Accelerated Likelihood Maximization for Diffusion-based Versatile Content Generation

Hyunsoo Lee[1], Inwoo Hwang[1], and Young Min Kim[1,2†]

[1] ECE, Seoul National University
[2] INMC & IPAI, Seoul National University
{philip21,inusu0818,youngmin.kim}@snu.ac.kr

Abstract. Generating diverse, coherent, and plausible content from partially given inputs remains a fundamental challenge for diffusion models. Existing approaches face clear limitations: training-based approaches offer strong task-specific results but require costly computation, and they generalize poorly across tasks. Training-free approaches offer better efficiency, but they do not explicitly optimize over unobserved variables, leading to globally inconsistent results. To address these limitations, we introduce Accelerated Likelihood Maximization (ALM), a novel training-free sampling strategy integrated into the reverse diffusion process that significantly extends the applicability of diffusion models beyond simple generation tasks. Unlike previous methods that implicitly influence missing regions through pre-generated region constraints, we directly optimize the unobserved region during the sampling process, enabling globally coherent and plausible generation. Furthermore, we incorporate an acceleration strategy that significantly improves computational efficiency without sacrificing performance. Experimental results demonstrate that ALM consistently outperforms state-of-the-art methods in various data domains and tasks, establishing a powerful paradigm for versatile content generation. Project website: http://hleephilip.github.io/ALM

Keywords: Diffusion Models · Versatile Generation · Synchronization

1 Introduction

Diffusion models [16, 51, 53] have demonstrated remarkable performance in visual synthesis, including images [11, 27, 41, 43, 46, 48, 57], videos [14, 19, 55, 61], and human motions [23, 54] learned from massive datasets [2, 5, 13, 49]. Within their training domains and task formulations, these models can produce highly realistic and semantically rich outputs. However, the generative power of the models is not directly applicable in practical scenarios that require generating content conditioned on partially observed or pre-generated inputs. Exemplar scenarios include: filling in missing regions [1, 9, 20, 37, 38, 69], extrapolating beyond pre-generated boundaries [24, 29, 31, 62], or lifting 2D image generation to view-consistent 3D texture generation [36, 45, 60, 63–65]. These real-world scenarios

† Corresponding author

involve completion and extension tasks that go far beyond the standard generate-from-scratch setting. We refer to this broader problem formulation as *versatile content generation*.

A naïve approach is retraining or fine-tuning pre-trained models for each specific task, which requires substantial computational resources and large-scale dedicated datasets. More importantly, these models are fundamentally limited in generalizability; they are trained for one specific task or modality and rarely transfer to others, even within the same modality. However, although task-specific tuning is inefficient, the pretrained generative models themselves remain extremely powerful. This observation motivates our goal: we aim to transform the pretrained generative models from simple synthesis frameworks to versatile content generators, without any additional training.

With a shared motivation, diffusion synchronization [24, 29, 31, 62] attempts to address this problem. These methods are designed to achieve two goals: (a) fill the unobserved region while maintaining consistency with the pre-generated content, and (b) generate high-quality unobserved regions without sacrificing the original performance of the model. However, existing methods are mostly heuristic or indirect: SyncTweedies [24] relies on extensive empirical search, while SyncSDE [29] provides a mathematical justification but adopts a strong Gaussian assumption. More importantly, SyncSDE's guidance is restricted to the pre-generated region and does not control the generation of unobserved regions. The unobserved region remains unconstrained, under the assumption that completions will naturally emerge during the diffusion process. However, this assumption does not hold in practice, especially when the unobserved region becomes large (*e.g.*, outpainting), thereby producing implausible outcomes. In essence, SyncSDE lacks a mechanism that enforces high-quality generation in the missing region, still failing to achieve the goal of synchronization.

To address these limitations, we propose *Accelerated Likelihood Maximization (ALM)*, a fully **training-free** and **widely applicable** sampling mechanism by modifying the reverse diffusion sampling process. ALM introduces a fundamentally different optimization principle from prior synchronization and posterior-guidance approaches [7, 12, 39, 52]. Instead of enforcing consistency only in pre-generated regions or sampling a single trajectory under posterior constraints, ALM is, to our knowledge, the first work to formulate synchronization as explicit likelihood maximization over the unobserved region to correlate multiple diffusion trajectories. By directly performing likelihood maximization over the unobserved variable with respect to the pre-generated context and the diffusion prior, ALM improves the plausibility of missing regions while preserving global coherence. This leads to a novel inference mechanism – adaptive, region-aware likelihood maximization that directly updates the diffusion trajectories. We use the term likelihood maximization in a score-based inference sense: the update follows score estimates of a composite log-probability objective over the unobserved variable while keeping the pretrained generative model fixed. Technically, we derive an acceleration strategy that handles iterative likelihood maximization in a single step, significantly improving inference efficiency without sacrificing quality.

We demonstrate that ALM is broadly applicable across various generative models [23, 46, 55, 67], ranging from large-scale diffusion models (SDXL [43]) to recent flow-matching frameworks (FLUX [27]), enabling them to handle versatile generation tasks. Further, ALM extends the application scope of pretrained models to include challenging tasks such as image inpainting, wide image generation, human motion completion, and 3D mesh texturing. As a result, ALM establishes a general paradigm for versatile content generation. Below are our contributions:

- We propose ALM, a fully training-free and model-agnostic sampling scheme that can directly transform pretrained generative models into versatile content generators.
- We formulate diffusion synchronization as explicit score-based likelihood maximization over the unobserved region, providing a principled optimization objective beyond prior synchronization-based approaches.
- We derive an accelerated one-step inference strategy and demonstrate broad applicability across images, human motion, 3D meshes, and videos, achieving state-of-the-art performance even compared to training-based baselines.

2 Related Works

Training-based Methods. Several methods require training to address specific tasks of versatile content generation [8, 20, 28, 32, 35, 59, 60, 69]. For image inpainting, BrushNet [20] presents a plug-and-play dual-branch architecture that separately processes masked image features from diffusion latents. Similarly, PowerPaint [69] introduces a framework with learnable task prompts, allowing a model to handle diverse inpainting challenges within the image domain. There also exist variants of Stable Diffusion [46] and SDXL [43] that are specifically fine-tuned for image inpainting. Beyond images, CondMDI [8] extends diffusion models to human motion [54] to perform human motion completion from partial keyframes, generating coherent and diverse motion sequences. While these methods achieve strong performance on specific tasks, their reliance on extensive task-specific training limits their generalization across diverse domains, making them unsuitable for versatile content generation. In contrast, ALM is fully training-free, shows broad generalizability across diverse tasks, and even outperforms training-based approaches.

Training-free Methods. To overcome the high computational cost required for training-based methods, several task-specific training-free approaches [1, 18, 36–38, 45, 65] have been proposed. HD-Painter [38] introduces prompt-aware attention and reweighted attention score guidance to guide the reverse diffusion process of inpainting models, combined with a tailored super-resolution module and Poisson blending [42]. Reconstruction guidance [18], originally proposed for long video generation, enforces consistency with pre-generated frames during denoising of the unobserved region using L2 loss. This strategy can be extended to other modalities, such as human motion, as discussed in [8]. In the 3D domain, TexPainter [65]

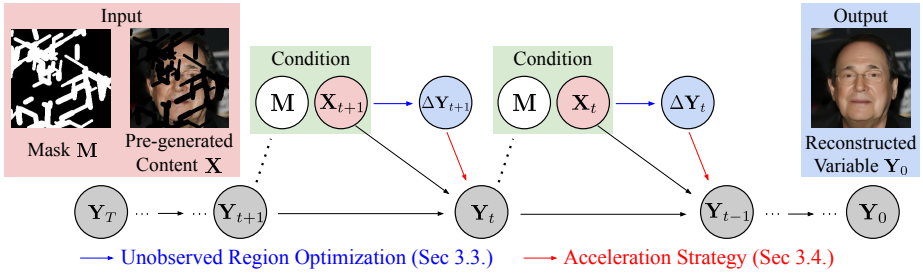

Fig. 1: Overview of the proposed method. ALM aims to adapt diffusion models to reconstruct the unobserved variable while preserving pre-generated content.

leverages the color blending scheme to ensure multi-view consistency during mesh texturing. However, most of these techniques are tailored to specific tasks and therefore remain limited in applicability. In contrast, we propose a unified framework that can be applied across modalities while achieving state-of-the-art performance.

Diffusion Synchronization. Synchronization-based methods [3, 24, 29, 31, 62] propose tailored strategies to model the correlations between diffusion trajectories. For instance, SyncTweedies [24] evaluates 60 strategies and shows that the averaging variables obtained using Tweedie's formula yield the best results, though its effectiveness relies largely on heuristics without a clear mathematical explanation. StochSync [62] relies on alternately sampled non-overlapping views, rather than explicitly modeling the relation between unobserved and pre-generated regions. Since it does not evaluate the interactions between trajectories within a single diffusion step, it inevitably results in global inconsistencies. SyncSDE [29] formulates the posterior distribution of the pre-generated content given the unobserved region, but still does not explicitly optimize the unobserved region and relies solely on guidance derived from the pre-generated content. It implicitly assumes that plausible completions will emerge without directly enforcing them. Such an assumption lacks a solid foundation, and consequently it fails to generate high-quality content. To overcome this limitation, we introduce a novel optimization objective that explicitly optimizes the unobserved region, thereby enhancing both consistency and overall fidelity.

3 Proposed Method

3.1 Overview

We aim to adapt diffusion-based generative models [16, 50, 51, 53] to versatile inpainting and outpainting tasks in a training-free manner, where the unobserved variables are sampled while conditioning on the given pre-generated content. We denote the pre-generated content as $\mathbf{X}$ and the binary mask indicating the unobserved region as $\mathbf{M}$. At diffusion timestep t, the noisy pre-generated content is

represented as $\mathbf{X}_t$, while the unobserved variable sampled by our method is written as $\mathbf{Y}_t$. We further define the blended variable $\mathbf{E}_t$ as $\mathbf{E}_t = \mathbf{X}_t \odot (\mathbf{1} - \mathbf{M}) + \mathbf{Y}_t \odot \mathbf{M}$. During the reverse diffusion process, we first update $\mathbf{Y}_t$ as follows:

$$\mathbf{Y}_t \leftarrow \mathbf{Y}_t + \mathbf{M} \odot (w_1(\epsilon_\theta(\mathbf{Y}_t, t, \mathbf{c}) - \epsilon_\theta(\mathbf{E}_t, t, \mathbf{c})) - w_2\epsilon_\theta(\mathbf{E}_t, t, \mathbf{c})), \tag{1}$$

where $\epsilon_\theta(\cdot, \cdot, \cdot)$ denotes the pretrained noise prediction network of the diffusion model, $\mathbf{c}$ is the conditioning variable (*e.g.* text embedding), and w_1, w_2 are tunable hyperparameters. Derivation of Eq. (1) is the core contribution of our work. We then sample $\mathbf{Y}_{t-1}$ by modified DDIM [51] reverse process as follows:

$$\begin{aligned}\mathbf{Y}_{t-1} \leftarrow & \sqrt{\alpha_{t-1}} \left(\frac{\mathbf{Y}_t - \sqrt{1-\alpha_t}\epsilon_\theta(\mathbf{Y}_t, t, \mathbf{c})}{\sqrt{\alpha_t}} \right) + \sqrt{1-\alpha_{t-1}}\epsilon_\theta(\mathbf{Y}_t, t, \mathbf{c}) \\ & + w_1(\mathbf{1} - \mathbf{M}) \odot (\mathbf{X}_t - \mathbf{Y}_t),\end{aligned} \tag{2}$$

Figure 1 shows the graphical diagram of the proposed method.

3.2 Preliminary: SyncSDE

A representative approach tackling training-free versatile content generation is diffusion synchronization [24, 29, 62]. SyncSDE, which provides a probabilistic explanation of why diffusion synchronization works, generates content by introducing a conditional probability term that couples different diffusion trajectories. Specifically, it factorizes the conditional score function of the diffusion model used during the sampling of $\mathbf{Y}_t$ as:

$$\nabla_{\mathbf{Y}_t} \log p(\mathbf{Y}_t \mid \mathbf{X}_t, \mathbf{c}) = \nabla_{\mathbf{Y}_t} \log p(\mathbf{Y}_t \mid \mathbf{c}) + \nabla_{\mathbf{Y}_t} \log p(\mathbf{X}_t \mid \mathbf{Y}_t, \mathbf{c}), \tag{3}$$

where the conditional probability of $\mathbf{X}_t$ given $\mathbf{Y}_t$ and $\mathbf{c}$ is modeled as:

$$p(\mathbf{X}_t \mid \mathbf{Y}_t, \mathbf{c}) := p(\mathbf{X}_t \mid \mathbf{Y}_t) \sim \mathcal{N}(\mathbf{Y}_t, \frac{\gamma_t}{w_1}(1-\alpha_t)(\mathbf{1} - \bar{\mathbf{M}})^{-1}), \tag{4}$$

with a diagonal precision matrix $\bar{\mathbf{M}}$, where pre-generated and unobserved entries are set to 0 and 1, respectively. The conditional score is then substituted into the DDIM reverse process, yielding the update rule derived in Eq. (2).

3.3 Unobserved Region Optimization

Analysis. The synchronization strategy discussed in Sec. 3.2 often yields suboptimal results, since the guidance mechanism of SyncSDE [29] focuses solely on optimizing the pre-generated region, $(\mathbf{1} - \mathbf{M}) \odot \mathbf{Y}_t$, without explicitly providing any information for the unobserved region, $\mathbf{M} \odot \mathbf{Y}_t$. In other words, it does not guarantee that the unobserved region will be harmonized with the pre-generated content; instead, it just assumes that diffusion will naturally produce a plausible outcome, which is typically insufficient and does not hold. To validate this analysis, we conduct an experiment on image inpainting. As shown in Figure 2 ("SyncSDE" column) and Figure 4 (1st row, "w/o ALM" columns), it often fails to synthesize coherent and high-quality outputs, where the unobserved regions contain inconsistent or arbitrarily generated content that does not harmonize with the pre-generated region, supporting our analysis.

Derivation. We aim to optimize the unobserved region of $\mathbf{Y}_t$ by imposing a novel sampling strategy. At each diffusion timestep t, we introduce an additional term $\Delta\mathbf{Y}_t$, which is added to $\mathbf{Y}_t$ for direct optimization. We design $\Delta\mathbf{Y}_t = \sum_{i=1}^{N} \Delta\mathbf{Y}_t^i$, where the sequence $\{\Delta\mathbf{Y}_t^i\}_{i=1}^N$ is constructed to iteratively minimize the following terms:

$$-\lambda_1 \log p(\mathbf{X}_t, \mathbf{M} \mid \mathbf{Y}_t^i + \mathbf{M} \odot \Delta\mathbf{Y}_t^i, \mathbf{c}) - \lambda_2 \log p(\mathbf{X}_t, \mathbf{M}, \mathbf{Y}_t^i + \mathbf{M} \odot \Delta\mathbf{Y}_t^i \mid \mathbf{c}), \quad (5)$$

with λ_1 and λ_2 being scalar hyperparameters ($\lambda_1 > \lambda_2$). Note that $\mathbf{Y}_t^i = \mathbf{Y}_t^{i-1} + \mathbf{M} \odot \Delta\mathbf{Y}_t^{i-1}$, and the initial values are set as $\mathbf{Y}_t^1 = \mathbf{Y}_t$ and $\{\Delta\mathbf{Y}_t^i\}_{i=1}^N = \{\mathbf{0}\}_{i=1}^N$. Here, the conditional likelihood term encourages contextual consistency by aligning the unobserved region with the pre-generated content, whereas the joint log-density term encourages the blended content to lie within high-density regions of the full data distribution, thereby harmonizing both regions into a globally realistic sample. We refer to the resulting procedure as likelihood maximization in a score-based sense, since the update follows score estimates of the corresponding composite log-probability objective over the unobserved region while keeping the pretrained model fixed. This separation enables our method to simultaneously promote local consistency and global harmonization. We verify that using both terms is essential for high-quality content generation in Sec. 4.2. The coefficients λ_1 and λ_2 act as weights in a composite energy function [53], allowing adaptive balancing between two terms for better performance.

We define $f(\Delta\mathbf{Y}_t^i)$ as the objective defined in Eq. (5). With the constraint of $\|\Delta\mathbf{Y}_t^i\| \ll 1$, we apply a Taylor expansion around $\mathbf{0}$. By taking a gradient descent on $\Delta\mathbf{Y}_t^i$ with step size of 1, we obtain:

$$\Delta\mathbf{Y}_t^i = \mathbf{M} \odot (\lambda_1 \nabla_{\mathbf{Y}_t^i} \log p(\mathbf{X}_t, \mathbf{M} \mid \mathbf{Y}_t^i, \mathbf{c}) + \lambda_2 \nabla_{\mathbf{Y}_t^i} \log p(\mathbf{X}_t, \mathbf{M}, \mathbf{Y}_t^i \mid \mathbf{c})). \quad (6)$$

Note that the small-magnitude constraint can be satisfied by choosing sufficiently small values of λ_1 and λ_2, which we detail in Sec. 3.4. Using Bayes' rule, we factorize the conditional log-likelihood term into $p(\mathbf{X}_t, \mathbf{M}, \mathbf{Y}_t^i \mid \mathbf{c})$ and $p(\mathbf{Y}_t^i \mid \mathbf{c})$. Following the score-based substitution technique [30, 53], the second term is calculated using the pretrained diffusion model:

$$\nabla_{\mathbf{Y}_t^i} \log p(\mathbf{Y}_t^i \mid \mathbf{c}) \simeq -\frac{1}{\sqrt{1-\alpha_t}} \epsilon_\theta(\mathbf{Y}_t^i, t, \mathbf{c}) \quad (7)$$

For the first term, we define $\nabla_{\mathbf{Y}_t^i} \log p(\mathbf{X}_t, \mathbf{M}, \mathbf{Y}_t^i \mid \mathbf{c}) \simeq \nabla_{\mathbf{Y}_t^i} \log p(\mathbf{E}_t^i \mid \mathbf{c})$. We justify that this score estimation works well in Appendix C, where it is interpreted as a surrogate for the joint score. Then we get

$$\nabla_{\mathbf{Y}_t^i} \log p(\mathbf{E}_t^i \mid \mathbf{c}) = \nabla_{\mathbf{E}_t^i} \log p(\mathbf{E}_t^i \mid \mathbf{c}) \odot \mathbf{M} \simeq -\frac{1}{\sqrt{1-\alpha_t}} \epsilon_\theta(\mathbf{E}_t^i, t, \mathbf{c}) \odot \mathbf{M}. \quad (8)$$

Putting these together, the closed form formula for $\Delta\mathbf{Y}_t^i$ becomes

$$\Delta\mathbf{Y}_t^i = \mathbf{M} \odot (\lambda_1(\epsilon_\theta(\mathbf{Y}_t^i, t, \mathbf{c}) - \epsilon_\theta(\mathbf{E}_t^i, t, \mathbf{c})) - \lambda_2 \epsilon_\theta(\mathbf{E}_t^i, t, \mathbf{c})), \quad (9)$$

up to a scaling factor of $1/\sqrt{1-\alpha_t}$.

3.4 Acceleration Strategy

From Eq. (9), we can choose λ_1 and λ_2 such that each $\Delta\mathbf{Y}_t^i$ remains sufficiently small for accurate Taylor expansion, then get a sequence $\{\Delta\mathbf{Y}_t^i\}_{i=1}^N$ with N iterations. However, this iterative process is computationally expensive, since its time complexity scales as $\mathcal{O}(N)$. To address this, we propose a *one-step approximation* strategy. For the rest of the derivation, we denote $\mathbf{Y}_t^i = \mathbf{Y}_t^{i-1} + \Delta\mathbf{Y}_t^{i-1}$ from the definition of $\Delta\mathbf{Y}_t^{i-1}$. Here, we present two claims for derivation:

Claim 1. $\Delta\mathbf{Y}_t^i$ is small enough for all $1 \leq i \leq N$. That is, λ_1 and λ_2 are chosen such that $\|\Delta\mathbf{Y}_t^i\| \ll 1$.
Claim 2. The noise prediction network $\epsilon_\theta(\cdot,\cdot,\cdot)$ of the pretrained diffusion model is L-Lipschitz [22, 25].

Using these claims, we analyze the difference between $\Delta\mathbf{Y}_t^i$ and $\Delta\mathbf{Y}_t^{i+1}$:

$$\begin{aligned}\|\Delta\mathbf{Y}_t^{i+1} - \Delta\mathbf{Y}_t^i\| &\leq \lambda_1\|\epsilon_\theta(\mathbf{Y}_t^i + \Delta\mathbf{Y}_t^i, t, \mathbf{c}) - \epsilon_\theta(\mathbf{Y}_t^i, t, \mathbf{c})\| \\ &\quad + (\lambda_1+\lambda_2)\|\epsilon_\theta(\mathbf{E}_t^i + \Delta\mathbf{Y}_t^i, t, \mathbf{c}) - \epsilon_\theta(\mathbf{E}_t^i, t, \mathbf{c})\| \\ &\leq L(2\lambda_1+\lambda_2)\|\Delta\mathbf{Y}_t^i\| = \mathcal{O}(\|\Delta\mathbf{Y}_t^i\|) \end{aligned} \tag{10}$$

From Claim 1, it follows that $\Delta\mathbf{Y}_t^{i+1} \simeq \Delta\mathbf{Y}_t^i$ for all i. Therefore, we approximate the iterative update with a one-step approximation as follows:

$$\Delta\mathbf{Y}_t \simeq N\Delta\mathbf{Y}_t^1 = \mathbf{M} \odot (w_1'(\epsilon_\theta(\mathbf{Y}_t, t, \mathbf{c}) - \epsilon_\theta(\mathbf{E}_t, t, \mathbf{c})) - w_2\epsilon_\theta(\mathbf{E}_t, t, \mathbf{c})), \tag{11}$$

where we define $w_1' = N\lambda_1$ and $w_2 = N\lambda_2$. In practice, we set $w_1 = w_1'$, yielding only two hyperparameters.

We justify that when Claim 1 holds, $\|\Delta\mathbf{Y}_t^{i+1} - \Delta\mathbf{Y}_t^i\| \simeq 0$, making the one-step approximation valid in Appendix D. Note that the use of $\mathcal{O}(\cdot)$ bounds for analyzing diffusion dynamics is not uncommon in the literature [25], supporting the reasonableness of our derivation with strong empirical results. Thanks to this approximation, instead of gradually refining the unobserved region through N iterations, we directly compute the outcome of the full optimization in a single update. This technique significantly reduces computation time *without sacrificing the performance* as verified in Sec. 4.2. In practice, we apply a decaying schedule to hyperparameters to better ensure the small-update assumption, defined as:

$$w_i = \sigma_t w_i^{\text{init}}, \quad \sigma_t = \sqrt{\frac{1-\alpha_{t-1}}{1-\alpha_t}}\sqrt{1-\frac{\alpha_t}{\alpha_{t-1}}} \tag{12}$$

where σ_t follows the same definition as in DDPM [16].

Additional methodological details are provided in the appendix. Appendix A describes the full derivation of ALM, including the Taylor expansion and acceleration formulation. Appendix C discusses the score estimation used to approximate the joint score, and Appendix D validates the one-step approximation.

4 Experiments

We comprehensively evaluate our approach on both inpainting and outpainting across diverse data modalities, highlighting its capability for versatile content

generation. Specifically, we assess image inpainting in Sec. 4.2 and wide image generation through outpainting in Sec. 4.3. Beyond the image domain, we extend our framework to human motion completion in Sec. 4.4, and further explore its applicability to 3D mesh texturing in Sec. 4.5. For each table, we **bold** and underline the best and second-best results, respectively. Additional experimental material is provided in the appendix. Appendix B provides task-specific details and additional results, including long video generation. Appendix E analyzes computational cost, and Appendix F discusses hyperparameter sensitivity.

4.1 Implementation Details

We implement our method based on PyTorch [40]. To ensure accurate score estimation in Eq. (7) and Eq. (8), we do not employ classifier-free guidance [17] during the calculation of Eq. (9). However, for fair comparison with baselines, we still apply classifier-free guidance in the reverse diffusion process of Eq. (2). For SyncSDE [29], since the official codebase does not support image inpainting scenarios, we reproduced it. For all other baselines, we run the official codes of each algorithm for fair comparison.

4.2 Image Inpainting

Comparison with Prior Works. We first compare ALM against a wide range of representative inpainting methods using the pretrained Stable Diffusion [46]. The baselines include training-based methods built upon Stable Diffusion, such as BrushNet [20], PowerPaint [69], and Stable Diffusion Inpainting (SDI) [46], as well as training-free methods such as SyncSDE [29] and HD-Painter [38].

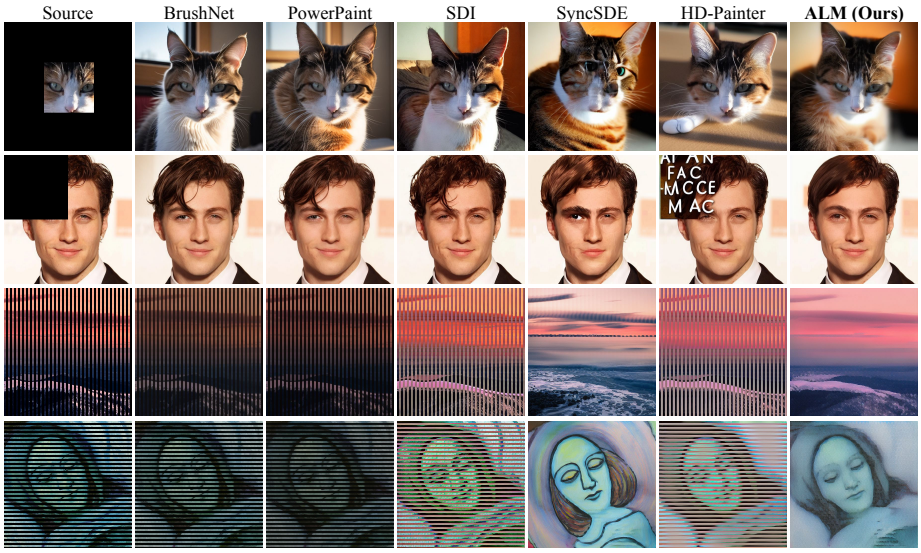

Fig. 2: Qualitative comparison of our method against Stable Diffusion-based [46] image inpainting methods [20, 29, 38, 46, 69]. ALM shows superior performance.

Table 1: Quantitative evaluation on image inpainting. Methods with * and † denote results obtained with pixel-level blending and super-resolution, respectively.

Method	Training-free	AFHQ [6]					CelebA-HQ [21]				
		LPIPS ↓	MSE ↓	M-SSIM ↑	MS-SSIM ↑	FSIM ↑	LPIPS ↓	MSE ↓	M-SSIM ↑	MS-SSIM ↑	FSIM ↑
BrushNet [20]	N	0.316	0.216	0.256	0.589	0.741	0.274	0.195	0.347	0.638	0.759
PowerPaint [69]	N	0.310	0.217	0.272	0.600	0.748	0.272	0.203	0.366	0.650	0.764
SDI [46]	N	0.292	**0.140**	0.295	0.623	0.757	0.268	0.130	0.368	0.659	0.763
SyncSDE [29]	Y	0.304	0.172	0.302	0.641	0.778	0.292	0.159	0.341	0.661	0.781
HD-Painter [38]	Y	0.301	0.146	0.285	0.610	0.741	0.286	0.146	0.344	0.642	0.747
ALM (Ours)	Y	**0.283**	0.143	**0.351**	**0.689**	**0.796**	**0.251**	**0.126**	**0.417**	**0.732**	**0.813**
BrushNet* [20]	N	0.286	0.201	0.270	0.622	0.765	0.250	0.183	0.360	0.664	0.779
HD-Painter*† [38]	Y	0.285	0.136	0.300	0.646	0.765	0.275	0.140	0.347	0.666	0.766
ALM* (Ours)	Y	**0.259**	**0.125**	**0.343**	**0.719**	**0.826**	**0.240**	**0.112**	**0.398**	**0.746**	**0.832**

By comparing our method against SyncSDE [29], we provide strong supporting evidence for our analysis of SyncSDE's limitations presented in Sec. 3.3. We use two distinct datasets for evaluation: AFHQ [6] and CelebA-HQ [21]. From each dataset, we sample 1,000 image–mask pairs to construct the test set. We measure the performance using five commonly adopted metrics: LPIPS [68], MSE, Masked SSIM (M-SSIM), Multi-Scale SSIM (MS-SSIM) [56], and FSIM [66]. Note that M-SSIM is computed over the unobserved region to better evaluate the quality of the generated region itself. Since both ALM and SyncSDE require the sequence $\{\mathbf{X}_t\}_{t=0}^{T}$, we apply DDIM [51] inversion with the masked source image. For all algorithms, the masked source image is provided as input and the model generates the target image from pure noise, which follows the standard inpainting setup.

As summarized in Table 1, our method demonstrates outstanding performance across all baselines. Notably, it consistently outperforms both training-free and training-based methods, regardless of the dataset. We also emphasize that our method achieves better M-SSIM with a significant margin, showing the effects of explicit optimization of the unobserved region. Figure 2 visualizes the qualitative comparisons, where our method consistently delivers superior visual quality. Our method also shows robust performance under diverse and complex mask geometries, demonstrating its generalizability.

Experiments across Diverse Backbones. To validate the robustness of ALM with respect to the underlying diffusion backbone, we conduct experiments with diverse models. We adopt several additional backbones: (a) an unconditional diffusion model trained on CelebA-HQ [21] from RePaint [37], (b) Stable Diffusion XL [43], and (c) FLUX [27]. FLUX uses flow matching [33, 34], and we detail the adaptation of our method to the flow matching framework in Appendix B.1. As shown in Figure 3, our method delivers high-quality inpainting results on every backbone. These findings highlight that ALM is model-agnostic, suggesting that our method transforms diverse pretrained models into versatile content generators without retraining, strengthening its importance.

Comparison with Large-Scale Models. To further validate the effectiveness of ALM against large-scale generative models, we compare it with SDXL-Inpainting [43], as well as FLUX [27] and Qwen-Image [57] adapted for inpainting via latent blending. Since these models are designed for high-resolution generation,

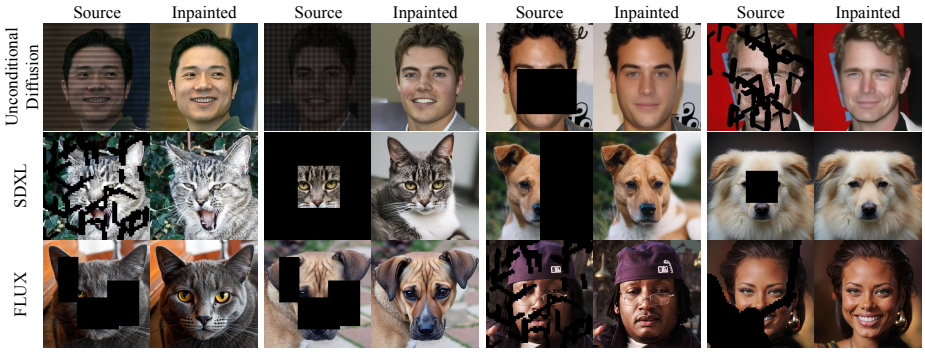

Fig. 3: Qualitative results of image inpainting with various backbones [27, 37, 43].

we adopt SDXL as the backbone for ALM to ensure a fair comparison at 1K resolution. As shown in Table 2, ALM not only outperforms but also achieves this without additional training, underscoring the significance of our approach.

Table 2: Quantitative comparison of ALM on image inpainting with large-scale generative models [27, 43, 57].

Method	Training-free	AFHQ [6]					CelebA-HQ [21]				
		LPIPS ↓	MSE ↓	M-SSIM ↑	MS-SSIM ↑	FSIM ↑	LPIPS ↓	MSE ↓	M-SSIM ↑	MS-SSIM ↑	FSIM ↑
SDXL-Inpainting [43]	N	0.303	0.191	0.309	0.656	0.780	0.313	0.232	0.332	0.657	0.768
FLUX-Inpainting [27]	Y	0.272	0.155	0.290	0.662	0.788	**0.212**	0.133	0.364	0.719	0.809
Qwen-Image-Inpainting [57]	Y	0.257	0.126	0.333	0.689	0.813	0.232	0.141	0.343	0.688	0.799
ALM (Ours)	Y	**0.254**	**0.112**	**0.410**	**0.754**	**0.841**	0.249	**0.130**	**0.442**	**0.772**	**0.843**

Analyzing the Effect of Components. We analyze the effect of each objective term described in Eq. (5), as well as the overall impact of ALM and the acceleration strategy. As shown in Table 3, each component plays an important role in generating high-quality results. In particular, our one-step approximation does not degrade performance, while reducing the runtime by approximately 185×. This shows that the acceleration strategy greatly improves efficiency without sacrificing quality. The exact runtime is reported in Appendix E.

We further present qualitative ablation results in Figure 4. The first row demonstrates that our method effectively mitigates a key limitation of the existing approach [29], which fails to harmonize the unobserved region with the pre-generated content. In contrast, by explicitly performing likelihood maximization over the unobserved region, our approach produces globally coherent samples that align well with the given context. The second and third rows visualize the effectiveness of the conditional likelihood and joint log-density terms, respectively. Specifically, the conditional likelihood is more effective when applied to pre-trained Stable Diffusion [46], whereas the joint log-density term has a significant impact on an unconditional diffusion model [37]. These results demonstrate that incorporating both terms enables ALM to generalize effectively across diverse diffusion backbones. The last row shows that the visual quality remains consistent irrespective of the use of the acceleration strategy.

Table 3: Quantitative ablation study results on image inpainting.

Method	AFHQ [6]					CelebA-HQ [21]				
	LPIPS ↓	MSE ↓	M-SSIM ↑	MS-SSIM ↑	FSIM ↑	LPIPS ↓	MSE ↓	M-SSIM ↑	MS-SSIM ↑	FSIM ↑
w/o ALM	0.295	0.169	0.300	0.650	0.782	0.287	0.161	0.332	0.663	0.782
w/o cond. term	0.295	0.162	0.323	0.661	0.787	0.291	0.156	0.357	0.673	0.786
w/o joint term	0.284	0.149	0.327	0.679	0.793	0.254	0.132	0.385	0.720	0.808
w/o acceleration	0.298	0.170	0.298	0.649	0.781	0.277	0.156	0.341	0.675	0.787
ALM (Ours)	**0.283**	**0.143**	**0.351**	**0.689**	**0.796**	**0.251**	**0.126**	**0.417**	**0.732**	**0.813**

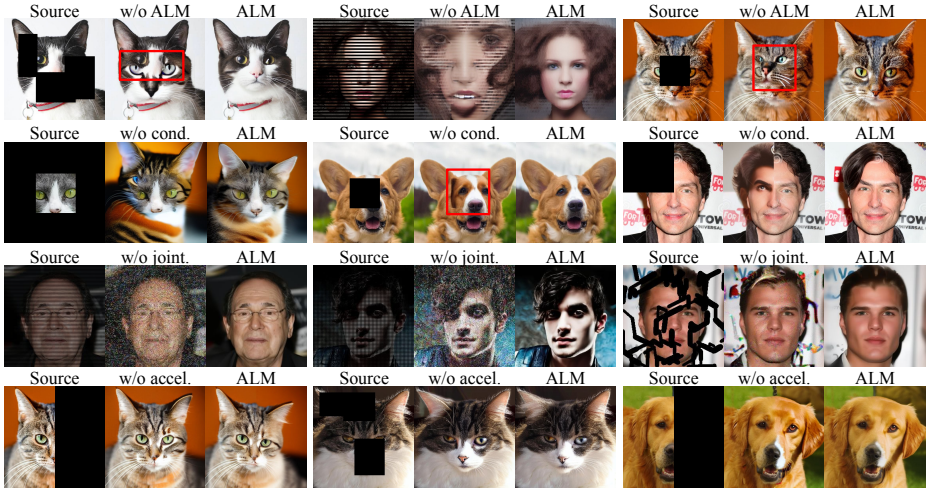

Fig. 4: Qualitative ablation study results. Each row shows the effect of ALM (Eq. 9), the conditional likelihood term (Eq. 5), the joint log-density term (Eq. 5), and the acceleration strategy (Eq. 11), respectively.

4.3 Wide Image Generation

Beyond image inpainting, our approach also naturally extends to the outpainting task, enabling the synthesis of wide, high-resolution images. We employ an autoregressive image outpainting strategy to generate wide images. Starting from 512×512 patch generated with the pretrained Stable Diffusion [46], subsequent overlapping patches are iteratively synthesized via outpainting. With a stride of 384 pixels, we generate five patches, resulting in a 2048×512 resolution image. The patches are overlapped such that the i-th patch is placed on top of the $(i+1)$-th one and decoded with the pretrained VAE [26] decoder. We compare our method against state-of-the-art diffusion synchronization approaches using 400 images, including SyncTweedies [24], SyncSDE [29], and StochSync [62]. For evaluation, we randomly crop the generated wide image into 512×512 image. We report FID [15] and KID [4] to evaluate distribution alignment, with Aesthetic Score [49] and Q-Align [58] to assess the perceptual quality and fidelity of the generated images.

As shown in Table 4 and Figure 5, our method achieves outstanding performance compared to the baselines. In particular, SyncSDE (Row 2) exhibits clear limitations in the wide image generation setting. Since it does not explicitly optimize the unobserved region, it achieves inferior aesthetic quality in terms

of aesthetic score and Q-Align. Furthermore, since its performance degrades significantly when the masked area becomes large, it requires a smaller stride between overlapping patches. This leads to color inconsistencies and clear edge artifacts.

Table 4: Quantitative evaluation on wide image generation. KID [4] is scaled by 10^3.

Method	FID ↓	KID ↓	Aesthetic Score ↑	Q-Align ↑
SyncTweedies [24]	85.95	58.36	6.104	<u>4.550</u>
SyncSDE [29]	<u>85.82</u>	<u>51.84</u>	<u>6.127</u>	4.542
StochSync [62]	113.21	92.10	6.026	4.546
ALM (Ours)	**83.41**	**42.98**	**6.133**	**4.581**

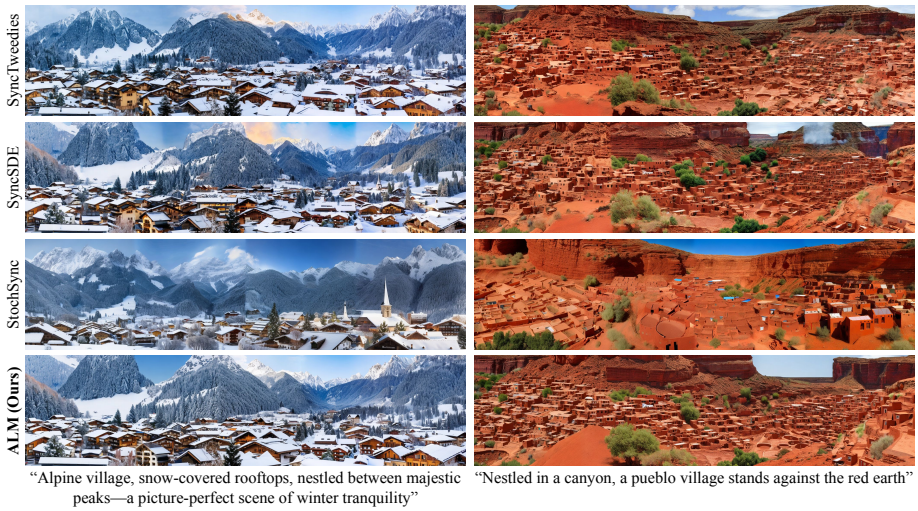

Fig. 5: Qualitative comparison of our method against state-of-the-art methods [24,29,62]. SyncSDE exhibits noticeable discontinuities across patches, while SyncTweedies produces blurry and inconsistent coloring. StochSync likewise tends to generate blurry and discontinuous wide images. In contrast, ALM successfully produces results without blurred or inconsistent regions.

4.4 Human Motion Completion

We further demonstrate the versatility of our method by extending it from images to human motion data. Specifically, we tackle the human motion completion task, where the goal is to reconstruct missing parts of a motion sequence. We evaluate across three distinct scenarios: "first-half prediction," predicting the initial segment from the latter half, "middle-half prediction," completing the central segment given the first and last quarters, and "last-half prediction," the inverse of the first-half setting. We use a U-Net-based [47] pretrained diffusion model [23] for text-to-motion synthesis.

We compare our method against the training-based method CondMDI [8] and training-free methods such as Reconstruction Guidance [18] and its imputation-based variant [54]. Basically, we follow the CondMDI setup that employs a

DDPM [16] sampler with 1,000 timesteps. For each completion scenario, we sample 1,000 motion sequences from the HumanML3D [13] dataset and report the average performance over 10 replications. We measure FID, Matching Score (Match.), Top-1 R-precision (R-prec.), and Diversity (Div.) metrics, which are widely adopted metrics in prior works [8,13]. Table 5 illustrates that the proposed method achieves superior performance with high versatility across various human motion completion scenarios. In Figure 6, the given frames are highlighted in orange, while the filled frames generated by the model are shown in blue.

Table 5: Quantitative evaluation on human motion completion using the motion sequences sampled from HumanML3D [13] dataset. Methods marked with * use imputation [54]. '→' indicates that closer alignment with the GT value is better.

Method	Training-free	First-half				Middle-half				Last-half			
		FID ↓	Match. ↓	R-prec. ↑	Div. →	FID ↓	Match. ↓	R-prec. ↑	Div. →	FID ↓	Match. ↓	R-prec. ↑	Div. →
CondMDI [8]	N	0.620	4.566	0.350	8.668	0.594	4.489	0.354	8.618	0.362	4.409	0.365	**9.050**
Recon. Gui. [18]	Y	11.342	5.230	0.284	6.101	12.806	5.152	0.299	6.041	7.322	4.866	0.319	6.804
Recon. Gui.* [18]	Y	3.547	4.395	0.361	7.774	4.250	4.415	0.366	7.599	0.576	**4.051**	0.400	8.797
ALM (Ours)	Y	**0.311**	4.143	0.396	**8.979**	**0.447**	4.175	**0.395**	**8.820**	**0.236**	4.085	**0.410**	9.041
ALM* (Ours)	Y	0.465	**4.140**	**0.398**	8.828	0.645	**4.159**	0.393	8.695	0.260	4.068	0.408	9.025
Ground Truth	-	0.001	3.243	0.453	9.299	0.001	3.243	0.453	9.299	0.001	3.243	0.453	9.299

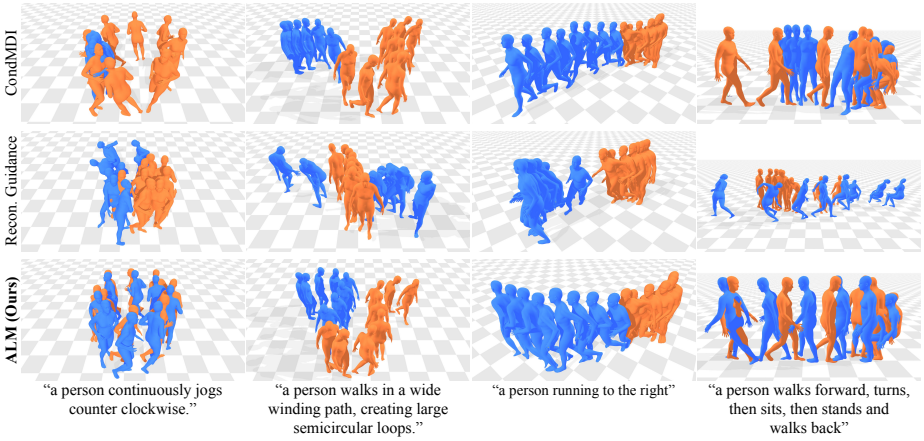

Fig. 6: Qualitative comparison of our method with baselines [8, 18] on human motion completion. While baselines show unrealistic or discontinuous motions, ALM generates plausible sequences that also align with the given text prompt.

4.5 3D Mesh Texturing

We extend our method to the 3D domain by applying it to the mesh texturing task. Following the standard setup of prior works [24, 29], we sample 10 partially overlapping viewpoints around each mesh. Iterating over the viewpoints, we generate a rendered image from the current view using the pretrained depth-conditioned ControlNet [67], where the overlapping regions of the pre-generated views are provided as additional conditioning. After obtaining multi-view renderings, we bake them into a single texture map.

For quantitative evaluation, we use 250 mesh–prompt pairs sampled from the Objaverse dataset [10]. We compare our method against both task-specific methods [45, 63, 65] and synchronization-based approaches [24, 29, 62]. Each textured mesh is rendered from 10 viewpoints, and the resulting images are used for evaluation. We report widely adopted metrics including FID [15], KID [4], and CLIP similarity [44] between rendered images and text prompts. Reference images used to compute FID and KID are also generated using depth-conditioned ControlNet, based on the depth maps rendered from the same 10 viewpoints. Results are presented in Table 6 and Figure 7-8. As shown, our method outperforms the baselines, further highlighting its effectiveness.

Table 6: Quantitative evaluation on 3D mesh texturing. KID [4] value is scaled by 10^3.

Method	Paint-it [63]	TexPainter [65]	TEXTure [45]	SyncTweedies [24]	SyncSDE [29]	StochSync [62]	ALM (Ours)
FID ↓	200.89	191.17	184.66	<u>156.78</u>	165.04	163.24	**155.46**
KID ↓	126.40	112.48	94.69	82.91	<u>82.75</u>	82.83	**74.41**
CLIP-Sim. ↑	0.287	0.284	0.289	**0.294**	0.290	0.289	<u>0.292</u>

Fig. 7: Qualitative comparison of ALM with baselines [24, 29, 45, 62, 63, 65] on 3D mesh texturing. ALM generates high-fidelity texture maps, outperforming prior works.

Fig. 8: Additional qualitative results of 3D mesh texturing. ALM generates diverse and high-fidelity textures, effectively handling detailed prompts.

5 Conclusion

In this work, we introduce a novel, training-free sampling strategy for diffusion-based versatile content generation. Such tasks go beyond simple generation by producing high-quality outputs conditioned on pre-generated inputs, encompassing a wide range of real-world applications. Although diffusion models achieve remarkable performance in standard generation tasks, adapting them to such conditional settings typically requires expensive task-specific training, which limits its generalizability. To overcome this limitation, we propose a broadly applicable mechanism that transforms a wide range of pretrained generative models into a flexible content generation framework. We synchronize pre-generated content with unobserved variables by maximizing a likelihood-based objective that combines a conditional likelihood term with a joint log-density term. Furthermore, we introduce an acceleration strategy to improve computational efficiency. Extensive experiments across diverse tasks and modalities demonstrate that our method achieves state-of-the-art performance.

Acknowledgements

This work was supported by the BK21 FOUR program of the Education and Research Program for Future ICT Pioneers, Seoul National University in 2026, the National Research Foundation of Korea(NRF) grant (No. RS-2026-25485899) and Institute of Information & communications Technology Planning & Evaluation (IITP) grant [RS-2021-II211343, Artificial Intelligence Graduate School Program (Seoul National University)] funded by the Korea government(MSIT).

References

1. Avrahami, O., Fried, O., Lischinski, D.: Blended latent diffusion. SIGGRAPH (2023)
2. Bain, M., Nagrani, A., Varol, G., Zisserman, A.: Frozen in time: A joint video and image encoder for end-to-end retrieval. In: ICCV (2021)
3. Bar-Tal, O., Yariv, L., Lipman, Y., Dekel, T.: Multidiffusion: Fusing diffusion paths for controlled image generation. In: ICML (2023)
4. Bińkowski, M., Sutherland, D.J., Arbel, M., Gretton, A.: Demystifying mmd gans. ICLR (2018)
5. Byeon, M., Park, B., Kim, H., Lee, S., Baek, W., Kim, S.: Coyo-700m: Image-text pair dataset. `https://github.com/kakaobrain/coyo-dataset` (2022), accessed: June 29, 2026
6. Choi, Y., Uh, Y., Yoo, J., Ha, J.W.: Stargan v2: Diverse image synthesis for multiple domains. In: CVPR (2020)
7. Chung, H., Kim, J., Mccann, M.T., Klasky, M.L., Ye, J.C.: Diffusion posterior sampling for general noisy inverse problems. ICLR (2023)
8. Cohan, S., Tevet, G., Reda, D., Peng, X.B., van de Panne, M.: Flexible motion in-betweening with diffusion models. In: SIGGRAPH (2024)
9. Corneanu, C., Gadde, R., Martinez, A.M.: Latentpaint: Image inpainting in latent space with diffusion models. In: WACV (2024)

10. Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., Farhadi, A.: Objaverse: A universe of annotated 3d objects. CVPR (2023)
11. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: ICML (2024)
12. Geyfman, D., Draxler, F., Groeneveld, J., Lee, H., Karaletsos, T., Mandt, S.: Calibrated test-time guidance for bayesian inference. ICML (2026)
13. Guo, C., Zou, S., Zuo, X., Wang, S., Ji, W., Li, X., Cheng, L.: Generating diverse and natural 3d human motions from text. In: CVPR (2022)
14. HaCohen, Y., Chiprut, N., Brazowski, B., Shalem, D., Moshe, D., Richardson, E., Levin, E., Shiran, G., Zabari, N., Gordon, O., Panet, P., Weissbuch, S., Kulikov, V., Bitterman, Y., Melumian, Z., Bibi, O.: Ltx-video: Realtime video latent diffusion. arXiv:2501.00103 (2024)
15. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. NIPS (2017)
16. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. NeurIPS (2020)
17. Ho, J., Salimans, T.: Classifier-free diffusion guidance. NeurIPS Workshop (2021)
18. Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., Fleet, D.J.: Video diffusion models. NeurIPS (2022)
19. Hong, W., Ding, M., Zheng, W., Liu, X., Tang, J.: Cogvideo: Large-scale pretraining for text-to-video generation via transformers. ICLR (2023)
20. Ju, X., Liu, X., Wang, X., Bian, Y., Shan, Y., Xu, Q.: Brushnet: A plug-and-play image inpainting model with decomposed dual-branch diffusion. In: ECCV (2024)
21. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. ICLR (2018)
22. Karras, T., Aittala, M., Aila, T., Laine, S.: Elucidating the design space of diffusion-based generative models. NeurIPS (2022)
23. Karunratanakul, K., Preechakul, K., Suwajanakorn, S., Tang, S.: Guided motion diffusion for controllable human motion synthesis. In: ICCV (2023)
24. Kim, J., Koo, J., Yeo, K., Sung, M.: Synctweedies: A general generative framework based on synchronized diffusions. NeurIPS (2024)
25. Kim, J., Kang, J., Choi, J., Han, B.: Fifo-diffusion: Generating infinite videos from text without training. NeurIPS (2024)
26. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv:1312.6114 (2013)
27. Labs, B.F.: Flux. `https://github.com/black-forest-labs/flux` (2024), accessed: June 29, 2026
28. Lai, Z., Zhao, Y., Zhao, Z., Yang, X., Huang, X., Huang, J., Yue, X., Guo, C.: Natex: Seamless texture generation as latent color diffusion. In: CVPR (2026)
29. Lee, H., Lee, H., Han, S.: Syncsde: A probabilistic framework for diffusion synchronization. In: CVPR (2025)
30. Lee, H., Kang, M., Han, B.: Conditional score guidance for text-driven image-to-image translation. NeurIPS (2023)
31. Lee, Y., Kim, K., Kim, H., Sung, M.: Syncdiffusion: Coherent montage via synchronized joint diffusions. NeurIPS (2023)
32. Liang, Y., Luo, K., Chen, X., Chen, R., Yan, H., Li, W., Liu, J., Tan, P.: Unitex: Universal high fidelity generative texturing for 3d shapes. CVPR (2026)
33. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. ICLR (2023)

34. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. ICLR (2023)
35. Liu, Y., Hou, X., Wu, J., Liu, B., Zhang, Y., Song, G., Liu, Y., Tian, C., Luo, G., You, H.: Blend-aware latent diffusion: Mitigating stitched seams in image inpainting. In: CVPR Findings (2026)
36. Liu, Y., Xie, M., Liu, H., Wong, T.T.: Text-guided texturing by synchronized multi-view diffusion. In: SIGGRAPH Asia (2024)
37. Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R., Van Gool, L.: Repaint: Inpainting using denoising diffusion probabilistic models. In: CVPR (2022)
38. Manukyan, H., Sargsyan, A., Atanyan, B., Wang, Z., Navasardyan, S., Shi, H.: Hd-painter: high-resolution and prompt-faithful text-guided image inpainting with diffusion models. In: ICLR (2025)
39. Pandey, K., Sofian, F.M., Draxler, F., Karaletsos, T., Mandt, S.: Variational control for guidance in diffusion models. ICML (2025)
40. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. NeurIPS (2019)
41. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: ICCV (2023)
42. Pérez, P., Gangnet, M., Blake, A.: Poisson image editing. In: Seminal Graphics Papers: Pushing the Boundaries, Volume 2 (2023)
43. Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., Rombach, R.: Sdxl: Improving latent diffusion models for high-resolution image synthesis. ICLR (2024)
44. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML (2021)
45. Richardson, E., Metzer, G., Alaluf, Y., Giryes, R., Cohen-Or, D.: Texture: Text-guided texturing of 3d shapes. In: SIGGRAPH (2023)
46. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022)
47. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention (2015)
48. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. NeurIPS (2022)
49. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al.: Laion-5b: An open large-scale dataset for training next generation image-text models. NeurIPS (2022)
50. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: ICML (2015)
51. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. ICLR (2021)
52. Song, J., Vahdat, A., Mardani, M., Kautz, J.: Pseudoinverse-guided diffusion models for inverse problems. In: ICLR (2023)
53. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. ICLR (2021)
54. Tevet, G., Raab, S., Gordon, B., Shafir, Y., Cohen-Or, D., Bermano, A.H.: Human motion diffusion model. ICLR (2023)

55. Wang, Y., Chen, X., Ma, X., Zhou, S., Huang, Z., Wang, Y., Yang, C., He, Y., Yu, J., Yang, P., et al.: Lavie: High-quality video generation with cascaded latent diffusion models. IJCV (2025)
56. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: The thrity-seventh asilomar conference on signals, systems & computers, 2003 (2003)
57. Wu, C., Li, J., Zhou, J., Lin, J., Gao, K., Yan, K., ming Yin, S., Bai, S., Xu, X., Chen, Y., Chen, Y., Tang, Z., Zhang, Z., Wang, Z., Yang, A., Yu, B., Cheng, C., Liu, D., Li, D., Zhang, H., Meng, H., Wei, H., Ni, J., Chen, K., Cao, K., Peng, L., Qu, L., Wu, M., Wang, P., Yu, S., Wen, T., Feng, W., Xu, X., Wang, Y., Zhang, Y., Zhu, Y., Wu, Y., Cai, Y., Liu, Z.: Qwen-image technical report (2025), https://arxiv.org/abs/2508.02324, accessed: June 29, 2026
58. Wu, H., Zhang, Z., Zhang, W., Chen, C., Liao, L., Li, C., Gao, Y., Wang, A., Zhang, E., Sun, W., et al.: Q-align: Teaching lmms for visual scoring via discrete text-defined levels. ICML (2024)
59. Wu, L., Yu, J., Jin, L., Wang, H., Zheng, B., Yang, X., Jiang, H., Xia, F., Ling, F., Deng, J., Jin, X.: Unifying precise keyframes and semantic control via multi-level diffusion. In: CVPR (2026)
60. Yan, D., Wu, L., Lin, J., Wang, L., Xu, T., Chen, Z., Yang, Z., Xu, L., Zhang, S., Chen, Y.: Flexpainter: Flexible and multi-view consistent texture generation. arXiv:2506.02620 (2025)
61. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., Yin, D., Yuxuan.Zhang, Wang, W., Cheng, Y., Xu, B., Gu, X., Dong, Y., Tang, J.: Cogvideox: Text-to-video diffusion models with an expert transformer. In: ICLR (2025)
62. Yeo, K., Kim, J., Sung, M.: Stochsync: Stochastic diffusion synchronization for image generation in arbitrary spaces. ICLR (2025)
63. Youwang, K., Oh, T.H., Pons-Moll, G.: Paint-it: Text-to-texture synthesis via deep convolutional texture map optimization and physically-based rendering. In: CVPR (2024)
64. Zeng, X., Chen, X., Qi, Z., Liu, W., Zhao, Z., Wang, Z., Fu, B., Liu, Y., Yu, G.: Paint3d: Paint anything 3d with lighting-less texture diffusion models. In: CVPR (2024)
65. Zhang, H., Pan, Z., Zhang, C., Zhu, L., Gao, X.: Texpainter: Generative mesh texturing with multi-view consistency. In: SIGGRAPH (2024)
66. Zhang, L., Zhang, L., Mou, X., Zhang, D.: Fsim: A feature similarity index for image quality assessment. IEEE transactions on Image Processing (2011)
67. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: ICCV (2023)
68. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018)
69. Zhuang, J., Zeng, Y., Liu, W., Yuan, C., Chen, K.: A task is worth one word: Learning with task prompts for high-quality versatile image inpainting. In: ECCV (2024)