Exploring Stable Diffusion in Computer Vision: Theory, Comparison, and Applications

Taha Enayat
ta.enayat@gmail.com

Mahshad Golafshant
golafshan.mahshad@gmail.com

Abstract

Diffusion models have gained popularity in computer vision for their ability to generate high-quality images through controlled noise injection and reversal processes. Among these models, latent diffusion models (LDMs) have emerged as a promising approach, leveraging pre-trained autoencoders to operate in a lower-dimensional latent space, thus enhancing efficiency without sacrificing quality. Stable diffusion, a subset of LDMs, offers stability and fine detail preservation, making it an attractive alternative to traditional pixel-based diffusion models. In this paper, we explore the theoretical foundations, comparative analysis, and practical applications of stable diffusion in computer vision. We investigate the underlying principles of diffusion models, highlight the advantages of stable diffusion within LDMs, and demonstrate its potential impact through empirical evaluations and practical demonstrations. By elucidating the capabilities and limitations of stable diffusion, we aim to foster a deeper understanding of diffusion models and their relevance in various computer vision domains.

1. Introduction

In recent years, diffusion models have emerged as powerful tools in the field of generative modeling, offering unique capabilities for image synthesis and data generation. These models, which encompass denoising diffusion probabilistic models (DDPMs), score-based generative models (SGMs), and stochastic differential equations (SDEs), operate by progressively degrading data through a controlled injection of noise, followed by a learned process to reverse this degradation and generate new samples.

Despite their effectiveness, traditional diffusion models pose significant computational challenges, requiring extensive computational resources and training time, particularly for high-resolution image synthesis. To address this problem, instead of pixel space, one can encode the image to a latent space with smaller size.

One such approach is the exploration of stable diffusion within the framework of latent diffusion models (LDMs). LDMs leverage pre-trained autoencoders to encode data into a lower-dimensional latent space, enabling more efficient training and generation processes. The stability of LDMs, coupled with their ability to preserve fine details and structure, has positioned them as promising alternatives to traditional pixel-based diffusion models.

In this paper, we present a comprehensive exploration of stable diffusion in computer vision, focusing on the theoretical foundations, comparative analysis, and practical applications of LDMs. We investigate the underlying principles of diffusion models, including DDPMs, SGMs, and SDEs, and highlight the advantages and challenges associated with each approach. Furthermore, we delve into the specifics of stable diffusion within the context of LDMs, examining the methodology, advantages, and potential applications of this innovative framework.

Through theoretical analysis, empirical evaluations, and practical demonstrations, we aim to provide insights into the capabilities and limitations of stable diffusion in computer vision. By elucidating the theoretical underpinnings and practical implications of LDMs, we seek to facilitate a deeper understanding of diffusion models and their potential impact on various domains within computer vision.

2. Background

In this section, the mathematical foundation of diffusion models are outlined. Most of this section is from the survey of L. Yang et al. in 2023 [7] and M. Chen et al. 2024 survey [1]. We start off from Denoising Diffusion Probabilistic Models (DDPMs) while score-based and stochastic differential equations can be explained easily after the description of DDPM.

In summary, A denoising diffusion probabilistic model (DDPM) makes use of two Markov chains: a forward chain that perturbs data to noise, and a reverse chain that converts noise back to data. The former is typically hand-designed with the goal to transform any data distribution into a simple prior distribution (e.g., standard Gaussian), while the latter Markov chain reverses the former by learning transition kernels parameterized by deep neural networks. New data points are subsequently generated by first sampling a random vector from the prior distribution (in this case Gaussian distribution), followed by ancestral sampling through the reverse Markov chain. Ancestral sampling is a technique in which samples from a distribution are generated sequentially, starting from the root of a generative model and proceeding through its hierarchy, following the conditional dependencies. A Markov chain is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.

Formally, given a data distribution $x_0 \sim q(x_0)$, the forward Markov process generates a sequence of random variables $x1, x2 ... x_T$ with transition kernel $q(x_t|x_{t-1})$ which has the form of Gaussian distribution:

$$q(x_t|x_{t-1}) = N(x_t; \sqrt{1-\beta_t}x_{t-1}; \beta_t I)$$

It states that given a previous state x_{t1}, the current state x_t is distributed according to a normal distribution, with a mean of $\sqrt{1-\beta_t}x_{t-1}$ and a covariance matrix of $\beta_t I$, where β_t is a variance schedule that controls the noise level at each timestep. Intuitively speaking, this forward process slowly injects noise to data until all structures are lost.

For generating new data samples, DDPMs start by first generating an unstructured noise vector from the prior distribution (which is typically trivial to obtain), then gradually remove noise therein by running a learnable Markov chain in the reverse time direction. Specifically, the reverse Markov chain is parameterized by a prior distribution $p(x_T) = N(x_T; 0, I)$ and a learnable transition kernel $p_\theta(x_{t-1}|x_t)$. The learnable transition kernel $p_\theta(x_{t-1}|x_t)$ takes the form of

$$p_\theta(x_{t-1}|x_t) = N(x_{t-1}; \mu_\theta(x_t, t); \Sigma_\theta(x_t, t))$$

where θ denotes model parameters, and the mean $\mu_\theta(x_t, t)$ and variance $\mu_\theta(x_t, t)$ are parameterized by deep neural networks. With this reverse Markov chain in hand, we can generate a data sample x_0 by first sampling a noise vector $x_T \sim p(x_T)$, then iteratively sampling from the learnable transition kernel $x_t \sim p_\theta(x_{t-1}|x_t)$until $t = 1$.

Key to the success of this sampling process is training the reverse Markov chain to match the actual time reversal of the forward Markov chain. That is, we have to adjust the parameter θ so that the joint distribution of the reverse Markov chain $p_\theta(x_1, x_2, ..., x_T)$ closely approximates that of the forward process $q(x_1, x_2, ..., x_T)$. This is achieved by minimizing the Kullback-Leibler (KL) divergence between these two which is our loss function in this problem. KL divergence, or Kullback-Leibler divergence, is a measure of how one probability distribution diverges from a second, reference probability distribution. But estimating $KL(q(x_1, x_2, ..., x_T)||p_\theta(x_1, x_2, ..., x_T))$ is not and easy task. As a result, instead of tackling the minimization of KL divergence, negative variational lower bound (VLB) is minimized:

$$-L_{VLB} = \mathbb{E}[-log(p_\theta(x_0)]$$

We can stop here and minimize L_{VLB} using Monte Carlo sampling which is very computationally expensive, but the genius of this model is that by expanding and rearranging the terms, a closed form loss function is achieved. Surprisingly, the final form of loss function is very simple and pretty much like the loss function of score-based diffusion models (explained in the following) and the general form is the following:

$$L = \mathbb{E}[||\epsilon - \epsilon_\theta(x_t, t)||^2]$$

where ϵ_θ is a deep neural network with parameter θ that predicts the noise vector ϵ given x_t and t.

Moving on to Score-based Generative Models (SGM) and Stochastic Differential Equations (SDE), these two models are very similar in the essence and We try to explain them together. Imagine the following differential equation known as Langevin Equation is given:

$$dX_t = -\frac{1}{2}g(t)X_t dt + \sqrt{g(t)}dW$$

where initial $X_0 \sim P_{data}$ follows the data distribution, $W_t(t \geq 0$ is a standard Wiener process, and $g(t)$ is a non-decreasing weighting function. It can be shown that this stochastic process is exactly the one for the DDPM model if $g(t) = \beta_t$, but this is not the main focus here. What this stochastic process does is that it corrupts and shrinks the magnitude of data until in $t = \infty$ the data becomes complete gaussian noise. In practice, the process terminates at sufficiently large time $t = T$. Then the diffusion models generate fake data by reversing the time, which leads to the following backward SDE:

$$d\overleftarrow{X_t} = [\frac{1}{2}\overleftarrow{X_t} + \nabla log p_{T-t}(\overleftarrow{X_t})]dt + d\bar{W}_t$$

where $\bar{W}_t$ is another Wiener process independent of W_t. $\nabla p_t(.)$ is the so-called “score function”. Note that score considered here is a function of the data x rather than the model parameters. But this score function is unknown to us. So, we estimate it with $\hat{s}_\theta(x, t)$ which is often parameterized by a deep neural network and takes data and time as input. Now, having the estimation of score function, we discretize the backward SDE and using ancestral sampling starting from standard Gaussian noise, we sample until $t = 1$ which gives us the fake generated data. The only thing that remained is how to estimate the score function. The loss function used to train score function obviously has the following form:

$$L = \mathbb{E}[||\nabla_x log p(x_t) - \hat{s}_\theta(x, t)||^2]$$

but as we are using Gaussian distribution (Wiener process), the loss can be written as:

$$L = \mathbb{E}[||\epsilon + \beta_t \hat{s}_\theta(x,t)||^2]$$

Comparing the loss function from DDPM model and SGM, the losses would be equivalent if the euqality $\epsilon_\theta(x,t) = -\beta_t \hat{s}_\theta(x,t)$ holds.

Generally, simulating the backward process for thousands of steps to generate a sample is time-consuming. As a result, one can accelerate the sampling speed of diffusion models using a pre-trained VAE to extract low-dimensional data representations and then implement diffusion processes; a model known as latent diffusion model. We will explore this model in method section.

3. Related Works

The field of Diffusion Models are relatively new models that the idea of such models originated back in 2015 [6] but later in 2020 [3] OpenAI implemented the this idea and trained it in large scale are started the hype for these family of models. Later in 2021, engineers again from OpenAI improved the model in two papers [2, 4] which resulted in effective and stable models. In 2022, a revolutionary paper from scientists of StabilityAI [5] company has been dropped which explored latent diffusion models, which stable diffusion is the offspring of this type of model. In this part, the outline of each paper is explored until the stable diffusion paper which is explored in the next section in more details.

The paper titled “Deep Unsupervised Learning using Nonequilibrium Thermodynamics” [6] addressed the long-standing dichotomy in probabilistic models between tractability and flexibility. Models that are tractable can be analytically evaluated and easily fit to data (e.g. a Gaussian or Laplace). However, these models are unable to aptly describe structure in rich datasets. On the other hand, models that are flexible can be molded to fit structure in arbitrary data. For example, we can define models in terms of any (non-negative) function $\phi(x)$ yielding the flexible distribution $p(x) = \frac{\phi(x)}{Z}$, where Z is a normalization constant. However, computing this normalization constant is generally intractable. Evaluating, training, or drawing samples from such flexible models typically requires a very expensive Monte Carlo process. The authors proposed a forward diffusion process to systematically degrade the structure in data, followed by a reverse diffusion process to restore it. This method allowed for the creation of deep generative models capable of learning, sampling, and evaluating probabilities efficiently, even with thousands of layers or time steps.

In “Denoising Diffusion Probabilistic Models,” [3] the authors presented a class of latent variable models that achieved high-quality image synthesis. The paper highlighted a connection between diffusion probabilistic models and denoising score matching with Langevin dynamics. The model’s performance was demonstrated through impressive results on datasets like CIFAR10 and LSUN, showcasing its ability to generate images comparable to those produced by ProgressiveGAN.

The “Improved Denoising Diffusion Probabilistic Models” [4] paper made significant advancements in DDPMs. The authors showed that with a few modifications, DDPMs could achieve competitive log-likelihoods while maintaining high sample quality. They introduced learning variances of the reverse diffusion process, which allowed for faster sampling with minimal impact on quality. The paper also compared the coverage of DDPMs and GANs, demonstrating the scalability of DDPMs with model capacity and training compute.

In their work “Diffusion Models Beat GANs on Image Synthesis,” [2] Dhariwal and Nichol demonstrated that diffusion models could surpass the image sample quality of state-of-the-art generative models. They achieved this through architectural improvements and the introduction of classifier guidance, which traded off diversity for fidelity. The paper reported superior FID scores on ImageNet and showed that diffusion models could maintain better coverage of the data distribution than BigGAN-deep.

4. Methods

Detailed description of the methods used and/or proposed, and clear justification of why these methods are used and not others.

advantages and disadvantages of stable diffusion compare to GAN a little

The groundbreaking paper “High-Resolution Image Synthesis with Latent Diffusion Models” [5] from StabilityAI, details a novel approach to high-resolution image synthesis by utilizing latent diffusion models (LDMs). These models operate in the latent space of powerful pretrained autoencoders, which allows for a significant reduction in computational requirements compared to traditional pixel-based diffusion models. By training diffusion models on this latent representation, the authors achieve a near-optimal balance between complexity reduction and detail preservation, enhancing visual fidelity.

The autoencoder’s architecture (see figure 1) is pivotal to the LDM’s performance. It comprises an encoder that compresses the input data into a latent representation and

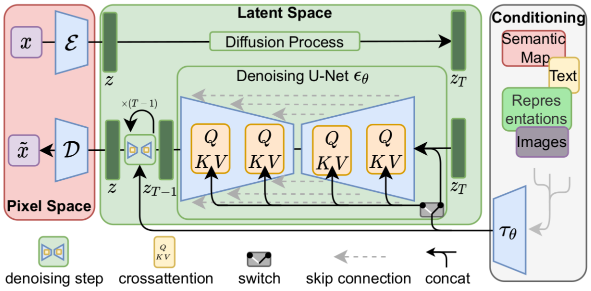

Figure 1. Overview of the model

a decoder that reconstructs the data from this compressed form. The addition of a discriminator at the end of the autoencoder, inspired by Generative Adversarial Networks (GANs), ensures the generation of high-quality latent representations. The discriminator's role is to differentiate between the actual data and the data generated by the autoencoder, thereby guiding the autoencoder to produce more accurate reconstructions. Here is the loss function used:

$$\begin{aligned} L_{\text{Autoencoder}} = \min_{\epsilon, D} \max_{\psi} L_{\text{rec}}(x, D(\epsilon(x))) \\ - L_{\text{adv}}(D(\epsilon(x))) + \log D_\psi(x) + L_{\text{reg}}(x; \epsilon, D) \end{aligned} \quad (1)$$

let's break this down. $L_{\text{rec}}(x, D(\epsilon(x))))$ is the reconstruction loss. It measures the difference between the original data (x) and the data reconstructed from the latent space $(D(\epsilon(x)))$ and it is just a MSE. $L_{\text{reg}}(x; \epsilon, D))$ is the regularization loss. It imposes additional constraints to ensure the stability of the training process and prevent overfitting. $L_{\text{adv}}(D(\epsilon(x)))$ and $\log D_\psi(x)$ are adversarial loss and log likelihood of discriminator respectively. The adverserial loss encourages the latent representation to be realistic enough to fool the discriminator and the log likelihood helps in aligning the generated samples with the actual data distribution.

The training process is bifurcated into two distinct phases. The first phase focuses on training the autoencoder to create and refine the latent space. The second phase involves training the diffusion model within this latent space to generate new data samples. The output of this process is a synthesized image, but to direct the model towards a specific type of output, conditioning is employed. The authors implement a conditioning mechanism using cross-attention, where the latent space is modulated based on encoded conditions, such as text descriptions.

One of the applications they showed is super-resolution. To achieve this, the authors train a separate autoencoder capable of operating at a higher resolution. The high-resolution decoder is then fed with the output from the diffusion model's decoder. Unlike the diffusion model's decoder, which transitions from latent space to pixel space, this high-resolution decoder operates entirely within pixel space, enhancing the detail and clarity of the final image output.

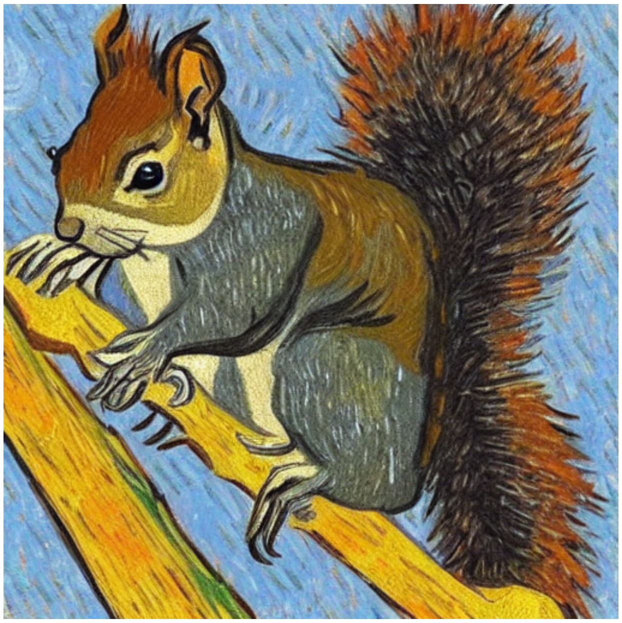

Figure 2. ”An image of a squirrel in Vincent van Gogh style”

5. Experiments

We use a Stable Diffusion model develop by Hugging Face, our obtaining results attached below with their queries. Figure 2 and Figure 3.

6. Conclusions

In conclusion, the exploration of latent diffusion models (LDMs) in this study underscores their significance as a pioneering advancement in generative modeling within computer vision. LDMs offer a compelling alternative to traditional methods such as Generative Adversarial Networks (GANs), showcasing superior capabilities in generating high-quality, detailed samples. Their stability in the training process, facilitated by a maximum likelihood estimation framework, further enhances their appeal, particularly in tasks necessitating fine details and sharpness, such as high-resolution image synthesis. Moreover, the versatility of LDMs extends beyond image data, as they can be effectively applied to a diverse range of data types, including audio.

Despite these notable strengths, the computational intensity and data requirements of LDMs present significant challenges, especially for users with limited resources.

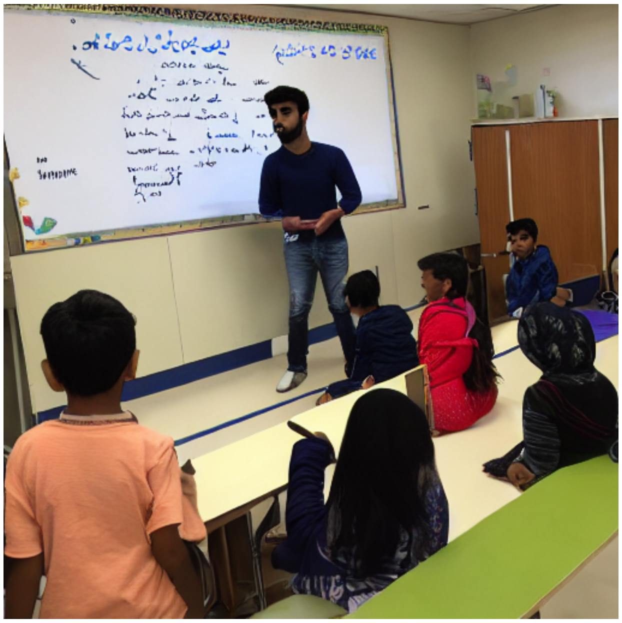

Figure 3. "A classroom in the spring where two students named Taha and Mahshad (a boy and a girl) are presenting a presentation about stable diffusion and a kind perfectionist professor is listening to them and giving them good score"

However, the immense potential of LDMs to push the boundaries of generative AI cannot be overlooked. As research progresses, further optimizations aimed at reducing computational demands and enhancing accessibility are anticipated. These advancements are poised to broaden the applicability and adoption of LDMs across various industries, catalyzing innovation and development in generative modeling.

The comparison with GANs provides valuable insights into the trade-offs between quality, training stability, and computational efficiency, guiding future research directions in the field of generative modeling. Moving forward, promising avenues for continued work include the exploration of novel architectures, optimization techniques, and applications of LDMs, as well as the development of scalable solutions to address computational constraints. By addressing these challenges and capitalizing on the strengths of LDMs, the field stands to benefit from enhanced generative modeling capabilities, with far-reaching implications for numerous domains within computer vision and beyond.

References

[1] Minshuo Chen, Song Mei, Jianqing Fan, and Mengdi Wang. An overview of diffusion models: Applications, guided generation, statistical rates and optimization. *arXiv preprint arXiv:2404.07771*, 2024. 1

[2] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 3

[3] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3

[4] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021. 3

[5] 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*, pages 10684–10695, 2022. 3

[6] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 3

[7] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4):1–39, 2023. 1