

---

# SPARKE: Scalable Prompt-Aware Diversity and Novelty Guidance in Diffusion Models via RKE Score

---

**Mohammad Jalali**<sup>1\*</sup> **Haoyu Lei**<sup>1\*</sup> **Amin Gohari**<sup>2</sup> **Farzan Farnia**<sup>1</sup>

<sup>1</sup>Department of Computer Science and Engineering, The Chinese University of Hong Kong

<sup>2</sup>Department of Information Engineering, The Chinese University of Hong Kong

{mjalali24, hylei22, farnia}@cse.cuhk.edu.hk, agohari@ie.cuhk.edu.hk

## Abstract

Diffusion models have demonstrated remarkable success in high-fidelity image synthesis and prompt-guided generative modeling. However, ensuring adequate diversity in generated samples of prompt-guided diffusion models remains a challenge, particularly when the prompts span a broad semantic spectrum and the diversity of generated data needs to be evaluated in a prompt-aware fashion across semantically similar prompts. Recent methods have introduced guidance via diversity measures to encourage more varied generations. In this work, we extend the diversity measure-based approaches by proposing the *Scalable Prompt-Aware Rényi Kernel Entropy Diversity Guidance (SPARKE)* method for prompt-aware diversity guidance. SPARKE utilizes conditional entropy for diversity guidance, which dynamically conditions diversity measurement on similar prompts and enables prompt-aware diversity control. While the entropy-based guidance approach enhances prompt-aware diversity, its reliance on the matrix-based entropy scores poses computational challenges in large-scale generation settings. To address this, we focus on the special case of *Conditional latent RKE Score Guidance*, reducing entropy computation and gradient-based optimization complexity from the  $\mathcal{O}(n^3)$  of general entropy measures to  $\mathcal{O}(n)$ . The reduced computational complexity allows for diversity-guided sampling over potentially thousands of generation rounds on different prompts. We numerically test the SPARKE method on several text-to-image diffusion models, demonstrating that the proposed method improves the prompt-aware diversity of the generated data without incurring significant computational costs. We release our code on the project page: <https://mjalali.github.io/SPARKE>.

## 1 Introduction

Diffusion models [1, 2, 3] have rapidly become a prominent class of generative models, achieving state-of-the-art results in several generative modeling tasks [2, 4, 5]. Notably, their ability to produce intricate and realistic image and video data has significantly advanced various content creation pipelines [6, 7, 8, 9]. Despite these successes, ensuring sufficient diversity in generated outputs remains an area of active research, particularly in prompt-guided diffusion models, where the diversity of generated samples needs to be evaluated and optimized while considering the variability of the input prompts.

Diversity in prompt-guided sample generation also plays a critical role in ensuring fairness and lack of mode collapse in response to a certain prompt category, implying that the outputs do not collapse in response to a group of prompts with similar components. Therefore, an adequately diverse

---

\*Contributed Equally

### Overview of SPARKE Method in Iteration $n$ :

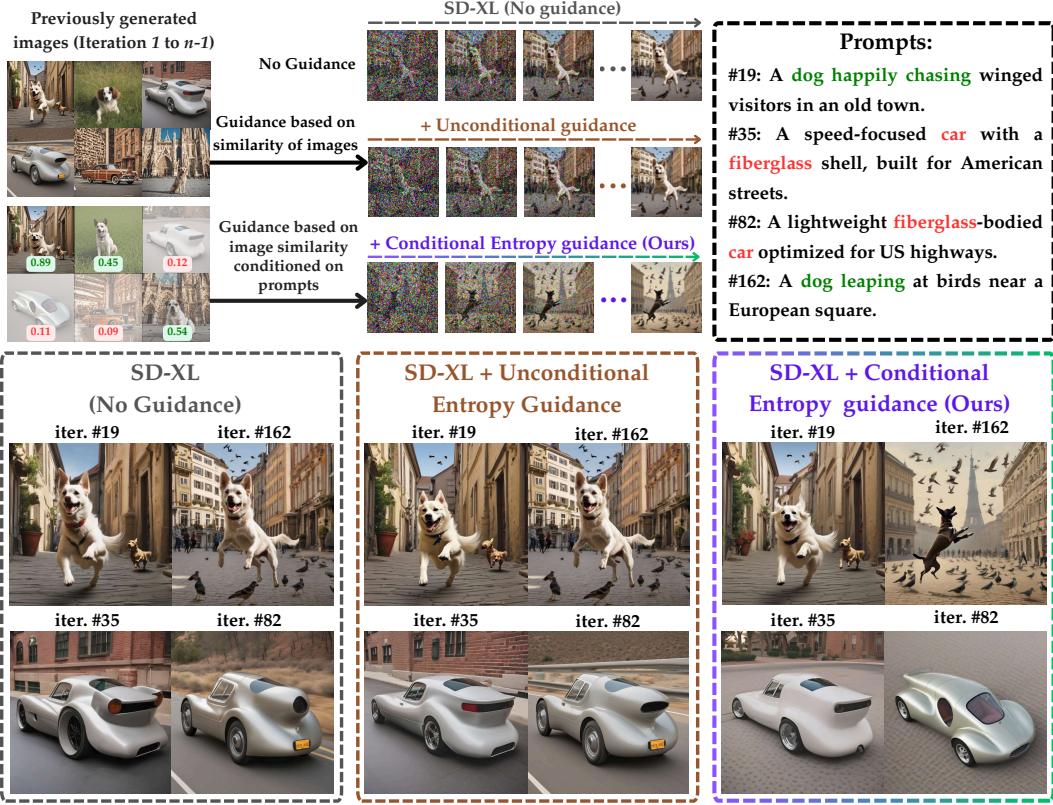


Figure 1: Overview of the proposed SPARKE method in generating images at different iterations in comparison to the vanilla Stable Diffusion-XL [5] model. We also compare the conditional-RKE guidance in SPARKE with the baseline Vendi Score guidance [10] (unconditional, in latent space).

prompt-guided generation model is supposed to provide varied and balanced outputs conditioned on each prompt category. For example, in geographically diverse image synthesis, the generation model is supposed to represent a broad spectrum of regional styles, preventing overrepresentation of dominant patterns while ensuring fair coverage across different locations [10]. Similarly, in data augmentation for visual recognition, generating a semantically varied set of samples enhances the model’s generalizability and reduces biases in prompt-guided sample creation [11, 12].

A recently explored strategy for diversity-aware generation in diffusion models is to utilize a diversity score for guiding the sample generation. One such method is the contextualized Vendi score guidance (c-VSG) [10]. The c-VSG approach builds upon the Vendi diversity measure [13], which is a kernel matrix-based entropy score evaluating diversity in generative models. The numerical results in [10] demonstrate the effectiveness of regularizing the Vendi score in guiding diffusion models toward more diverse sample distributions over multiple rounds of data generation. Another method, CADS [14] tends to improve sample diversity through the perturbation of conditional inputs, typically resulting in an increased Vendi score.

Although the mentioned diversity-score-based diffusion sampling methods apply smoothly to the unconditional sample generation without a varying input prompt, a scalable prompt-aware extension to prompt-guided diffusion models remains a challenge. As discussed in [10], one feasible framework extension to prompt-guided generation can leverage a pre-clustering of prompts to apply the Vendi diversity guidance separately within every prompt cluster. However, in successive queries to prompt-guided generative models, prompts often vary in subtle details, making their pre-clustering into a fixed number of hard clusters challenging. In addition, relying on small clusters of highly similar prompts could lead to limited diversity across prompt clusters, due to missing the partial similarities of the prompts in different clusters.

## Diversity Guidance in Latent Diffusion Models

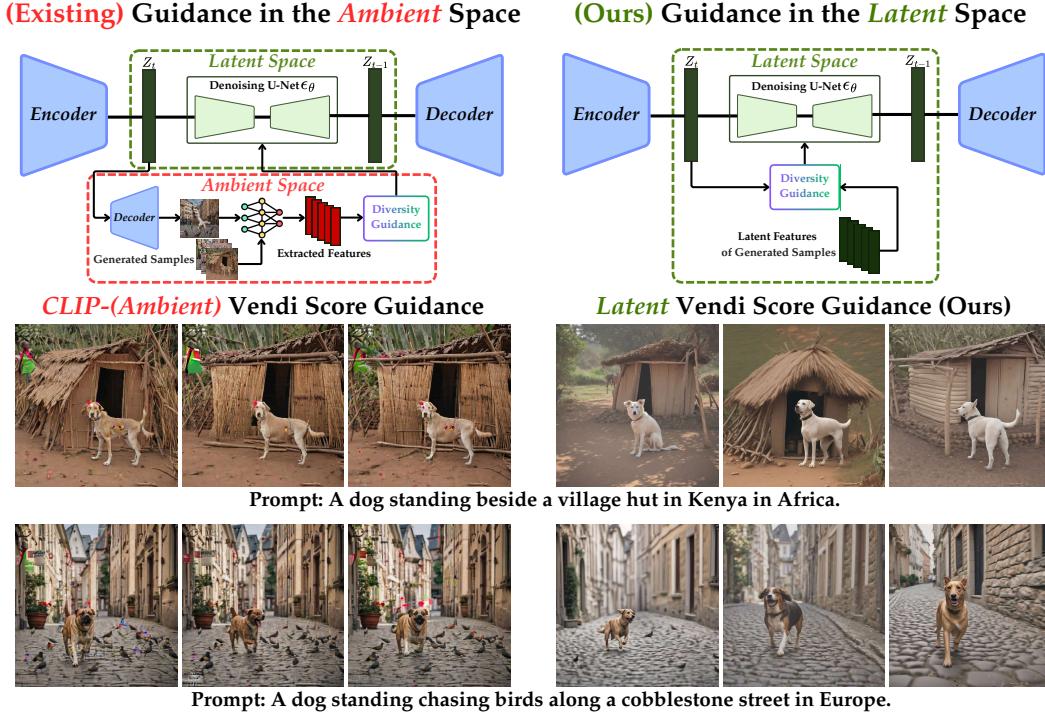
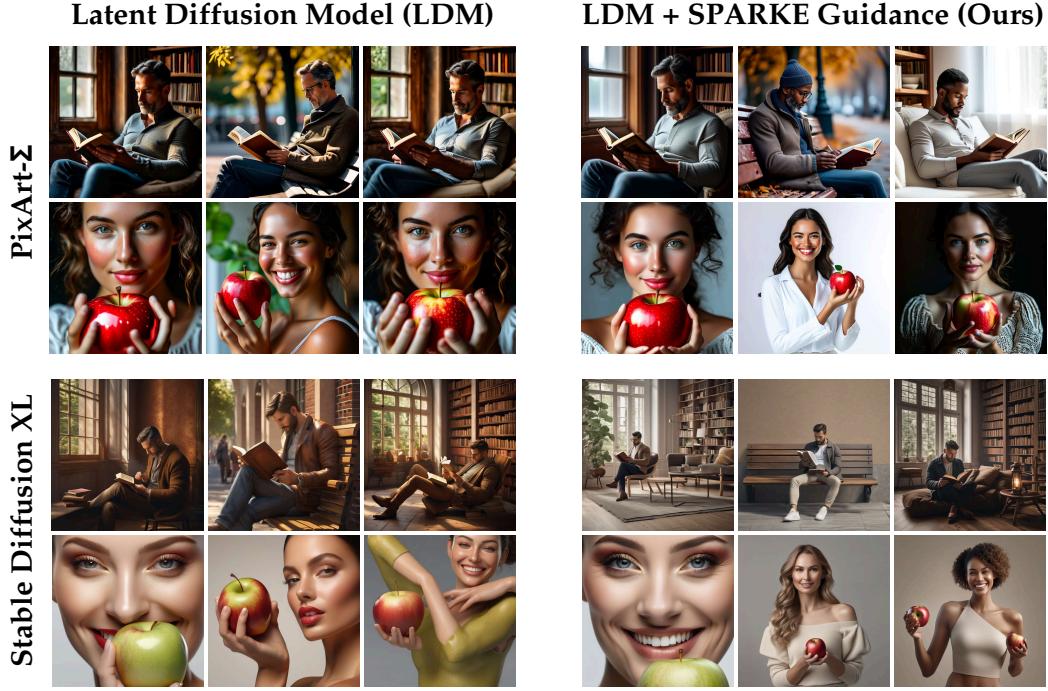


Figure 2: Comparison of latent entropy-based diversity guidance (ours) vs. ambient entropy diversity guidance in Latent Diffusion Models (LDMs). The experiment is performed with the SD-XL LDM.

In this work, we propose ***Scalable Prompt-Aware Rényi Kernel Entropy Diversity Guidance (SPARKE)*** in diffusion models, applying the conditional entropy-based diversity score family [15] of the latent representation in the latent diffusion models (LDMs). As illustrated in Figure 1, SPARKE directly incorporates prompts into the diversity calculation without pre-clustering. This extension enables a more dynamic control over the diversity calculation with varying prompts, which can be interpreted as an adaptive kernel-based similarity evaluation without grouping the prompts into clusters. According to the conditional entropy guidance in SPARKE, we update the kernel matrix of generated data by taking the Hadamard product with the kernel matrix of input prompts. Applying an appropriate kernel function for text prompts, the updated similarity matrix will assign a higher weight to the pairwise interaction of samples with higher prompt similarity.

While the conditional entropy guidance offers flexibility by relaxing the requirement for explicit hard clustering of the prompts, it relies on computing the gradient of the matrix-based entropy of the kernel matrix, which will be computationally expensive in large-scale generation tasks. Since the entropy score is computed for an  $n \times n$  kernel similarity matrix of  $n$  samples, the entropy estimation of its eigenvalues would need an eigendecomposition, leading to  $\mathcal{O}(n^3)$  complexity. This computational cost would be heavy in settings where thousands of samples need to be diversified across a large set of prompts. Thus, while conditioning diversity on prompts improves the adaptability of our proposed approach, we further need to develop a computationally scalable solution to extend the approach to a larger-scale generation of diverse data.

To improve scalability while preserving diversity-aware generation, we propose applying the order-2 Rényi matrix-based entropy, defined as the RKE Score [16], in the SPARKE approach. This RKE-based formulation replaces eigenvalue decomposition with a Frobenius norm-based entropy measure, which significantly reduces computational overhead. As a result, the complexity of entropy estimation reduces to  $\mathcal{O}(n^2)$ , while the gradient-based optimization in guided diffusion models can be performed even more efficiently in  $\mathcal{O}(n)$  time due to the canceled terms with zero gradients. By integrating these computational improvements, we introduce SPARKE framework, which maintains the advantages of prompt-conditioned diversity guidance while making large-scale sample generation feasible.



Prompts (row 1): 1. A man sitting quietly and reading a book, 2. A man reading book on a bench, 3. A man reading a book in a relaxed setting.  
 Prompts (row 2): 1. A young woman holding an apple, 2. A female holding an apple and smiling, 3. A woman showing apple with both hands.

Figure 3: Qualitative Comparison of samples generated by the base latent diffusion model (LDM), PixArt- $\Sigma$ , and Stable Diffusion XL, vs. LDM guided via our proposed SPARKE guidance.

To evaluate the effectiveness of our approach, we perform several numerical experiments on standard text-to-image generation models. Our results indicate that Conditional RKE Score Guidance not only scales efficiently to a large number of prompts but also maintains high sample diversity while preserving fidelity. Compared to prior methods, our approach achieves a more balanced trade-off between diversity and computational efficiency, making it suitable for real-world generative modeling applications that require scalability. The following is a summary of the contributions of our work:

- We introduce the conditional entropy score guidance as a prompt-aware diversity promotion tool in sampling from diffusion models.
- We propose using the order-2 Renyi kernel entropy (RKE) score to reduce the computational complexity of the entropy-based diversity guidance.
- We propose the application of entropy guidance in the latent space of latent diffusion models, improving the efficiency and performance in the entropy-based diversity guidance (See Figure 2).
- We test our method on several state-of-the-art diffusion models (see Figures 3) and text-to-image benchmarks, indicating its ability to improve sample diversity and computational efficiency.

## 2 Related Works

**Standard and Latent Diffusion Models.** Diffusion generative models [1, 2, 17, 3] learns to reverse an iterative noising process, effectively estimating the gradient of the data log-density (the Stein score [18]) to generate new data samples. This approach has shown remarkable capabilities in synthesizing high-fidelity images [6, 19, 20, 7]. Despite the impressive results, a primary limitation of diffusion models was the substantial computational cost, particularly when operating directly in high-dimensional spaces like pixel space. To address this challenge, Latent Diffusion Models (LDMs) [4, 21] perform the forward and denoising processes in an encoded latent space, enabling high-quality images such as Stable Diffusion [4, 5] and video [22, 23, 8] synthesis at a large scale.

**Conditional Generation with Guidance.** The ability to control generative processes with specific conditions is increasingly crucial for practical applications, based on conditional inputs like text-

guided [20, 24, 25], class labels [6], style images [26, 27], or human motions [28], etc. Methods for conditional generation with guidance are categorized as either training-based or training-free. Training-based approaches either learn a time-dependent classifier that guides the noisy sample  $\mathbf{x}_t$  towards the condition  $\mathbf{y}$  [6, 24, 29, 25], or directly train the conditional denoising model  $\epsilon_\theta(\mathbf{x}_t, t, \mathbf{y})$  via few-shot adaptation [26, 4, 30]. In contrast, training-free guidance aims at zero-shot conditional generation by leveraging a pre-trained differentiable target predictor without requiring any training. This predictor can be a classifier, loss function, or energy function quantifying the alignment of a sample with the target condition [31, 32, 33, 34]. Our work can be included as a training-free guidance approach that applies conditional entropy scores guidance to enhance the diversity of samples.

**Quantifying Diversity and Novelty.** Diversity is quantified using both reference-based [35, 36] and reference-free metrics. Reference-free metrics include the Vendi Score [13, 37, 38], the RKE score [16] for unconditional models, and the Conditional-RKE [15] and Scendi [39] scores for conditional models. Also, the diversity metrics have been extended to online and distributed model selection tasks [40, 41, 42, 43]. For novelty, prior work [44, 45, 46] analyzes how generated samples differ from a reference model, with [45, 46] proposing a spectral method to measure the entropy of novel modes. [47, 48, 49, 50] also introduce kernel-based methods to compare and align two embeddings. In this work, we propose a novelty guidance approach that operates with respect to a reference dataset.

**Guidance for Improving Diversity.** A common strategy in diffusion-based generative modeling is the use of guidance mechanisms to balance quality and diversity [51, 52, 53]. For example, classifier-free guidance methods [52] considerably enhance prompt alignment and image quality but often compromise diversity due to overly deterministic conditioning. Several works have attempted to address this diversity challenge. To encourage diversity, [54] introduced a strategy that samples from the data manifold’s low-density regions, however, their method operates directly in pixel space, posing challenges in adapting it effectively to latent diffusion frameworks. Another line of work is fine-tuning. In [55], the authors provide a finetuning method using Reinforcement Learning to improve the diversity of generated samples using a diversity reward function.

Recent works tackle this problem in the denoising phase. The CADS framework [14] shows that adding Gaussian noise to the conditioning signals during inference increases sample diversity. Particle Guidance (PG) [56] employs non-IID sampling from the joint distribution defined by a diffusion model combined with a potential function that maximizes pairwise dissimilarity across all samples, independent of semantic context. Similar to PG, ProCreate [57] uses DreamSim embeddings to find similar images via log energy and maximizes embedding-space distances. The concurrent method SPELL [58] adds repellency terms during sampling to prevent samples in a batch from being too close. These methods maximize across all samples without considering the prompts. However, these methods are prompt-agnostic, while ours dynamically conditions diversity guidance on the input.

The recent work [10] introduces contextualized Vendi Score Guidance (c-VSG), enhancing generative diversity during the denoising process using Vendi Score [13]. Their approach, however, requires the same prompts, restricting its applicability across diverse prompt scenarios. To tackle this issue, [15] uses Conditional-Vendi score guidance for a prompt-aware diversity guidance. On the other hand, our method leverages the RKE Score [16] to significantly enhance computational efficiency and sample complexity. Additionally, we propose prompt-aware guidance inspired by [15], enabling adaptive soft-clustering and effective conditioning on semantically distinct prompts. Moreover, unlike [10], which applies guidance based on latent encoded features of reference images, our strategy directly applies guidance within the diffusion model’s latent space, saving computational costs and enabling a prompt-aware diversity improvement.

### 3 Preliminaries

#### 3.1 Kernel function and Vendi diversity scores

Consider a sample  $\mathbf{x} \in \mathcal{X}$  in the support set  $\mathcal{X}$ . A function  $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  is called a kernel function, if for every integer  $n \in \mathbb{N}$  and sample set  $\{x_1, \dots, x_n\} \in \mathcal{X}$ , the following kernel similarity matrix  $K \in \mathbb{R}^{n \times n}$  is positive semi-definite (PSD):

$$K_X = \begin{bmatrix} k(x_1, x_1) & \cdots & k(x_1, x_n) \\ \vdots & \ddots & \vdots \\ k(x_n, x_1) & \cdots & k(x_n, x_n) \end{bmatrix} \quad (1)$$

We assume that the kernel function is normalized, i.e.,  $k(x, x) = 1$  for every  $x \in \mathcal{X}$ . For a general (potentially unnormalized) kernel  $k$ , one can define its normalized counterpart as  $\tilde{k}(x, y) = k(x, y) / \sqrt{k(x, x) k(y, y)}$ , which we apply whenever normalization is required. The Rényi Kernel Entropy (RKE) diversity score [16] is defined and analyzed as the inverse Frobenius norm-squared of the trace-normalized kernel matrix:

$$\text{RKE}(x_1, \dots, x_n) := \exp\left(H_2\left(\frac{1}{n}K_X\right)\right) = \left\|\frac{1}{n}K_X\right\|_F^{-2} \quad (2)$$

where  $\|\cdot\|_F$  denotes the Frobenius norm. As theoretically shown in [16], the RKE score can be regarded as a mode count of a mixture distribution with multiple modes.

To extend this entropy-based diversity scores to conditional (i.e., prompt-aware) diversity measurement for prompt-guided generative models, [15] propose the application of conditional kernel matrix entropy measures, resulting in the following definition of order-2 Conditional-RKE given the Hadamard product  $\frac{1}{n}K_X \odot K_T$  of the output data  $K_X$  and input prompt  $K_T$  kernel matrices:

$$\text{Conditional-RKE}(x_1, \dots, x_n; t_1, \dots, t_n) := \frac{\|K_T\|_F^2}{\|K_X \odot K_T\|_F^2}, \quad (3)$$

Note that here the Hadamard product of the prompt  $(t_1, \dots, t_n)$  kernel matrix  $K_T$  and output  $(x_1, \dots, x_n)$  kernel matrix  $K_X$  is defined as the elementwise product of the matrices, which is guaranteed to be a PSD matrix by the Schur product theorem.

### 3.2 Conditional Latent Diffusion Models

Latent Diffusion Models (LDMs) [4] can achieve scalability and efficiency by performing the diffusion process within a compressed latent space, rather than directly in pixel space. A pre-trained autoencoder is used in LDMs, consisting of an encoder  $\mathcal{E}$  and a decoder  $\mathcal{D}$ . The encoder  $\mathcal{E}$  maps high-dimensional images  $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$  to a lower-dimensional latent representation  $\mathbf{z}_0 = \mathcal{E}(\mathbf{x}) \in \mathbb{R}^{h \times w \times c}$ , while the decoder  $\mathcal{D}$  reconstructs the images  $\tilde{\mathbf{x}} = \mathcal{D}(\mathbf{z}_0)$  from a denoised latent variable  $\mathbf{z}_0$ . The forward diffusion process applies in the latent space as  $\mathbf{z}_t = \sqrt{\alpha_t} \mathbf{z}_0 + \sqrt{1 - \alpha_t} \boldsymbol{\epsilon}_t$ , where  $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  and  $\alpha_t$  are predefined noise schedule parameters. Then the conditional LDMs with conditions  $\mathbf{y}$  are trained to predict the noise  $\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t, \mathbf{y})$  at each time step  $t$ , also learn the score of  $p_t(\mathbf{z}_t | \mathbf{y})$  [17, 3]:

$$\min_{\theta} \mathbb{E}_{\mathbf{z}_t, \boldsymbol{\epsilon}_t, t, \mathbf{y}} \left[ \|\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t, \mathbf{y}) - \boldsymbol{\epsilon}_t\|_2^2 \right] = \min_{\theta} \mathbb{E}_{\mathbf{z}_t, \boldsymbol{\epsilon}_t, t, \mathbf{y}} \left[ \left\| \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t, \mathbf{y}) + \sqrt{1 - \alpha_t} \nabla_{\mathbf{z}_t} \log p_t(\mathbf{z}_t | \mathbf{y}) \right\|_2^2 \right]. \quad (4)$$

## 4 Scalable Prompt-Aware Diversity Guidance in Diffusion Models

Utilizing the kernel-based entropy diversity scores and latent diffusion models (LDMs) described in Section 3.2, we introduce a scalable conditional entropy-based framework extending the recent Vendi score-based approach [10] in diversity-guided generative modeling to prompt-aware diversity enhancement that suits the conditional text-to-image models. Specifically, we first propose extending the order-1 Vendi score to the general matrix-based order- $\alpha$  Rényi entropy and subsequently to the conditional order- $\alpha$  Rényi entropy for achieving higher prompt-conditioned diversity. In addition, we demonstrate a computationally efficient special case in this family of diversity score functions by considering the order-2 entropy measures. This choice of entropy function leads to the RKE-guidance and Conditional-RKE guidance approaches for enhancing overall diversity and prompt-aware diversity in sample generation via diffusion models.

### 4.1 Diversity guidance via general order- $\alpha$ Rényi Entropy and RKE scores

The existing diversity score-based guidance approach, including the Vendi guidance in [10], optimizes the diversity score of the embedded version of the generated outputs, e.g. the CLIP embedding [59] of image data generated by the prompt-guided diffusion models. However, such a guidance process by considering an embedding on top of the diffusion model is computationally expensive and may not lead to semantically diverse samples. As discussed in [13, 37], computing the order-1 Vendi score and its gradients requires at least  $\Omega(n^{2.367})$  computations for  $n$  samples, and in practice involves

$O(n^3)$  computations for the eigen-decompositions of kernel matrices. This computational complexity limits practical window sizes of the existing Vendi guidance approach to a few hundred samples, i.e. the guidance function cannot be computed by a standard GPU processor for a sample size  $n > 500$ .

To reduce computational complexity, we extend the order-1 Vendi score guidance to the general order- $\alpha$  Rényi kernel entropy, and adopt the specific case  $\alpha = 2$ , referred to as the RKE score [16]. This formulation significantly lowers computational cost and is defined as:

$$\mathcal{L}_{\text{RKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}) = \left\| \frac{1}{n} K_Z \right\|_F^{-2} = \left( \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n k^2(\mathbf{z}^{(i)}, \mathbf{z}^{(j)}) \right)^{-1}, \quad (5)$$

with kernel matrix  $(K_Z)_{ij} = k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})$  computed directly for the latent representations and  $n$  denotes the total number of generated samples. Therefore, applying  $\mathcal{L}_{\text{RKE}}$  as the guidance potential function reduces the computational complexity from  $O(n^3)$  in order-1 Vendi score to  $O(n^2)$ , enabling significantly larger batch sizes during the guided sampling process and resulting in higher efficiency.

## 4.2 Diversity-Guided Sampling in Latent Diffusion Models

In contrast to previous work using order-1 Vendi score as the diversity guidance in ambient image space [10], we integrate efficient IRKE score diversity guidance directly into latent-space sampling. Standard LDMs employ classifier-free guidance (CFG) [52] for conditional sampling, iteratively denoising latents  $\mathbf{z}_{t-1}$  from the noisy latents  $\mathbf{z}_t$  at time step  $t$  via reverse samplers:

$$\mathbf{z}_{t-1} \leftarrow \text{Sampler}(\mathbf{z}_t, \hat{\epsilon}_\theta(\mathbf{z}_t, t, \mathbf{y})), \quad (6)$$

To efficiently promote diversity within the latent space, we optimize the Inverse-RKE (IRKE) score loss. Proposition 1 defines this loss and its gradient:

**Proposition 1.** *Let  $Z = \{\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \dots, \mathbf{z}^{(n)}\}$  denote a set of  $n$  generated data. Let kernel function  $k : \mathcal{Z} \times \mathcal{Z} \rightarrow \mathbb{R}$  be symmetric and normalized, i.e.  $k(\mathbf{z}, \mathbf{z}) = 1$  for every  $\mathbf{z} \in \mathcal{Z}$ . Then, we observe that the function  $\mathcal{L}_{\text{RKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)})$  in Eq. (5) changes monotonically with the Inverse-RKE function  $\mathcal{L}_{\text{IRKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}) = 1/\mathcal{L}_{\text{RKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)})$ , whose gradient with respect to  $\mathbf{z}^{(n)}$  is:*

$$\nabla_{\mathbf{z}^{(n)}} \mathcal{L}_{\text{IRKE}} \propto \sum_{i=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \nabla_{\mathbf{z}^{(n)}} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}). \quad (7)$$

The above Proposition 1 implies that an optimization objective maximizing  $\mathcal{L}_{\text{RKE}}$  to promote diversity can be equivalently pursued by minimizing  $\mathcal{L}_{\text{IRKE}}$ , reducing the computational complexity from  $O(n^2)$  to  $O(n)$ . Leveraging this computational efficacy, our approach directly updates the gradient of  $\mathcal{L}_{\text{IRKE}}$  rather than computing the gradient of  $\mathcal{L}_{\text{RKE}}$  to the latent  $\mathbf{z}^{(n)}$ , allowing for an efficient, explicit diversity-promoting update. Specifically, for Latent Diffusion Models (LDMs), we introduce the following diversity-guided sampling via Eq. (7) in each time step  $t$ :

$$\mathbf{z}_t \leftarrow \mathbf{z}_t - \eta \cdot \nabla_{\mathbf{z}^{(n)}} \mathcal{L}_{\text{IRKE}} \quad (\text{IRKE Diversity Guidance}) \quad (8)$$

where  $\eta$  represents the guidance scale and  $n$  denotes the number of previously generated samples.

## 4.3 Conditional Prompt-Aware Diversity Guidance

To explicitly capture prompt-conditioned diversity and focus on the computational efficiency of order-2, we introduce the Conditional RKE loss:

$$\mathcal{L}_{\text{Cond-RKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}; \mathbf{y}^{(1)}, \dots, \mathbf{y}^{(n)}) := \frac{\|K_Y\|_F^2}{\|K_Y \odot K_Z\|_F^2}, \quad (9)$$

where  $(K_Y)_{ij} = k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)})$  denotes the similarity among condition (e.g., prompts) embeddings. Similarly, we reduce the computational complexity from  $O(n^2)$  to  $O(n)$ , by computing the gradient of Conditional Inverse-RKE (Cond-IRKE) defined in Proposition 2:

**Proposition 2.** *Let  $Z$  denote a set of  $n$  generated data, and  $Y = \{\mathbf{y}^{(1)}, \mathbf{y}^{(2)}, \dots, \mathbf{y}^{(n)}\}$  is the set of corresponding conditions. Let the kernel function  $k$  be symmetric and normalized. Then, we observe that the function  $\mathcal{L}_{\text{Cond-RKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}; \mathbf{y}^{(1)}, \dots, \mathbf{y}^{(n)})$  in Eq. (9) changes monotonically with*

---

**Algorithm 1:** Scalable Prompt-Aware Latent IRKE Diversity Guidance

---

**Input:** Latents  $\{z^{(1)}, \dots, z^{(n-1)}\}$ , prompt features  $\{y^{(1)}, \dots, y^{(n-1)}\}$ , kernel functions  $k_Z$ ,  $k_Y$ , denoising model  $\epsilon_\theta$ , diffusion reverse sampler, guidance scale  $w$ , diversity guidance scale  $\eta$ , decoder  $\mathcal{D}$

**Output:** Diverse generated samples  $\{\tilde{x}^{(n)}\}$  with prompt features  $y^{(n)}$

```

1 for  $t = T$  to 1 do
2   Compute CFG guided noise:  $\hat{\epsilon}_\theta^{(n)} = (1 + w) \cdot \epsilon_\theta(z_t^{(n)}, t, y^{(n)}) - w \cdot \epsilon_\theta(z_t^{(n)}, t)$  ;
3   Perform one-step denoising:  $z_{t-1}^{(n)} \leftarrow \text{Sampler}(z_t^{(n)}, \hat{\epsilon}_\theta^{(n)})$  ;
4   Compute  $K_Z$ ,  $K_Y$ , and  $\mathcal{L}_{\text{Cond-IRKE}} = \frac{1}{n^2} \|K_Z \odot K_Y\|_F^2$  ;
5   Compute the gradients  $g^{(n)} = \nabla_{z^{(n-1)}} \mathcal{L}_{\text{Cond-IRKE}}$  ;
6   Update latents:  $z_{t-1}^{(n)} \leftarrow z_{t-1}^{(n)} - \eta \cdot g^{(n)}$  ;
7 Decode final samples:  $\tilde{x}^{(n)} = \mathcal{D}(z_0^{(n)})$ 

```

---

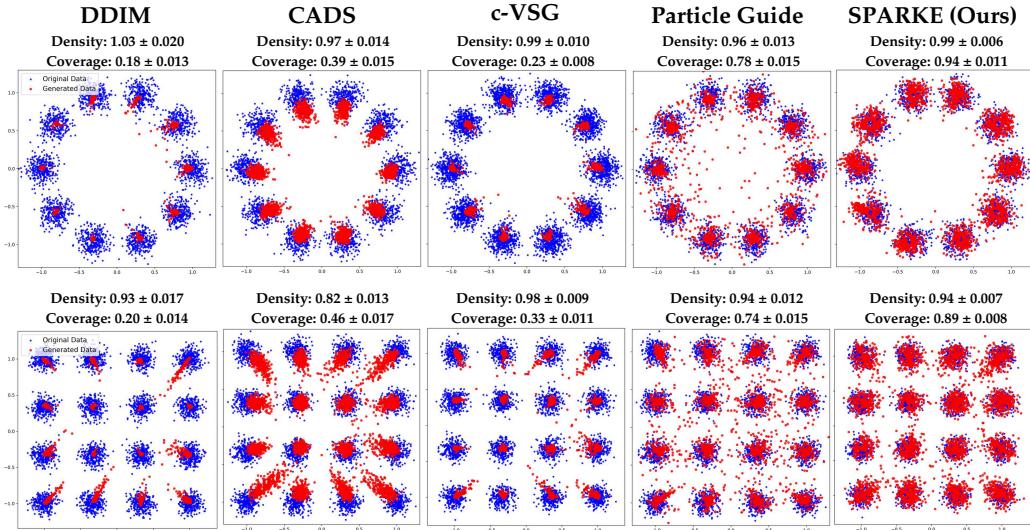


Figure 4: Comparison of SPARKE (Conditional RKE) Guidance with baselines on 2D GMMs.

the Conditional Inverse-RKE function  $\mathcal{L}_{\text{Cond-IRKE}}(z^{(1)}, \dots, z^{(n)}; y^{(1)}, \dots, y^{(n)}) = \|\tilde{K}_Z \odot \tilde{K}_Y\|_F^2$ , whose gradient with respect to  $z^{(n)}$  is:

$$\nabla_{z^{(n)}} \mathcal{L}_{\text{Cond-IRKE}} \propto \sum_{i=1}^{n-1} k_Z(z^{(i)}, z^{(n)}) k_Y(y^{(i)}, y^{(n)})^2 \nabla_{z^{(n)}} k_Z(z^{(i)}, z^{(n)}). \quad (10)$$

Then we propose the following prompt-aware diversity-guided sampling via Eq. (10) in time step  $t$ :

$$z_t \leftarrow z_t - \eta \cdot \nabla_{z^{(n)}} \mathcal{L}_{\text{Cond-IRKE}} \quad (\text{Cond-IRKE Diversity Guidance}) \quad (11)$$

Algorithm 1 provides an explicit outline of this procedure. This Cond-IRKE guidance explicitly aligns the diversity of generated samples with respective prompts, while significantly maintaining the computational efficiency in gradient calculation. In Appendix B.3, we provide a theoretical interpretation based on a stochastic differential equation.

## 5 Numerical Results

We evaluated the performance of the SPARKE framework on various conditional diffusion models, where our results support that SPARKE can boost output diversity without compromising fidelity scores considerably. For the complete set of our numerical results, we refer to the Appendix D.

**Baselines.** We compare our method with CADS [14], Particle Guidance [56], and Contextualized Vendi Score Guidance (c-VSG) [10]. In experiments without a reference, we use VSG [10].

Table 1: Quantitative comparison of guidance methods on Stable Diffusion 2.1.

Method	CLIPScore $\uparrow$	KD $\times 10^2 \downarrow$	Cond-Vendi Score $\uparrow$	Vendi Score $\uparrow$	In-batch Sim. $\times 10^2 \downarrow$
SD (No Guidance) [60]	31.20	52.37	26.54	369.37	80.36
c-VSG [10]	28.75	56.25	27.91	376.48	79.82
CADS [14]	29.89	55.08	28.73	380.08	79.44
<b>latent RKE Guidance (Ours)</b>	30.18	55.37	29.88	387.59	79.01
<b>SPARKE: latent Cond-RKE (Ours)</b>	30.96	53.15	32.57	405.51	75.68

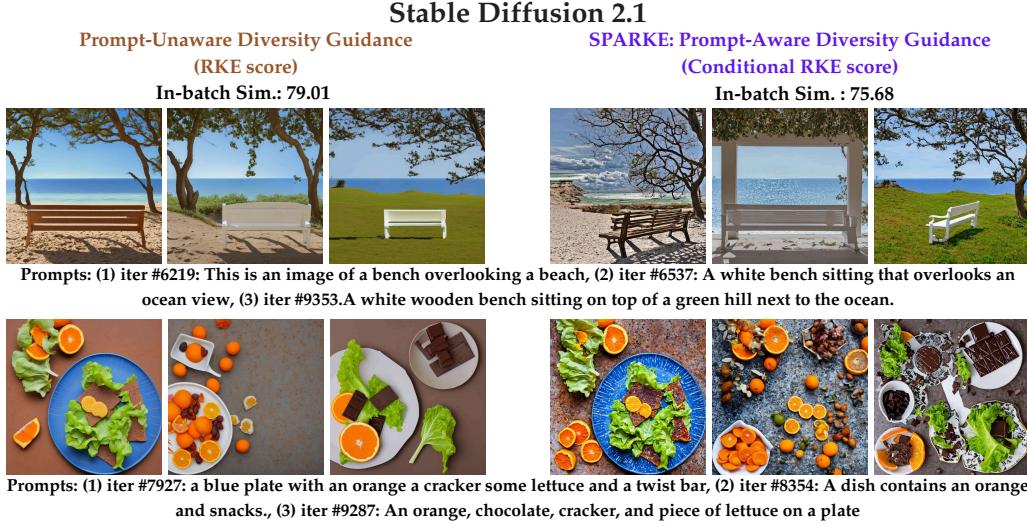


Figure 5: Comparison of SPARKE prompt-aware diversity guidance via conditional-RKE score vs. diversity-unaware diversity guidance using the RKE score on SD 2.1 text-to-image generation.

**Models.** We used Stable Diffusion [60] and the larger-scale Stable Diffusion XL (SDXL) [5], and PixArt- $\Sigma$  [61] in our experiments on text-to-image generation. We also provide additional experimental results on different SOTAs in the Appendix.

**Evaluation.** We compared our method with baselines in terms of output diversity and fidelity. Fidelity was measured using CLIPScore [62], and KID [63] to evaluate prompt quality and consistency, and Density [36] where a reference dataset was available. For diversity, we used Vendi [13] and Conditional-Vendi [15] scores, and we used Coverage [36] when a reference dataset existed. We measured diversity within each prompt cluster using the in-batch similarity score [56], calculated as the average pairwise cosine similarity of image features in a batch.

**Synthetic Datasets.** We compare our method on 2D Gaussian mixture benchmarks [64], training a diffusion model with the DDIM sampler [65] and fine-tuning hyperparameters (see Appendix C). Figure 4 shows that CADS covered the modes’ support but with a slight distribution shift, while c-VSG better adhered to the modes at the cost of lower diversity. PG and SPARKE both covered the modes well, with SPARKE producing fewer noisy samples due to its clustering-based structure.

**Comparison of Entropy Guidance Effects in Latent and Ambient Spaces.** We compared latent-space guidance (SPARKE) with CLIP-embedded ambient-space guidance from c-VSG [10] using modified GeoDE categories [66]. As qualitatively shown in Figure 2, CLIP-ambient led to visual artifacts and offered less improvement in output visual diversity, while latent guidance produced more semantically diverse outputs. We also provided quantitative results in Table 5, which show that latent guidance achieves higher diversity. Also, latent guidance requires significantly less GPU memory ( $\approx 20\text{GB}$  vs.  $\approx 35\text{GB}$ ), offering substantially better computational efficiency.

**Comparison of Prompt-Aware Diversity Guidance vs. Unconditional Guidance.** We generated images for 10,000 MS-COCO prompts [67] using Stable Diffusion 2.1, comparing latent RKE conditional guidance (prompt-aware) against unconditional RKE guidance (prompt-unaware). Figure 5 shows unconditional diversity guidance becomes less effective over time, while SPARKE’s prompt-aware approach remained effective throughout all 40k iterations. Table 1 confirms that prompt-aware

Table 2: Comparison of diversity and fidelity metrics across different models and methods.

Method	CLIPScore $\uparrow$	KD $\times 10^2 \downarrow$	Cond-Vendi Score $\uparrow$	Vendi Score $\uparrow$	In-batch Sim. $\times 10^2 \downarrow$
SDXL	31.17	66.37	27.54	309.54	81.67
<b>SDXL + SPARKE</b>	<b>30.47</b>	<b>62.62</b>	<b>31.17</b>	<b>313.80</b>	<b>77.75</b>
PixArt- $\Sigma$	31.01	65.37	26.45	307.36	83.84
<b>PixArt-<math>\Sigma</math> + SPARKE</b>	<b>30.66</b>	<b>63.86</b>	<b>32.14</b>	<b>322.25</b>	<b>78.81</b>

### Novelty Guidance Using SPARKE



Figure 6: Novelty guidance with SPARKE. Using the prompt "A young lady walking with an animal in the park," samples are generated with respect to a reference set of cat and dog images, resulting in more novel samples compared to the base model.

methods achieve higher diversity while preserving image quality and prompt alignment. Also, Table 2 provides the quantitative results for SD-XL and PixArt- $\Sigma$ .

**Novelty Guidance Using SPARKE.** We evaluated novelty guidance by using the SPARKE method to guide an SD-XL model with a reference dataset of cat and dog images. As shown in Figure 6, this approach successfully reduced the generation of cats and dogs, yielding more novel samples with respect to the reference dataset and resulting in a more balanced distribution of other animals.

**Computational Efficiency Comparison.** We evaluated SPARKE’s computational efficiency by comparing its runtime and peak memory with the baselines over 1000 text-to-image generations using a 50-step diffusion process on an NVIDIA RTX 4090. The results, as shown in Table 3, indicate that SPARKE requires significantly lower runtime and less memory (as it functions in the latent space).

Table 3: Comparison of runtime and GPU memory usage for different guidance methods.

Model + Guidance Method	Runtime per sample (s)	GPU Memory Peak (GB)
Stable Diffusion v1.5	$1.620 \pm 0.265$	$3.178 \pm 0.001$
SD-1.5 + SPARKE (Ours)	$1.752 \pm 0.287$	$3.230 \pm 0.025$
SD-1.5 + c-VSG	$3.079 \pm 0.296$	$8.665 \pm 0.116$
SD-1.5 + Particle Guide (Pixel)	$4.912 \pm 0.312$	$7.531 \pm 0.032$
SD-1.5 + Particle Guide (DINOv2)	$9.102 \pm 0.271$	$20.133 \pm 0.038$

## 6 Conclusion and Limitations

In this work, we proposed the prompt-aware SPARKE diversity guidance approach for prompt-based diffusion models. The SPARKE method aims to improve the diversity of output data conditioned on the prompt, so that the diversity guidance process takes into account the similarity of the prompts. We also proposed the application of RKE and Conditional-RKE scores in the latent space of LDMs to boost the scalability in the SPARKE method. Our numerical results of applying SPARKE to the stable-diffusion LDMs indicate the method’s qualitative and quantitative improvement of variety in generated samples. A limitation of our numerical evaluation is its primary focus on image generation diffusion models. Future exploration can extend SPARKE’s application to other modalities, such as video and text diffusion models. Furthermore, combining SPARKE’s entropy guidance with other diversity-enhancement techniques for diffusion models presents another interesting future direction.

## Acknowledgments

The work of Farzan Farnia is partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China, Project 14209920, and is partially supported by CUHK Direct Research Grants with CUHK Project No. 4055164 and 4937054. The work of Amin Gohari is supported by CUHK Direct Research Grants with CUHK Project No. 4055270. The work is also supported by a grant under 1+1+1 CUHK-CUHK(SZ)-GDSTC Joint Collaboration Fund. Finally, the authors would like to sincerely thank the anonymous reviewers for their insightful feedback and constructive suggestions.

## References

- [1] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. pmlr, 2015.
- [2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020.
- [3] 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*, 2020.
- [4] 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 (CVPR)*, pages 10684–10695, 2022.
- [5] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations*, 2024.
- [6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.
- [7] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research*, 23(47):1–33, 2022.
- [8] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.
- [9] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2837–2845, 2021.
- [10] Reyhane Askari Hemmat, Melissa Hall, Alicia Sun, Candace Ross, Michal Drozdzal, and Adriana Romero-Soriano. Improving geo-diversity of generated images with contextualized vendi score guidance. In *European Conference on Computer Vision*, pages 213–229. Springer, 2024.
- [11] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10844–10853, 2024.
- [12] Connor Shorten, Taghi M. Khoshgoftaar, and Borko Furht. A survey on image data augmentation for deep learning. *Journal of Big Data*, 8(1):1–48, 2021.
- [13] Dan Friedman and Adji Boussou Dieng. The vendi score: A diversity evaluation metric for machine learning. *arXiv preprint arXiv:2210.02410*, 2022.

[14] Seyedmorteza Sadat, Jakob Buhmann, Derek Bradley, Otmar Hilliges, and Romann M. Weber. CADS: Unleashing the diversity of diffusion models through condition-annealed sampling. In *The Twelfth International Conference on Learning Representations*, 2024.

[15] Mohammad Jalali, Azim Ospanov, Amin Gohari, and Farzan Farnia. Conditional Vendi Score: Prompt-aware diversity evaluation for text-guided generative ai models. In *Proceedings of The 29th International Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research. PMLR, 02–05 May 2026.

[16] Mohammad Jalali, Cheuk Ting Li, and Farzan Farnia. An information-theoretic evaluation of generative models in learning multi-modal distributions. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[17] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[18] Charles Stein. A bound for the error in the normal approximation to the distribution of a sum of dependent random variables. In *Proceedings of the sixth Berkeley symposium on mathematical statistics and probability, volume 2: Probability theory*, volume 6, pages 583–603. University of California Press, 1972.

[19] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

[20] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2426–2435, 2022.

[21] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Advances in neural information processing systems*, 34:11287–11302, 2021.

[22] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. *arXiv preprint arXiv:2211.13221*, 2022.

[23] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22563–22575, 2023.

[24] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

[25] Xihui Liu, Dong Huk Park, Samaneh Azadi, Gong Zhang, Arman Chopikyan, Yuxiao Hu, Humphrey Shi, Anna Rohrbach, and Trevor Darrell. More control for free! image synthesis with semantic diffusion guidance. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 289–299, 2023.

[26] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 4296–4304, 2024.

[27] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023.

[28] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022.

[29] Min Zhao, Fan Bao, Chongxuan Li, and Jun Zhu. Egsde: Unpaired image-to-image translation via energy-guided stochastic differential equations. *Advances in Neural Information Processing Systems*, 35:3609–3623, 2022.

[30] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023.

[31] Yutong He, Naoki Murata, Chieh-Hsin Lai, Yuhta Takida, Toshimitsu Uesaka, Dongjun Kim, Wei-Hsiang Liao, Yuki Mitsufuji, J Zico Kolter, Ruslan Salakhutdinov, et al. Manifold preserving guided diffusion. *arXiv preprint arXiv:2311.16424*, 2023.

[32] 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*, pages 843–852, 2023.

[33] Jiwen Yu, Yinhui Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. Freedom: Training-free energy-guided conditional diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23174–23184, 2023.

[34] Haotian Ye, Haowei Lin, Jiaqi Han, Minkai Xu, Sheng Liu, Yitao Liang, Jianzhu Ma, James Y Zou, and Stefano Ermon. Tfg: Unified training-free guidance for diffusion models. *Advances in Neural Information Processing Systems*, 37:22370–22417, 2024.

[35] Tuomas Kynkänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2019.

[36] Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In *International conference on machine learning*, pages 7176–7185. PMLR, 2020.

[37] Azim Ospanov, Jingwei Zhang, Mohammad Jalali, Xuenan Cao, Andrej Bogdanov, and Farzan Farnia. Towards a scalable reference-free evaluation of generative models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[38] Azim Ospanov and Farzan Farnia. Do vendi scores converge with finite samples? truncated vendi score for finite-sample convergence guarantees. In *The 41st Conference on Uncertainty in Artificial Intelligence*, 2025.

[39] Azim Ospanov, Mohammad Jalali, and Farzan Farnia. Scendi score: Prompt-aware diversity evaluation via schur complement of clip embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16927–16937, October 2025.

[40] Parham Rezaei, Farzan Farnia, and Cheuk Ting Li. Be more diverse than the most diverse: Optimal mixtures of generative models via mixture-UCB bandit algorithms. In *The Thirteenth International Conference on Learning Representations*, 2025.

[41] Xiaoyan Hu, Ho-Fung Leung, and Farzan Farnia. A multi-armed bandit approach to online selection and evaluation of generative models. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pages 1864–1872. PMLR, 2025.

[42] Xiaoyan Hu, Ho-Fung Leung, and Farzan Farnia. PAK-UCB contextual bandit: An online learning approach to prompt-aware selection of generative models and LLMs. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 24447–24481. PMLR, 2025.

[43] Zixiao Wang, Farnia Farzan, Zhenghao Lin, Yunheng Shen, and Bei Yu. On the distributed evaluation of generative models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 7644–7653, 2025.

[44] Marco Jiralerpong, Joey Bose, Ian Gemp, Chongli Qin, Yoram Bachrach, and Gauthier Gidel. Feature likelihood score: Evaluating the generalization of generative models using samples. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[45] Jingwei Zhang, Cheuk Ting Li, and Farzan Farnia. An interpretable evaluation of entropy-based novelty of generative models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 59148–59172. PMLR, 21–27 Jul 2024.

[46] Jingwei Zhang, Mohammad Jalali, Cheuk Ting Li, and Farzan Farnia. Unveiling differences in generative models: A scalable differential clustering approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[47] Mohammad Jalali, Bahar Dibaei Nia, and Farzan Farnia. Towards an explainable comparison and alignment of feature embeddings. In *Forty-second International Conference on Machine-Learning*, 2025.

[48] Shizhan Gong, Yankai Jiang, Qi Dou, and Farzan Farnia. Kernel-based unsupervised embedding alignment for enhanced visual representation in vision-language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 19912–19931. PMLR, 13–19 Jul 2025.

[49] Shizhan Gong, Haoyu Lei, Qi Dou, and Farzan Farnia. Boosting the visual interpretability of clip via adversarial fine-tuning. In *International Conference on Learning Representations*, 2025.

[50] Youqi Wu, Jingwei Zhang, and Farzan Farnia. When kernels multiply, clusters unify: Fusing embeddings with the kronecker product. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[51] Seyedmorteza Sadat, Manuel Kansy, Otmar Hilliges, and Romann M. Weber. No training, no problem: Rethinking classifier-free guidance for diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025.

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

[53] Matina Mahdizadeh Sani, Nima Jamali, Mohammad Jalali, and Farzan Farnia. Training-free distribution adaptation for diffusion models via maximum mean discrepancy guidance. *arXiv preprint arXiv:2601.08379*, 2026.

[54] Vikash Sehwag, Caner Hazirbas, Albert Gordo, Firat Ozgenel, and Cristian Canton Ferrer. Generating high fidelity data from low-density regions using diffusion models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11482–11491, 2022.

[55] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10844–10853, June 2024.

[56] Gabriele Corso, Yilun Xu, Valentin De Bortoli, Regina Barzilay, and Tommi S. Jaakkola. Particle Guidance: non-I.I.D. diverse sampling with diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024.

[57] Jack Lu, Ryan Teehan, and Mengye Ren. ProCreate, Don’t Reproduce! propulsive energy diffusion for creative generation. In *Computer Vision - ECCV 2024 - 18th European Conference, Milano, Italy, September 29 - October 27, 2024*, 2024.

[58] Michael Kirchhof, James Thornton, Pierre Ablin, Louis Béthune, Eugene Ndiaye, and Marco Cuturi. Sparse repellency for shielded generation in text-to-image diffusion models. *arXiv preprint arXiv:2410.06025*, 2024.

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

[60] 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 (CVPR)*, pages 10684–10695, June 2022.

[61] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. PixArt- $\Sigma$ : Weak-to-strong training of diffusion transformer for 4k text-to-image generation. In *Computer Vision – ECCV 2024*, pages 74–91, Cham, 2025.

[62] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528. Association for Computational Linguistics, November 2021.

[63] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations*, 2018.

[64] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017.

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

[66] Vikram V. Ramaswamy, Sing Yu Lin, Dora Zhao, Aaron B. Adcock, Laurens van der Maaten, Deepti Ghadiyaram, and Olga Russakovsky. Geode: a geographically diverse evaluation dataset for object recognition. In *NeurIPS Datasets and Benchmarks*, 2023.

[67] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham, 2014. Springer International Publishing.

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

[69] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

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

[71] Herbert E Robbins. An empirical bayes approach to statistics. In *Breakthroughs in Statistics: Foundations and basic theory*, pages 388–394. Springer, 1992.

[72] Yifei Shen, Xinyang Jiang, Yifan Yang, Yezhen Wang, Dongqi Han, and Dongsheng Li. Understanding and improving training-free loss-based diffusion guidance. *Advances in Neural Information Processing Systems*, 37:108974–109002, 2024.

[73] William Gaviria Rojas, Sudnya Diamos, Keertan Kini, David Kanter, Vijay Janapa Reddi, and Cody Coleman. The Dollar Street Dataset: Images representing the geographic and socioeconomic diversity of the world. In *Advances in Neural Information Processing Systems*, volume 35, pages 12979–12990. Curran Associates, Inc., 2022.

[74] OpenAI. GPT-4o mini: advancing cost-efficient intelligence, July 2024.

## A Preliminaries on Diffusion Models

### A.1 Denoising Diffusion Probabilistic Models (DDPMs)

Denoising diffusion probabilistic models (DDPMs) [1, 2] define a generative process by reversing a fixed Markovian forward diffusion that progressively adds Gaussian noise to data. Given a data sample  $\mathbf{x}_0 \sim p_0(\mathbf{x}_0)$ , a time step  $t \in [T] \triangleq \{1, \dots, T\}$ , the forward process gradually adds noise to construct a noisy data point  $\mathbf{x}_t = \sqrt{\alpha_t} \mathbf{x}_0 + \sqrt{1 - \alpha_t} \epsilon_t$ , where  $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  is standard Gaussian noise and  $\alpha_t \in [0, 1]$  monotonically decreases with time step  $t$  to control the noise level. The diffusion model  $\epsilon_\theta : \mathcal{X} \times [T] \mapsto \mathcal{X}$  is trained to predict the noise  $\epsilon_t$  at each time step  $t$ , also learn the score of  $p_t(\mathbf{x}_t)$  [17, 3]:

$$\min_{\theta} \mathbb{E}_{\mathbf{x}_t, \epsilon_t, t} \left[ \|\epsilon_\theta(\mathbf{x}_t, t) - \epsilon_t\|_2^2 \right] = \min_{\theta} \mathbb{E}_{\mathbf{x}_t, \epsilon_t, t} \left[ \|\epsilon_\theta(\mathbf{x}_t, t) + \sqrt{1 - \alpha_t} \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)\|_2^2 \right], \quad (12)$$

The reverse process obtained by  $\mathbf{x}_{t-1} \sim p_{t-1|t}(\mathbf{x}_{t-1}|\mathbf{x}_t)$  is not directly computable in practice, multiple efficient reverse samplers are proposed [65, 68, 69]. In the commonly-used DDIM [65] sampler, we sample  $\mathbf{x}_{t-1}$  by:

$$\mathbf{x}_{t-1} = \sqrt{\alpha_{t-1}} \tilde{\mathbf{x}}_{0|t} + \sqrt{1 - \alpha_{t-1} - \sigma_t^2} \frac{\mathbf{x}_t - \sqrt{\alpha_t} \tilde{\mathbf{x}}_{0|t}}{\sqrt{1 - \alpha_t}} + \sigma_t \epsilon_t, \quad (13)$$

where  $\sigma_t$  is the DDIM parameter, and the clean sample  $\tilde{\mathbf{x}}_{0|t}$  given  $\mathbf{x}_t$  is estimated according to Tweedie's Formula [70, 71]:

$$\tilde{\mathbf{x}}_{0|t} = \frac{\mathbf{x}_t - \sqrt{1 - \alpha_t} \epsilon_\theta(\mathbf{x}_t, t)}{\sqrt{\alpha_t}}. \quad (14)$$

### A.2 Conditional Generation via Guidance.

In conditional generation tasks like text-to-image synthesis, diffusion models learn to approximate the conditional distribution  $p(\mathbf{x}_0|\mathbf{y})$  given conditions  $\mathbf{y}$  (e.g., text prompts). From the view of score functions [17, 3], we denote the conditional score as:

$$\underbrace{\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{y})}_{\text{Conditional Score}} = \underbrace{\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)}_{\text{Unconditional Score}} + \underbrace{\nabla_{\mathbf{x}_t} \log p_t(\mathbf{y}|\mathbf{x}_t)}_{\text{Guidance Score}}, \quad (15)$$

The strategies of conditional generation via guidance can be roughly divided into two categories: Training-based methods and training-free methods.

**Training-based Guidance.** Training-based guidance methods include several strategies. One approach, Classifier-Guidance, initially proposed in [3, 6], requires training an additional time-dependent classifier to estimate the guidance score  $f(\mathbf{x}_t, t) \triangleq \mathbb{E}_{\mathbf{x}_0 \sim p_{0|t}(\cdot|\mathbf{x}_t)} f(\mathbf{x}_0) \approx f(\mathbf{x}_0)$ . Alternatively, other training-based techniques involve few-shot fine-tuning of base models or the use of adapters [4, 30] to achieve conditional control.

A distinct yet training-based approach is Classifier-Free Guidance (CFG), introduced by [52]. Unlike methods requiring a separate classifier, CFG integrates the condition  $\mathbf{y}$  as a direct input to the conditional denoising network  $\epsilon_\theta(\mathbf{x}_t, t, \mathbf{y})$ . It is enabled by a joint training procedure where the model also learns to make unconditional predictions by randomly dropping the condition  $\mathbf{y}$  with a specific probability during each training iteration. During inference, CFG estimates the conditional noise prediction as follows:

$$\hat{\epsilon}_\theta(\mathbf{x}_t, t, \mathbf{y}) = (1 + w) \cdot \epsilon_\theta(\mathbf{x}_t, t, \mathbf{y}) - w \cdot \epsilon_\theta(\mathbf{x}_t, t), \quad (16)$$

where the guidance scale  $w > 0$  adjusts the strength of conditional guidance. These various training-based guidance methods have demonstrated considerable effectiveness with the availability of training resources.

**Training-free Guidance.** An alternative category for conditional guidance is training-free guidance. Instead of requiring additional training, these methods directly introduce a time-independent conditional predictor  $f$  on the estimated clean data  $f(\tilde{\mathbf{x}}_{0|t}, \mathbf{y})$ , which can be a classifier, loss function, or energy function quantifying the alignment of a generated sample with the target condition [32, 33, 34, 72]. Then the estimation of the guidance score via training-free approaches is:

$$\nabla_{\mathbf{x}_t} \log p_t(\mathbf{y}|\mathbf{x}_t) := -\nabla_{\mathbf{x}_t} f(\tilde{\mathbf{x}}_{0|t}, \mathbf{y}), \quad (17)$$

where the clean sample for the predictor is estimated by Tweedie's Formula [70, 71]:

$$\tilde{\mathbf{x}}_{0|t} = \frac{\mathbf{x}_t - \sqrt{1 - \alpha_t} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{\alpha_t}}. \quad (18)$$

## B Proofs

### B.1 Proof of Proposition 1

*Proof.* Let  $K_Z$  be the kernel matrix with entries  $(K_Z)_{ij} = k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})$ . The RKE loss is given by:

$$\mathcal{L}_{\text{RKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}) = \frac{1}{\|\tilde{K}_Z\|_F^2} = \frac{\text{Tr}(K_Z)^2}{\sum_{i=1}^n \sum_{j=1}^n k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2}, \quad (19)$$

The Inverse-RKE function  $\mathcal{L}_{\text{IRKE}}$  is defined by:

$$\mathcal{L}_{\text{IRKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}) = \frac{1}{\mathcal{L}_{\text{RKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)})} = \frac{\sum_{i=1}^n \sum_{j=1}^n k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2}{\text{Tr}(K_Z)^2}, \quad (20)$$

Given that the kernel function  $k$  is normalized, i.e.,  $k(\mathbf{z}, \mathbf{z}) = 1$  for all  $\mathbf{z} \in Z$ , the trace of the kernel matrix  $K_Z$  is:

$$\text{Tr}(K_Z) = \sum_{i=1}^n k(\mathbf{z}^{(i)}, \mathbf{z}^{(i)}) = \sum_{i=1}^n 1 = n,$$

Substituting this into the expression for  $\mathcal{L}_{\text{IRKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)})$  in Eq. (20):

$$\mathcal{L}_{\text{IRKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}) = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2, \quad (21)$$

Now, we want to compute the gradient of  $\mathcal{L}_{\text{IRKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)})$  with respect to  $\mathbf{z}^{(n)}$ :

$$\nabla_{\mathbf{z}^{(n)}} \mathcal{L}_{\text{IRKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}) = \nabla_{\mathbf{z}^{(n)}} \left( \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 \right), \quad (22)$$

Since  $1/n^2$  is a constant with respect to  $\mathbf{z}^{(n)}$ :

$$\nabla_{\mathbf{z}^{(n)}} \mathcal{L}_{\text{IRKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}) = \frac{1}{n^2} \nabla_{\mathbf{z}^{(n)}} \left( \sum_{i=1}^n \sum_{j=1}^n k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 \right), \quad (23)$$

Let  $S = \sum_{i=1}^n \sum_{j=1}^n k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2$ . We need to find  $\nabla_{\mathbf{z}^{(n)}} S$ . The terms in the sum  $S$  that involve  $\mathbf{z}^{(n)}$  are those where  $i = n$  or  $j = n$  (or both). We can split the sum into:

$$\begin{aligned} S &= \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 \quad (\text{Terms not involving } \mathbf{z}^{(n)}) \\ &\quad + \sum_{i=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)})^2 \quad (j = n, i \neq n) \\ &\quad + \sum_{j=1}^{n-1} k(\mathbf{z}^{(n)}, \mathbf{z}^{(j)})^2 \quad (i = n, j \neq n) \\ &\quad + k(\mathbf{z}^{(n)}, \mathbf{z}^{(n)})^2, \quad (i = n, j = n) \end{aligned}$$

Using the symmetry of the kernel,  $k(\mathbf{z}^{(n)}, \mathbf{z}^{(j)}) = k(\mathbf{z}^{(j)}, \mathbf{z}^{(n)})$ , the second and third sums are identical:

$$S = \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 + 2 \sum_{i=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)})^2 + k(\mathbf{z}^{(n)}, \mathbf{z}^{(n)})^2, \quad (24)$$

Now, we compute the gradient of  $S$  with respect to  $\mathbf{z}^{(n)}$ :

$$\nabla_{\mathbf{z}^{(n)}} S = \nabla_{\mathbf{z}^{(n)}} \left( \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 \right) + \nabla_{\mathbf{z}^{(n)}} \left( 2 \sum_{i=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)})^2 \right) + \nabla_{\mathbf{z}^{(n)}} \left( k(\mathbf{z}^{(n)}, \mathbf{z}^{(n)})^2 \right), \quad (25)$$

The first term is zero because it does not depend on  $\mathbf{z}^{(n)}$ . For the second term:

$$\begin{aligned} \nabla_{\mathbf{z}^{(n)}} \left( 2 \sum_{i=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)})^2 \right) &= 2 \sum_{i=1}^{n-1} \nabla_{\mathbf{z}^{(n)}} \left( k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)})^2 \right) \\ &= 2 \sum_{i=1}^{n-1} 2k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \nabla_{\mathbf{z}^{(n)}} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \\ &= 4 \sum_{i=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \nabla_{\mathbf{z}^{(n)}} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}), \end{aligned} \quad (26)$$

For the third term, since  $k(\mathbf{z}, \mathbf{z}) = 1$  (normalized kernel), this term is constant:

$$\nabla_{\mathbf{z}^{(n)}} \left( k(\mathbf{z}^{(n)}, \mathbf{z}^{(n)})^2 \right) = \nabla_{\mathbf{z}^{(n)}} (1^2) = \nabla_{\mathbf{z}^{(n)}} (1) = \mathbf{0}, \quad (27)$$

According to Eq. (26) and Eq. (27), we have the gradient for  $S$ :

$$\nabla_{\mathbf{z}^{(n)}} S = 4 \sum_{i=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \nabla_{\mathbf{z}^{(n)}} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}), \quad (28)$$

Substituting this back into the gradient of  $\mathcal{L}_{\text{IRKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)})$  in Eq. (23):

$$\begin{aligned} \nabla_{\mathbf{z}^{(n)}} \mathcal{L}_{\text{IRKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}) &= \frac{1}{n^2} \left( 4 \sum_{i=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \nabla_{\mathbf{z}^{(n)}} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \right) \\ &= \frac{4}{n^2} \sum_{i=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \nabla_{\mathbf{z}^{(n)}} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}), \end{aligned} \quad (29)$$

Since  $4/n^2$  is a positive constant (for  $n \geq 1$ ), we have the following:

$$\nabla_{\mathbf{z}^{(n)}} \mathcal{L}_{\text{IRKE}} \propto \sum_{i=1}^{n-1} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \nabla_{\mathbf{z}^{(n)}} k(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}). \quad (30)$$

This proves the form of the gradient.

Regarding the monotonic relationship  $\mathcal{L}_{\text{IRKE}} = 1/\mathcal{L}_{\text{RKE}}$ ,  $\mathcal{L}_{\text{RKE}}$  is always positive ( $\text{Tr}(K_Z)^2 = n^2 > 0$  and the denominator  $\sum k_{ij}^2 \geq 0$ ; for it to be non-zero, not all kernel values can be zero), then  $\mathcal{L}_{\text{RKE}}^2 > 0$ . Therefore,  $\frac{d\mathcal{L}_{\text{IRKE}}}{d\mathcal{L}_{\text{RKE}}} = -\frac{1}{\mathcal{L}_{\text{RKE}}^2} < 0$ , which means that  $\mathcal{L}_{\text{IRKE}}$  is a strictly monotonically decreasing function of  $\mathcal{L}_{\text{RKE}}$ . This completes the proof of Proposition 1.  $\square$

## B.2 Proof of Proposition 2

*Proof.* Let  $K_Z$  be the kernel matrix for the data  $Z$  with entries  $(K_Z)_{ij} = k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})$ . Let  $K_Y$  be the kernel matrix for the conditions  $Y$  with entries  $(K_Y)_{ij} = k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)})$ . The normalized kernel matrices are  $\tilde{K}_Z = K_Z/\text{Tr}(K_Z)$  and  $\tilde{K}_Y = K_Y/\text{Tr}(K_Y)$ . Given that the kernel functions  $k_Z$  and  $k_Y$  are normalized, their traces are:

$$\begin{aligned} \text{Tr}(K_Z) &= \sum_{i=1}^n k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(i)}) = \sum_{i=1}^n 1 = n \\ \text{Tr}(K_Y) &= \sum_{i=1}^n k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(i)}) = \sum_{i=1}^n 1 = n \end{aligned}$$

So  $\tilde{K}_Z = K_Z/n$  and  $\tilde{K}_Y = K_Y/n$ .

The Conditional Inverse-RKE function is defined as:

$$\mathcal{L}_{\text{Cond-IRKE}}(\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n)}; \mathbf{y}^{(1)}, \dots, \mathbf{y}^{(n)}) = \|\tilde{K}_Z \odot \tilde{K}_Y\|_F^2, \quad (31)$$

where  $\odot$  denotes the Hadamard (element-wise) product. The entries of  $\tilde{K}_Z \odot \tilde{K}_Y$  are:

$$\begin{aligned} (\tilde{K}_Z \odot \tilde{K}_Y)_{ij} &= (\tilde{K}_Z)_{ij} (\tilde{K}_Y)_{ij} \\ &= \frac{1}{n} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(j)}) \cdot \frac{1}{n} k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)}) \\ &= \frac{1}{n^2} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(j)}) k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)}), \end{aligned} \quad (32)$$

The squared Frobenius norm is the sum of the squares of its elements, then we have the following according to Eq. (31):

$$\begin{aligned} \mathcal{L}_{\text{Cond-IRKE}}(Z; Y) &= \sum_{i=1}^n \sum_{j=1}^n \left( \frac{1}{n^2} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(j)}) k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)}) \right)^2 \\ &= \frac{1}{n^4} \sum_{i=1}^n \sum_{j=1}^n k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)})^2, \end{aligned} \quad (33)$$

Now, we want to compute the gradient of  $\mathcal{L}_{\text{Cond-IRKE}}(Z; Y)$  with respect to  $\mathbf{z}^{(n)}$  according to Eq. (33):

$$\nabla_{\mathbf{z}^{(n)}} \mathcal{L}_{\text{Cond-IRKE}}(Z; Y) = \frac{1}{n^4} \nabla_{\mathbf{z}^{(n)}} \left( \sum_{i=1}^n \sum_{j=1}^n k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)})^2 \right), \quad (34)$$

Let  $S_C = \sum_{i=1}^n \sum_{j=1}^n k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)})^2$ . We need to find  $\nabla_{\mathbf{z}^{(n)}} S_C$ . The terms in the sum  $S_C$  that involve  $\mathbf{z}^{(n)}$  (through  $k_Z$ ) are those where  $i = n$  or  $j = n$  (or both), and the terms  $k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)})$  do not depend on  $\mathbf{z}^{(n)}$ . We can split the sum as follows:

$$\begin{aligned} S_C &= \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)})^2 \quad (\text{terms not involving } \mathbf{z}^{(n)}) \\ &\quad + \sum_{i=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)})^2 k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 \quad (j = n, i \neq n) \\ &\quad + \sum_{j=1}^{n-1} k_Z(\mathbf{z}^{(n)}, \mathbf{z}^{(j)})^2 k_Y(\mathbf{y}^{(n)}, \mathbf{y}^{(j)})^2 \quad (i = n, j \neq n) \\ &\quad + k_Z(\mathbf{z}^{(n)}, \mathbf{z}^{(n)})^2 k_Y(\mathbf{y}^{(n)}, \mathbf{y}^{(n)})^2, \quad (i = n, j = n) \end{aligned} \quad (35)$$

Using the symmetry of the kernels,  $k_Z(\mathbf{z}^{(n)}, \mathbf{z}^{(j)}) = k_Z(\mathbf{z}^{(j)}, \mathbf{z}^{(n)})$  and  $k_Y(\mathbf{y}^{(n)}, \mathbf{y}^{(j)}) = k_Y(\mathbf{y}^{(j)}, \mathbf{y}^{(n)})$ , the second and third sums are identical. Then we have:

$$\begin{aligned} S_C &= \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)})^2 \\ &\quad + 2 \sum_{i=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)})^2 k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 + k_Z(\mathbf{z}^{(n)}, \mathbf{z}^{(n)})^2 k_Y(\mathbf{y}^{(n)}, \mathbf{y}^{(n)})^2, \end{aligned} \quad (36)$$

Now, we compute the gradient of  $S_C$  with respect to  $\mathbf{z}^{(n)}$ :

$$\begin{aligned} \nabla_{\mathbf{z}^{(n)}} S_C &= \nabla_{\mathbf{z}^{(n)}} \left( \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})^2 k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(j)})^2 \right) \\ &\quad + \nabla_{\mathbf{z}^{(n)}} \left( 2 \sum_{i=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)})^2 k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 \right) \\ &\quad + \nabla_{\mathbf{z}^{(n)}} \left( k_Z(\mathbf{z}^{(n)}, \mathbf{z}^{(n)})^2 k_Y(\mathbf{y}^{(n)}, \mathbf{y}^{(n)})^2 \right), \end{aligned} \quad (37)$$

The first term is zero because it does not depend on  $\mathbf{z}^{(n)}$ . For the second term:

$$\begin{aligned}
\nabla_{\mathbf{z}^{(n)}} \left( 2 \sum_{i=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)})^2 k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 \right) &= 2 \sum_{i=1}^{n-1} k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 \nabla_{\mathbf{z}^{(n)}} \left( k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)})^2 \right) \\
&= 2 \sum_{i=1}^{n-1} k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 \left( 2k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \nabla_{\mathbf{z}^{(n)}} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \right) \\
&= 4 \sum_{i=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 \nabla_{\mathbf{z}^{(n)}} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}),
\end{aligned} \tag{38}$$

For the third term, since  $k_Z(\mathbf{z}^{(n)}, \mathbf{z}^{(n)}) = 1$  and  $k_Y(\mathbf{y}^{(n)}, \mathbf{y}^{(n)}) = 1$  (normalized kernels), this term is constant:

$$\nabla_{\mathbf{z}^{(n)}} \left( k_Z(\mathbf{z}^{(n)}, \mathbf{z}^{(n)})^2 k_Y(\mathbf{y}^{(n)}, \mathbf{y}^{(n)})^2 \right) = \nabla_{\mathbf{z}^{(n)}} (1^2 \cdot 1^2) = \nabla_{\mathbf{z}^{(n)}} (1) = \mathbf{0}, \tag{39}$$

So we have:

$$\nabla_{\mathbf{z}^{(n)}} S_C = 4 \sum_{i=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 \nabla_{\mathbf{z}^{(n)}} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}), \tag{40}$$

Substituting this back into the gradient of  $\mathcal{L}_{\text{Cond-IRKE}}(Z; Y)$  in Eq. (33):

$$\begin{aligned}
\nabla_{\mathbf{z}^{(n)}} \mathcal{L}_{\text{Cond-IRKE}}(Z; Y) &= \frac{1}{n^4} \left( 4 \sum_{i=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 \nabla_{\mathbf{z}^{(n)}} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) \right) \\
&= \frac{4}{n^4} \sum_{i=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 \nabla_{\mathbf{z}^{(n)}} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}),
\end{aligned} \tag{41}$$

Since  $4/n^4$  is a positive constant (for  $n \geq 1$ ), we finally get:

$$\nabla_{\mathbf{z}^{(n)}} \mathcal{L}_{\text{Cond-IRKE}} \propto \sum_{i=1}^{n-1} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}) k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(n)})^2 \nabla_{\mathbf{z}^{(n)}} k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(n)}). \tag{42}$$

This proves the form of the gradient.

Regarding the monotonic relationship: From Eq. (9), we have:

$$\mathcal{L}_{\text{Cond-RKE}}(Z; Y) = \frac{\|\tilde{K}_Y\|_F^2}{\left\| \frac{K_Y \odot K_Z}{n} \right\|_F^2}, \tag{43}$$

We have shown that  $\text{Tr}(K_Y \odot K_Z) = n$ , since  $(K_Y \odot K_Z)_{ii} = k_Y(\mathbf{y}^{(i)}, \mathbf{y}^{(i)}) k_Z(\mathbf{z}^{(i)}, \mathbf{z}^{(i)}) = 1 \cdot 1 = 1$ . Also,  $\|\tilde{K}_Y\|_F^2 = \|K_Y/n\|_F^2 = \|K_Y\|_F^2/n^2$ . Substituting these into the formulation for  $\mathcal{L}_{\text{Cond-RKE}}$  in Eq. (43):

$$\mathcal{L}_{\text{Cond-RKE}}(Z; Y) = \frac{\frac{1}{n^2} \|K_Y\|_F^2}{\left\| \frac{K_Y \odot K_Z}{n} \right\|_F^2} = \frac{\frac{1}{n^2} \|K_Y\|_F^2}{\frac{1}{n^2} \|K_Y \odot K_Z\|_F^2} = \frac{\|K_Y\|_F^2}{\|K_Y \odot K_Z\|_F^2}, \tag{44}$$

From the definition of  $\mathcal{L}_{\text{Cond-IRKE}}(Z; Y)$  in Eq. (31), we have:

$$\mathcal{L}_{\text{Cond-IRKE}}(Z; Y) = \|\tilde{K}_Z \odot \tilde{K}_Y\|_F^2 = \left\| \frac{K_Z}{n} \odot \frac{K_Y}{n} \right\|_F^2 = \frac{1}{n^4} \|K_Z \odot K_Y\|_F^2, \tag{45}$$

Let  $C_Y = \|K_Y\|_F^2$ . This term depends only on the conditions  $Y$  and is constant with respect to  $\mathbf{z}^{(n)}$ .  $C_Y$  is always positive since the kernel function  $K_Y$  is positive semi-definite and not all kernel values can be 0. From the formulation for  $\mathcal{L}_{\text{Cond-IRKE}}$  in Eq. (31), we have  $\|K_Z \odot K_Y\|_F^2 = n^4 \mathcal{L}_{\text{Cond-IRKE}}(Z; Y)$ . Substituting this into the formulation for  $\mathcal{L}_{\text{Cond-RKE}}$ :

$$\mathcal{L}_{\text{Cond-RKE}}(Z; Y) = \frac{C_Y}{n^4 \mathcal{L}_{\text{Cond-IRKE}}(Z; Y)}. \tag{46}$$

Since  $C_Y/n^4$  is a positive constant, then  $\mathcal{L}_{\text{Cond-RKE}}(Z; Y)$  is a strictly monotonically decreasing function of  $\mathcal{L}_{\text{Cond-IRKE}}(Z; Y)$ . Thus, they change monotonically with respect to each other. This completes the proof of Proposition 2.  $\square$

### B.3 Interpretation of SPARKE

Our method, SPARKE, can be theoretically interpreted as an interacting particle system where sequential samples  $\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \dots, \mathbf{z}^{(n)}$  evolve along a gradient flow to minimize an energy potential, thus improving prompt-aware diversity. This process is described by the following sequential stochastic differential equation:

$$d\mathbf{z}^{(n)} = \left[ -\mathbf{f}(\mathbf{z}^{(n)}, t') + \frac{g^2(t')}{2} \left( \nabla_{\mathbf{z}^{(n)}} \log p_{t'}(\mathbf{z}^{(n)}) - \eta \nabla_{\mathbf{z}^{(n)}} L_{\text{Cond-IRKE}}(\mathbf{z}_1, \dots, \mathbf{z}_n) \right) \right] dt + g(t') d\mathbf{w}, \quad (47)$$

The core insight of SPARKE is to employ the Conditional Rényi Kernel Entropy (Conditional RKE) as a differentiable objective for promoting diversity. Following [16], the order-2 RKE score serves as a differentiable objective for counting the modes of a particle distribution. Furthermore, the Conditional RKE score in [15] quantifies the internal diversity conditioning on the prompt categories. SPARKE defines a gradient flow that optimizes the Conditional RKE score, and sequentially drives samples to cover distinct modes, while the conditional mechanism ensures the prompt-aware diversity guidance potential.

## C Implementation Details and Hyperparameters

In the kernel-based guidance experiments of SPARKE and the baselines with kernel entropy diversity scores, we considered a Gaussian kernel, which consistently led to higher output scores in comparison to the other standard cosine similarity kernel (see Section D for ablation studies). We used the same Gaussian kernel bandwidth  $\sigma$  in the RKE and Vendi experiments, and the bandwidth parameter choice matches the selected value in [15, 13]. The numerical experiments were conducted on 4×NVIDIA GeForce RTX 4090 GPUs, each of which has 22.5 GB of memory.

### C.1 Experimental Configuration for Table 1

We evaluated the methods listed in Table 1 in the following setting. We used Stable Diffusion 2.1 with a resolution of  $1024 \times 1024$ , a fixed classifier-free guidance scale of  $W_{\text{CFG}} = 7.5$ , and 50 inference steps using the DPM solver. We used the first 10,000 prompts of the MS-COCO 2014 validation set and fixed the generation seed to be able to compare the effect of the methods. For the methods, we used the following configuration to generate the results reported in Table 1.

The hyperparameter tuning was performed by performing cross-validation on the in-batch similarity score, selecting the hyperparameter values that optimized this alignment-based metric. Note that the in-batch similarity score accounts for both text-image consistency and inter-sample diversity as discussed in [56]. Due to GPU memory requirements (as mentioned in Section B.2 in [56]), we were unable to evaluate the Particle Guide baseline [56] on SD v2.1. Following the provided implementation of this baseline for SD v1.5, we conducted this baseline’s experiments only on SD v1.5.

**Stable Diffusion.** We used the standard CFG guidance and DPM solver with no additional diversity-related guidance.

**CADS.** Following the discussion in Table 13 of [14], we set the threshold parameters as  $\tau_1 = 0.6$  and  $\tau_2 = 0.9$ ,  $\psi = 1$ , and used a noise scale of 0.25.

**c-VSG.** We note that the reference [10] considered GeoDE [66] and DollarStreet [73] datasets, in which multiple samples exist per input prompt. On the other hand, in our experiments, we considered the standard MSCOCO prompt set where for each prompt corresponds we access a single image, making the contextualized Vendi guidance baseline in [10] not directly applicable. Therefore, we simulated the non-contextualized version of VSG. For selecting the Vendui score guidance scale, we performed validation over the set  $\{0, 0.04, 0.05, 0.06, 0.07\}$ , following the procedure in [10]. A guidance frequency of 5 was used, consistent with the original implementation. To maintain stable gradient computation for the Vendui score, we implemented a sliding window of 150 most recently generated samples, as gradient calculations became numerically unstable for some steps beyond this threshold.

**latent RKE Guidance.** We used a Gaussian kernel with bandwidth  $\sigma_{img} = 0.8$  and used  $\eta = 0.03$  as the weight of RKE guidance. To balance the effects of the diversity guidance in sample generation,

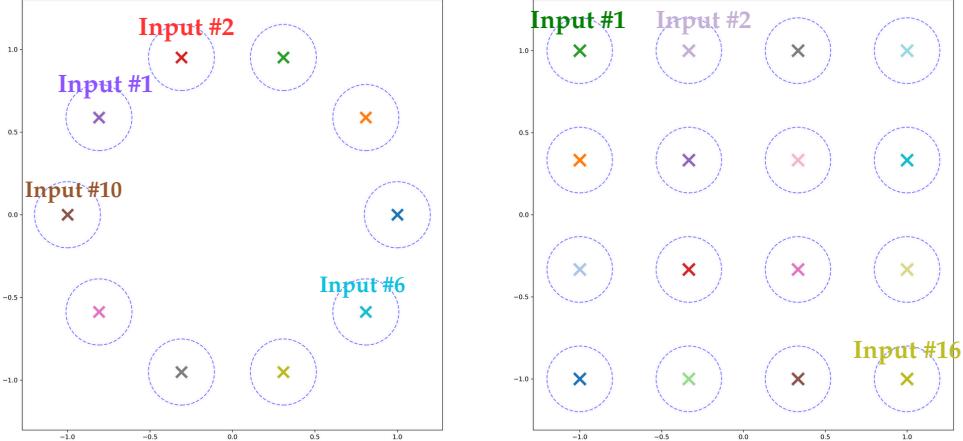


Figure 7: Illustration of input-based Gaussian output used for the conditional diffusion model in Figure 4. As displayed, the output 2D Gaussian vector is generated centered around the mean vector specified by the input prompt number.

the RKE guidance update was applied every 10 reverse-diffusion steps in the diffusion process, which is similar to the implementation of Vendi score guidance in [10]. Unlike VSG, which requires a sample window limit, RKE guidance operates without window size constraints, leveraging the complete history of generated samples for gradient computation.

**SPARKE (latent Conditional RKE Guidance).** We considered the same Gaussian kernel for the image generation with bandwidth  $\sigma_{img} = 0.8$  and used bandwidth parameter  $\sigma_{text} = 0.3$  for the text kernel. The guidance hyperparameter was set to  $\eta = 0.03$ , as in RKE guidance. Similar to the RKE and Vendi guidance, the SPARKE diversity guidance was applied every 10 reverse-diffusion steps. Unlike VSG but similar to RKE, SPARKE uses the complete history of generated samples for guidance.

## C.2 Experiment Settings in the results of Table 4 and Figure 4

We conducted additional experiments using Stable Diffusion v1.5 with a resolution of  $512 \times 512$ . We created a prompt set by performing K-Means clustering on the MSCOCO 2014 validation prompts. Specifically, we clustered the MS-COCO prompt dataset via the spectral clustering in the CLIP embedding space, into 40 groups and randomly drawn 50 prompts from each cluster, resulting in a total of 2,000 prompts. To evaluate the performance of our method with the baselines with different seeds, we generated five samples per prompt using seeds 0, 1, 2, 3, and 4, yielding a total of 10,000 images for each method. The complete list of prompts is included in the supplementary materials.

For all the tested methods, we considered the settings described in Section C.1, including a classifier-free guidance scale of  $W_{CFG} = 7.5$ , 50 reverse-diffusion steps, and the DPM solver. For the Particle Guide baseline, we used the following settings:

**Particle Guide.** We applied the original implementation provided by the repository of [56]. Since the method operates on repeated generations of the same prompt, we generated five samples per prompt using seeds 0 through 4. The coefficient parameter was set to 30, following the original implementation.

**Gaussian Mixtures.** We used conditional diffusion models in Figure 4. In Figure 7, we illustrate how the centers of clusters were used as inputs to the diffusion model to simulate conditional generation.

## D Additional Numerical Results

In this section, we provide additional numerical results for the SPARKE guidance method.

**Ablation studies on the kernel function choice in SPARKE.** In Figure 9, we compared the outputs of applying SPARKE with the cosine similarity kernel and the Gaussian kernel in the latent space of



Figure 8: Qualitative results for various diversity guidance scales from  $\eta = 0$  (no guidance) to  $\eta = 0.09$ . The results are generated using Stable Diffusion XL with the same Gaussian kernel.

SDXL. As shown in the figure, the Gaussian kernel seems able to exhibit more diverse features, and therefore, the diversity of its images looks higher than the outputs of the cosine similarity kernel.

**Hyperparameter selection for SPARKE.** In Figure 8, we analyzed the impact of different diversity guidance scales  $\eta$ , ranging from 0 to 0.09. This range was explored to investigate the trade-off between generation quality and diversity. We observed that as  $\eta$  gradually increased, there was a corresponding increase in image diversity and a relative decrease in image quality. As a result, we chose  $\eta = 0.03$  in our experiments to show the balanced trade-off.

**Effect of Classifier-Free Guidance scale on SPARKE guidance.** We study the Classifier-free guidance's impact on SPARKE for the Stable Diffusion 1.5 model, demonstrating how our method alleviates the known quality-diversity trade-off [52]. We changed the CFG scale from 2 to 8 while using the same setting of SD 1.5 as mentioned in Section C.2. We evaluated SPARKE's effect on the quality-diversity trade-off using Precision/Recall metrics in Figure 12 and Density/Coverage metrics in Figure 11. Our results show that SPARKE guidance maintains diversity in a high CFG scale while preserving acceptable generation quality. Additionally, the Vendi score and Kernel Distance (KD) evaluations in Figure 10 show the same trend. Qualitative comparisons of generated samples at CFG

**Gaussian Kernel****Cosine Kernel**

Prompt: A dog standing lying in front of a crackling fireplace in a chalet in Europe.



Prompt: A man casually drinking a hot cup of coffee.



Prompt: A cat resting quietly on a wooden chair.



Prompt: A dog standing beside a cherry blossom tree in full bloom in East Asia.



Prompt: A doctor wearing a white coat in a clinic.



Prompt: A dog standing guarding the entrance of a historic palace in East Asia.

Figure 9: Qualitative comparison between SPARKE numerical results applying Gaussian (RBF) kernel and Cosine similarity kernel on Stable Diffusion XL.

scales  $w = 4, 6, 8$  are presented in Figure 13, demonstrating that SPARKE produces diverse outputs without compromising quality.

**Additional quantitative comparison with the baseline diversity-guided diffusion-based sample generations.** Similar to Table 1, we compare different guidance methods on Stable Diffusion 1.5 [60] in Table 4.

**Latent space vs. ambient space.** We compared latent-space guidance (as in SPARKE) with the CLIP-embedded ambient-space guidance in c-VSG [10] using three categories from the GeoDE dataset [66] and modified it with GPT-4o [74]. We use a similar prompt template as mentioned in Table 7 in [10] to add more details to the prompt. We provide additional samples in Figure 14. We

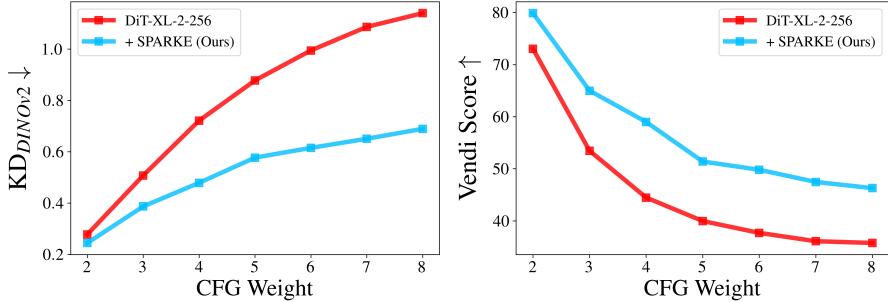


Figure 10: Comparison of image generation diversity and quality in the DiT-XL-2-256 model, analyzing the impact of SPARKE guidance through Vendi Score and Kernel Distance.

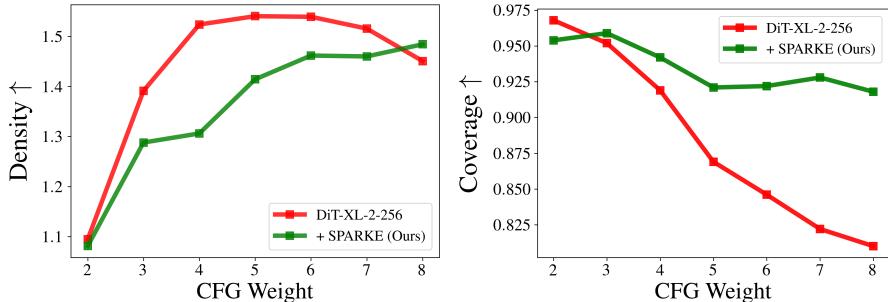


Figure 11: Comparison of image generation quality and diversity in the DiT-XL-2-256 model, analyzing the impact of SPARKE guidance through density and coverage metrics.

also provide quantitative results in Table 5 shows that although CLIP-based Vendi scores were slightly higher for CLIP-embedded ambient guidance, which could be due to optimizing Vendi with the CLIP features, the DINOv2-based Vendi scores and in-batch similarity scores indicate higher diversity for the latent Vendi entropy guidance that we proposed for LDMs. Furthermore, latent guidance can be performed with a considerably lower GPU memory (in our implementation  $\approx 20$ GB for the latent case vs.  $\approx 35$  GB for the ambient case), resulting in higher computational efficiency.

We use the following template:

Prompt to GPT-4o: "You are an expert prompt optimizer for text-to-image models. Text-to-image models take a text prompt as input and generate images depicting the prompt as output. You translate prompts written by humans into better prompts for the text-to-image models. Your answers should be concise and effective. Your task is to optimize this prompt template written by a human: "object in region". This prompt template is used to generate many images of objects such as dogs, chairs, and cars in regions such as Africa, Europe, and East Asia. Generate one sentence of the initial prompt templates that contains the keywords "object" and "region" but increases the diversity of the objects depicted in the image."

**Additional qualitative comparisons of vanilla, baseline diversity guided, and SPARKE prompt-aware diversity guided diffusion models.** We compared the SPARKE prompt-aware diversity guidance method with state-of-the-art conditional latent diffusion models, including Stable Diffusion 2.1, Stable Diffusion XL [5] and PixArt- $\Sigma$  [61].

Additional qualitative comparisons between prompt-aware (conditional RKE score) and prompt-unaware (RKE score) diversity guidance methods are presented on SD 2.1 (Figure 15). Building on these observations, our results demonstrate that SPARKE improves prompt-aware diversity more effectively than prompt-unaware RKE guidance, as quantified by In-batch Similarity [56].

Furthermore, we showed additional results applied to PixArt- $\Sigma$  (Figure 16) and Stable Diffusion XL (Figure 17). The qualitative comparison of SPARKE and without guidance shows SPARKE guidance significantly improves prompt-aware diversity relative to the baseline, highlighting its efficacy across varied latent diffusion model architectures.

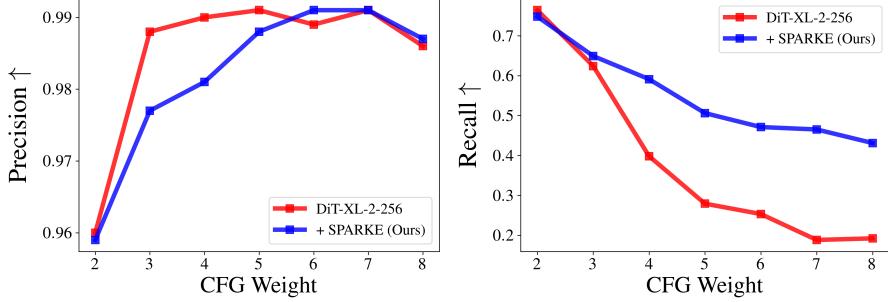


Figure 12: Comparison of image generation quality and diversity in the DiT-XL-2-256 model, analyzing the impact of SPARKE guidance through precision and recall metrics.

Table 4: Quantitative comparison of guidance methods on Stable Diffusion 1.5 (Table 2 in the main text, extended to SD 1.5).

Method	CLIPScore ↑	KD $\times 10^2$ ↓	AuthPct ↑	Cond-Vendi ↑	Vendi ↑	In-batch Sim. $\times 10^2$ ↓
SD v1.5 [60]	30.15	1.045	73.86	25.41	350.28	81.25
c-VSG [10]	27.80	1.078	74.92	26.78	357.39	79.71
CADS [14]	28.94	1.049	75.44	27.60	360.99	78.23
Particle Guide [56]	29.50	1.056	72.58	27.24	345.50	79.50
<b>latent RKE (Ours)</b>	29.23	1.081	78.02	28.75	369.50	78.30
<b>SPARKE (Ours)</b>	29.31	1.071	80.92	31.44	386.42	76.67

Table 5: Comparison of diversity metrics between (CLIP-embedded) ambient and latent guidance.

Guidance Method	IS ↑	Vendi Score <sub>CLIP</sub> ↑	Vendi Score <sub>DINOv2</sub> ↑	In-batch Sim. ↓
(CLIP-embedded) Ambient VSG [10]	5.45	5.34	13.33	0.28
Latent VSG	7.24	4.42	25.22	0.15

DiT-XL-2 Model

DiT-XL-2 + SPARKE (Ours)

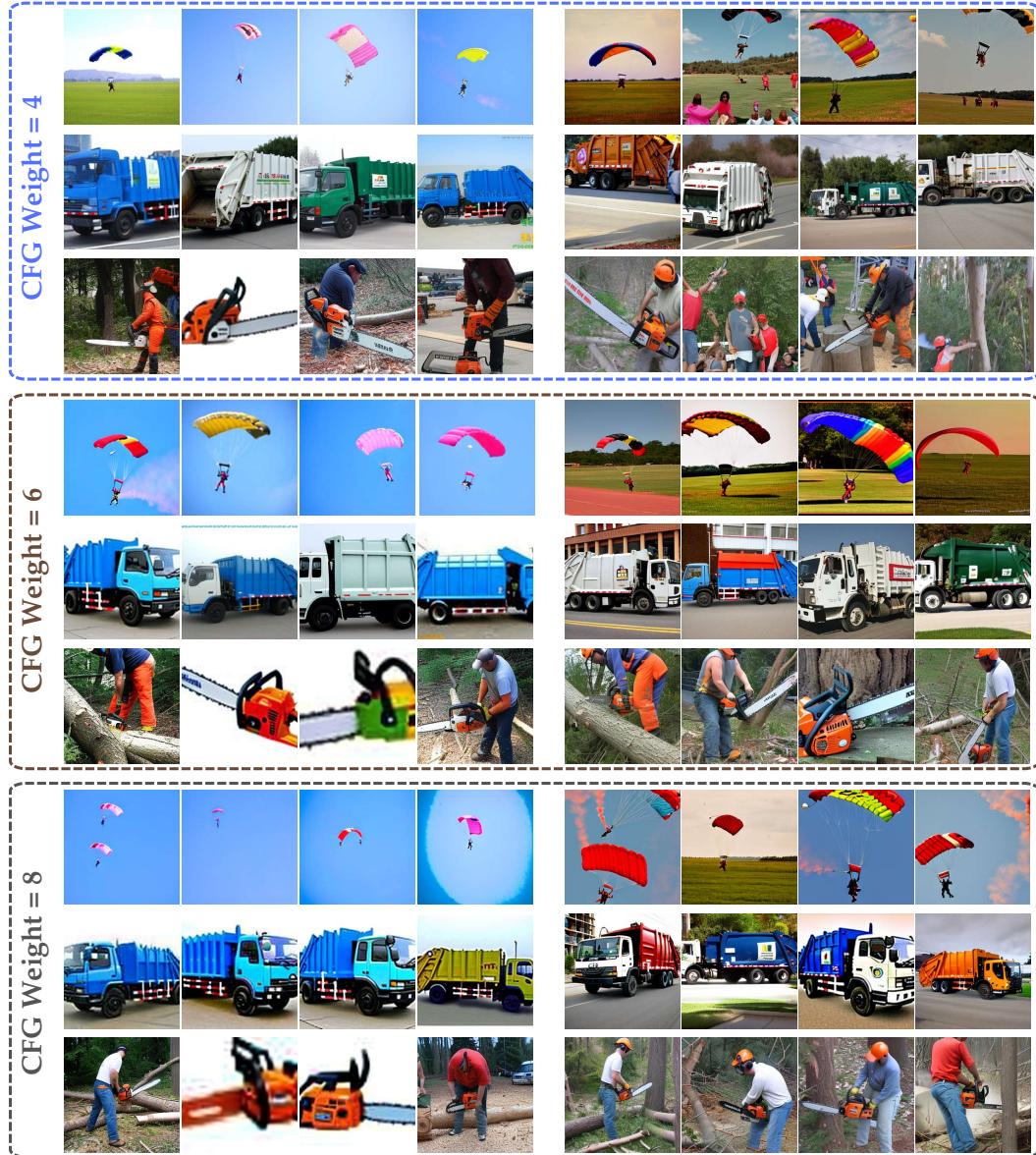


Figure 13: Qualitative comparison of DiT-XL-2 with different classifier free guidance scales: (Left side) standard DiT-XL-2 (Right side) DiT-XL-2 with SPARKE prompt-aware diversity guidance.



Figure 14: Comparison of latent entropy guidance vs. CLIP-(ambient) entropy guidance on SD-XL. Additional Samples for Figure 2.

## Stable Diffusion 2.1

**SPARKE: Prompt-Aware Diversity Guidance  
(Conditional RKE score)**

**Prompt-Unaware Diversity Guidance  
(RKE score)**

In-batch Sim. : 75.68

In-batch Sim.: 79.01



Prompts: (1) iter #6219: This is an image of a bench overlooking a beach, (2) iter #6537: A white bench sitting that overlooks an ocean view, (3) iter #9353: A white wooden bench sitting on top of a green hill next to the ocean.



Prompts: (1) iter #1300: A bird on a branch on the ground, (2) iter #1375: A very cute little bird standing on some rocks, (3) iter #3234: A small bird standing on top of a rock near grass.



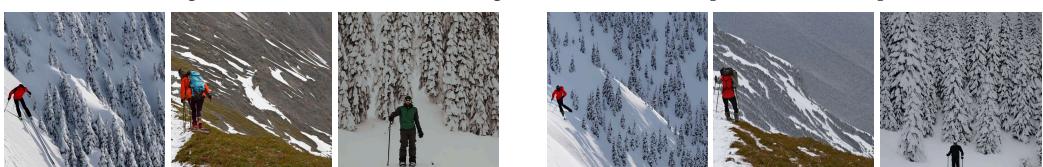
Prompts: (1) iter #242: Two dogs are looking up while they stand near the toilet in the bathroom, (2) iter #3475: Two small dogs standing in a restroom next to a toilet, (3) iter #4875: Small dogs standing in a restroom next to a toilet.



Prompts: (1) iter #6365: A wooden bench with snow in a forest, (2) iter #7348: a wood bench is outside covered in snow, (3) iter #9237: A winter scene of a park bench covered in snow among the trees



Prompts: (1) iter #7927: a blue plate with an orange a cracker some lettuce and a twist bar, (2) iter #8354: A dish contains an orange and snacks., (3) iter #9287: An orange, chocolate, cracker, and piece of lettuce on a plate



Prompts: (1) iter #8034: A person wearing winter gears skis down a mountain, (2) iter #8527: A woman hiking across a mountain with skis., (3) iter #9275: A man on ski's that is standing in the snow.

Figure 15: Comparison of SPARKE prompt-aware diversity guidance via conditional-RKE score vs. diversity-unaware diversity guidance using the RKE score on SD 2.1 text-to-image generation. Additional Samples for Figure 5.

**PixArt- $\Sigma$**



**PixArt- $\Sigma$  + SPARKE Guidance**



Prompts: 1. A photo of a man posing for a portrait, 2. A portrait of a young male facing forward, 3. An upper body portrait of a man.



Prompts: 1. A relaxed cat sitting still on a chair, 2. A fluffy cat sitting calmly on a seat, 3. A peaceful cat on top of a simple chair.



Prompts: 1. A bird sitting calmly among the branches, 2. A bird resting peacefully on a twig, 3. A tree branch holding a bird.



Prompts: 1. An old man walking down a street, 2. An elderly man strolling through a street, 3. An old gentleman taking a walk on the road.



Prompts: 1. A small kid coloring on a table, 2. A child sitting and drawing on a piece of paper, 3. A kid making a drawing on a paper.



Prompts: 1. A man sitting quietly and reading a book, 2. A man reading book on a bench, 3. A man reading a book in a relaxed setting



Prompts: 1. A young woman holding an apple, 2. A female holding an apple and smiling, 3. A woman showing apple with both hands.

Figure 16: Qualitative Comparison of samples generated by standard PixArt- $\Sigma$  vs. SPARKE diversity guided PixArt- $\Sigma$ .



Figure 17: Qualitative Comparison of samples generated by standard Stable Diffusion XL vs. SPARKE diversity guided Stable Diffusion XL.

## NeurIPS Paper Checklist

### 1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: We made the main claims in the abstract and Section 1.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

### 2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: We stated the limitations in Section 6.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

### 3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: We provided the assumptions and proofs in Section 4 and the Appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

#### 4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: **[Yes]**

Justification: We disclosed all the information to reproduce the experimental results in Section 5 and the Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

#### 5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [\[Yes\]](#)

Justification: We provided access to the code in the supplementary materials.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

## 6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [\[Yes\]](#)

Justification: We specified the details in Section 5 and the Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

## 7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [\[Yes\]](#)

Justification: We provided statistical significance in Section 5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer “Yes” if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [\[Yes\]](#)

Justification: We provided sufficient information on the computer resources in Section 5 and the Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

## 9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [\[Yes\]](#)

Justification: We conducted the code of ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

## 10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [\[Yes\]](#)

Justification: We discussed improving diversity in generative models ,which would have potentially positive societal impacts.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

## 11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: We don't have risks of misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

## 12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We credited and explicitly mentioned the assets in Section 5 and the Appendix.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets) has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

### 13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: We don't release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

### 14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: We do not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

### 15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: We do not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

### 16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorosity, or originality of the research, declaration is not required.

Answer: [NA]

Justification: We do not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.