Asynchronous Denoising Diffusion Models for Aligning Text-to-Image Generation

Anonymous authors
Paper under double-blind review

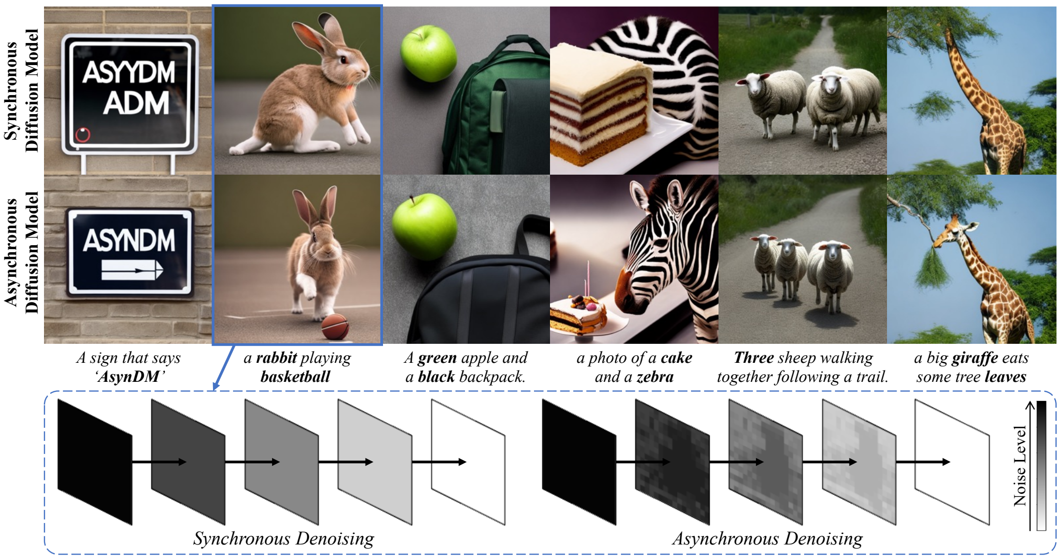

Synchronous Diffusion Model
Asynchronous Diffusion Model
ASYYDM ADM
ASYNDM
A sign that says 'AsynDM'
a **rabbit** playing **basketball**
A **green** apple and a **black** backpack.
a photo of a **cake** and a **zebra**
Three sheep walking together following a trail.
a big **giraffe** eats some tree **leaves**
Synchronous Denoising
Asynchronous Denoising
Noise Level

Figure 1: Existing diffusion models generate images through synchronous denoising, where all pixels are simultaneously denoised step-by-step from noises to images, hindering text-to-image alignment. Asynchronous diffusion models denoise the prompt-related regions more gradually than other regions, thereby receiving clearer inter-pixel context and ultimately achieving improved alignment.

Abstract

Diffusion models have achieved impressive results in generating high-quality images. Yet, they often struggle to faithfully align the generated images with the input prompts. This limitation is associated with synchronous denoising, where all pixels simultaneously evolve from random noise to clear images. As a result, during generation, the prompt-related regions can only reference the unrelated regions at the same noise level, failing to obtain clear context and ultimately impairing text-to-image alignment. To address this issue, we propose asynchronous diffusion models—a novel framework that allocates distinct timesteps to different pixels and reformulates the pixel-wise denoising process. By dynamically modulating the timestep schedules of individual pixels, prompt-related regions are denoised more gradually than unrelated regions, thereby allowing them to leverage clearer inter-pixel context. Consequently, these prompt-related regions achieve better alignment in the final images. Extensive experiments demonstrate that our asynchronous diffusion models can significantly improve text-to-image alignment across diverse prompts.

1 Introduction

Diffusion models have achieved remarkable success across a wide range of domains, such as robotics (Chi et al., 2024; Wolf et al., 2025), classification (Li et al., 2023a; Tong et al., 2025a), image segmentation (Amit et al., 2021), text generation (Austin et al., 2021; Nie et al., 2025) and visual generation (Yang et al., 2023; Wang et al., 2025). Among these, text-to-image generation has emerged as the most widely recognized application, with the generated images demonstrating impressive diversity and high fidelity (Ho et al., 2020; Rombach et al., 2022). Despite their success, even the most advanced diffusion models still struggle with the issue of *text-to-image misalignment* (Hinz et al., 2020; Ramesh et al., 2022; Feng et al., 2023; Chefer et al., 2023), where the generated images often fail to faithfully match the user-provided prompts, for example with respect to text, color, or count, as illustrated in Figure 1.

We argue that misalignment in diffusion models is closely associated with the issue of *synchronous denoising*. That is, under the formulation of a Markov decision process (Ho et al., 2020; Song et al., 2022), all pixels in an image simultaneously evolve from random noise to a clear state, following the same timestep schedule. At each denoising step, pixels interact by leveraging one another as contextual references, ultimately forming a coherent and harmonious image.

Beyond this, an image is composed of diverse regions. Some of these regions correspond directly to the objects described in the prompt, while others serve as background. For aligned generation, prompt-related regions typically demand more gradual refinement to accurately capture fine-grained semantics. In contrast, prompt-unrelated regions involve fewer semantic constraints and mainly provide supporting context, allowing them to be denoised into a clear state relatively quickly. However, synchronous denoising treats all pixels equally, overlooking the heterogeneous nature of different regions. Consequently, these prompt-related regions always rely on other regions at the same noise level for contextual references. This raises the concern that *synchronuous denoising limits the effective utilization of inter-pixel context, and ultimately hinders text-to-image alignment*.

Based on the above motivation, we propose **Asyn**chronous **D**iffusion **M**odels (AsynDM), a plug-and-play and tuning-free framework that reformulates the denoising process of pre-trained diffusion models. Instead of denoising all pixels simultaneously, the asynchronous diffusion model allows different pixels to be denoised according to varying timestep schedules, as shown in Figure 1. In particular, prompt-unrelated regions can be denoised more quickly, while prompt-related regions are denoised more gradually to ensure sufficient refinement for capturing prompt semantics. These clearer unrelated regions prevent noisy and ambiguous context from bringing uncertainty to the related regions (*e.g.*, undetermined style, shape, etc.). As a result, the related regions can better focus on the content specified by the prompt, thereby enhancing text-to-image alignment.

Moreover, we introduce a method that dynamically identifies the prompt-related regions and modulates the timestep schedules along the denoising process. Specifically, the cross-attention modules (Vaswani et al., 2017) in diffusion models encapsulate rich information about the shapes and structures of the generated images. At each denoising step, we can extract a mask from the cross-attention modules, which highlights the objects in the prompt. Guided by this mask, the asynchronous diffusion model adaptively modulates the timestep schedules of different regions. The highlighted regions (*i.e.*, prompt-related regions) are modulated to be denoised more gradually than other regions (*i.e.*, prompt-unrelated regions), thereby receiving clearer inter-pixel context.

We conduct experiments on four sets of commonly used prompts and compare with advanced baselines. The results show that AsynDM can effectively improve text-to-image alignment both qualitatively and quantitatively. Meanwhile, AsynDM maintains comparable sampling efficiency to the vanilla diffusion model, as it only requires the additional encoding of pixel-wise timesteps.

The main contributions of this paper can be summerized as follows: (1) We highlight that synchronous denoising is a primary reason for the text-to-image misalignment in existing diffusion models. (2) We propose asynchronous diffusion models that introduces pixel-level timesteps, and adaptively modulate the timestep schedules of different pixels, to address the above issue. (3) Comprehensive experiments demonstrate that asynchronous diffusion models consistently improve text-to-image alignment across diverse prompts.

2 Background

2.1 Text-to-Image Diffusion Models

Diffusion Model Formulation. Diffusion models have emerged as a powerful family of text-to-image generative models. DDPM (Ho et al., 2020) formulates the generation process as a Markovian

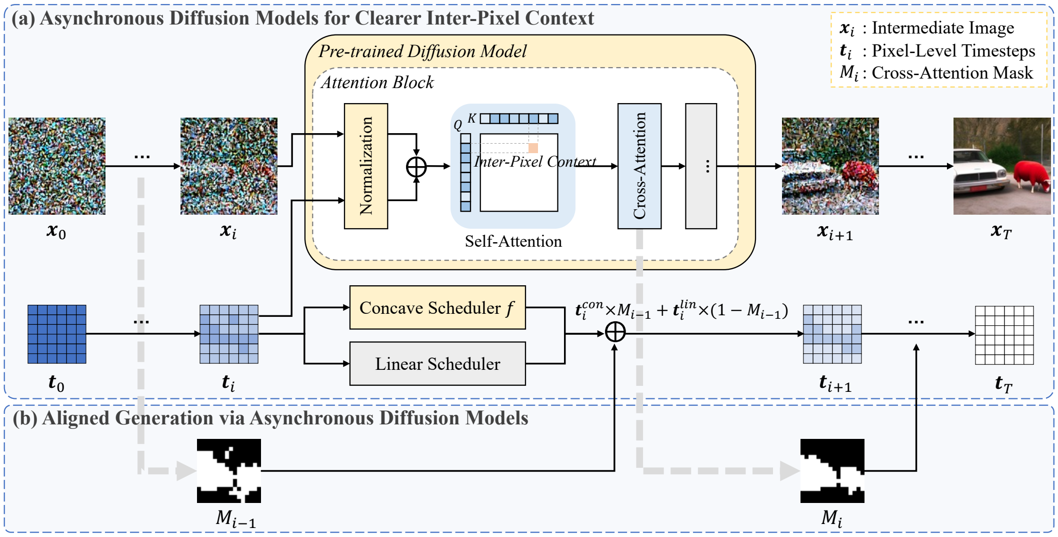

(a) Asynchronous Diffusion Models for Clearer Inter-Pixel Context
x_i : Intermediate Image
t_i : Pixel-Level Timesteps
M_i : Cross-Attention Mask
Pre-trained Diffusion Model
Attention Block
Normalization
Q
K
Inter-Pixel Context
Self-Attention
Cross-Attention
x_0
x_i
x_{i+1}
x_T
t_0
t_i
t_{i+1}
t_T
Concave Scheduler f
Linear Scheduler
$t_i^{con}\times M_{i-1} + t_i^{lin}\times(1-M_{i-1})$
(b) Aligned Generation via Asynchronous Diffusion Models
M_{i-1}
M_i

Figure 2: Asynchronous diffusion models improve text-to-image alignment by (a) assigning distinct timesteps to different pixels, where faster-denoised regions provide clearer context, serving as better references for slower ones, and (b) using masks extracted from cross-attention to identify prompt-related regions and dynamically modulate pixel-level timestep schedules.

sequence of latent states. By denoising step by step, these models progressively transform random noise into a coherent image. Based on the DDPM sampler, at each denoising step, the model predicts the last intermediate state $\mathbf{x}_{t-1}$ from current state $\mathbf{x}_t$ according to:

$$p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{c}) = \mathcal{N}(\mathbf{x}_{t-1} \mid \mu_\theta(\mathbf{x}_t, t, \mathbf{c}), \sigma_t^2 \mathbf{I}), \tag{1}$$

$$\text{with} \quad \mu_\theta(\mathbf{x}_t, t, \mathbf{c}) = \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}})\epsilon_\theta(\mathbf{x}_t, t, \mathbf{c}), \tag{2}$$

where ϵ_θ denotes the denoising model paramiterized by θ, $\mathbf{c}$ is the prompt, and σ_t, α_t and β_t are timestep-dependent constants. Subsequent extensions, such as DDIM (Song et al., 2022) and DPM-Solver (Lu et al., 2022), further enhance the efficiency and sample quality. These formulations act as the foundation of most modern diffusion-based generative models (Rombach et al., 2022).

Attention Module in Diffusion Models. The attention mechanism (Vaswani et al., 2017) has played an important role not only in large language models (Zhao et al., 2025; Han et al., 2025), but also in text-to-image diffusion models (Hertz et al., 2023; Tumanyan et al., 2023). Both UNet-based (Rombach et al., 2022; Podell et al., 2023) and DiT-based (Peebles & Xie, 2023; Esser et al., 2024) diffusion models employ attention blocks to enhance expressiveness. A typical attention block includes a self-attention part and a cross-attention part, and can be formally expressed as:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^\top}{\sqrt{d_\text{key}}})V, \tag{3}$$

where $Q \in \mathbb{R}^{m\times d_\text{key}}$ denotes queries projected from image features, and $K \in \mathbb{R}^{n\times d_\text{key}}$, $V \in \mathbb{R}^{n\times d_\text{value}}$ denote keys and values, projected either from image features (in self-attention) or from prompt embeddings (in cross-attention). Cross-attention allows the models to condition image generation on textual prompts, while self-attention further enables the models to capture long-range dependencies across the pixels.

2.2 Diffusion Model Alignment

Text-to-image misalignment has been a longstanding challenge across various generative models, including VAEs, GANs and diffusion models (Zhang & Peng, 2018; Wang et al., 2021; Liao et al., 2022). Early diffusion model studies have explored methods for conditioning generation on specific factors, such as class labels (Dhariwal & Nichol, 2021; Wang et al., 2023b), styles (Sohn et al., 2023) and layouts (Zheng et al., 2023). The incorporation of text encoders has endowed diffusion models with the capability to generate images from textual descriptions(Rombach et al., 2022). Following this development, recent studies therefore focus on the challenge of text-to-image misalignment in diffusion models, which is essential for the reliable deployment.

On the one hand, researchers have sought to achieve better alignment through fine-tuning. Some studies focus on directly fine-tuning the model (Lee et al., 2023; Tong et al., 2025b), among which reinforcement learning-based methods stand out (Fan et al., 2023; Hu et al., 2025a;b). Others optimize different components without altering the main model parameters. For example, some progressively refine the intermediate noisy images during the denoising process (Chefer et al., 2023; Li et al., 2023c; Rassin et al., 2023), while others optimize prompts to be more precise and informative (Wang et al., 2023a; Mañas et al., 2024).

On the other hand, some studies investigate alignment techniques that do not require fine-tuning. For instance, Z-Sampling (LiChen et al., 2025) enhances alignment by introducing zigzag diffusion step. SEG (Hong, 2024) exploits the energy-based perspective of self-attention to improve image generation. S-CFG (Shen et al., 2024) and CFG++ (Chung et al., 2025) improve text-to-image alignment by refining the classifier-free guidance technique (Ho & Salimans, 2022).

3 Asynchronous Denoising for Clearer Inter-Pixel Context

In this section, we first introduce the rationale and methodology for allocating distinct timesteps to pixels. We then describe our approach to scheduling the pixel-level timesteps in asynchronous diffusion models. The overview of this section is shown in Figure 2 (a).

3.1 Pixel-Level Timestep Allocation

It is reasonable to allocate distinct timesteps to different pixels. During the denoising process of diffusion models, image features establish inter-pixel dependencies through the attention mechanism, thus pixels can interact with each other and form a coherent image. Notably, timestep information is embedded into the features in a pixel-wise manner external to the attention modules, rather than being directly injected into the attention. In other words, timesteps are involved only in intra-pixel computations, which naturally allows different pixels to be associated with distinct timesteps.

We present the pixel-level timestep formulation of the DDPM sampler, as follows[1]. We adopt $i \in [0, T]$ as the new index of the denoising process, since different pixels have distinct timesteps t. This formulation performs denoising from 0 to T, rather than from T to 0. Accordingly, the model predicts the next state $\mathbf{x}_{i+1}$ from current state $\mathbf{x}_i$, where $\mathbf{x}_i, \mathbf{x}_{i+1} \in \mathbb{R}^{n_c \times h \times w}$.

$$p_\theta(\mathbf{x}_{i+1} \mid \mathbf{x}_i, \mathbf{c}) = \mathcal{N}(\mathbf{x}_{i+1} \mid \mu_\theta(\mathbf{x}_i, \mathbf{t}_i, \mathbf{c}), \sigma_i^2 \mathbf{I}), \tag{4}$$

$$\text{with } \mu_\theta(\mathbf{x}_i, \mathbf{t}_i, \mathbf{c}) = \frac{1}{\sqrt{\alpha_{\mathbf{t}_i}}}(\mathbf{x}_i - \frac{\beta_{\mathbf{t}_i}}{\sqrt{1 - \bar{\alpha}_{\mathbf{t}_i}}})\epsilon_\theta(\mathbf{x}_i, \mathbf{t}_i, \mathbf{c}), \tag{5}$$

where $\mathbf{t}_i \in \mathbb{R}^{h \times w}$ denotes the timestep states assigned to individual pixels. Specifically, $\alpha_{\mathbf{t}_i}$, $\beta_{\mathbf{t}_i}$ and $\bar{\alpha}_{\mathbf{t}_i}$ denote element-wise indexing, where each entry of $\mathbf{t}_i$ selects corresponding scalar value, yielding matrices of the same shape as $\mathbf{t}_i$. These constant matrices are automatically broadcast along the channel dimension, enabling joint computations with $\mathbf{x}_i$. Moreover, the denoising model ϵ_θ can be seamlessly extended to handle pixel-level timesteps by independently encoding them and incorporating the resulting embeddings into the original computation on a per-pixel basis.

The above formulation enables diffusion models to incorporate pixel-level timesteps. Importantly, the asynchronous diffusion model still preserves the *Markov property*. In the asynchronous setting, $\mathbf{t}_i$ becomes a tensor with the same height and width as $\mathbf{x}_i$, serving as a state within the Markov chain, rather than its original role as the reverse-time index.

3.2 Timestep Scheduling in Asynchronous Diffusion Models

During the denoising process of diffusion models, the noise level of individual pixels gradually decreases as the timestep progresses from T to 0. In conventional diffusion models, all pixels share

[1] The pixel-level timestep formulation can generalize across diverse diffusion samplers. We also provide the formulation of DDIM sampler in Appendix A.2.

the same timestep scheduler from T to 0, and commonly used samplers, such as DDPM and DDIM, typically implement this progression linearly. In this subsection, we schedule the timesteps and allow certain regions to evolve more slowly than others. This scheduling enables these regions to accumulate clearer inter-pixel context, thereby achieving more gradual refinement.

We adopt the concave function $t = f(i)$ as the scheduler, according to Proposition 1.

Proposition 1. (See proof in Appendix A.1) *Let $f(i) : [0, T] \to \mathbb{R}$ be a concave function with $f(0) = T$ and $f(T) = 0$. For any i_0 with $0 < i_0 < T$ and any t_0 with $T - i_0 \leq t_0 \leq f(i_0)$, there exist unique constants a, b such that the shifted function $f(i - a) + b$ satisfies:*

$$f(i_0 - a) + b = t_0, \qquad f(T - a) + b = 0. \tag{6}$$

As illustrated in Figure 3, this proposition states that any point located within the shaded area can reach $t = 0$ along the appropriately shifted concave function. In the asynchronous diffusion model, pixels within the target regions (*i.e.*, the prompt-related regions in text-to-image alignment task) are denoised according to the concave function. By applying only a shift to the concave function, regions selected earlier as targets are denoised at a slower rate. Other regions, in contrast, are denoised following a linear function (or a less concave function in some samplers). Therefore, the target regions can be denoised more gradually, thus receive clearer inter-pixel context.

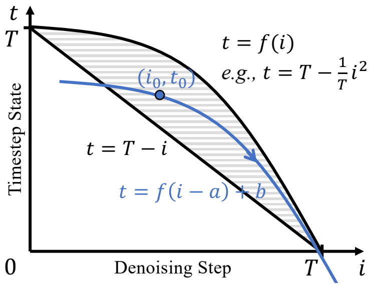


Figure 3: Any point located within the shaded area can reach $t = 0$ along appropriately shifted f.

From the perspective of a Markov decision process, in the conventional synchronous diffusion models, the state $\mathbf{x}_t$ transitions to the next state $\mathbf{x}_{t-1}$ under the policy distribution p_θ. Differently, the state in the asynchronous diffusion model is composed of $(\mathbf{x}_i, \mathbf{t}_i)$, which transitions to the next state $(\mathbf{x}_{i+1}, \mathbf{t}_{i+1})$ under the policy distribution (p_θ, f). In our experiments, we simply adopt a quadratic function $f(i) = T - \frac{1}{T}i^2$ as the scheduling function.

4 Aligned Generation via Asynchronous Diffusion Models

In this section, we introduce a method that dynamically identifies the prompt-related regions and modulates the timestep schedules of individual pixels along the denoising process.

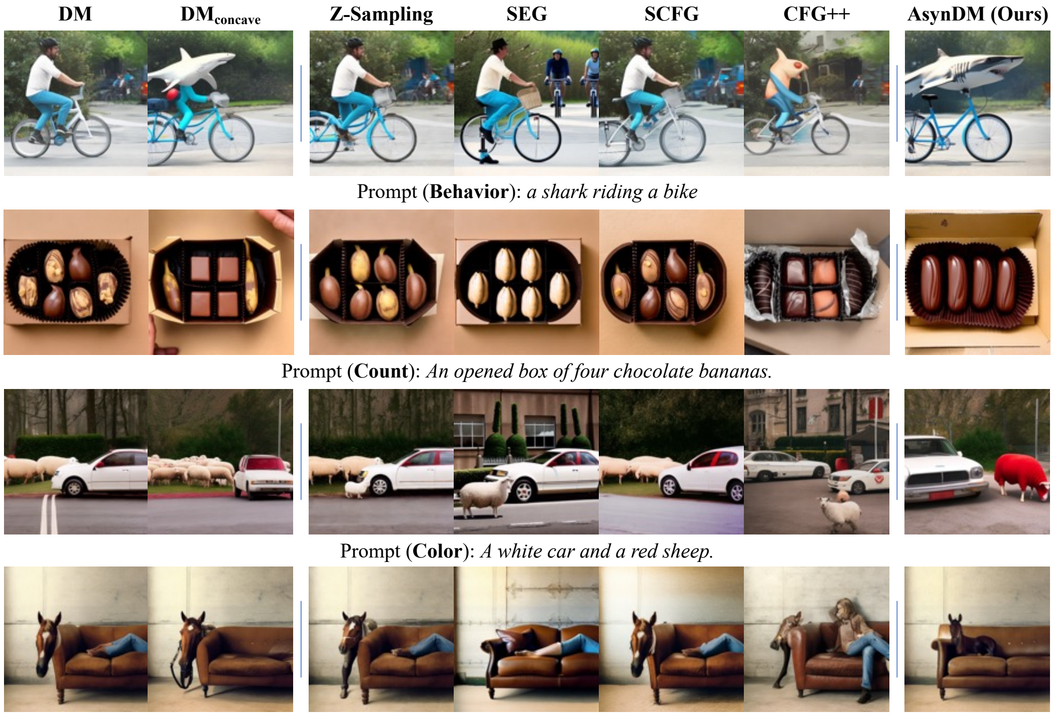
Prompt-Related Region Extraction. In most text-to-image diffusion models, cross-attention is employed to condition image generation on textual prompts. Even for DiT-based models that rely solely on self-attention, the prompt embeddings are concatenated with image features, thereby enabling implicit cross-attention computations within the self-attention modules (Peebles & Xie, 2023).

In cross-attention computation, the term softmax$(\frac{QK^\top}{\sqrt{d_{\text{key}}}})$ is commonly referred to as cross-attention maps, denoted by $A \in \mathbb{R}^{|\mathbf{c}| \times h \times w}$, where $|\mathbf{c}|$ is the number of tokens in prompt $\mathbf{c}$. Previous studies (Tang et al., 2023; Hertz et al., 2023; Cao et al., 2023) show that cross-attention maps encapsulate rich information about the shapes and structures of the generated images. Specifically, the o-th map in A, denoted by A^o, highlights the pixels most influenced by the o-th token. This property allows us to extract a mask that identifies the image regions most relevant to the prompt, as follows:

$$M = \bigvee_{o \in \mathcal{O}_\mathbf{c}} \{\mathbf{1}[A^o > A^o_{\text{mean}}]\}, \tag{7}$$

where $\mathcal{O}_\mathbf{c}$ denotes the set of token indices corresponding to the objects described in prompt $\mathbf{c}$. For each token o, A^o_{mean} represents the average value of its cross-attention map A^o. $\mathbf{1}[\cdot]$ is the indicator function that produces a binary mask based on the given condition, and the operator $\bigvee$ indicates an element-wise logical OR across the resulting masks. This formula ultimately yields a mask that highlights the prompt-related regions.

Mask-Guided Asynchronous Denoising. At each denoising step i, we can extract a mask M_i according to Eq.(7). As illustrated in Figure 2 (b), each mask serves as a guidance signal for the next denoising step, where the highlighted regions follow the concave scheduler, and the remaining



Prompt (**Behavior**): *a shark riding a bike*

Prompt (**Count**): *An opened box of four chocolate bananas.*

Prompt (**Color**): *A white car and a red sheep.*

Prompt (**Co-occurrence**): *a photo of a couch and a horse*

Figure 4: The samples generated by AsynDM and baseline methods across diverse prompts. The images generated by AsynDM show better text-to-image alignment.

regions follow the linear scheduler. As denoising progresses, the image gradually becomes clearer in a coarse-to-fine manner (Park et al., 2023; Rissanen et al., 2023), and the mask correspondingly evolves to precisely indicate the shapes and positions of the objects. Consequently, the object-related regions are dynamically modulated to denoise more slowly and gradually, thereby receiving clearer inter-pixel context. The clearer context enables these object-related regions to better focus on the content specified by the prompt, ultimately yielding more faithful and aligned image generation.

5 Experiments

In this section, we first introduce our experimental setting. Next, we demonstrate the effectiveness of AsynDM in improving text-to-image alignment, providing both qualitative and quantitative results across diverse prompts and in comparison with multiple baselines. Finally, we conduct ablation on the mask and the concave scheduler, demonstrating the effectiveness and robustness of AsynDM.

5.1 Experimental Setting

Diffusion Models. We adopt Stable Diffusion (SD) 2.1-512-base (Rombach et al., 2022), one of the commonly used UNet-based diffusion models, as the foundation model of our experiments. The total timesteps T is set to 50. We employ the DDIM sampler (Song et al., 2022), and the noise weight η is set to 1.0, which determines the extent of randomness at each denoising step. We also conduct experiments on more advanced diffusion models, including the UNet-based SDXL-base-1.0 (Podell et al., 2023) and DiT-based SD3.5-medium (Esser et al., 2024). The experimental results on these models are shown in Appendix D.1.

Prompts. We adopt four commonly used prompt sets in our experiments. (1) *Animal activity* (Black et al., 2023). This prompt set has the form "*a(n) [animal] [activity]*", where the activities come from humans, such as "*riding a bike*". (2) *Drawbench* (Saharia et al., 2022). This prompt set consists of 11 categories with approximately 200 prompts, including aspects such as color and count. (3) *GenEval* (Ghosh et al., 2023). This prompt set incorporates 553 prompts, including aspects such as

Table 1: Text-to-image alignment performance of AsynDM compared with baseline methods across diverse prompts.

Prompt Set	Method	BERTScore↑	CLIPScore↑	ImageReward↑	QwenScore↑
Animal Activity	DM	0.6353	0.3685	0.7543	4.9445
	$\text{DM}_{\text{concave}}$	0.6381 (+0.0028)	0.3715 (+0.0030)	0.8544 (+0.1001)	5.0695 (+0.1250)
	Z-Sampling	0.6353 (+0.0000)	0.3708 (+0.0023)	0.8283 (+0.0740)	5.0242 (+0.0797)
	SEG	0.6309 (-0.0044)	0.3605 (-0.0080)	0.6493 (-0.1050)	4.7632 (-0.1813)
	S-CFG	0.6383 (+0.0030)	0.3716 (+0.0031)	0.8653 (+0.1110)	5.0421 (+0.0976)
	CFG++	0.6249 (-0.0104)	0.3565 (-0.0120)	0.3284 (-0.4259)	4.4484 (-0.4961)
	AsynDM	**0.6414 (+0.0061)**	**0.3750 (+0.0065)**	**0.9219 (+0.1676)**	**5.5218 (+0.5773)**
Drawbench	DM	0.6968	0.3659	0.3943	4.7406
	$\text{DM}_{\text{concave}}$	0.6970 (+0.0002)	0.3670 (+0.0011)	0.4152 (+0.0209)	4.8179 (+0.0773)
	Z-Sampling	0.6979 (+0.0011)	0.3676 (+0.0017)	0.4505 (+0.0562)	4.7656 (+0.0250)
	SEG	0.6925 (-0.0043)	0.3527 (-0.0132)	0.2478 (-0.1465)	4.6695 (-0.0711)
	S-CFG	0.6972 (+0.0004)	0.3693 (+0.0034)	0.4398 (+0.0455)	4.8750 (+0.1344)
	CFG++	0.6938 (-0.0030)	0.3539 (-0.0120)	0.1644 (-0.2299)	4.6210 (-0.1196)
	AsynDM	**0.7007 (+0.0039)**	**0.3701 (+0.0042)**	**0.4560 (+0.0617)**	**4.9804 (+0.2398)**
GenEval	DM	0.7030	0.3620	0.1541	4.9390
	$\text{DM}_{\text{concave}}$	0.7039 (+0.0009)	0.3637 (+0.0017)	0.1979 (+0.0438)	4.9976 (+0.0586)
	Z-Sampling	0.7046 (+0.0016)	0.3626 (+0.0006)	0.1757 (+0.0216)	4.9179 (-0.0211)
	SEG	0.7005 (-0.0025)	0.3493 (-0.0127)	0.0689 (-0.0852)	4.9125 (-0.0265)
	S-CFG	0.7031 (+0.0001)	0.3630 (+0.0010)	0.1819 (+0.0278)	4.8968 (-0.0422)
	CFG++	0.6992 (-0.0038)	0.3482 (-0.0138)	-0.1344 (-0.2885)	4.5835 (-0.3555)
	AsynDM	**0.7081 (+0.0051)**	**0.3683 (+0.0063)**	**0.2895 (+0.1354)**	**5.3390 (+0.4000)**
MSCOCO	DM	0.6995	0.3388	0.2696	5.8507
	$\text{DM}_{\text{concave}}$	0.7004 (+0.0009)	0.3395 (+0.0007)	0.2917 (+0.0221)	5.9632 (+0.1125)
	Z-Sampling	0.6999 (+0.0004)	0.3377 (-0.0011)	0.2946 (+0.0250)	5.8289 (-0.0218)
	SEG	0.6952 (-0.0043)	0.3295 (-0.0093)	0.1667 (-0.1029)	5.8320 (-0.0187)
	S-CFG	0.6995 (+0.0000)	0.3409 (+0.0021)	0.3316 (+0.0620)	5.9328 (+0.0821)
	CFG++	0.6975 (-0.0020)	0.3348 (-0.0040)	0.1471 (-0.1225)	5.6921 (-0.1586)
	AsynDM	**0.7055 (+0.0060)**	**0.3420 (+0.0032)**	**0.3339 (+0.0643)**	**6.2601 (+0.4094)**

co-occurrence, color and count. (4) *MSCOCO* (Lin et al., 2014). This prompt set is derived from the captions of the MSCOCO 2014 validation set and consists of descriptions of real-world images. For each set, we randomly select 40 prompts for our experiments.

Metrics. In our experiments, we employ four metrics to evaluate text-to-image alignment. (1) *BERTScore* (Zhang et al., 2020). This metric leverages a multimodal large language model to generate a description for the image, and then employs BERT-based recall to quantify the semantic similarity between the prompt and the generated description. In our implementation, we use Qwen2.5-VL-7B-Instruct (Wang et al., 2024) to generate descriptions and DeBERTa xlarge model (He et al., 2021) to compute similarity. (2) *CLIPScore*. This metric measures the similarity between the text embeddings and image embeddings encoded by CLIP model (Radford et al., 2021). We use ViT-H-14 CLIP model in our implementation. (3) *ImageReward* (Xu et al., 2023). This metric employs a pre-trained model to estimate human preferences, in which alignment serves as a key factor. (4) *QwenScore*. We employ Qwen2.5-VL-7B-Instruct (Wang et al., 2024) to score text-to-image alignment directly, ranging from 0 to 9. The prompts fed to Qwen are provided in Appendix B.4.

Baselines. We sample the diffusion model using both the standard scheduler and the concave scheduler, denoted as DM and $\text{DM}_{\text{concave}}$, respectively. In addition, we compare AsynDM with the most advanced methods, including Z-Sampling (LiChen et al., 2025), SEG (Hong, 2024), S-CFG (Shen et al., 2024) and CFG++ (Chung et al., 2025).

5.2 Qualitative Evaluation

We first provide the qualitative results of AsynDM in comparison with multiple baselines, as shown in Figure 4. We select several representative prompts that encompass object behavior, count, color, and co-occurrence. The vanilla diffusion model (*i.e.*, DM and $\text{DM}_{\text{concave}}$) fails to generate images that are well aligned with the prompts. In contrast, AsynDM effectively generates well-aligned

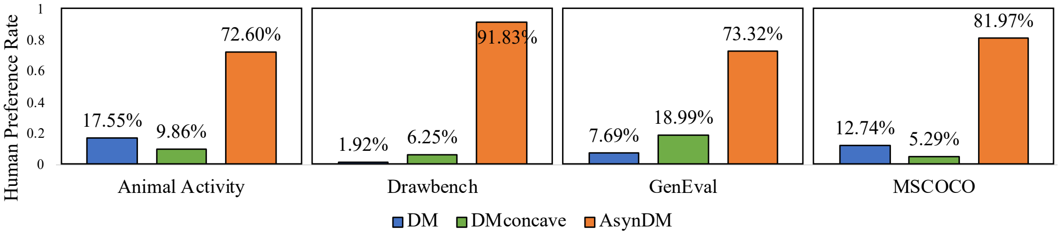


Figure 5: Human preference rates for text-to-image alignment of the images generated by DM, $\text{DM}_{\text{concave}}$ and AsynDM.

Table 2: Text-to-image alignment performance of AsynDM when employing different concave schedulers and using fixed masks, across prompts from animal activity set.

Scheduler	Method	BERTScore↑	CLIPScore↑	ImageReward↑	QwenScore↑
	DM	0.6353	0.3685	0.7543	4.9445
Quadratic	$\text{DM}_{\text{concave}}$	0.6381 (+0.0028)	0.3715 (+0.0030)	0.8544 (+0.1001)	5.0695 (+0.1250)
	AsynDM	**0.6414 (+0.0061)**	**0.3750 (+0.0065)**	**0.9219 (+0.1676)**	**5.5218 (+0.5773)**
	+fixed mask	0.6405 (+0.0052)	0.3722 (+0.0037)	0.8642 (+0.1099)	5.2593 (+0.3148)
Piecewise Linear	$\text{DM}_{\text{concave}}$	0.6338 (-0.0015)	0.3667 (-0.0018)	0.7043 (-0.0500)	4.7406 (-0.2039)
	AsynDM	**0.6401 (+0.0048)**	**0.3724 (+0.0039)**	**0.8472 (+0.0929)**	**5.2335 (+0.2890)**
	+fixed mask	0.6383 (+0.0030)	0.3705 (+0.0020)	0.7504 (-0.0039)	5.0812 (+0.1367)
Exponential	$\text{DM}_{\text{concave}}$	0.6352 (-0.0001)	0.3689 (+0.0004)	0.7981 (+0.0438)	4.9289 (-0.0156)
	AsynDM	**0.6408 (+0.0055)**	**0.3715 (+0.0030)**	**0.8686 (+0.1143)**	**5.2367 (+0.2922)**
	+fixed mask	0.6386 (+0.0033)	0.3714 (+0.0029)	0.8374 (+0.0831)	5.2023 (+0.2578)

images with the same random seeds. Additional qualitative examples, together with those from SDXL and SD 3.5, can be found in Appendix E.

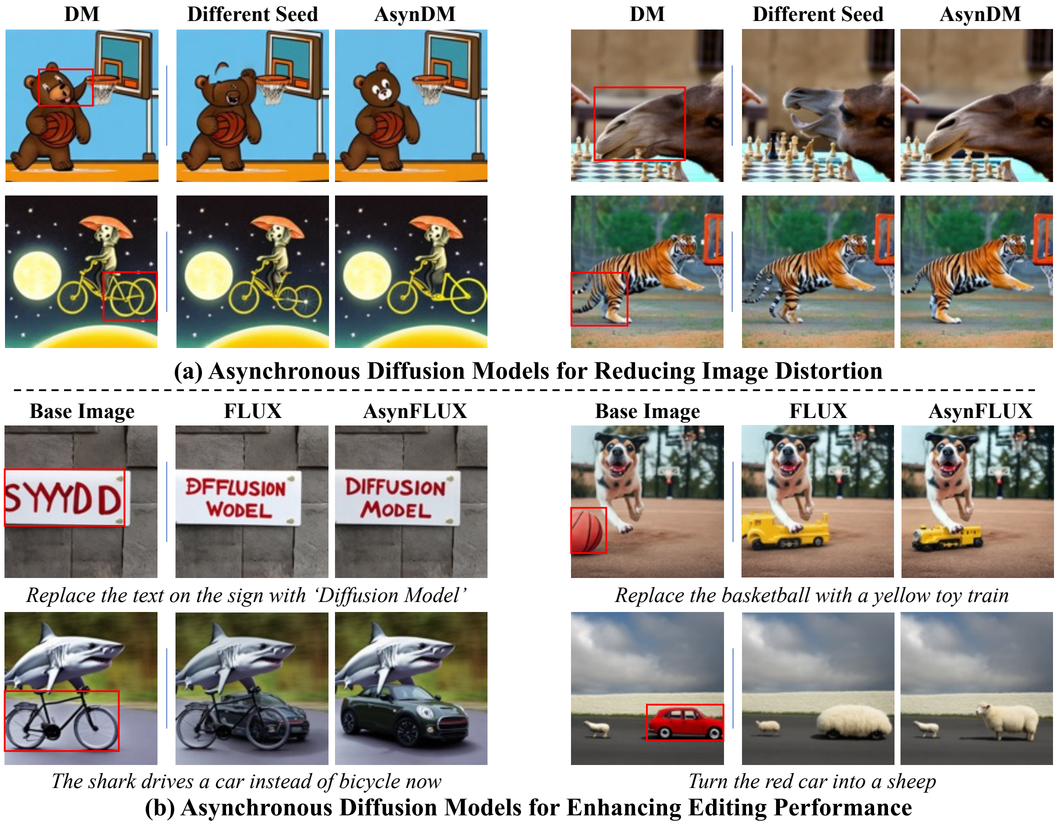
5.3 Quantitative Evaluation

We also quantitatively demonstrate the text-to-image alignment performance of AsynDM compared with baseline methods. As shown in Table 1, we sample 1,280 images for each of the four prompt sets, using the same random seeds across different methods. The generated images are then evaluated with four metrics. The results demonstrate that AsynDM consistently achieves better alignment across all prompt sets. Meanwhile, sampling 1,280 images takes 78 minutes using the vanilla diffusion model, compared to 86 minutes using AsynDM, which indicates that AsynDM achieves improvements without significantly sacrificing efficiency. In addition, we conduct a human evaluation. We invite 52 participants to choose the image they consider best aligned with the prompt from each group of three candidates, corresponding to DM, $\text{DM}_{\text{concave}}$ and AsynDM. As shown in Figure 5, the results further demonstrate that AsynDM improves text-to-image alignment.

We also evaluate the image quality of AsynDM using FID-30K (↓). FID-30K refers to the Frechet Inception Distance calculated using 30,000 images from the MSCOCO 2024 validation set as the reference dataset (Pavlov et al., 2023; Lin et al., 2014). We merge all four prompt sets and generate 16,000 images with each of DM, $\text{DM}_{\text{concave}}$, and AsynDM. The resulting FID-30K scores are 48.63 for DM, 49.29 for $\text{DM}_{\text{concave}}$, and 49.38 for AsynDM. These results indicate that our method can largely preserve the image quality of the pretrained diffusion model.

5.4 Ablation Study

Ablation on Mask. In this ablation study, we replace the dynamically updated mask with a fixed mask. This fixed mask is extracted from the average cross-attention map of DM during its denoising process, following Eq.(7). Due to the use of the same random seed, the mask derived from DM can *roughly* highlight the prompt-related regions in the image generated by AsynDM. The results are shown in the Table 2. Despite the fixed mask being imperfect, AsynDM still improves text-to-image alignment compared with the base model, demonstrating its robustness to inaccurate masks.



(a) Asynchronous Diffusion Models for Reducing Image Distortion

Replace the text on the sign with 'Diffusion Model'

Replace the basketball with a yellow toy train

The shark drives a car instead of bicycle now

Turn the red car into a sheep

(b) Asynchronous Diffusion Models for Enhancing Editing Performance

Figure 6: We further employ AsynDM to reduce image distortion and enhance editing performance.

Ablation on Concave Scheduler. In addition to the quadratic scheduler, we also employ the piecewise linear scheduler and the exponential scheduler to AsynDM, as follows:

$$f(i) = \min(T - \frac{1}{2}i, \frac{3}{2}T - \frac{3}{2}i), \qquad \text{(Piecewise Linear Scheduler)}$$

$$f(i) = \frac{T}{e-1}(e - e^{\frac{1}{T}i}). \qquad \text{(Exponential Scheduler)}$$

As shown in Table 2, AsynDM consistently improves image alignment across different schedulers. This is because, across all the variants, these concave schedulers enable the prompt-related regions to receive clearer inter-pixel context. These results further demonstrate the effectiveness and robustness of AsynDM. The image samples of these two ablation studies are provided in Appendix E.

6 Further Exploration and Discussion

Asynchronous Diffusion Models for Reducing Image Distortion. Diffusion-generated images often suffer from distortions, such as abnormal limb shapes. As shown in Figure 6 (a), inpainting the distorted regions under different random seeds yields limited improvements. In contrast, applying AsynDM with a mask over the distorted regions, while using the same seed, generates improved images. This suggests that AsynDM has the potential to mitigate image distortions.

Asynchronous Diffusion Models for Enhancing Editing Performance. FLUX.1 Kontext is a DiT-based diffusion model that unifies image generation and editing (Labs et al., 2025). However, as shown in Figure 6 (b), even this advanced model can produce edits that mismatch the user prompts. By manually annotating the regions to be edited and applying the concave scheduler during the editing process, the resulting images align more closely with user expectations. This observation suggests that AsynDM has the potential to further enhance the performance of image editing models.

Limitations and Future Work. (1) In this work, we employ a fixed concave function to guide the transition of timestep states. A promising direction for future research is to replace this fixed function with a learnable model that can adaptively predict the next timestep state for each pixel (*e.g.*, Ye et al. (2025); Li et al. (2023b)), potentially leading to more flexible and accurate transitions. (2) We only distinguish between prompt-related and unrelated regions. A natural extension would be to capture more complex object relationships by sorting the objects or constructing a directed acyclic graph (Han et al., 2024; Kong et al., 2025). Assigning different objects with varying concave schedulers may further lead to improved performance. (3) When timestep states across pixels differ extremely, the faster denoised regions may be affected by noisy regions, causing the final image to retain a considerable amount of noise (See Appendix D.2 for an example). We attribute this limitation to the training-free nature of AsynDM, which makes it less robust to large disparities in noise levels. Future work could address this issue through fine-tuning or pre-training.

7 Conclusion

In this work, we propose the asynchronous denoising diffusion models to improve text-to-image alignment. The AsynDM allocates distinct timesteps to individual pixels and schedules them using a concave function. Guided by the masks that highlight the prompt-related regions, these regions can be denoised more slowly than unrelated ones, allowing them to receive clearer inter-pixel context. The clearer context can help the related regions better capture the content specified by the prompts, thereby generating more aligned images. Our empirical results demonstrate the effectiveness and robustness of the proposed asynchronous diffusion models.

Ethics Statement

This paper aims to advance the broader field of text-to-image alignment in diffusion models. While our method focuses on improving controllability and semantic faithfulness in generative models, it may have societal implications similar to those associated with image synthesis technologies. We do not identify any concerns unique to our approach that require special emphasis. For a more extensive discussion of the ethical considerations and broader impacts surrounding diffusion models and text-to-image generation, we refer interested readers to Po et al. (2024).

Reproducibility Statement

We have taken several measures to ensure the reproducibility of our work. The detailed experimental setting is provided in Section 5.1 of the main paper, and the Appendix B includes comprehensive implementation details, such as hyperparameters. To further ensure reproducibility, we provide pseudo-code that outlines the proposed method step by step in Appendix C.

References

Tomer Amit, Tal Shaharbany, Eliya Nachmani, and Lior Wolf. Segdiff: Image segmentation with diffusion probabilistic models. *arXiv preprint arXiv:2112.00390*, 2021.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

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

Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 22560–22570, 2023.

Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM transactions on Graphics (TOG)*, 42(4):1–10, 2023.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion, 2024. URL https://arxiv.org/abs/2303.04137.

Hyungjin Chung, Jeongsol Kim, Geon Yeong Park, Hyelin Nam, and Jong Chul Ye. CFG++: Manifold-constrained classifier free guidance for diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=E77uvbOTtp.

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

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

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

Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Reddy Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PUIqjT4rzq7.

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

Kairong Han, Kun Kuang, Ziyu Zhao, Junjian Ye, and Fei Wu. Causal agent based on large language model, 2024. URL https://arxiv.org/abs/2408.06849.

Kairong Han, Wenshuo Zhao, Ziyu Zhao, JunJian Ye, Lujia Pan, and Kun Kuang. Cat: Causal attention tuning for injecting fine-grained causal knowledge into large language models, 2025. URL https://arxiv.org/abs/2509.01535.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=XPZIaotutsD.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=_CDixzkzeyb.

Tobias Hinz, Stefan Heinrich, and Stefan Wermter. Semantic object accuracy for generative text-to-image synthesis. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1552–1565, 2020.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. URL https://arxiv.org/abs/2207.12598.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Susung Hong. Smoothed energy guidance: Guiding diffusion models with reduced energy curvature of attention. *Advances in Neural Information Processing Systems*, 37:66743–66772, 2024.

Zijing Hu, Fengda Zhang, Long Chen, Kun Kuang, Jiahui Li, Kaifeng Gao, Jun Xiao, Xin Wang, and Wenwu Zhu. Towards better alignment: Training diffusion models with reinforcement learning against sparse rewards. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 23604–23614, 2025a.

Zijing Hu, Fengda Zhang, and Kun Kuang. D-fusion: Direct preference optimization for aligning diffusion models with visually consistent samples. In *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=WVlEwFiDGH.

Lingjing Kong, Guangyi Chen, Biwei Huang, Eric P. Xing, Yuejie Chi, and Kun Zhang. Learning discrete concepts in latent hierarchical models, 2025. URL https://arxiv.org/abs/2406.00519.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space, 2025. URL https://arxiv.org/abs/2506.15742.

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

Alexander C Li, Mihir Prabhudesai, Shivam Duggal, Ellis Brown, and Deepak Pathak. Your diffusion model is secretly a zero-shot classifier. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2206–2217, 2023a.

Lijiang Li, Huixia Li, Xiawu Zheng, Jie Wu, Xuefeng Xiao, Rui Wang, Min Zheng, Xin Pan, Fei Chao, and Rongrong Ji. Autodiffusion: Training-free optimization of time steps and architectures for automated diffusion model acceleration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7105–7114, 2023b.

Yumeng Li, Margret Keuper, Dan Zhang, and Anna Khoreva. Divide & bind your attention for improved generative semantic nursing. *arXiv preprint arXiv:2307.10864*, 2023c.

Wentong Liao, Kai Hu, Michael Ying Yang, and Bodo Rosenhahn. Text to image generation with semantic-spatial aware gan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18187–18196, 2022.

Bai LiChen, Shitong Shao, zikai zhou, Zipeng Qi, zhiqiang xu, Haoyi Xiong, and Zeke Xie. Zigzag diffusion sampling: Diffusion models can self-improve via self-reflection. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=MKvQH1ekeY.

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 *European conference on computer vision*, pp. 740–755. Springer, 2014.

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.

Oscar Mañas, Pietro Astolfi, Melissa Hall, Candace Ross, Jack Urbanek, Adina Williams, Aishwarya Agrawal, Adriana Romero-Soriano, and Michal Drozdzal. Improving text-to-image consistency via automatic prompt optimization. *arXiv preprint arXiv:2403.17804*, 2024.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models, 2025. URL https://arxiv.org/abs/2502.09992.

Yong-Hyun Park, Mingi Kwon, Jaewoong Choi, Junghyo Jo, and Youngjung Uh. Understanding the latent space of diffusion models through the lens of riemannian geometry. *Advances in Neural Information Processing Systems*, 36:24129–24142, 2023.

I. Pavlov, A. Ivanov, and S. Stafievskiy. Text-to-Image Benchmark: A benchmark for generative models. https://github.com/boomb0om/text2image-benchmark, September 2023. Version 0.1.0.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Ryan Po, Wang Yifan, Vladislav Golyanik, Kfir Aberman, Jonathan T Barron, Amit Bermano, Eric Chan, Tali Dekel, Aleksander Holynski, Angjoo Kanazawa, et al. State of the art on diffusion models for visual computing. In *Computer graphics forum*, volume 43, pp. e15063. Wiley Online Library, 2024.

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, 2023. URL https://arxiv.org/abs/2307.01952.

Alec Radford, Jong Wook Kim, Chris Hallacy, A. Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. URL https://arxiv.org/abs/2204.06125.

Royi Rassin, Eran Hirsch, Daniel Glickman, Shauli Ravfogel, Yoav Goldberg, and Gal Chechik. Linguistic binding in diffusion models: Enhancing attribute correspondence through attention map alignment. *Advances in Neural Information Processing Systems*, 36:3536–3559, 2023.

Severi Rissanen, Markus Heinonen, and Arno Solin. Generative modelling with inverse heat dissipation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=4PJUBT9f2Ol.

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

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

Dazhong Shen, Guanglu Song, Zeyue Xue, Fu-Yun Wang, and Yu Liu. Rethinking the spatial inconsistency in classifier-free diffusion guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9370–9379, 2024.

Kihyuk Sohn, Nataniel Ruiz, Kimin Lee, Daniel Castro Chin, Irina Blok, Huiwen Chang, Jarred Barber, Lu Jiang, Glenn Entis, Yuanzhen Li, et al. Styledrop: Text-to-image generation in any style. *arXiv preprint arXiv:2306.00983*, 2023.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. URL https://arxiv.org/abs/2010.02502.

Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Türe. What the daam: Interpreting stable diffusion using cross attention. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5644–5659, 2023.

Yunze Tong, Fengda Zhang, Zihao Tang, Kaifeng Gao, Kai Huang, Pengfei Lyu, Jun Xiao, and Kun Kuang. Latent score-based reweighting for robust classification on imbalanced tabular data. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025a.

Yunze Tong, Fengda Zhang, Didi Zhu, Jun Xiao, and Kun Kuang. Decoding correlation-induced misalignment in the stable diffusion workflow for text-to-image generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2025b.

Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1921–1930, 2023. doi: 10.1109/CVPR52729.2023.00191.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Chen Wang, Hao-Yang Peng, Ying-Tian Liu, Jiatao Gu, and Shi-Min Hu. Diffusion models for 3d generation: A survey. *Computational Visual Media*, 11(1):1–28, 2025. doi: 10.26599/CVM.2025.9450452.

Hao Wang, Guosheng Lin, Steven CH Hoi, and Chunyan Miao. Cycle-consistent inverse gan for text-to-image synthesis. In *Proceedings of the 29th ACM international conference on multimedia*, pp. 630–638, 2021.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Yunlong Wang, Shuyuan Shen, and Brian Y Lim. Reprompt: Automatic prompt editing to refine ai-generative art towards precise expressions. In *Proceedings of the 2023 CHI conference on human factors in computing systems*, pp. 1–29, 2023a.

Zhendong Wang, Yifan Jiang, Huangjie Zheng, Peihao Wang, Pengcheng He, Zhangyang Wang, Weizhu Chen, Mingyuan Zhou, et al. Patch diffusion: Faster and more data-efficient training of diffusion models. *Advances in neural information processing systems*, 36:72137–72154, 2023b.

Rosa Wolf, Yitian Shi, Sheng Liu, and Rania Rayyes. Diffusion models for robotic manipulation: A survey, 2025. URL `https://arxiv.org/abs/2504.08438`.

Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion GANs. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=JprM0p-q0Co`.

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

Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation. *Entropy*, 25(10):1469, 2023.

Zilyu Ye, Zhiyang Chen, Tiancheng Li, Zemin Huang, Weijian Luo, and Guo-Jun Qi. Schedule on the fly: Diffusion time prediction for faster and better image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23412–23422, June 2025.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6613–6623, 2024.

Chenrui Zhang and Yuxin Peng. Stacking vae and gan for context-aware text-to-image generation. In *2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM)*, pp. 1–5. IEEE, 2018.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert, 2020. URL `https://arxiv.org/abs/1904.09675`.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2025. URL `https://arxiv.org/abs/2303.18223`.

Guangcong Zheng, Xianpan Zhou, Xuewei Li, Zhongang Qi, Ying Shan, and Xi Li. Layoutdiffusion: Controllable diffusion model for layout-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22490–22499, 2023.

The **Appendix** is organized as follows:

- **Appendix A:** provides the proof of the proposition in the main text and the formulation of the DDIM sampler.
- **Appendix B:** provides more details on implementation.
- **Appendix C:** provides the pseudo-code of employing AsynDM to generate images.
- **Appendix D:** presents more experimental results.
- **Appendix E:** presents more image samples generated by AsynDM.
- **Appendix F:** describes the **role of the large language models (LLMs)** in preparing this paper.

A Theoretical Derivations

A.1 Proof of Proposition 1

From the second equation in Eq.(6) we obtain $b = -f(T-a)$. Substituting into the first equation yields the single-variable condition $f(i_0 - a) - f(T-a) = t_0$. Define:

$$g(a) = f(i_0 - a) - f(T-a), \qquad a \in [0, i_0]. \tag{8}$$

The domain $[0, i_0]$ ensures that both $i_0 - a$ and $T - a$ lie in $[0, T]$.

Since f is concave on $[0, T]$, then f is continuous, hence g is continuous on $[0, i_0]$. Moreover, concavity implies that the slope of f is nonincreasing, which in turn gives:

$$g'(a) = f'(T-a) - f'(i_0 - a) \leq 0, \tag{9}$$

whenever f is differentiable. Therefore, g is nonincreasing on $[0, i_0]$, and strictly decreasing unless f is linear.

At the endpoints, we have:

$$g(0) = f(i_0) - f(T) = f(i_0), \qquad g(i_0) = f(0) - f(T - i_0) = T - f(T - i_0). \tag{10}$$

Therefore, the range of g is exactly the interval $[T - f(T - i_0), f(i_0)]$.

Moreover, since f is concave on $[0, T]$, then:

$$f(T - i_0) = f(\frac{i_0}{T} \cdot 0 + \frac{T - i_0}{T} \cdot T) \geq \frac{i_0}{T} \cdot f(0) + \frac{T - i_0}{T} \cdot f(T) = i_0. \tag{11}$$

Hence $T - f(T - i_0) \leq T - i_0$.

According to the *intermediate value theorem*, for any $t_0 \in [T - i_0, f(i_0)]$, there exists some $a \in [0, i_0]$, such that $g(a) = t_0$. Monotonicity of g guarantees that this solution is unique. Finally, since a is uniquely determined, then $b = -f(T-a)$ is also uniquely determined.

Therefore, the constants a, b exist and are unique.

A.2 Asynchronous Denoising with DDIM Sampler

The vanilla DDIM sampler predicts next intermediate state $\mathbf{x}_{t-1}$ according to:

$$\mathbf{x}_{t-1} = \sqrt{\alpha_{t-1}} \cdot \hat{\mathbf{x}}_0 + \sqrt{1 - \alpha_{t-1} - \sigma_t^2} \cdot \epsilon_\theta(\mathbf{x}_t, t, \mathbf{c}) + \sigma_t \epsilon_t, \tag{12}$$

$$\text{with } \hat{\mathbf{x}}_0 = \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t - \sqrt{1 - \alpha_t} \cdot \epsilon_\theta(\mathbf{x}_t, t, \mathbf{c})), \tag{13}$$

where $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The pixel-level timestep formulation of the DDIM sampler is given as follow:

$$\mathbf{x}_{i+1} = \sqrt{\alpha_{\mathbf{t}_{i+1}}} \cdot \hat{\mathbf{x}}_0 + \sqrt{1 - \alpha_{\mathbf{t}_{i+1}} - \sigma_i^2} \cdot \epsilon_\theta(\mathbf{x}_i, \mathbf{t}_i, \mathbf{c}) + \sigma_i \epsilon_i, \tag{14}$$

$$\text{with } \hat{\mathbf{x}}_0 = \frac{1}{\sqrt{\alpha_{\mathbf{t}_i}}}(\mathbf{x}_i - \sqrt{1 - \alpha_{\mathbf{t}_i}} \cdot \epsilon_\theta(\mathbf{x}_i, \mathbf{t}_i, \mathbf{c})), \tag{15}$$

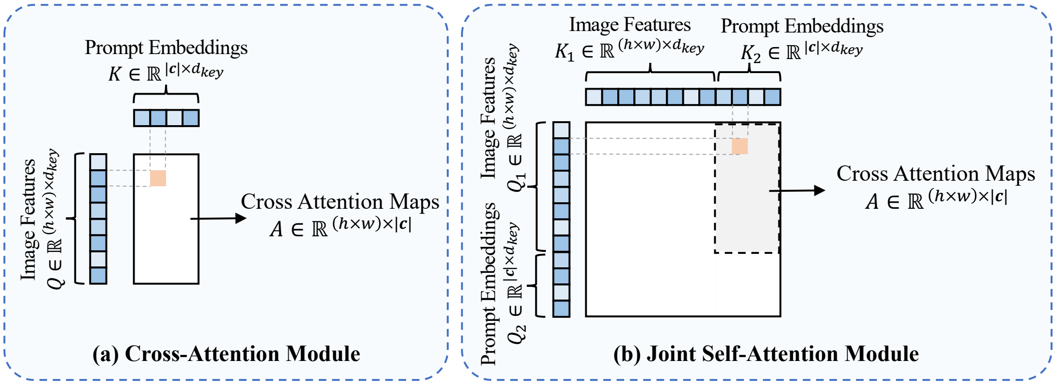

Prompt Embeddings
$K \in \mathbb{R}^{|c| \times d_{key}}$
Image Features
$Q \in \mathbb{R}^{(h \times w) \times d_{key}}$
Cross Attention Maps
$A \in \mathbb{R}^{(h \times w) \times |c|}$
(a) Cross-Attention Module
Image Features
$K_1 \in \mathbb{R}^{(h \times w) \times d_{key}}$
Prompt Embeddings
$K_2 \in \mathbb{R}^{|c| \times d_{key}}$
Image Features
$Q_1 \in \mathbb{R}^{(h \times w) \times d_{key}}$
Prompt Embeddings
$Q_2 \in \mathbb{R}^{|c| \times d_{key}}$
Cross Attention Maps
$A \in \mathbb{R}^{(h \times w) \times |c|}$
(b) Joint Self-Attention Module

Figure 7: Extracting cross-attention masks from attention modules.

B Implementation Details

B.1 Implementation Details of AsynDM

Mask Extraction. In Section 4, we have described how to extract prompt-related regions from cross-attention maps. However, a model typically contains multiple cross-attention layers, each producing its own set of attention maps. For DiT-based diffusion models, we average the cross-attention maps across all layers and then extract the mask following the procedure outlined in Section 4. In contrast, UNet-based diffusion models comprise layers with varying spatial resolutions. Let $h \times w$ represent the image resolution of $\mathbf{x}_t$, and $h_l \times w_l$ represent the resolution at layer l of the UNet. Inspired by prior work (Hertz et al., 2023; Cao et al., 2023), we only use the cross-attention maps from layers at resolution $h_l \times w_l = \frac{h}{4} \times \frac{w}{4}$. The maps from these layers are averaged to obtain the mask, and subsequently upsampled to the resolution $h \times w$.

Scheduler Reweighting. As shown in Appendix D.2, when timestep states across pixels differ extremely, the prompt-unrelated regions in the final image might retain a considerable amount of noise. Therefore, constraining the maximum disparity of timestep states across pixels is fundamental to ensuring that any concave function can be reliably applied for denoising. To achieve this, we adopt a straightforward yet effective strategy by weighting the concave function f with the standard denoising function g (*e.g.*, the linear function). Consequently, the concave function employed for state transitions becomes $f' = \omega \cdot f + (1 - \omega) \cdot g$, where $\omega \in (0, 1)$. The function f' not only retains the concavity, but also mitigates its maximum disparity with respect to the standard function.

B.2 Details of Human Evaluation

The human study was conducted with 52 participants from eight universities, including 23 females and 29 males, whose academic backgrounds ranged from undergraduates to Ph.D. students. Each participant was asked to complete a form. At the beginning of the form, we provided the following instruction: "*Image–prompt alignment refers to how well an image matches the given textual description. For each group of three images, please select the one you believe best matches the given textual description.*" The form contained 64 groups of images in total, corresponding to four prompt sets, each comprising 16 groups. These 64 groups were randomly selected from the images sampled during the evaluation of results reported in Table 1. Each group consisted of three images generated by DM, $\text{DM}_{\text{concave}}$ and AsynDM under the same random seed, accompanied by the corresponding text prompt used for generation. All participants received the same set of images, but the presentation order was randomized, ensuring that participants were unaware of which method each image originated from. The entire form was presented in English.

B.3 Extracting Cross-attention Masks from DiT-based Models

As shown in Figure 7 (a), in the cross-attention modules, we first obtain the cross-attention maps directly via $A = \text{softmax}(\frac{QK^\top}{\sqrt{d_{\text{key}}}})$, and subsequently derive the corresponding masks using Eq.(7).

However, DiT-based diffusion models typically do not include dedicated cross-attention modules. Instead, they rely on implicit cross-attention computation within the self-attention modules to enable the image to be guided by the prompt. As illustrated in Figure 7 (b), the queries Q and keys K are formed by concatenating the image features with the prompt embeddings. During the attention operation, the resulting attention maps A_{joint} has a size of $(h \times w + |\mathbf{c}|) \times (h \times w + |\mathbf{c}|)$. By extracting the submatrix corresponding to the interactions between the image-feature queries and the prompt-embedding keys, we can obtain the cross-attention maps (*i.e.*, $A = A_{\text{joint}}[: (h \times w), (h \times w) :]$). The cross-attention masks are then computed using Eq.(7) [2].

B.4 Prompt for Qwen

We employ Qwen2.5-VL-7B-Instruct (Wang et al., 2024) to score text-to-image alignment with the following prompt: "*You are given an image and a description. Please evaluate how well the image matches the description on a scale from 0 to 9, where 0 means completely unrelated and 9 means perfectly aligned. Return only the score as a single integer without explanation.*\n *Description: [prompt used to generate the image]*".

B.5 Experimental Resources

The experiments were conducted on 24GB NVIDIA 3090 GPUs. It tooks approximately 78 minutes for the vanilla diffusion model (SD2.1-512-base) to generate 1,280 images, and approximately 86 minutes for the asynchronous diffusion model.

B.6 Hyperparameters

The full hyperparameter list of our experiments is presented in Table 3.

Table 3: Hyperparameters of our experiments.

	Patameter	**Value**
Sampling	Denoising steps T	50
	Noise weight η	1.0
	Classifier-free guidance	True
	Guidance scale	5.0
	Batch size	8
	Batch count	160
Z-Sampling	Inversion guidance γ_2	0.0
	Zigzag steps	49
	Number of rounds T_{max}	1
SEG	SEG guidance γ_{seg}	3.0
	Blurred weight σ	1.0
CFG++	CFG++ guidance λ	0.4

C Pseudo-Code

The pseudo-code of employing the asynchronous diffusion model to generate text-aligned images is shown in Algorithm 1.

[2] The cross-attention maps A has a size of $(h \times w) \times |\mathbf{c}|$. The dimension $|\mathbf{c}| \times h \times w$ mentioned in the main text corresponds to its transposed and reshaped form, which is presented to facilitate clearer understanding for the readers.

Algorithm 1: Pseudo-code of employing the asynchronous diffusion model to generate text-aligned images.

```
Input  : Total denoising timesteps $T$, number of samples $N$, prompt list $C$, pre-trained
         diffusion model $\epsilon_\theta$, linear/standard scheduler $g$, concave scheduler $f$.
$D_{sample} = [\,]$ ;
for $n \leftarrow 0$ to $N - 1$ do
    $\mathbf{c} \leftarrow C_n$ ;
    // Initialize $\mathbf{x}_i$, $\mathbf{t}_i$ and $M$
    Randomly choose $\mathbf{x}_0$ from $\mathcal{N}(\mathbf{0}, \mathbf{I})$ ;
    $\mathbf{t}_0 \leftarrow \text{tensor}(\text{shape}(\mathbf{x}_0), \text{fill} = T)$ ;
    $M \leftarrow \text{tensor}(\text{shape}(\mathbf{x}_0), \text{fill} = 1)$ ;
    for $i \leftarrow 0$ to $T - 1$ do
        // Transition of $\mathbf{t}_i$
        $\mathbf{t}_{i+1}^{lin} \leftarrow$ Calculate the next state of $\mathbf{t}_i$ using $g$ ;
        $\mathbf{t}_{i+1}^{con} \leftarrow$ Calculate the next state of $\mathbf{t}_i$ using $f$ ;
        $\mathbf{t}_{i+1} \leftarrow M \times \mathbf{t}_{i+1}^{con} + (1 - M) \times \mathbf{t}_{i+1}^{lin}$;
        // Transition of $\mathbf{x}_i$
        $\epsilon \leftarrow \epsilon_\theta(\mathbf{x}_i, \mathbf{t}_i, \mathbf{c})$, and extract the cross-attention map $A$ ;
        Calculate $\mathbf{x}_{i+1}$ according to the chosen sampler (e.g., Eq.(4) for DDPM) ;
        // Update $M$
        Update $M$ using Eq.(7) ;
    end
    $D_{sample}$.append($\mathbf{x}_T$) ;
end
Output: $D_{sample}$
```

D More Experimental Results

D.1 Experiments on SDXL and SD3.5

We also quantitatively demonstrate the text-to-image alignment performance of AsynDM compared with baseline methods on SDXL and SD 3.5, as shown in Table 4 and Table 5 respectively. For experiments conducted on SD 3.5, we have not included comparisons with Z-Sampling or CFG++. This is because Z-Sampling relies on DDIM inversion, and CFG++ makes modifications to DDIM. However, SD 3.5 is a flow model that is not directly compatible with the DDIM sampler. The experimental results demonstrate that AsynDM consistently achieves better alignment across all prompt sets. The image samples for these experiments are shown in Figure 12 and Figure 13.

Table 4: Text-to-image alignment performance of AsynDM compared with baseline methods on animal activity prompt set. The base model is SDXL-base-1.0 (Podell et al., 2023).

Prompt Set	Method	BERTScore↑	CLIPScore↑	ImageReward↑	QwenScore↑
Animal Activity	DM	0.6671	0.3976	1.6552	6.6562
	DM$_{\text{concave}}$	0.6695 (+0.0024)	0.3993 (+0.0017)	1.6768 (+0.0216)	6.8421 (+0.1859)
	Z-Sampling	0.6674 (+0.0003)	0.4022 (+0.0046)	1.6677 (+0.0125)	6.7320 (+0.0758)
	SEG	0.6673 (+0.0002)	0.3963 (-0.0013)	1.6417 (-0.0135)	6.8085 (+0.1523)
	S-CFG	0.6670 (-0.0001)	0.3981 (+0.0005)	1.6481 (-0.0071)	6.6367 (-0.0195)
	CFG++	0.6581 (-0.0090)	0.3879 (-0.0097)	1.3748 (-0.2804)	6.4078 (-0.2484)
	AsynDM	**0.6829 (+0.0158)**	**0.4026 (+0.0050)**	**1.6893 (+0.0341)**	**7.2781 (+0.6219)**

D.2 Ablation on Maximum Timestep Difference

Given an extreme concave scheduler $f(i) = \min(T, 2T - 2i)$ and a standard linear scheduler $g(i) = T - i$, the maximum timestep difference between pixels within the same denoising step can reach $\frac{T}{2}$. By interpolating the two schedulers as $f' = \omega \cdot f + (1 - \omega) \cdot g$, we obtain a concave scheduler whose

Table 5: Text-to-image alignment performance of AsynDM compared with baseline methods on animal activity prompt set. The base model is SD3.5-medium (Esser et al., 2024).

Prompt Set	Method	BERTScore↑	CLIPScore↑	ImageReward↑	QwenScore↑
Animal Activity	DM	0.6590	0.3906	1.5091	6.8812
	$\text{DM}_{\text{concave}}$	0.6603 (+0.0013)	0.3928 (+0.0022)	1.6385 (+0.1294)	7.0656 (+0.1844)
	SEG	0.6570 (-0.0020)	0.3740 (-0.0166)	1.4022 (-0.1069)	7.1250 (+0.2438)
	S-CFG	0.6629 (+0.0039)	0.3908 (+0.0002)	1.6227 (+0.1136)	7.0125 (+0.1313)
	AsynDM	**0.6663 (+0.0073)**	**0.3941 (+0.0035)**	**1.6418 (+0.1327)**	**7.2171 (+0.3359)**

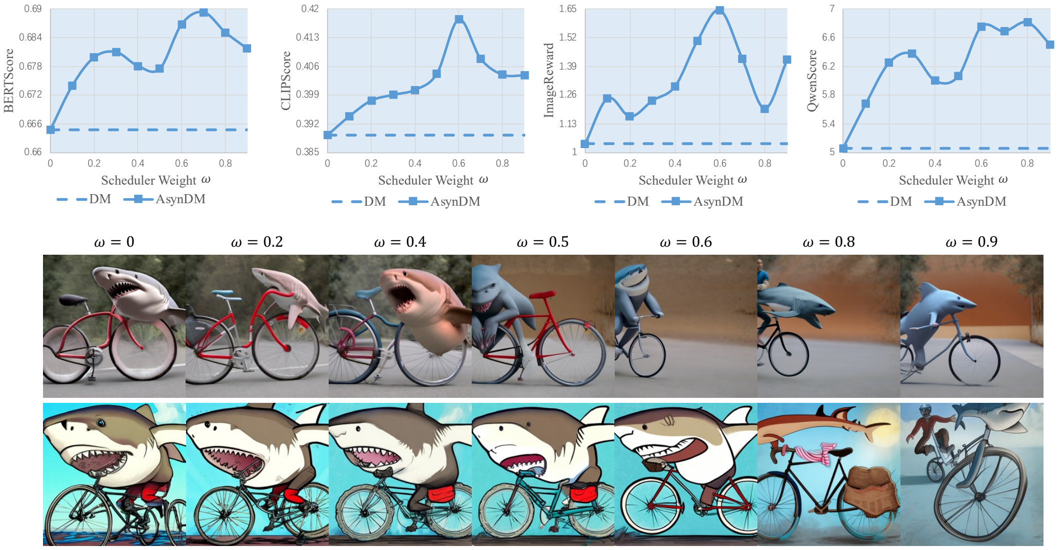


Figure 8: As ω increases, the maximun timestep difference increases, and the alignment first improves and then degrades. The extreme differences cause faster-denoised regions to retain noise for contextual consistency, leading to blurry and noisy background in final images (*e.g.*, $\omega = 0.8, 0.9$).

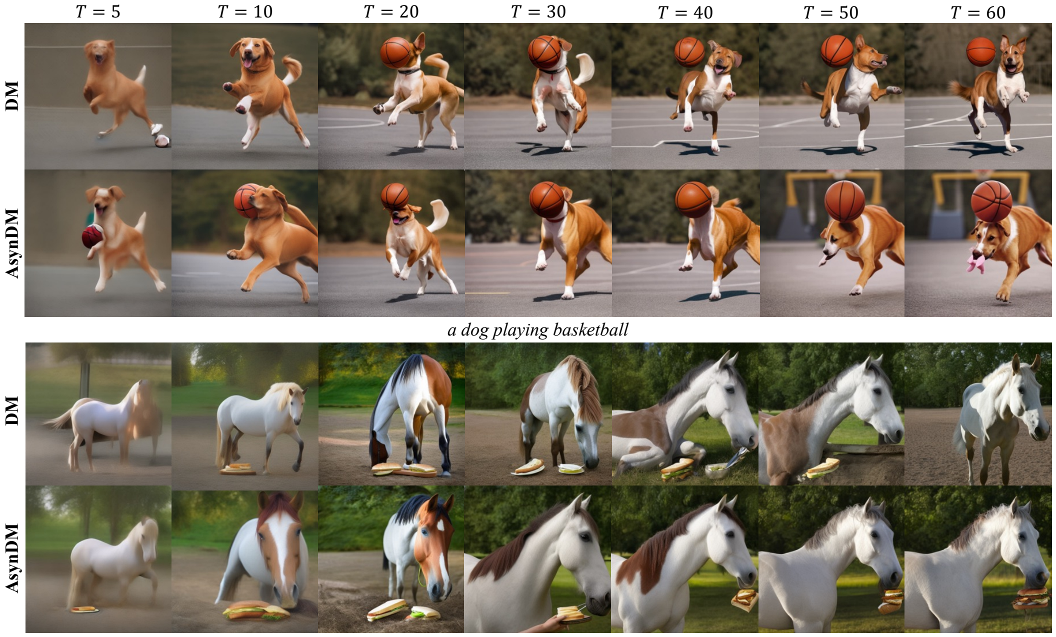
maximum timestep difference can be flexibly controlled. As a case study, we consider the prompt "*a shark riding a bike*", and sample 32 images for each value of ω to evaluate text-to-image alignment. As shown in Figure 8, the results indicate that as ω increases (*i.e.*, the maximum timestep difference increases), the alignment first improves and then degrades. The degradation occurs because, when timestep states across pixels differ extremely, the faster denoised regions may be affected by noisy regions, which continue to provide noisy context even at later denoising steps. Consequently, these faster denoised regions tend to preserve a considerable amount of noise in order to remain consistent with the context. This effect is particularly evident at $\omega = 0.8$ and $\omega = 0.9$, where the generated images exhibit blurry and noisy background regions.

D.3 Ablation on Denoising Steps

A growing body of work has focused on enabling diffusion models to generate high-quality images with only a small number of denoising steps (Xiao et al., 2022; Yin et al., 2024). Motivated by this line of research, we further evaluate the performance of AsynDM under different total denoising steps T. Specifically, we set the steps T to 5, 10, 20, 30, 40, 50 and 60, and generate 1,280 images for each setting on the animal activity prompt set. The results are summarized in Table 6. Across all denoising-step configurations, AsynDM consistently improves text-to-image alignment. Figure 9 provides some examples. These results further demonstrate the effectiveness of our method.

Table 6: Alignment performance of AsynDM for different denoising steps T, across prompts on animal activity prompt set. The base model is SD2.1-512-base.

Metric	Method	$T = 5$	$T = 10$	$T = 20$	$T = 30$	$T = 40$	$T = 50$	$T = 60$
BERTScore↑	DM	0.5924	0.6221	0.6311	0.6330	0.6371	0.6353	0.6346
	AsynDM	**0.5987**	**0.6280**	**0.6364**	**0.6372**	**0.6402**	**0.6414**	**0.6412**
CLIPScore↑	DM	0.3111	0.3574	0.3681	0.3672	0.3689	0.3685	0.3691
	AsynDM	**0.3260**	**0.3636**	**0.3708**	**0.3699**	**0.3729**	**0.3750**	**0.3752**
ImageReward↑	DM	-1.0882	0.0926	0.5801	0.6458	0.7606	0.7543	0.7668
	AsynDM	**-0.7556**	**0.3087**	**0.6732**	**0.7561**	**0.8692**	**0.9219**	**0.9123**
QwenScore↑	DM	2.8703	4.0750	4.7148	4.7359	4.8992	4.9445	4.9382
	AsynDM	**3.3164**	**4.5718**	**4.9976**	**5.0218**	**5.2859**	**5.5218**	**5.3234**



a dog playing basketball

a horse eating a sandwich

Figure 9: Samples generated by AsynDM compared with DM for different denoising steps T. The base model is SD2.1-512-base.

D.4 Standard Deviations of Quantitative Results

Here we provide the standard deviations of the results reported in Table 1, Table 2, Table 4 and Table 5, as shown in Table 7, Table 8, Table 9 and Table 10 respectively.

E More Samples

In this section, we present additional samples generated by AsynDM, alongside those from baseline methods. Specifically, Figure 10 presents more samples on SD 2.1 across diverse prompts. Figure 11 presents the samples of the ablation studies in Section 5.4. Figure 12 and Figure 13 present the samples on SDXL and SD 3.5, respectively.

F Declaration of LLM Usage

In preparing this manuscript, we used the large language model (LLM) as a general-purpose writing assistant. Specifically, the LLM was employed to (1) check grammar and correctness of the text, and (2) suggest more natural and fluent wording. When using the LLM, we first wrote an initial draft of the sentence, and then asked the LLM to check and polish it. The LLM did not contribute to research ideas, methods, experiments, or results. The authors take full responsibility for the content of this paper.

Table 7: Standard deviations of Table 1.

Prompt Set	Method	BERTScore↑	CLIPScore↑	ImageReward↑	QwenScore↑
Animal Activity	DM	0.0321	0.0391	0.9657	2.6160
	$\text{DM}_{\text{concave}}$	0.0323	0.0383	0.9300	2.6090
	Z-Sampling	0.0312	0.0385	0.9179	2.4824
	SEG	0.0333	0.0435	1.0453	2.7407
	S-CFG	0.0330	0.0370	0.9149	2.5524
	CFG++	0.0318	0.0466	1.1062	2.7309
	AsynDM	0.0322	0.0334	0.8611	2.4162
Drawbench	DM	0.0226	0.0354	0.7256	1.9625
	$\text{DM}_{\text{concave}}$	0.0234	0.0364	0.7397	2.0889
	Z-Sampling	0.0228	0.0340	0.7208	1.9152
	SEG	0.0243	0.0384	0.7783	2.2053
	S-CFG	0.0237	0.0347	0.7138	1.8801
	CFG++	0.0253	0.0360	0.7828	2.0983
	AsynDM	0.0232	0.0324	0.7045	1.8936
GenEval	DM	0.0238	0.0396	0.8833	2.4434
	$\text{DM}_{\text{concave}}$	0.0242	0.0398	0.9051	2.4122
	Z-Sampling	0.0245	0.0397	0.8905	2.4414
	SEG	0.0248	0.0423	0.9348	2.4923
	S-CFG	0.0245	0.0402	0.9012	2.4150
	CFG++	0.0249	0.0427	0.9148	2.3647
	AsynDM	0.0244	0.0383	0.8702	2.3426
MSCOCO	DM	0.0267	0.0291	0.6696	2.1744
	$\text{DM}_{\text{concave}}$	0.0266	0.0287	0.6403	2.1839
	Z-Sampling	0.0260	0.0297	0.6771	2.1776
	SEG	0.0262	0.0303	0.6933	2.1959
	S-CFG	0.0268	0.0284	0.6551	2.1360
	CFG++	0.0268	0.0307	0.7182	2.2440
	AsynDM	0.0265	0.0274	0.6323	2.0151

Table 8: Standard deviations of Table 2.

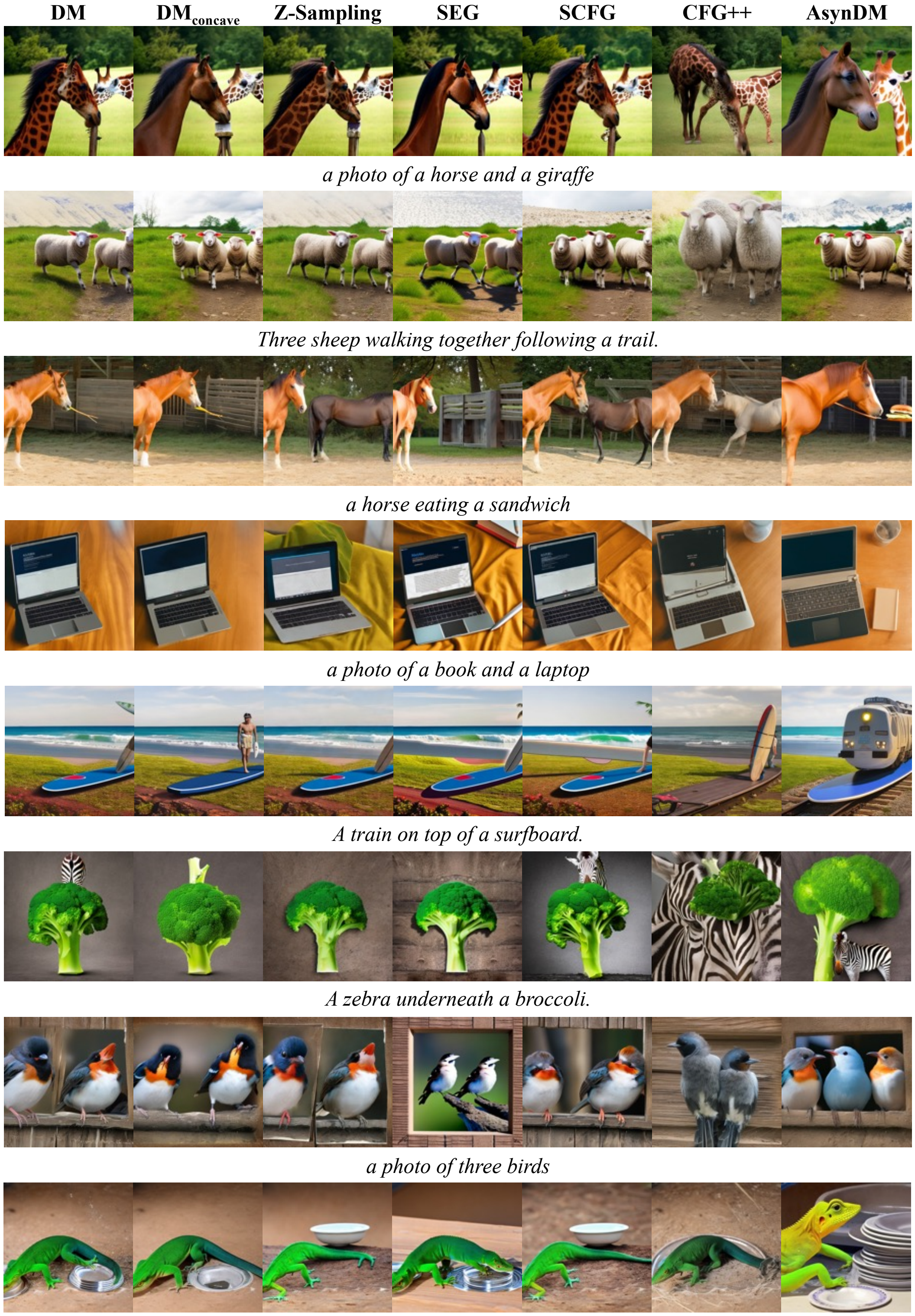
Scheduler	Method	BERTScore↑	CLIPScore↑	ImageReward↑	QwenScore↑
	DM	0.0321	0.0391	0.9657	2.6160
Quadratic	$\text{DM}_{\text{concave}}$	0.0323	0.0383	0.9300	2.6090
	AsynDM	0.0322	0.0334	0.8611	2.4162
	+*fixed mask*	0.0314	0.0385	0.9489	2.5205
Piecewise Linear	$\text{DM}_{\text{concave}}$	0.0317	0.0391	0.9282	2.5570
	AsynDM	0.0317	0.0376	0.9394	2.5430
	+*fixed mask*	0.0319	0.0399	0.9916	2.5752
Exponential	$\text{DM}_{\text{concave}}$	0.0320	0.0385	0.9643	2.6149
	AsynDM	0.0320	0.0382	0.9422	2.5430
	+*fixed mask*	0.0327	0.0400	0.9799	2.5657

Table 9: Standard deviations of Table 4.

Prompt Set	Method	BERTScore↑	CLIPScore↑	ImageReward↑	QwenScore↑
Animal Activity	DM	0.0258	0.0160	0.3061	1.6000
	$\text{DM}_{\text{concave}}$	0.0280	0.0159	0.2814	1.5290
	Z-Sampling	0.0272	0.0151	0.2961	1.5937
	SEG	0.0289	0.0175	0.3480	1.5495
	S-CFG	0.0265	0.0159	0.3171	1.6653
	CFG++	0.0293	0.0230	0.5516	1.7666
	AsynDM	0.0268	0.0163	0.3027	1.5785

Table 10: Standard deviations of Table 5.

Prompt Set	Method	BERTScore↑	CLIPScore↑	ImageReward↑	QwenScore↑
Animal Activity	DM	0.0282	0.0205	0.4052	1.4930
	$\text{DM}_{\text{concave}}$	0.0289	0.0185	0.3438	1.3569
	SEG	0.0343	0.0252	0.6220	1.4515
	S-CFG	0.0283	0.0175	0.3192	1.3775
	AsynDM	0.0288	0.0198	0.3941	1.3657



a photo of a horse and a giraffe

Three sheep walking together following a trail.

a horse eating a sandwich

a photo of a book and a laptop

A train on top of a surfboard.

A zebra underneath a broccoli.

a photo of three birds

a lizard washing dishes

Figure 10: More samples generated by AsynDM compared with baseline methods. The images sampled by AsynDM show higher text-to-image alignment. The base model used to sample these images is SD2.1-512-base.

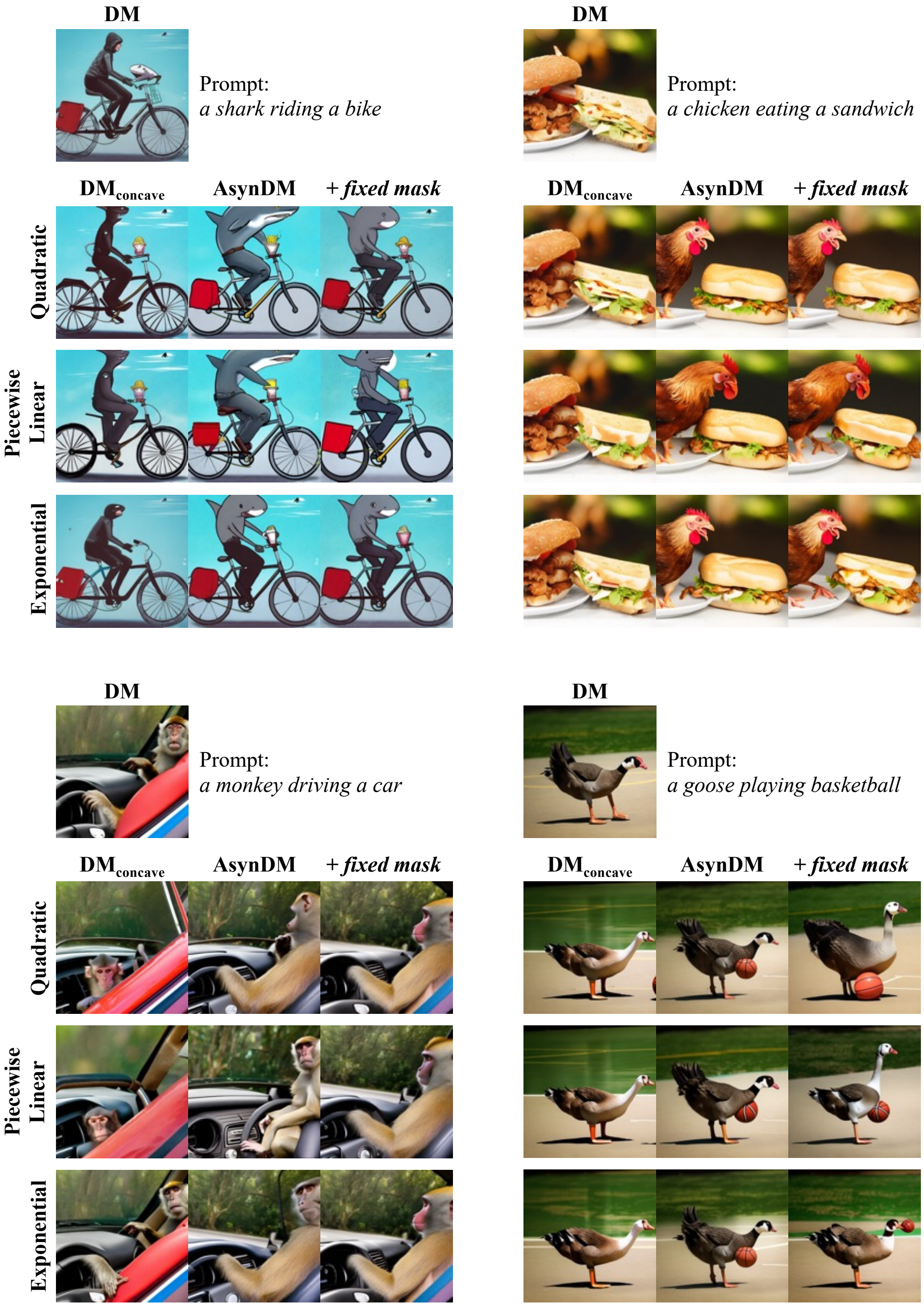
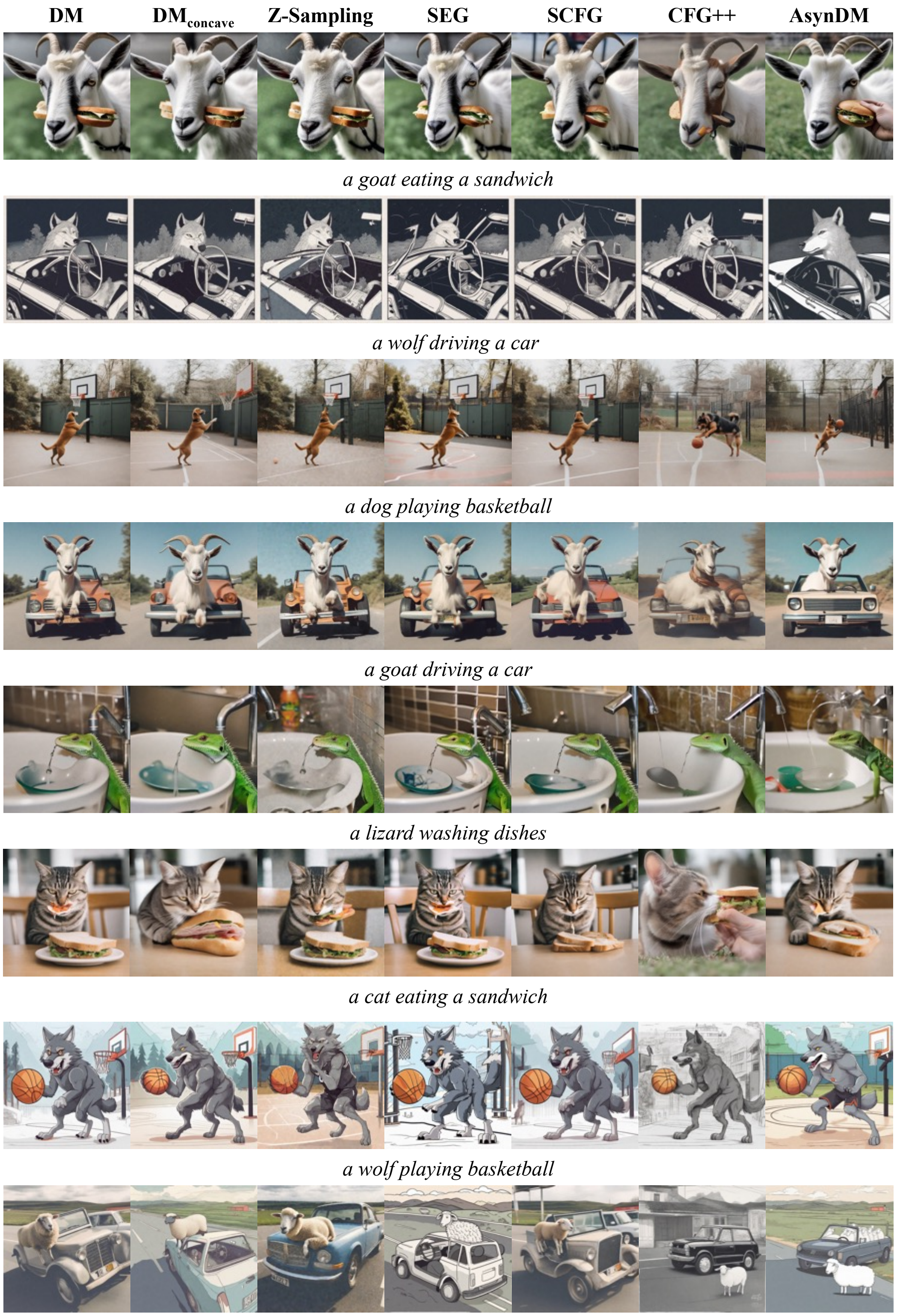


Figure 11: Samples generated by AsynDM when employing different concave schedulers and using fixed masks. The base model used to sample these images is SD2.1-512-base.



a goat eating a sandwich

a wolf driving a car

a dog playing basketball

a goat driving a car

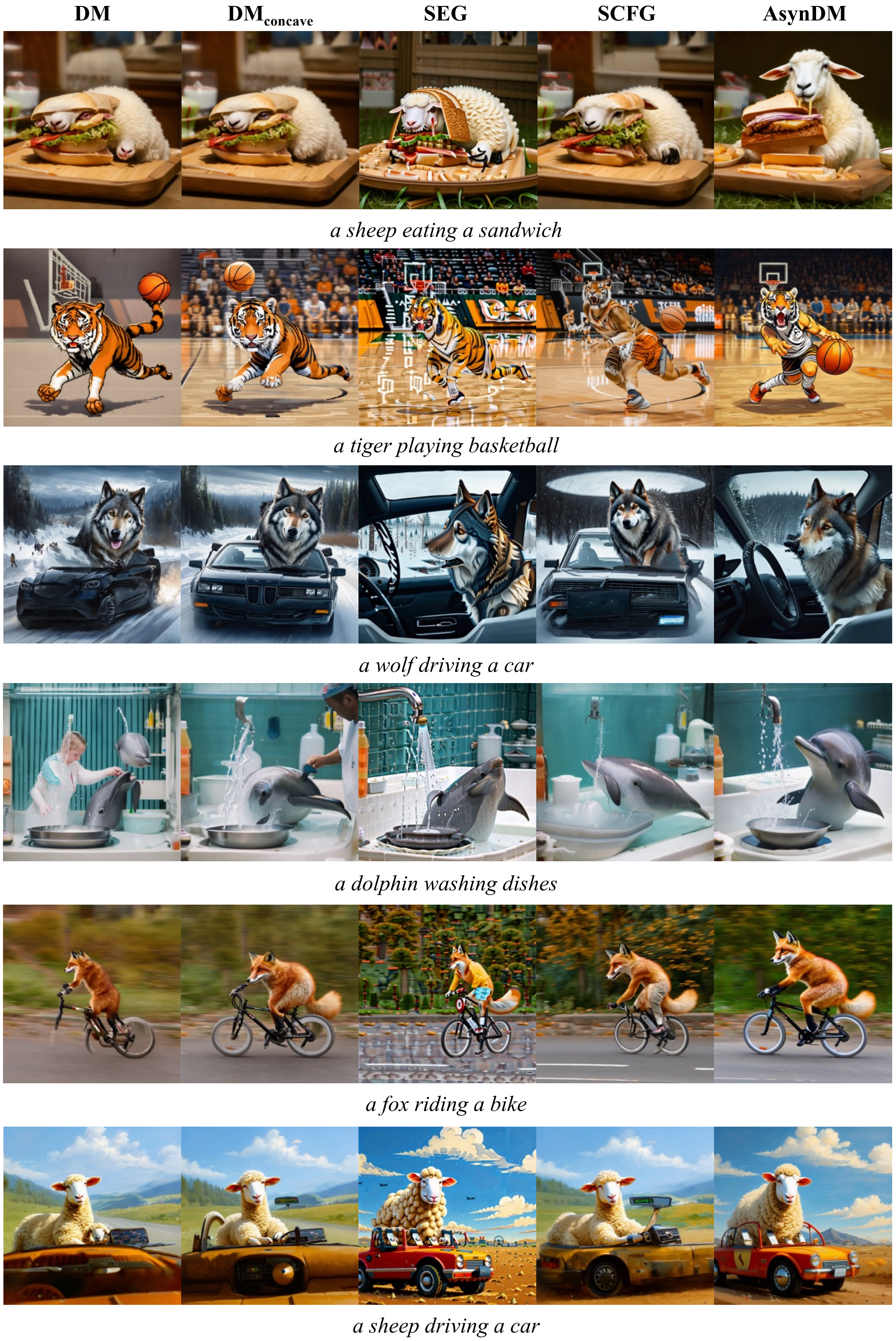
a lizard washing dishes

a cat eating a sandwich

a wolf playing basketball

a sheep driving a car

Figure 12: Samples generated by AsynDM compared with baseline methods when using SDXL-base-1.0. The images sampled by AsynDM show higher text-to-image alignment.



a sheep eating a sandwich

a tiger playing basketball

a wolf driving a car

a dolphin washing dishes

a fox riding a bike

a sheep driving a car

Figure 13: Samples generated by AsynDM compared with baseline methods when using SD3.5-medium. The images sampled by AsynDM show higher text-to-image alignment.