 2 EvoSearch at Denoising States

Input: Pre-trained model p_θ, starting states $\boldsymbol{x}_{t'}$, population list $\mathcal{P}$, reward list $\mathcal{R}$, evolution schedule $\mathcal{T} = \{T, \cdots, t_j, \cdots, t_n\}$, population size schedule $\mathcal{K} = \{k_T, \cdots, k_j, \cdots, k_n\}$, generation g, elites size m.
Set $\text{idx} = \text{g}$
Set population size $k = \mathcal{K}[g+1]$
for $t = t', t'-1, \cdots, 1$ **do**
  if t in $\mathcal{T}$ **then**
    $\mathcal{P}[\text{idx}] = \text{cat}(\mathcal{P}[\text{idx}], \boldsymbol{x}_t)$
    $\text{idx} \leftarrow \text{idx} + 1$
  end if
  $\boldsymbol{x}_{t-1} = \text{denoise}(\boldsymbol{x}_t, t)$
end for
Calculate rewards r via fully denoised $\boldsymbol{x}_0$ in Eq. equation 2
for $i = g, \cdots, \text{len}(\mathcal{R}) - 1$ **do**
  $\mathcal{R}[i] = \text{cat}(\mathcal{R}[i], r)$ // Compute a single reward per $\boldsymbol{x}_0$
end for
Select elites $e = \mathcal{P}[g]\,[\text{topk}(\mathcal{R}[g], m)]$
Select $k - m$ parents p from $\mathcal{P}[g]$ via tournament selection (Goldberg & Deb, 1991)
if g=0 **then**
  Mutate parents $p = \sqrt{1-\beta^2} \times p + \varepsilon \times \beta, \;\; \varepsilon \sim \mathcal{N}(0, I)$
else
  Mutate parents $p = p + \sigma_t \times \varepsilon, \;\; \varepsilon \sim \mathcal{N}(0, I)$
  // σ_t is the diffusion coefficient in the SDE denoising process
end if
Get children $c \leftarrow \text{cat}(e, p)$
Output: Children c, $\mathcal{P}$, $\mathcal{R}$

A.1.2 Implementation Details of Baselines

Best of N. Best of N generates a batch of N candidate samples (images or videos), from which the highest-quality sample is selected according to a predefined guidance reward function. In practice, we use the same guidance reward for EvoSearch and all baselines to ensure fair comparison.

Particle Sampling. Particle-based sampling methods have demonstrated significant effectiveness in enhancing the generative performance of diffusion models during inference. For our implementation, we leverage the generalist particle-based sampling framework proposed by (Singhal et al., 2025a), utilizing their publicly available codebase. Their approach introduces a flexible methodology that accommodates diverse potential functions, sampling algorithms, and reward models, leading to improved performance across a broad spectrum of text-to-image generation tasks. We adopt the *Max* potential schedule for resampling at intermediate states, which empirically demonstrated superior performance in the original study. Other hyperparameters, such as the *resampling interval*, are carefully tuned to establish a robust baseline performance.

A.2 Theoretical Analysis of Intermediate State Mutation

Definition 1 (SDE Denoising Process)**.** *Let $\{x_t\}_{t=0}^T$ denote the state sequence in a stochastic differential equation (SDE) denoising process. The reverse-time transition from x_t to x_{t-1} follows:*

$$x_{t-1}^{parent} = x_t - f_t(x_t) + \sigma_t \varepsilon_1, \quad \varepsilon_1 \sim \mathcal{N}(0, \mathbf{I}) \tag{5}$$

where $f_t : \mathbb{R}^d \to \mathbb{R}^d$ is a drift function, $\sigma_t > 0$ is the diffusion coefficient at timestep t, and ε_1 is standard Gaussian noise.

Theorem 1 (Validity of Mutation Scheme)**.** *The proposed mutation operator $\mathcal{M} : \mathbb{R}^d \to \mathbb{R}^d$ defined as*

$$\mathcal{M}(x_{t-1}^{parent}) = x_{t-1}^{parent} + \sigma_t \varepsilon_2, \quad \varepsilon_2 \sim \mathcal{N}(0, \mathbf{I}) \tag{6}$$

satisfies the following properties:

1. *Well-definedness: $\mathcal{M}$ generates valid state transitions.*

2. SDE consistency: Mutated states adhere to the reverse-time SDE dynamics.

Proof. Let $x_{t-1}^{\text{child}} = \mathcal{M}(x_{t-1}^{\text{parent}})$. Substituting Definition 1 into the mutation operator:

$$\begin{aligned} x_{t-1}^{\text{child}} &= [x_t - f_t(x_t) + \sigma_t \varepsilon_1] + \sigma_t \varepsilon_2 \\ &= x_t - f_t(x_t) + \sigma_t(\varepsilon_1 + \varepsilon_2). \end{aligned} \tag{7}$$

Since $\varepsilon_1 \sim \mathcal{N}(0, \mathbf{I})$ and $\varepsilon_2 \sim \mathcal{N}(0, \mathbf{I})$ are independent, their sum follows:

$$\varepsilon \triangleq \varepsilon_1 + \varepsilon_2 \sim \mathcal{N}(0, 2\mathbf{I}). \tag{8}$$

By substitution, we have:

$$x_{t-1}^{\text{child}} = x_t - f_t(x_t) + \sigma_t \varepsilon. \tag{9}$$

Therefore, the marginal distribution $p_t(x_{t-1}^{\text{child}})$ after mutation remains Gaussian:

$$p_t(x_{t-1}^{\text{child}}) = \mathbb{E}_{x_t}\left[\mathcal{N}\left(x_{t-1}\,;\, x_t - f_t(x_t),\, 2\sigma_t^2 \mathbf{I}\right)\right]. \tag{10}$$

This matches the SDE transition form with a modified diffusion coefficient $\sqrt{2}\sigma_t$, which expands the exploration space without hindering the denoising process, as the diffusion coefficient can be chosen freely within the stochastic interpolant framework (Ma et al., 2024; Albergo et al., 2023). □

A.3 Evaluation Metrics

Image Evaluation Metrics. (i) **ImageReward** is a text-to-image human preference reward model (Xu et al., 2023), which takes an image and its corresponding prompt as inputs and outputs a preference score. (ii) **CLIPScore** is a reference-free evaluation metric derived from the CLIP model (Hessel et al., 2021), which aligns visual and textual embeddings in a shared latent space. By computing the cosine similarity between an image embedding and its associated text prompt embedding, CLIPScore quantifies semantic coherence without requiring ground-truth images. (iii) **HPSv2** is a preference prediction model that reflects human perceptual preferences for text-to-image generation (Wu et al., 2023b). (iv) **Aesthetic** quantifies the visual appeal of images, often independent of text prompts (Schuhmann et al., 2022).

Video Evaluation Metrics. (i) **Dynamic** evaluates a model's ability to follow complex prompts and simulate dynamic changes (i.e., color, size, lightness, and material). This evaluation metric includes prompts of *Dynamic Attribute* form VBench2.0. Scores are calculated following the original codes (Zheng et al., 2025). (ii) **Semantic** evaluates the model's ability to follow long prompts, which involve at least 150 words. This evaluation metric includes the prompts of *Complex Plot* and *Complex Landscape* from VBench2.0. (iii) **Human Fidelity** evaluates both the structural correctness and temporal consistency of human figures in generated videos. This evaluation metric includes the prompts of *Human Anatomy*, *Human Clothes*, and *Human Identities* from VBench2.0. (iv) **Composition** evaluates the model's ability to generate complex, impossible compositions beyond real-world constraints. This evaluation metric includes the prompts of *Composition* from VBench 2.0. (v) **Physics** evaluates whether models follow basic real-world physical principles (e.g., gravity). This evaluation metric includes the prompts of *Mechanics* from VBench2.0. (vi) **Aesthetic** evaluates the aesthetic values perceived by humans towards each video frame using the LAION aesthetic predictor (Schuhmann et al., 2022). This evaluation metric includes the prompts of *Aesthetic Quality* from VBench.

B Additional Experimental Results

B.1 Ablation on Population Size Schedule

To ablate the effect of population size schedules under the same inference-time computation budget, we set different population size schedules for the Stable Diffusion 2.1 model with approximately 140×50 inference-time NFEs. Here, 50 is the length of the denoising steps for each generation. We report the DrawBench results in Fig. 11. We observe that different population size schedules perform similarly with little reward difference. The most significant factor is the value of k_{start}, which represents the population size of the initial Gaussian noises. A larger value of k_{start} benefits a strong initialization for the subsequent search process, while a small value of k_{start} would affect the performance a lot.

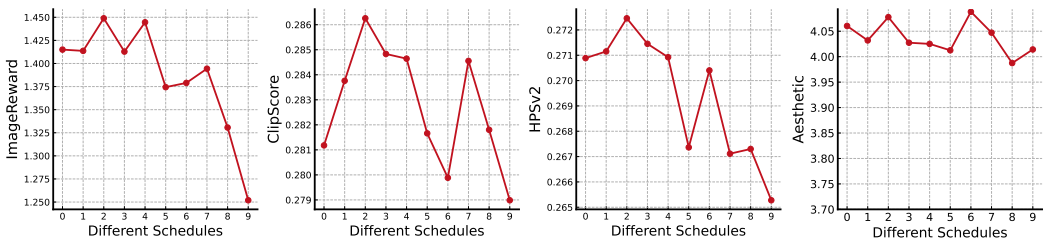


Figure 11: Ablation study on the population size schedule $\mathcal{K}$. We denote the population size schedule $\mathcal{K} = \{k_{\text{start}}, k_T, \cdots, k_j, \cdots, k_n\}$, where k_{start} is the size of the initial sampled Gaussian noises. We use Stable Diffusion 2.1 to conduct EvoSearch on DrawBench, employing ImageReward as the guidance reward function during search, and the denoising step is 50. From left to right of the x-axis, the population size schedule $\mathcal{K}$ is configured as: 0) $\{60, 40, 50\}$; 1) $\{70, 30, 50\}$; 2) $\{80, 20, 50\}$; 3) $\{62, 62, 20\}$; 4)$\{58, 58, 30\}$; 5) $\{54, 54, 40\}$; 6) $\{46, 46, 60\}$;7) $\{40, 60, 50\}$; 8) $\{30, 70, 50\}$; 9) $\{20, 80, 50\}$, where we maintain the evolution schedule as $\{50, 40\}$.

B.1.1 Ablation on Evolution Schedule

We further ablate the effect of the evolution schedule. From the results shown in Fig. 12, we find that the evolution schedule $\mathcal{T}$ exhibits less significant influence compared to the population size schedule $\mathcal{K}$. Our analysis demonstrates that an evolution schedule with uniform intervals yields superior performance. Additionally, larger initial population sizes k_{start} help increase the performance.

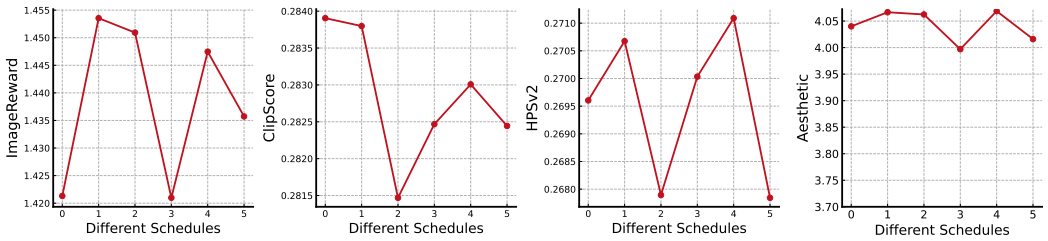


Figure 12: Ablation study on the evolution schedule $\mathcal{T}$. We use Stable Diffusion 2.1 to conduct EvoSearch on the DrawBench, employing ImageReward as the guidance reward function during search. We denote the evolution schedule $\mathcal{T} = \{T, \cdots, t_m, \cdots, t_n\}$. From left to right of the x-axis, the evolution schedule is 0) $\{50, 30\}$; 1) $\{50, 20\}$; 2) $\{50, 10\}$; 3) $\{50, 30\}$; 4) $\{50, 20\}$; 5) $\{50, 10\}$. To keep the same test-time scaling computation budget across different evolution schedules, each population size schedule is adjusted as 0) $\{60, 50, 50\}$; 1) $\{70, 50, 50\}$; 2) $\{80, 50, 50\}$; 3) $\{55, 55, 50\}$; 4) $\{60, 60, 50\}$; 5) $\{75, 75, 50\}$.

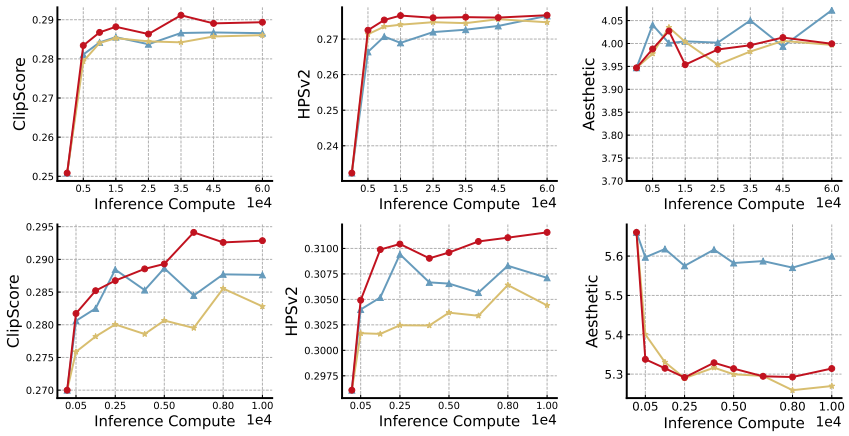
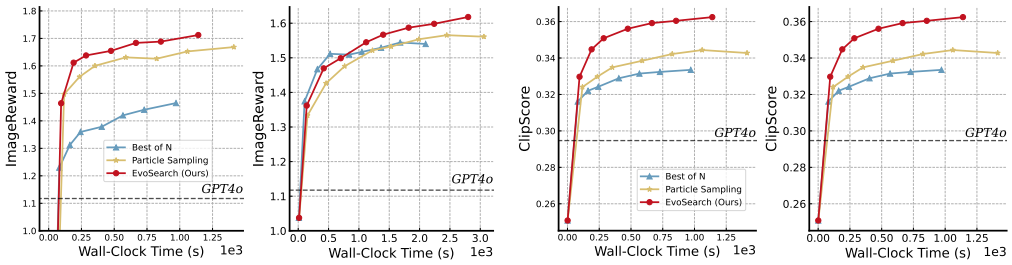


Figure 13: EvoSearch can generalize to unseen metrics, where ImageReward is set as the guidance reward function during search. *Top row*: DrawBench results on SD2.1. *Bottom row*: DrawBench results on Flux.1-dev.

B.2 Wall-Clock Time Analysis

We show the wall-clock time required for different methods in Fig. 14. We observe that EvoSearch achieves superior performance compared with baselines given the same wall-clock time, demonstrating the efficiency and effectiveness of our proposed method. Both particle sampling and Best-of-N can rapidly fall into a plateau, while EvoSearch continues to improve the base models' performance with increased computation. The efficiency of EvoSearch lies in its progressive evolution framework: (1) EvoSearch only needs a single reward evaluation at the end of each evolution generation, while particle sampling requires multiple reward computations at intermediate steps per particle. (2) EvoSearch uniquely caches all intermediate samples at evolution timesteps $t \in \mathcal{T}$, creating a rich pool of parent candidates for subsequent evolution generations. This mechanism avoids repeatedly denoising from x_T across each evolution branch and eliminates redundant denoising computations.



(a) ImageReward as target guidance. *Left:* SD2.1; *Right:* Flux.1-dev

(b) ClipScore as target guidance. *Left:* SD2.1; *Right:* Flux.1-dev

Figure 14: We compare EvoSearch with baselines based on wall-clock time per prompt. We record the time for different methods on the same hardware and GPU card to ensure fairness.

B.3 Results on GenEval and DPGBench

To further showcase the effects of EvoSearch generalizing to different evaluation metrics and benchmarks, we compare EvoSearch with baselines on GenEval (Ghosh et al., 2023) and DPGBench (Hu et al., 2024), which include fine-grained assessment across multiple dimensions (e.g., color, count) and carefully designed prompts. As shown in Table 4, EvoSearch improves SD2.1's score on GenEval by 83.6%, finally surpassing GPT4o. The results provided in Table 5 demonstrate that EvoSearch continues to outperform all the baselines on DPGBench, and we find that Flux.1-dev with EvoSearch can also surpass GPT4o.

Table 4: Following the official evaluation pipeline of GenEval (Ghosh et al., 2023), we compare EvoSearch to baselines with 200× NFEs available at test-time. SD2.1 is used as the base model. We employ the scores defined in GenEval as the guidance rewards during search.

Methods	Geneval Score
SD2.1	0.50
GPT4o	0.84
EvoSearch w/ SD2.1	**0.92**
Particle Sampling w/ SD2.1	0.86
Best of N w/ SD2.1	0.83

Table 5: We evaluate EvoSearch and the baselines with $10\times$ NFEs allocated at test time on the 1065 prompts provided by DPGBench (Hu et al., 2024), leveraging the pre-defined DPG score as the guidance reward during search. Flux.1-dev is used as the base model, with 50 denoising steps per generation.

Methods	DPG Score
Flux.1-dev	83.84
GPT4o	85.15
EvoSearch w/ Flux.1-dev	**93.51**
Particle Sampling w/ Flux.1-dev	89.32
Best of N w/ Flux.1-dev	90.06

B.4 Qualitative Results

We present extensive qualitative results for both image and video generation as follows. The images of GPT4o are generated by the OpenAI API following the default configuration. To ensure fair comparison, the prompts given to GPT4o remain the same as those of other models.

B.4.1 Results for Image Generation

Please refer to Fig. 15, Fig. 16, and Fig. 17 for comparison between EvoSearch and baselines. These examples clearly demonstrate that EvoSearch significantly enhances image generation performance while requiring lower computational resources.

B.4.2 Results for Video Generation

Please refer to Fig. 18, Fig. 19, Fig. 20, and Fig. 21 for comparison between EvoSearch and baselines in the context of video generation. We find that EvoSearch outperforms all the baselines with higher efficacy and efficiency. Please refer to Fig. 22, Fig. 23, Fig. 24, Fig. 25, Fig. 26, Fig. 27, and Fig. 28 for comparison between Wan14B and Wan1.3B enhanced with EvoSearch. For more details, please visit the anonymous website `evosearch.github.io`. The results demonstrate that by increasing the test-time computation budget of Wan1.3B to match the inference latency of Wan14B, the smaller model outperforms its $10\times$ larger counterpart across a diverse range of input prompts.

C Discussions

Limitations and Future Work. EvoSearch has demonstrated significant effectiveness in exploring high-reward regions of novel states, which opens promising directions for future research. The exploration ability of EvoSearch relies on the strength of the mutation rate β and σ_t. A higher mutation rate will effectively expand the search space to find high-quality candidates, while a low mutation rate can restrict the exploration space, which represents a trade-off. In addition, we rely on Gaussian noise to mutate the selected parents. While this approach provides robust exploration across diverse image and video generation tasks, developing more informative mutation strategies with prior knowledge can further improve the search efficiency. The inherent complexity of interpreting denoising states makes it an interesting open research question. Our findings also suggest promising future directions in understanding the shared structure between ”golden” noise

and "golden" intermediate denoising states, which may provide valuable insights for future test-time scaling research.

Broader Impacts. This work proposes a novel test-time scaling method, called EvoSearch, for image and video generation tasks across both diffusion-based and flow-based models. EvoSearch draws inspiration from biological evolution (Ao, 2005), which significantly improves both the quality and diversity of generated samples through enhanced exploration during the search process. Our proposed method is promising to provide insights for test-time scaling in other areas, like large language models (LLMs). Therefore, our proposed method can further enhance the research of test-time scaling and inference-time alignment in the general area of machine learning. No significant negative broader impacts were identified that warrant specific emphasis in this paper.

D Comparison and Discussion with Gradient-based Methods

Compared with gradient-based methods like ReNO (Eyring et al., 2024) and D-Flow (Ben-Hamu et al., 2024), our proposed method, EvoSearch, has the following advantages: (1) **Universality**: Not all reward functions are differentiable (e.g., rewards from proprietary APIs, discrete metrics, or human feedback). A gradient-free approach makes EvoSearch universally applicable. (2) **Memory & Efficiency**: Backpropagating gradients through the diffusion process is memory-intensive and computationally expensive. Training-free methods like test-time scaling are significantly cheaper, allowing us to explore a wider search space within the same wall-clock time. (3) **Avoiding Local Optima**: Gradient ascent on noise latent space is prone to getting stuck in local optima or generating "adversarial" examples (e.g., high reward score but poor visual quality), which is a well-known problem in the literature. Our proposed EvoSearch, which is based on a gradient-free evolutionary algorithm, is better suited for non-convex landscapes as it maintains population diversity explicitly.

We compare EvoSearch against ReNO (Eyring et al., 2024), a representative gradient-based method. From the results shown in Table 6, we observe that EvoSearch significantly outperforms ReNO given the same wall-clock time. ReNO suffers from diminishing returns, improving only slightly (0.65 $\rightarrow$ 0.68) even when computation is increased by $10\times$. In contrast, EvoSearch effectively converts increased compute into quality, reaching a 0.77 score.

Methods	Computation / Time	GenEval Score
SDXL-Turbo (Base)	-	0.54
+ReNO	50 iters / 33s	0.65
+EVOSearch	190 NFEs / 33s	**0.71**
+ReNO	500 iters / 330s	0.68
+EVOSearch	1000 NFEs / 172s	**0.77**

Table 6: Both ReNO and EvoSearch employ the combination of the reward models (HPS, ImageReward, CLIP, PickScore) as the guidance, with SDXL-Turbo as the base model. We compare based on Wall-Clock Time to ensure a fair assessment of practical utility.

E Comparison with Rollover Budget Forcing (RBF)

To further demonstrate the advantages of EvoSearch, we have added Rollover Budget Forcing (RBF) (Kim et al., 2025a) for comparison. Following RBF's official implementation, we employ Flux as the base model with ImageReward as the guidance. We fix the total number of function evaluations (NFEs) to 500 and set the number of denoising steps to 10. We evaluate both EvoSearch and RBF on DrawBench. We provide the results in Table 7. While the numerical margins may appear modest, we emphasize that on the DrawBench scale, a consistent improvement across 200 prompts on three different metrics (including both target and unseen metrics) confirms that EvoSearch generates strictly superior samples.

Methods	ImageReward (target)	ClipScore (o.o.d.)	HPSv2(o.o.d.)
RBF	1.38	0.281	0.305
EvoSearch	**1.41**	**0.284**	**0.310**

Table 7: EvoSearch outperforms RBF given the same test-time NFEs, including both target reward and unseen reward.

F THE USE OF LARGE LANGUAGE MODELS (LLMS)

In compliance with ICLR 2026 policies on large language model usage, we disclose that LLMs are mainly used for writing polish in this work. We utilized LLMs to polish the paper's writing at the syntactic and grammatical levels. All LLM-generated content has undergone thorough human review and verification to ensure accuracy, appropriateness, and compliance with academic standards.

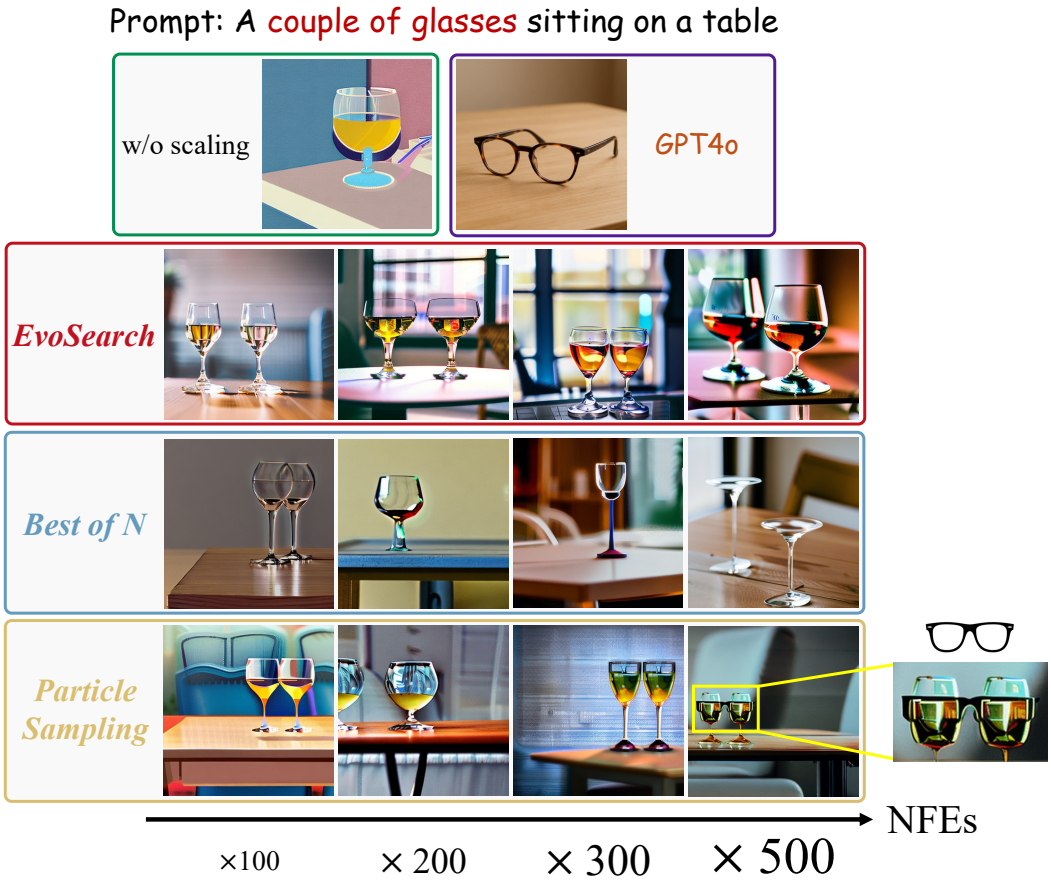


Figure 15: Comparative analysis of test-time scaling methods for Stable Diffusion 2.1. **EvoSearch** demonstrates consistent improvements in image quality and text-prompt alignment as NFEs increase, achieving accurate interpretations of the challenging prompt with high computational efficiency. In contrast, **Best-of-N** fails to produce semantically correct results even with increased NFEs, while **Particle Sampling** introduces semantic ambiguity at higher NFEs (e.g., confusing wine glasses and eyeglasses). Notably, **EvoSearch** further enables SD2.1 to outperform **GPT4o**.

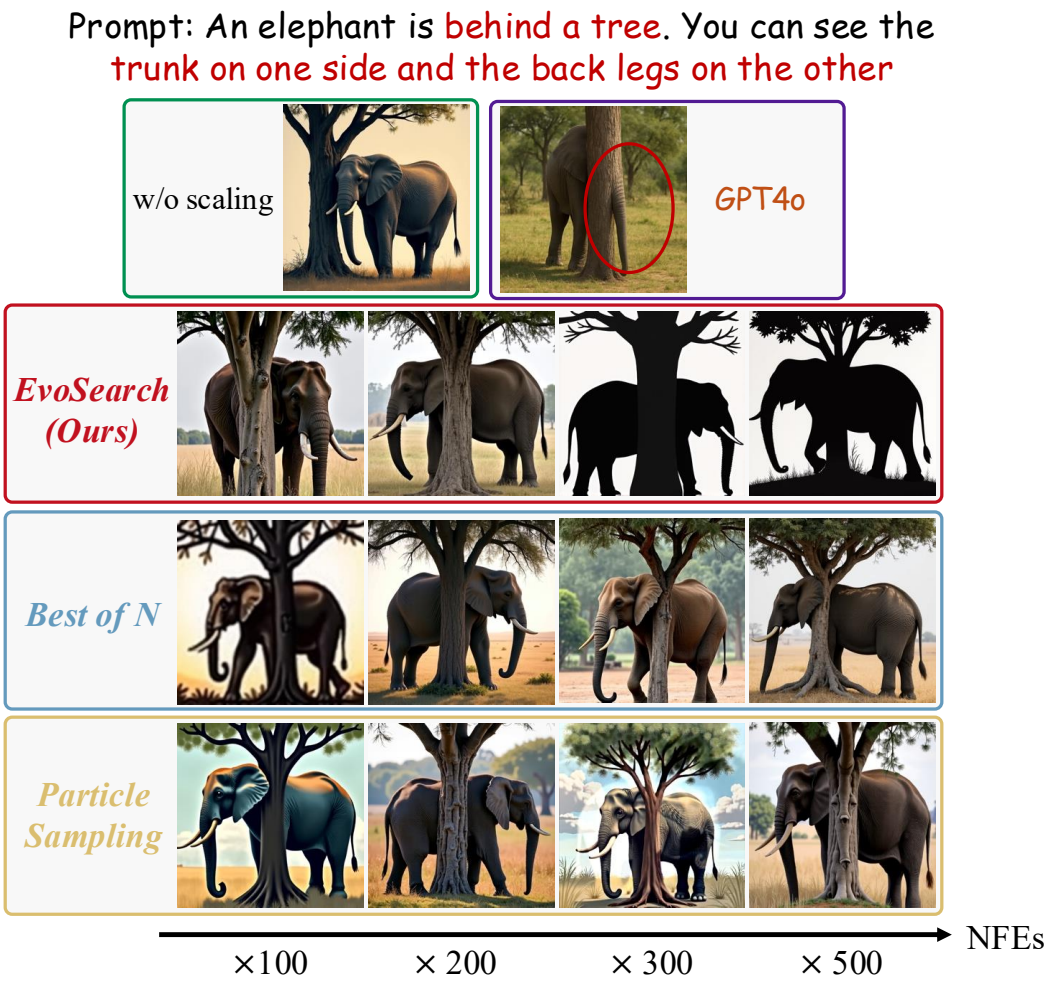


Figure 16: Results of test-time scaling for Flux.1-dev. **EvoSearch** demonstrates significant exploration ability, enabling the generation of images with diverse styles, while both **Best-of-N** and **Particle Sampling** generate images with reduced diversity.

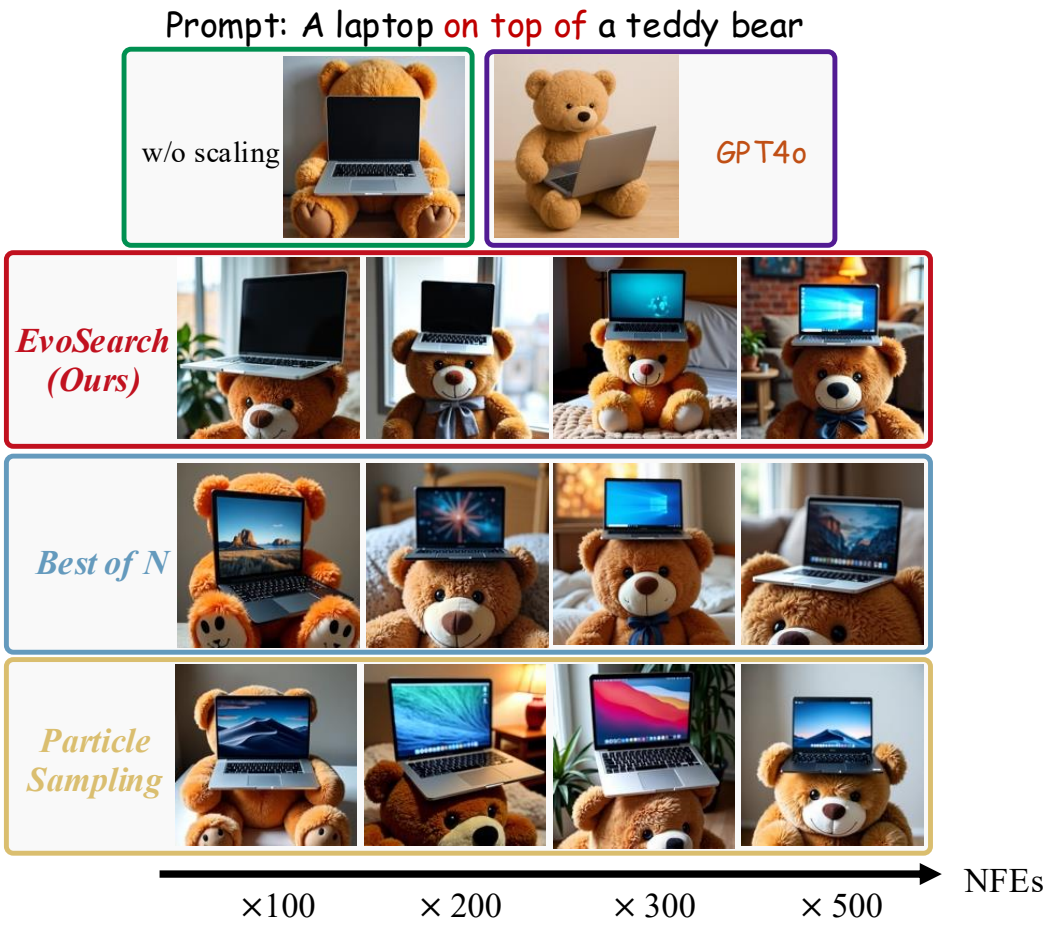


Figure 17: Results of test-time scaling for Flux.1-dev. **EvoSearch** can even achieve accurate spatial relationship interpretation with only $10\times$ scaled computation budget, while consistently improving image quality through higher NFEs.

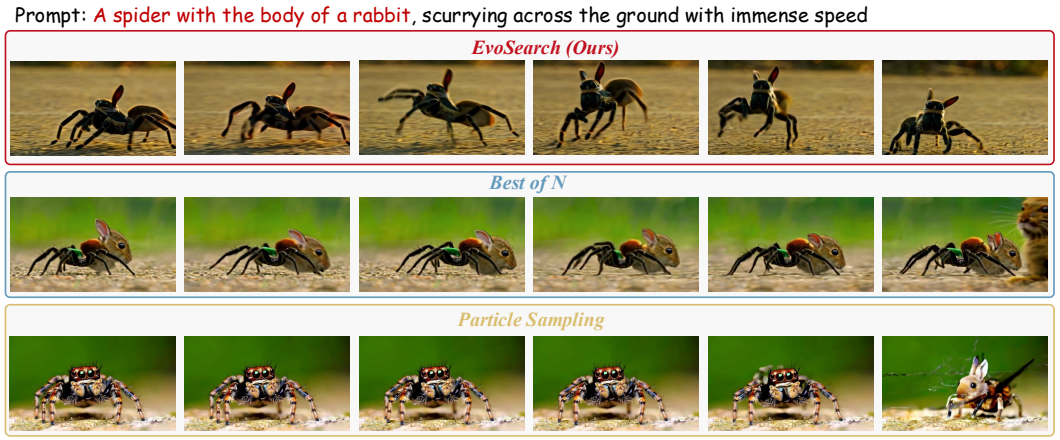


Figure 18: Results of test-time scaling for Hunyuan 13B. The denoising step is 30, and we scale up the test-time computation by $20\times$. Only **EvoSearch** generates high-quality video aligned closely with the text prompt.

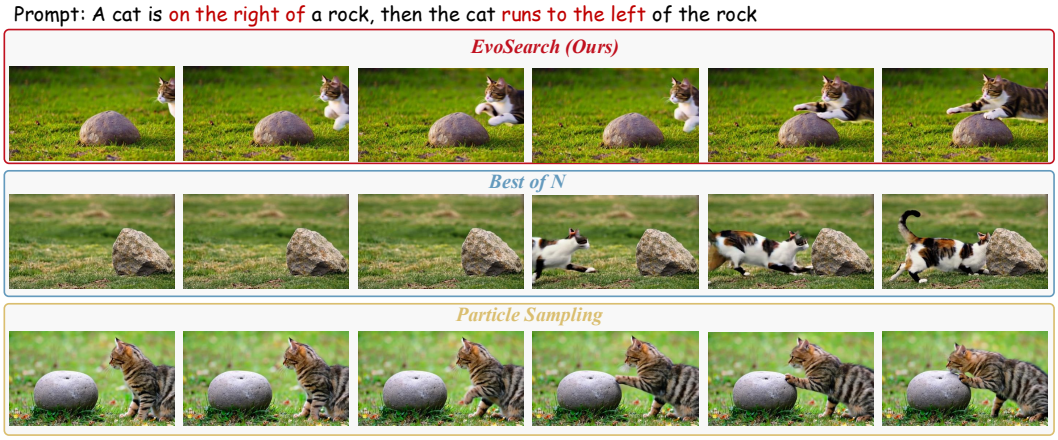


Figure 19: Results of test-time scaling for Hunyuan 13B. The denoising step is 30, and we scale up the test-time computation by 20×. **EvoSearch** successfully follows the text prompt while both **Best-of-N** and **Particle Sampling** fail.

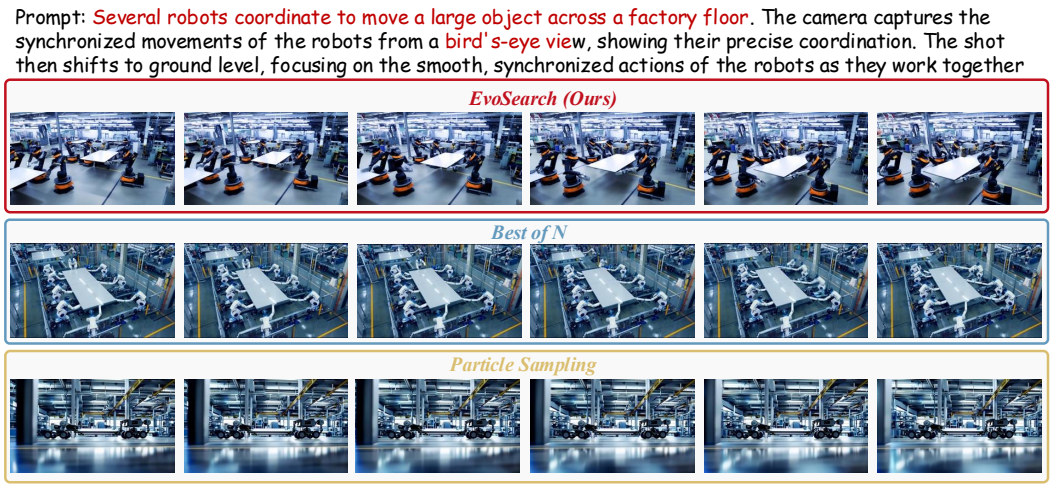


Figure 20: Results of test-time scaling for Hunyuan 13B. The denoising step is 30, and we scale up the test-time computation by 20×. **EvoSearch** demonstrates superior text alignment and higher-quality generation compared to baselines.

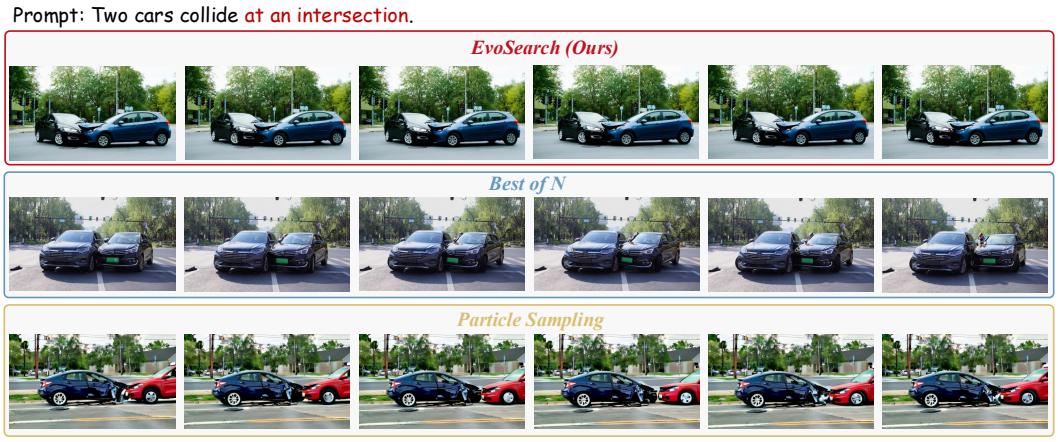


Figure 21: Results of test-time scaling for Hunyuan 13B. The denoising step is 30, and we scale up the test-time computation by 20×. The video generated by **EvoSearch** demonstrates better image quality and text alignment.

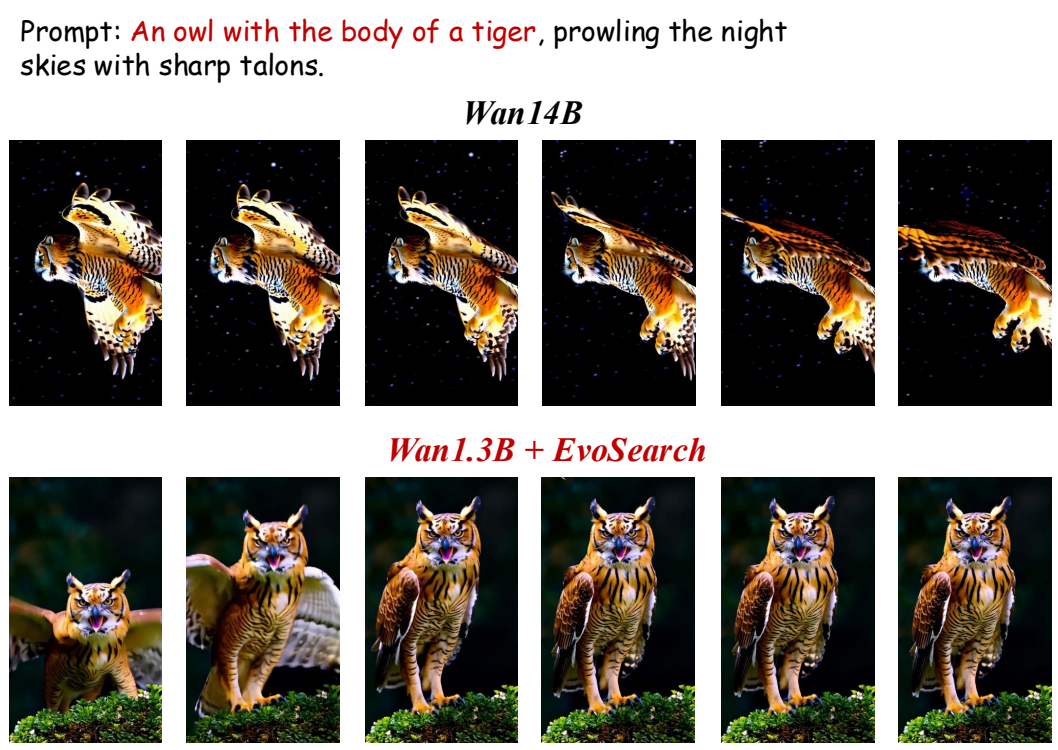


Figure 22: We scale up the test-time computation of Wan1.3B by 5×, ensuring equivalent inference times between Wan14B and **Wan1.3B+EvoSearch**. Qualitative results demonstrate that **EvoSearch** enables Wan1.3B to outperform Wan14B, its 10× larger counterpart.

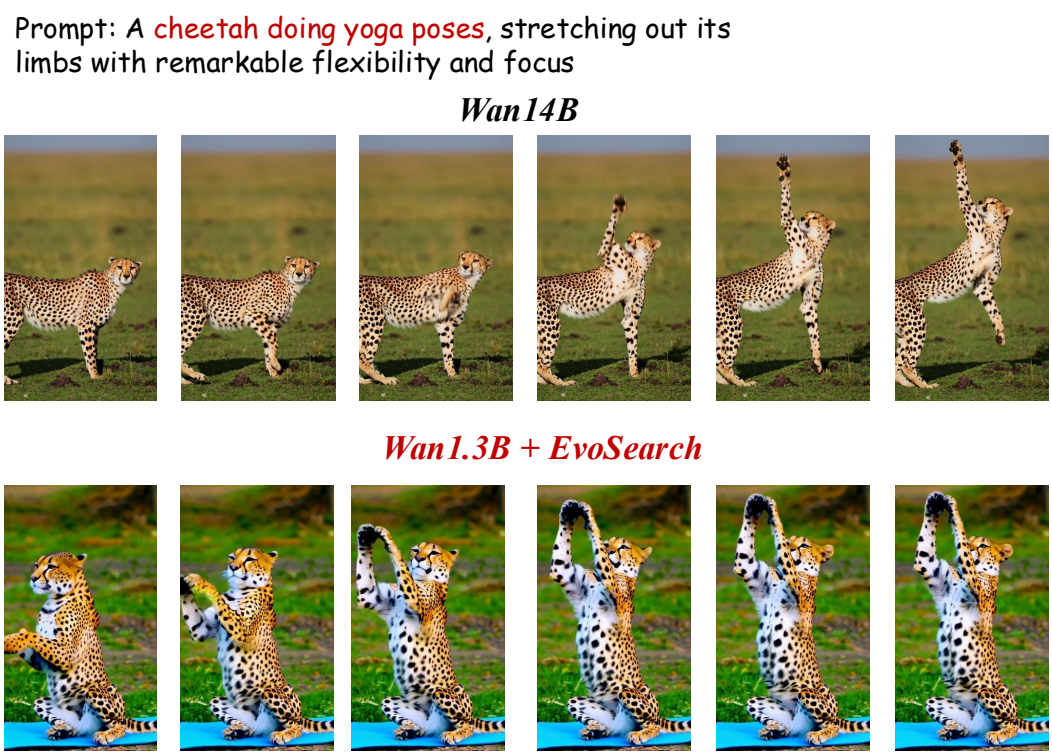


Figure 23: We scale up the test-time computation of Wan1.3B by 5×, ensuring equivalent inference times between Wan14B and **Wan1.3B+EvoSearch**. **EvoSearch** enables smaller models to achieve not only competitive but superior performance compared to their larger counterparts.

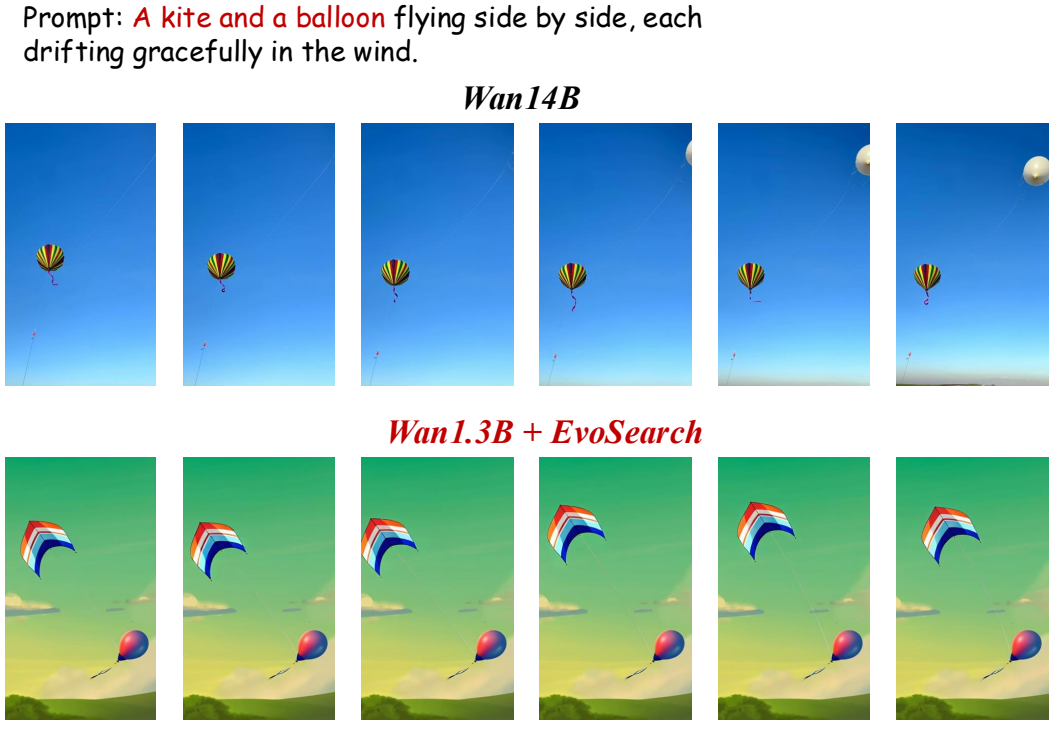


Figure 24: We scale up the test-time computation of Wan1.3B by 5×, ensuring equivalent inference times between Wan14B and **Wan1.3B+EvoSearch**. **EvoSearch** demonstrate superior text-alignment performance.

Figure 25: We scale up the test-time computation of Wan1.3B by $5\times$, ensuring equivalent inference times between Wan14B and **Wan1.3B+EvoSearch**. **EvoSearch** enhances Wan1.3B's capability in dynamic-attribute video generation.

Figure 26: We scale up the test-time computation of Wan1.3B by $5\times$, ensuring equivalent inference times between Wan14B and **Wan1.3B+EvoSearch**. **EvoSearch** enhances Wan1.3B's capability in handling challenging prompts, outperforming Wan14B given the same inference time.

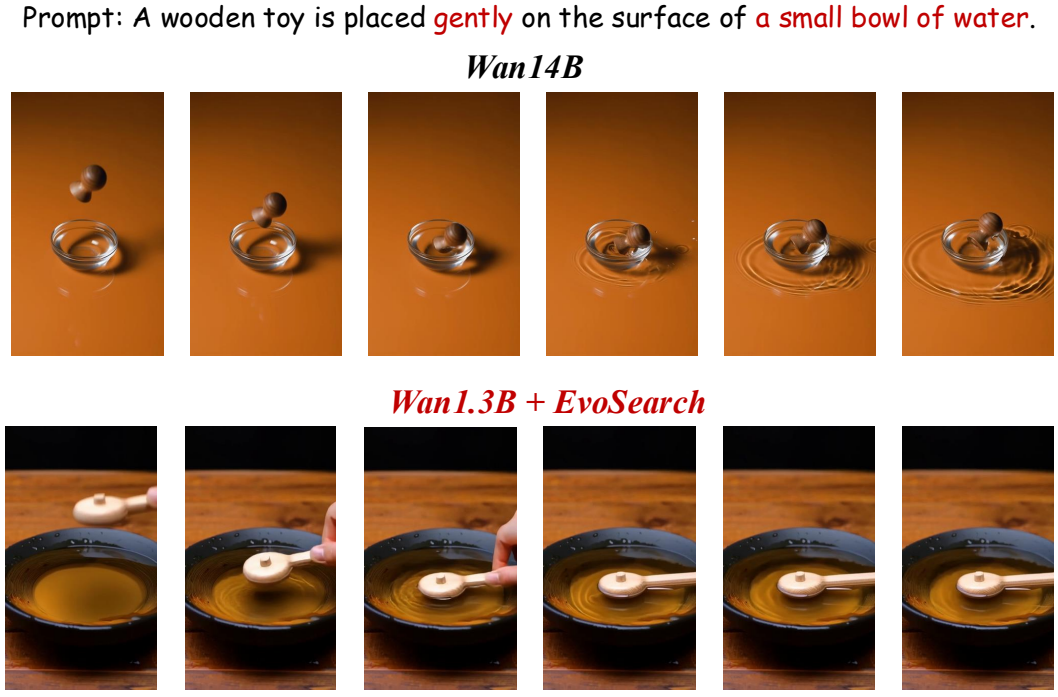


Figure 27: We scale up the test-time computation of Wan1.3B by $5\times$, ensuring equivalent inference times between Wan14B and **Wan1.3B+EvoSearch**. The video generated by **EvoSearch** follows the text instruction more closely, exhibiting improved logical consistency.

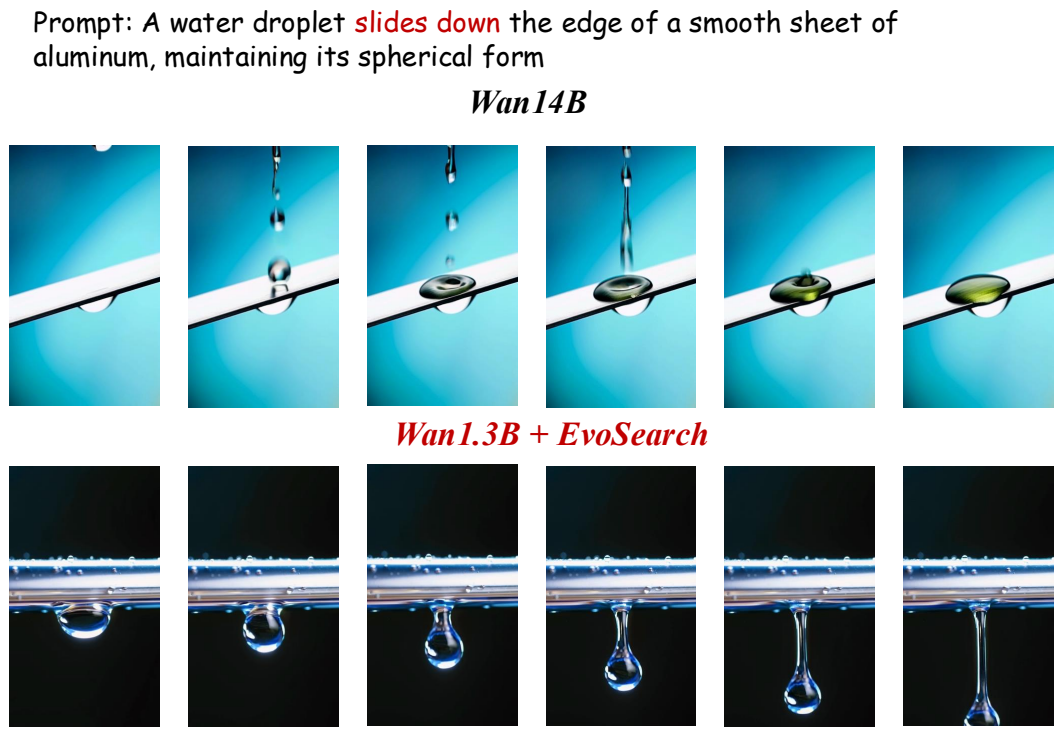


Figure 28: We scale up the test-time computation of Wan1.3B by $5\times$, ensuring equivalent inference times between Wan14B and **Wan1.3B+EvoSearch**. **EvoSearch**significantly improves the generation quality with superior semantic alignment.