교육학석사 학위논문

Understanding the Latent Space of Diffusion Models through the Lens of Riemannian Geometry

리만 기하학을 통한 확산 모형의 잠재 공간에 대한 이해

August 2024

서울대학교 대학원

교육대학원 · 과학교육과 물리전공

박 용 현

Understanding the Latent Space of Diffusion Models through the Lens of Riemannian Geometry

리만 기하학을 통한 확산 모형의 잠재 공간에 대한 이해

지도교수 조 정 효

이 논문을 교육학석사 학위 논문으로 제출함

2024년 8월

서울대학교 대학원

교육대학원 · 과학교육과 물리전공

박 용 현

박용현의 교육학석사 학위 논문을 인준함

2024년 8월

위 원 장	채 승 철	(인)
부위원장	이 경 호	(인)
위 원	조 정 효	(인)

국 문 초 록

확산 모델의 성공에도 불구하고, 우리는 여전히 그 잠재 공간에 대한 철저한 이해가 부족한 실정이다. 본 연구에서는 잠재 공간 $\mathbf{x}_t \in \mathcal{X}$ 를 이해하기 위해 기하학적 관점에서 분석을 수행하였다. 우리의 접근 방식은 모델 내부 표상과 연관된 풀백 메트릭 (pullback metric) 을 활용하여 $\mathcal{X}$ 내의 국소 잠재 기저를 도출하는 것이다. 주목할 만한 점은, 우리가 발견한 국소 잠재 기저가 특정 타임스텝에서 기저 벡터를 따라 잠재 변수 $\mathbf{x}_t$ 를 이동시킴으로써 이미지 편집 기능을 가능케 한다는 것이다. 또한, 확산 모델의 생성 과정에 걸쳐 잠재 변수 주변 기하학적 구조가 어떻게 변화하고 서로 다른 텍스트 조건에 따라 어떻게 달라지는지 분석하였다. 이를 통해 이미 알려진 굵직한 특성으로부터 세밀한 질감으로의 생성 현상을 확인함과 동시에, 잠재 변수 위치에 따른 공간 불일치, 데이터셋 복잡성의 영향, 그리고 시간에 따른 텍스트 프롬프트의 영향 변화와 같은 새로운 통찰을 얻을 수 있었다. 우리가 아는 한, 본 논문은 $\mathbf{x}$ 조작 및 추가 훈련 없이 특정 타임스텝 t 에서 단 한 번의 편집만으로 가능한 방법, 그리고 확산 모형의 잠재 구조에 대한 철저한 분석을 최초로 제시하고 있다.

주요어: 확산모형, 기하 구조, 이미지 편집

학 번: 2024-25129

Abstract

Despite the success of diffusion models (DMs), we still lack a thorough understanding of their latent space. To understand the latent space $\mathbf{x}_t \in \mathcal{X}$, we analyze them from a geometrical perspective. Our approach involves deriving the local latent basis within $\mathcal{X}$ by leveraging the pullback metric associated with their encoding feature maps. Remarkably, our discovered local latent basis enables image editing capabilities by moving $\mathbf{x}_t$, the latent space of DMs, along the basis vector at specific timesteps. We further analyze how the geometric structure of DMs evolves over diffusion timesteps and differs across different text conditions. This confirms the known phenomenon of coarse-to-fine generation, as well as reveals novel insights such as the discrepancy between $\mathbf{x}_t$ across timesteps, the effect of dataset complexity, and the time-varying influence of text prompts. To the best of our knowledge, this paper is the first to present image editing through $\mathbf{x}$-space traversal, editing only once at specific timestep t without any additional training, and providing thorough analyses of the latent structure of DMs.

Keywords: Diffusion model, Geometry, Image editing

Student Number: 2024-25129

Contents

List of Figures

List of Tables

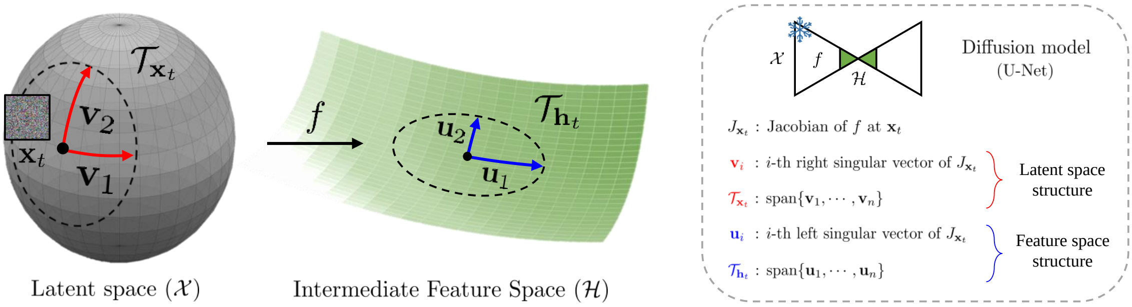
Chapter 1

Introduction

The diffusion models (DMs) are powerful generative models that have demonstrated impressive performance [Ho et al., 2020, Song et al., 2020a,b, Dhariwal and Nichol, 2021, Nichol and Dhariwal, 2021]. DMs have shown remarkable applications, including text-to-image synthesis [Ramesh et al., 2022, Rombach et al., 2022, Balaji et al., 2022, Nichol et al., 2021], inverse problems [Chung et al., 2022, Lugmayr et al., 2022], and image editing [Hertz et al., 2022, Tumanyan et al., 2022, Parmar et al., 2023, Mokady et al., 2022].

Despite their achievements, the research community lacks a comprehensive understanding of the latent space of DMs and its influence on the generated results. So far, the completely diffused images are considered as latent variables but it does not have useful properties for controlling the results. For example, traversing along a direction from a latent produces weird changes in the results. Fortunately, Kwon et al. [2022] consider the intermediate feature space of the diffusion kernel, referred to as $\mathcal{H}$, as a semantic latent space and show its usefulness on controlling generated images. In the similar sense, some works investigate the feature maps of the self-attention or cross-attention operations for controlling the results [Hertz et al., 2022, Tumanyan et al., 2022, Parmar et al., 2023], improving sample quality [Chefer et al., 2023], or downstream tasks such as semantic segmentation [Ma et al., 2023, Xu et al., 2023].

Still, the structure of the space $\mathcal{X}_t$ where latent variables $\{\mathbf{x}_t\}$ live remains unexplored despite its crucial role in understanding DMs. It is especially challenging because 1) the model is trained to estimate the forward noise which does not depend on the input, as opposed to other typical supervisions such as classification or similarity, and 2) there are lots of latent variables over multiple recursive timesteps. In this paper, we aim to analyze $\mathcal{X}$



(a) Conceptual illustration (b) Notation

Figure 1.1: **Conceptual illustration of local geometric structure.** (a) The local basis $\{\mathbf{v}_1, \mathbf{v}_2, \cdots\}$ of the local latent subspace $\mathcal{T}_{\mathbf{x}_t}$ within the latent space $\mathcal{X}$ is interlinked with the local basis $\{\mathbf{u}_1, \mathbf{u}_2, \cdots\}$ of the local tangent space $\mathcal{T}_{\mathbf{h}_t}$ in the feature space $\mathcal{H}$. (b) The derivation of these local bases is facilitated through the singular value decomposition (SVD) of the Jacobian, which emanates from the U-Net responsible for encoding the feature map f, linking $\mathcal{X}$ and $\mathcal{H}$.

in conjunction with its corresponding representation $\mathcal{H}$, by incorporating a local geometry to $\mathcal{X}$ using the concept of a *pullback metric* in Riemannian geometry.

First, we discover the local latent basis for $\mathcal{X}$ and the corresponding local tangent basis for $\mathcal{H}$. The local basis is obtained by performing singular value decomposition (SVD) of the Jacobian of the mapping from $\mathcal{X}$ to $\mathcal{H}$. To validate the discovered local latent basis, we demonstrate that walking along the basis can edit real images in a semantically meaningful way. Furthermore, we can use the discovered local latent basis vector to edit other samples by using parallel transport, when they exhibit comparable local geometric structures. Note that existing editing methods manipulate the self-attention map or cross-attention map over multiple timesteps [Hertz et al., 2022, Tumanyan et al., 2022, Parmar et al., 2023]. On the other hand, we manipulate only $\mathbf{x}_t$ once at a specific timestep.

Second, we investigate how the latent structures differ across different timesteps and samples as follows. The frequency domain of the local latent basis shifts from low-frequency to high-frequency along the generative process. We explicitly confirm it using power spectral density analysis. The difference between local tangent spaces of different samples becomes larger along the generative process. The local tangent spaces at various diffusion timesteps are similar to each other if the model is trained on aligned datasets such as CelebA-HQ or Flowers. However, this homogeneity does not occur on complex datasets such as ImageNet.

Finally, we examine how the prompts affect the latent structure of text-to-image DMs as follows. Similar prompts yield similar latent structures. Specifically, we find a positive correlation between the similarity of prompts and the similarity of local tangent spaces. The influence of text on the local tangent space becomes weaker along the generative process.

Our work examines the geometry of $\mathcal{X}$ and $\mathcal{H}$ using Riemannian geometry. We discover the latent structure of $\mathcal{X}$ and how it evolves during the generative process and is influenced by prompts. This geometric exploration deepens our understanding of DMs.

Chapter 2

Related Work

2.1 Diffusion Models.

Recent advances in DMs make great progress in the field of image synthesis and show state-of-the-art performance [Sohl-Dickstein et al., 2015, Ho et al., 2020, Song et al., 2020a]. An important subject in the diffusion model is the introduction of gradient guidance, including classifier-free guidance, to control the generative process [Dhariwal and Nichol, 2021, Sehwag et al., 2022, Avrahami et al., 2022, Liu et al., 2021, Nichol et al., 2021, Rombach et al., 2022]. The work by Song et al. [2020b] has facilitated the unification of DMs with score-based models using SDEs, enhancing our understanding of DMs as a reverse diffusion process. However, the latent space is still largely unexplored, and our understanding is limited.

2.2 The study of latent space in GANs.

The study of latent spaces has gained significant attention in recent years. In the field of Generative Adversarial Networks (GANs), researchers have proposed various methods to manipulate the latent space to achieve the desired effect in the generated images [Ramesh et al., 2018, Patashnik et al., 2021, Abdal et al., 2021, Härkönen et al., 2020, Shen and Zhou, 2021, Yüksel et al., 2021, Pan et al., 2023]. More recently, several studies [Zhu et al., 2021, Choi et al., 2021a] have examined the geometrical properties of latent space in GANs and utilized these findings for image manipulations. These studies bring the advantage of better understanding the characteristics of the latent space and facilitating the analysis and utilization of GANs. In contrast, the latent space of DMs remains poorly understood, making it difficult to fully utilize their capabilities.

2.3 Image manipulation in DMs.

Early works include Choi et al. [2021b] and Meng et al. [2021] have attempted to manipulate the resulting images of DMs by replacing latent variables, allowing the generation of desired random images. However, due to the lack of semantics in the latent variables of DMs, current approaches have critical problems with semantic image editing. Alternative approaches have explored the potential of using the feature space within the U-Net for semantic image manipulation. For example, Kwon et al. [2022] have shown that the bottleneck of the U-Net, $\mathcal{H}$, can be used as a semantic latent space. Specifically, they used CLIP [Radford et al., 2021] to identify directions within $\mathcal{H}$ that facilitate genuine image editing. Baranchuk et al. [2021] and Tumanyan et al. [2022] use the feature map of the U-Net for semantic segmentation and maintaining the structure of generated images. Unlike previous works, our editing method finds the editing direction without supervision, and directly traverses the latent variable along the latent basis.

2.4 Riemannain Geometry.

Some studies have applied Riemannian geometry to analyze the latent spaces of deep generative models, such as Variational Autoencoders (VAEs) and GANs [Arvanitidis et al., 2017, Shao et al., 2018, Chen et al., 2018, Arvanitidis et al., 2020, Lee and Park, 2023, Lee et al., 2022, Yonghyeon et al., 2021]. Shao et al. [2018] proposed a pullback metric on the latent space from image space Euclidean metric to analyze the latent space's geometry. This method has been widely used in VAEs and GANs because it only requires a differentiable map from latent space to image space. However, no studies have investigated the geometry of latent space of DMs utilizing the pullback metric.

Chapter 3

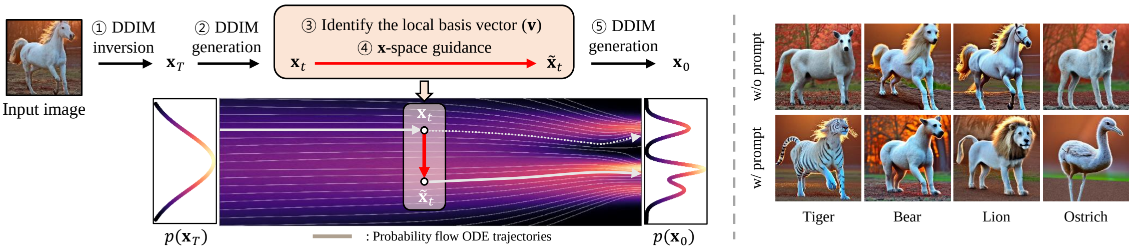
Method

In this chapter, we explain how to extract a latent structure of $\mathcal{X}$ using differential geometry. First, we introduce a key concept in our method: the *pullback metric*. Next, by adopting the local Euclidean metric of $\mathcal{H}$ and utilizing the pullback metric, we discover the local latent basis of the $\mathcal{X}$. Moreover, although the direction we found is *local*, we show how it can be applied to other samples via parallel transport. Finally, we introduce $\mathbf{x}$-space guidance for editing data in the $\mathcal{X}$ to enhance the quality of image generation.

3.1 Pullback metric

We consider a curved manifold, $\mathcal{X}$, where our latent variables $\mathbf{x}_t$ exist. The differential geometry represents $\mathcal{X}$ through patches of tangent spaces, $\mathcal{T}_{\mathbf{x}}$, which are vector spaces defined at each point $\mathbf{x}$. Then, all the geometrical properties of $\mathcal{X}$ can be obtained from the inner product of $||d\mathbf{x}||^2 = \langle d\mathbf{x}, d\mathbf{x} \rangle_{\mathbf{x}}$ in $\mathcal{T}_{\mathbf{x}}$. However, we do not have any knowledge of $\langle d\mathbf{x}, d\mathbf{x} \rangle_{\mathbf{x}}$. It is definitely not a Euclidean metric. Furthermore, samples of $\mathbf{x}_t$ at intermediate timesteps of DMs include inevitable noise, which prevents finding semantic directions in $\mathcal{T}_{\mathbf{x}}$.

Fortunately, Kwon et al. [2022] observed that $\mathcal{H}$, defined by the bottleneck layer of the U-Net, exhibits local linear structure. This allows us to adopt the Euclidean metric on $\mathcal{H}$. In differential geometry, when a metric is not available on a space, *pullback metric* is used. If a smooth map exists between the original metric-unavailable domain and a metric-available codomain, the pullback metric is used to measure the distances in the domain space. Our idea is to use the pullback Euclidean metric on $\mathcal{H}$ to define the distances



(a) Method overview (b) Editing results

Figure 3.1: **Image editing with the discovered latent basis.** (a) Schematic depiction of our image editing procedure. ① An input image is subjected to DDIM inversion, resulting in an initial noisy sample $\mathbf{x}_T$. ② The sample $\mathbf{x}_T$ is progressively denoised until reaching the point t through DDIM generation. ③ Subsequently, the local latent basis $\{\mathbf{v}_1, \cdots, \mathbf{v}_n\}$ is identified by using the pullback metric. ④ This enables the manipulation of the sample $\mathbf{x}_t$ along one of the basis vectors using $\mathbf{x}$-space guidance. ⑤ The DDIM generation concludes with the progression from the modified latent variable $\tilde{\mathbf{x}}_t$. (b) Examples of edited images using a selected basis vector. The latent basis vector could be conditioned on prompts and it facilitates text-aligned manipulations.

between the samples in $\mathcal{X}$.

DMs are trained to infer the noise ϵ_t from a latent variable $\mathbf{x}_t$ at each diffusion timestep t. Each $\mathbf{x}_t$ has a different internal representation $\mathbf{h}_t$, the bottleneck representation of the U-Net, at different t's. The differentiable map between $\mathcal{X}$ and $\mathcal{H}$ is denoted as $f : \mathcal{X} \to \mathcal{H}$. Hereafter, we refer to $\mathbf{x}_t$ as $\mathbf{x}$ for brevity unless it causes confusion. It is important to note that our method can be applied at any timestep in the denoising process. We consider a linear map, $\mathcal{T}_\mathbf{x} \to \mathcal{T}_\mathbf{h}$, between the domain and codomain tangent spaces. This linear map can be described by the *Jacobian* $J_\mathbf{x} = \nabla_\mathbf{x}\mathbf{h}$ which determines how a vector $\mathbf{v} \in \mathcal{T}_\mathbf{x}$ is mapped into a vector $\mathbf{u} \in \mathcal{T}_\mathbf{h}$ by $\mathbf{u} = J_\mathbf{x}\mathbf{v}$.

Using the local linearity of $\mathcal{H}$, we assume the metric, $||d\mathbf{h}||^2 = \langle d\mathbf{h}, d\mathbf{h}\rangle_\mathbf{h} = d\mathbf{h}^\mathsf{T} d\mathbf{h}$ as a usual dot product defined in the Euclidean space. To assign a geometric structure to $\mathcal{X}$, we use the pullback metric of the corresponding $\mathcal{H}$. In other words, the norm of $\mathbf{v} \in \mathcal{T}_\mathbf{x}$ is measured by the norm of corresponding codomain tangent vector:

$$||\mathbf{v}||^2_{\text{pb}} \triangleq \langle \mathbf{u}, \mathbf{u}\rangle_\mathbf{h} = \mathbf{v}^\mathsf{T} {J_\mathbf{x}}^\mathsf{T} J_\mathbf{x}\mathbf{v}. \tag{3.1}$$

3.2 Discovering the latent basis of DMs

Using the pullback metric, we define the local latent vector $\mathbf{v} \in \mathcal{T}_\mathbf{x}$ that shows a large variability in $\mathcal{T}_\mathbf{h}$. We find a unit vector $\mathbf{v}_1$ that maximizes $||\mathbf{v}||^2_{\text{pb}}$. By maximizing $||\mathbf{v}||^2_{\text{pb}}$

while remaining orthogonal to $\mathbf{v}_1$, one can obtain the second unit vector $\mathbf{v}_2$. This process can be repeated to have n latent directions of $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n\}$ in $\mathcal{T}_\mathbf{x}$. In practice, $\mathbf{v}_i$ corresponds to the i-th right singular vector from the singular value decomposition (SVD) of $J_\mathbf{x} = U\Lambda V^\mathsf{T}$, i.e., $J_\mathbf{x}\mathbf{v}_i = \Lambda_i \mathbf{u}_i$. Since the Jacobian of too many parameters is not tractable, we use a *power method* [Golub and Van Loan, 2013, Miyato et al., 2018, Haas et al., 2023] to approximate the SVD of $J_\mathbf{x}$ (See Appendix D for the time complexity and Appendix F for the detailed algorithm).

Henceforth, we refer to $\mathcal{T}_\mathbf{x}$ as a local latent subspace, and $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n\}$ as the corresponding local latent basis.

$$\mathcal{T}_\mathbf{x} \triangleq \mathrm{span}\{\mathbf{v}_1, \mathbf{v}_2 \cdots, \mathbf{v}_n\}, \text{ where } \mathbf{v}_i \text{ is } i\text{-th right singular vector of } J_\mathbf{x}. \tag{3.2}$$

Using the linear transformation between $\mathcal{T}_\mathbf{x}$ and $\mathcal{T}_\mathbf{h}$ via the Jacobian $J_\mathbf{x}$, one can also obtain corresponding directions in $\mathcal{T}_\mathbf{h}$. In practice, $\mathbf{u}_i$ corresponds to the i-th left singular vector from the SVD of $J_\mathbf{x}$. After selecting the top n (e.g., $n = 50$) directions of large eigenvalues, we can approximate any vector in $\mathcal{T}_\mathbf{h}$ with a finite basis, $\{\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_n\}$. When we refer to a local tangent space henceforth, it means the n-dimensional low-rank approximation of the original tangent space.

$$\mathcal{T}_\mathbf{h} \triangleq \mathrm{span}\{\mathbf{u}_1, \mathbf{u}_2 \cdots, \mathbf{u}_n\}, \text{ where } \mathbf{u}_i \text{ is the } i\text{-th left singular vector of } J_\mathbf{x}. \tag{3.3}$$

The collection of local latent basis vectors, $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n\}$, obtained through our proposed method, can be interpreted as a *signal* that the model is highly response to for a given $\mathbf{x}$. On the other hand, the basis of the local tangent space, denoted as $\{\mathbf{u}_1, \mathbf{u}_2 \cdots, \mathbf{u}_n\}$, can be viewed as the corresponding *representation* associated with the signal.

In Stable Diffusion, the prompt also influences the Jacobian, which means that the local basis also depends on it. We can utilize any prompt to obtain a local latent basis, and different prompts create distinct geometrical structures. For the sake of brevity, we will omit the word *local* unless it leads to confusion.

3.3 Generating edited images with x-space guidance

A naïve approach for manipulating a latent variable $\mathbf{x}$ using a latent vector $\mathbf{v}$ is through simple addition, specifically $\mathbf{x}+\gamma\mathbf{v}$. However, using the naïve approach sometime leads to noisy image generation. To address this issue, instead of directly using the basis for manipulation, we use a basis vector that has passed through the decoder once for manipulation. The $\mathbf{x}$-space guidance is defined as follows

$$\tilde{\mathbf{x}}_{\mathrm{XG}} = \mathbf{x} + \gamma[\epsilon_\theta(\mathbf{x}+\mathbf{v}) - \epsilon_\theta(\mathbf{x})] \tag{3.4}$$

where γ is a hyper-parameter controlling the strength of editing and ϵ_θ is a diffusion model. Equation 3.4 is inspired by classifier-free guidance, but the key difference is that it is directly applied in the latent space $\mathcal{X}$. Our $\mathbf{x}$-space guidance provides qualitatively similar results to direct addition, while it shows better fidelity. (See Appendix C for ablation study.)

3.4 The overall process of image editing

In this section, we summarize the entire editing process with five steps: 1) The input image is inverted into initial noise $\mathbf{x}_T$ using DDIM inversion. 2) $\mathbf{x}_T$ is gradually denoised until t through DDIM generation. 3) Identify local latent basis $\{\mathbf{v}_1, \cdots, \mathbf{v}_n\}$ using the pullback metric at t. 4) Manipulate $\mathbf{x}_t$ along the one of basis vectors using the $\mathbf{x}$-space guidance. 5) The DDIM generation is then completed with the modified latent variable $\tilde{\mathbf{x}}_t$. Figure 3.1 illustrates the entire editing process.

In the context of a text-to-image model, such as Stable Diffusion, it becomes possible to include textual conditions while deriving local basis vectors. Although we do not use any text guidance during DDIM inversion and generation, a local basis with text conditions enables semantic editing that matches the given text. Comprehensive experiments can be found in Section 4.1.

It is noteworthy that our approach needs no extra training and simplifies image editing by only adjusting the latent variable within a single timestep.

3.5 Editing various samples with parallel transport

Let us consider a scenario where our aim is to edit ten images, changing straight hair to curly hair. Due to the nature of the unsupervised image editing method, it is becomes imperative to manually check the semantic relevance of the latent basis vector in the edited results. Thus, to edit every samples, we have to manually find a straight-to-curly basis vector for individual samples.

One way to alleviate this tedious task is to apply the curly hair vector obtained from one image to other images. However, the basis vector $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}$ obtained at $\mathbf{x}$ cannot be used for the other sample $\mathbf{x}'$ because $\mathbf{v} \notin \mathcal{T}_{\mathbf{x}'}$. Thus, in order to apply the direction we obtained to another sample, it is necessary to relocate the extracted direction to a new tangent space.

To achieve this, we use parallel transport that moves $\mathbf{v}_i$ onto the new tangent space $\mathcal{T}_{\mathbf{x}'}$. In the realm of differential geometry, parallel transport is a technique to relocate a tangent vector to different position with minimal direction change, while keeping the vector tangent on the manifold [Shao et al., 2018]. It is notable that in curved space, parallel transport can significantly modify the original vector. Therefore, it is beneficial to apply the parallel transport in $\mathcal{H}$, where relatively flatter than $\mathcal{X}$.

The process of moving a tangent vector $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}$ to $\mathbf{v}' \in \mathcal{T}_{\mathbf{x}'}$ through parallel transport in $\mathcal{H}$ is summarized as follows. First, we convert the latent direction $\mathbf{v}_i \in \mathcal{T}_{\mathbf{x}}$ to the corresponding direction of $\mathbf{u}_i \in \mathcal{T}_{\mathbf{h}}$. Second, we apply the parallel transport $\mathbf{u}_i \in \mathcal{T}_{\mathbf{h}}$ to $\mathbf{u}'_i \in \mathcal{T}_{\mathbf{h}'}$, where $\mathbf{h}' = f(\mathbf{x}')$. In the general case, parallel transport involves iterative projection and normalization on the tangent space along the path connecting two points [Shao et al., 2018]. However, in our case, we assume that $\mathcal{H}$ has Euclidean geometry. Therefore, we move $\mathbf{u}$ directly onto $\mathcal{T}_{\mathbf{h}'}$ through projection, without the need for an iterative process. Finally, transform $\mathbf{u}'_i$ into $\mathbf{v}'_i \in \mathcal{X}$. Using this parallel transport of $\mathbf{v}_i \to \mathbf{v}'_i$ via $\mathcal{H}$, we can apply the local latent basis obtained from $\mathbf{x}$ to edit or modify the input $\mathbf{x}'$.

Chapter 4

Experiment

In this chapter, we analyze the geometric structure of DMs with our method. § 4.1 demonstrates that the latent basis found by our method can be used for image editing. In § 4.2, we investigate how the geometric structure of DMs evolves as the generative process progresses. Lastly, in § 4.3, we examine how the geometric properties of the text-condition model change with a given text.

The implementation details of our work are provided on Appendix B. The source code for our experiments can be found at `https://github.com/enkeejunior1/Diffusion-Pullback`.

4.1 Image editing with the latent basis

In this section, we demonstrate the capability of our discovered latent basis for image editing. To extract the latent variables from real images for editing purposes, we use DDIM inversion. In experiments with Stable Diffusion (SD), we do not use guidance, i.e., unconditional sampling, for both DDIM inversion and DDIM sampling processes. This ensures that our editing results solely depend on the latent variable, not on other factors such as prompt conditions.

Figures 3.1 and 4.1 illustrate the example results edited by the latent basis found by our method. The latent basis clearly contains semantics such as age, gender, species, structure, and texture. Note that editing at timestep T yields coarse changes such as age and species. On the other hand, editing at timestep $0.6T$ leads to fine changes, such as nose color and facial expression.

Figure 4.2 demonstrates the example results edited by the various latent basis vectors.

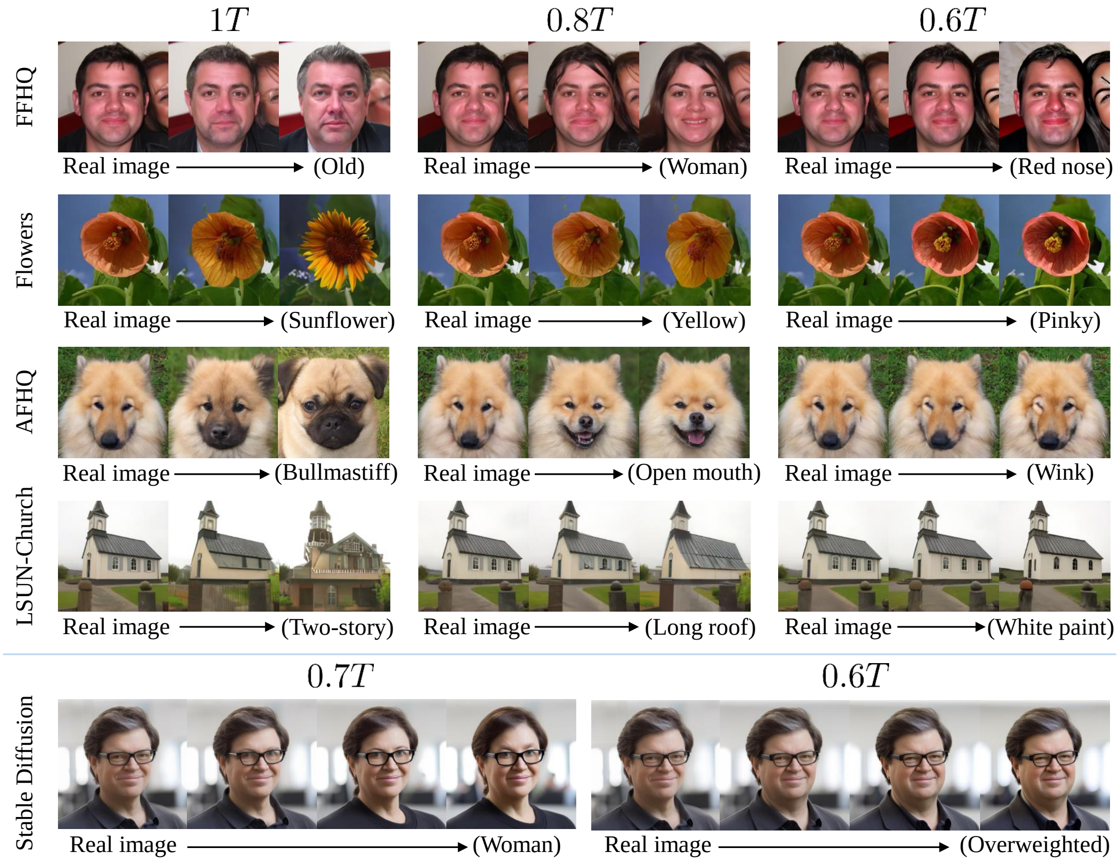


Figure 4.1: **Examples of image edition using the latent basis.** The attributes are manually interpreted since the editing directions are not intentionally supervised. For Stable Diffusion, we used an empty string as a prompt. Each column represents edits made at different diffusion timesteps ($0.6T$, $0.8T$, and T for the unconditional diffusion model; $0.6T$ and $0.7T$ for Stable Diffusion).

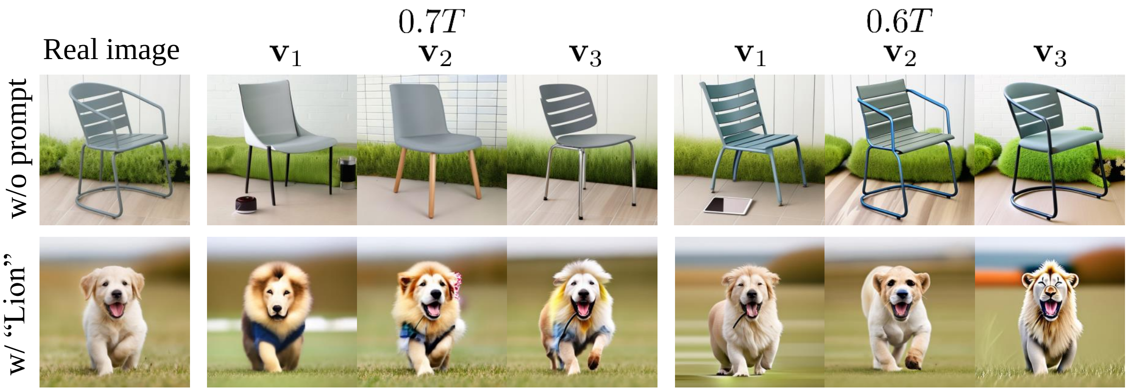


Figure 4.2: **Examples of image edition using top-3 latent basis vectors.** Each column is edited using a different latent vector $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$. Each group of columns represents edits made at different diffusion timesteps ($0.6T$ and $0.7T$). Notably, when given the "Lion" prompt, it is evident that all the top latent basis vectors align with the direction of the prompt.

Interestingly, using the text "lion" as a condition, the entire latent basis captures lion-related attributes. Furthermore, Figure 4.3 shows that the latent basis aligns with the text

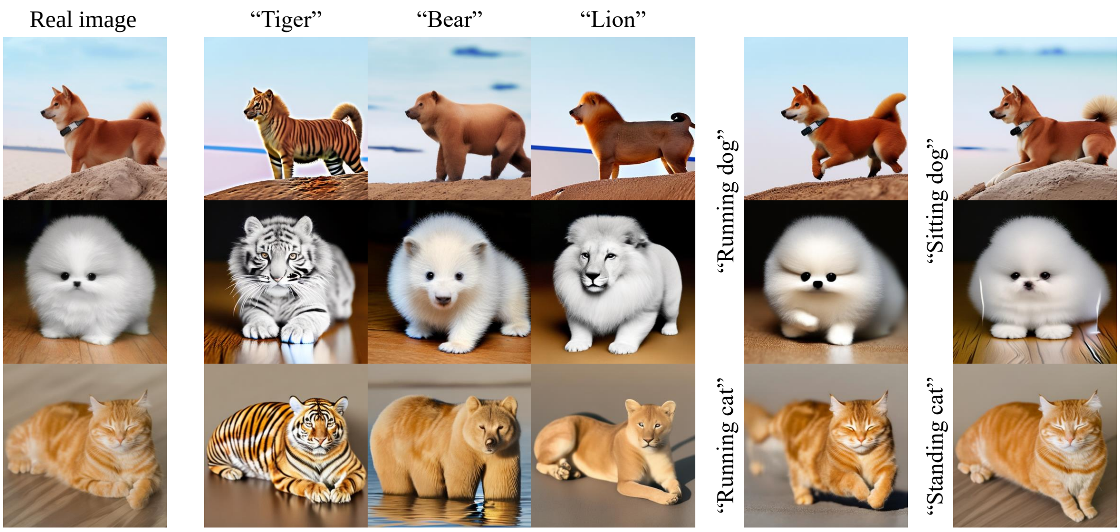


Figure 4.3: **Examples of image edition using latent basis vectors discovered with various prompts.** Each column is edited using the latent basis vector obtained from a different text prompt. Importantly, our method employs each prompt only once to derive the local latent basis.

not only in terms of object types but also in relation to pose or action. For a qualitative comparison with other state-of-the-art image editing methods, refer to Appendix D. For more examples of editing results, refer to Appendix G.

4.2 Evolution of latent structures during generative processes

In this section, we demonstrate how the latent structure evolves during the generative process and identify three trends. 1) The frequency domain of the latent basis changes from low to high frequency. It agrees with the previous observation that DMs generate samples in coarse-to-fine manner. 2) The difference between the tangent spaces of different samples increases over the generative process. It implies finding generally applicable editing direction in latent space becomes harder in later timesteps. 3) The differences of tangent spaces between timesteps depend on the complexity of the dataset.

4.2.1 Latent bases gradually evolve from low- to high-frequency structures.

Figure 4.4 is the power spectral density (PSD) of the discovered latent basis over various timesteps. The early timesteps contain a larger portion of low frequency than the later

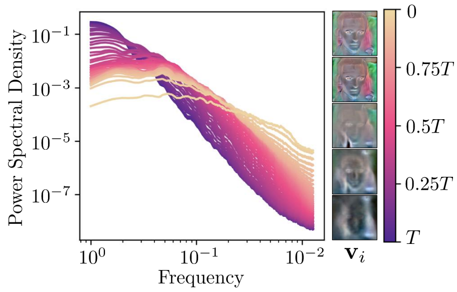


Figure 4.4: **Power Spectral Density (PSD) of latent basis.** The PSD at $t = T$ (purple) exhibits a greater proportion of low-frequency signals, while the PSD at smaller t (beige) reveals a larger proportion of high-frequency signals. The latent vectors $\mathbf{v}_i$ are min-max normalized for visual clarity.

timesteps and the later timesteps contain a larger portion of high frequency.

This suggests that the model focuses on low-frequency signals at the beginning of the generative process and then shifts its focus to high-frequency signals over time. This result strengthens the common understanding about the coarse-to-fine behavior of DMs over the generative process [Choi et al., 2022, Daras and Dimakis, 2022].

4.2.2 The discrepancy of tangent spaces from different samples increases along the generative process.

To investigate the geometry of the tangent basis, we employ a metric on the Grassmannian manifold. The Grassmannian manifold is a manifold where each point is a vector space, and the metric defined above represents the distortion across various vector spaces. We use *geodesic metric* [Choi et al., 2021a, Ye and Lim, 2016] to define the discrepancy between two subspaces $\{\mathcal{T}^{(1)}, \mathcal{T}^{(2)}\}$:

$$D_{\text{geo}}(\mathcal{T}^{(1)}, \mathcal{T}^{(2)}) = \sqrt{\sum_k \theta_k^2}, \tag{4.1}$$

where θ_k denotes the k-th principle angle between $\mathcal{T}^{(1)}$ and $\mathcal{T}^{(2)}$. Intuitively, the concept of geodesic metric can be understood as an angle between two vector spaces. Here, the comparison between two different spaces was conducted for $\{\mathcal{T}_{\mathbf{h}_1}, \mathcal{T}_{\mathbf{h}_2}\}$. Unlike the $\mathcal{X}$, the $\mathcal{H}$ assumes a Euclidean space which makes the computation of geodesic metric that requires an inner product between tangent spaces easier. The relationship between tangent space and latent subspace is covered in more detail in Appendix E.

Figure 4.5 demonstrates that the tangent spaces of the different samples are the most

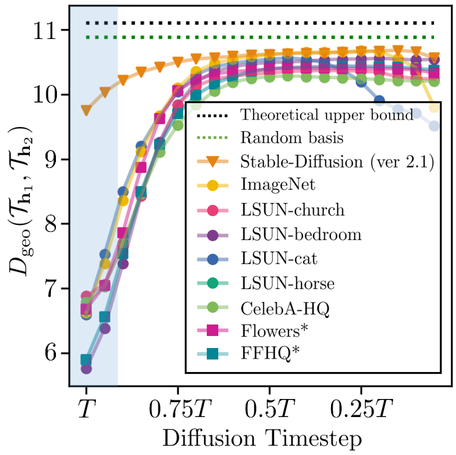


Figure 4.5: **Geodesic distance across tangent space of different samples at various diffusion timesteps.** Each point represents the average geodesic distance between pairs of 15 samples. It is notable that the similarity of tangent spaces among different samples diminishes as the generative process progresses.

similar at $t = T$ and diverge as timestep becomes zero.

Moreover, the similarity across tangent spaces allows us to effectively transfer the latent basis from one sample to another through parallel transport as shown in Figure 4.6. In T, where the tangent spaces are homogeneous, we consistently obtain semantically aligned editing results. On the other hand, parallel transport at $t = 0.6T$ does not lead to satisfactory editing because the tangent spaces are hardly homogeneous. Thus, we should examine the similarity of local subspaces to ensure consistent editing across samples.

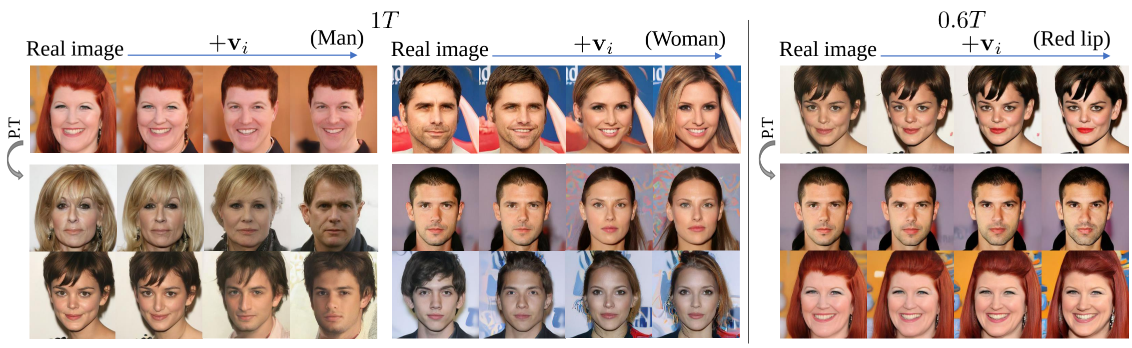


Figure 4.6: **Examples of image edition using parallel transport.** The first row demonstrates the results of editing with their respective latent vectors, while the subsequent rows exhibit the results of editing through the parallel transport (P.T) of the latent vectors used in the first row. The latent vector performs effectively when $t = T$ (left and middle), but comparatively less satisfactorily for $0.6T$ (right).

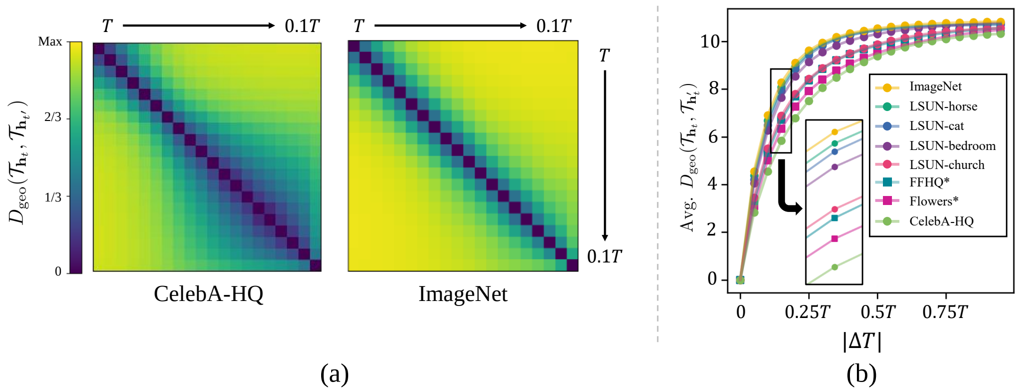


Figure 4.7: **Simpler datasets lead to more similar tangent spaces across diffusion timesteps.** (a) Distance matrix visualization of tangent space measured by geodesic metric across various timesteps. (b) Average geodesic distance based on timestep differences, indicating that the complexity of the dataset correlates with greater distances between tangent spaces.

4.2.3 DMs trained on simpler datasets exhibit more consistent tangent spaces over time.

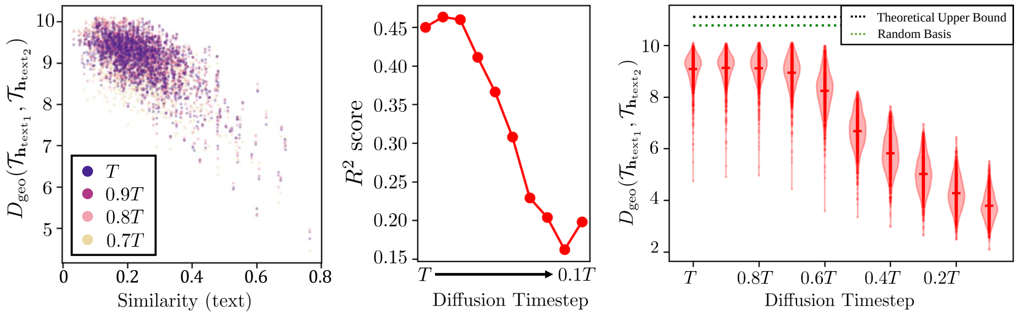
In Figure 4.7 (a), we provide a distance matrix of the tangent spaces across different timesteps, measured by the geodesic metric. We observe that the tangent spaces are more similar to each other when a model is trained on CelebA-HQ, compared to ImageNet. To verify this trend, we measure the geodesic distances between tangent spaces of different timesteps and plot the average distances of the same difference in timestep in Figure 4.7 (b). As expected, we find that DMs trained on datasets, that are generally considered simpler, have similar local tangent spaces over time.

4.3 Effect of conditioning prompts on the latent structure

In this section, we aim to investigate how prompts influence the generative process from a geometrical perspective. We randomly sampled 50 captions from the MS-COCO dataset [Lin et al., 2014] and used them as text conditions.

4.3.1 Similar text conditions induce similar tangent spaces.

In Figure 4.8 (a), we observe a negative correlation between the CLIP similarity of texts and the distance between tangent spaces. In other words, when provided with similar texts, the tangent spaces are more similar. The linear relationship between the text and the discrepancy of the tangent spaces is particularly strong in the early phase of the



(a) Tangent space distance and text similarity (b) Coefficient of determination (c) Geodesic distance across timesteps

Figure 4.8: **Similar prompts create similar tangent spaces, and the impact of the prompt decreases as the generative process progresses.** (a) The horizontal axis represents the CLIP similarity between two different prompts, while the vertical axis represents the geodesic distance in the tangent space from each prompt. Different colors represent various diffusion timesteps. A negative relationship is observed between prompt similarity and tangent space distance. (b) The R^2 score of the linear regression between clip similarity and geodesic distance of tangent spaces decreases throughout the generative process. (c) Each point represents the distance between tangent spaces created from different prompts. Until around $t = 0.7T$, the distance between tangent spaces is very large, but it gradually decreases thereafter. This indicates that the influence of the prompt on the tangent space diminishes.

generative process as shown by R^2 score in Figure 4.8 (b).

4.3.2 The generative process depends less on text conditions in later timesteps.

Figure 4.8 (c) illustrates the distances between local tangent spaces for given different prompts with respect to the timesteps. Notably, as the diffusion timestep approaches values below $0.7T$, the distances between the local tangent spaces start to decrease. It implies that the variation due to walking along the local tangent basis depends less on the text conditions, i.e., the text less influences the generative process, in later timesteps. It is a possible reason why the correlation between the similarity of prompts and the similarity of tangent spaces reduces over timesteps.

Chapter 5

Limitation

It is interesting that our latent basis usually conveys disentangled attributes even though we do not adopt attribute annotation to enforce disentanglement. We suppose that decomposing the Jacobian of the encoder in the U-Nets naturally yields disentanglement to some extent. However, it does not guarantee the perfect disentanglement and some directions are entangled. For example, the editing for beard converts a female subject to a male as shown in Figure 5.1 (a). This kind of entanglement often occurs in other editing methods due to the dataset bias: female faces seldom have beard.

While our method has shown effectiveness in Stable Diffusion, more research is needed to fully validate its potential. We have observed that some of the discovered latent vector occasionally leads to abrupt changes during the editing process in Stable Diffusion, as depicted in Figure 5.1 (b). This observation highlights the complex geometry of $\mathcal{X}$ in achieving seamless editing. Exploring this topic in future research is an interesting area to delve into.

Our approach is broadly applicable when the feature space in the DM adheres to a Euclidean metric, as demonstrated by $\mathcal{H}$. This characteristic has been observed in the context of U-Net within Kwon et al. [2022]. It would be intriguing to investigate if other architectural designs, especially those similar to transformer structures as introduced in [Peebles and Xie, 2023, Tevet et al., 2022], also exhibit a Euclidean metric.

Despite these limitations, our method provides a significant advance in the field of image editing for DMs, and provides a deep understanding of DM through several experiments.

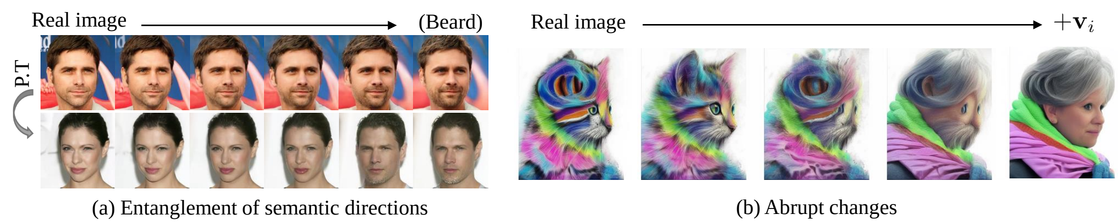


Figure 5.1: **Limitations.** (a) Entanglement between attributes due to dataset biases (b) Abrupt changes in Stable Diffusion.

Chapter 6

Conclusion

We have analyzed the latent space of DMs from a geometrical perspective. We used the pullback metric to identify the latent and tangent bases in $\mathcal{X}$ and $\mathcal{H}$. The latent basis found by the pullback metric allows editing images by traversal along the basis. We have observed properties of the bases in two aspects. First, we discovered that 1) the latent bases evolve from low- to high-frequency components; 2) the discrepancy of tangent spaces from different samples increases along the generative process; and 3) DMs trained on simpler datasets exhibit more consistent tangent spaces over timesteps. Second, we investigated how the latent structure changes based on the text conditions in Stable Diffusion, and discovered that similar prompts make tangent space analogous but its effect becomes weaker over timesteps. We believe that a better understanding of the geometry of DMs will open up new possibilities for adopting DMs in useful applications.

Appendices

A Societal Impacts & Ethics Statements

Our research endeavors to unravel the geometric structures of the diffusion model and facilitate high-quality image editing within its framework. While our primary application resides within the creative realm, it is important to acknowledge that image manipulation techniques, such as the one proposed in our method, hold the potential for misuse, including the dissemination of misinformation or potential privacy implications. Therefore, the continuous advancement of technologies aimed at thwarting or identifying manipulations rooted in generative models remains of utmost significance.

B Implementation details

Models and datasets We validate our method and provide analyses on various models using the official code and pre-trained checkpoints. The available combinations of the models and the datasets are: DDPM [Ho et al., 2020] on ImageNet [Deng et al., 2009], LSUN-church/bedroom/cat/horse [Yu et al., 2015], and CelebA-HQ [Karras et al., 2018]; and DDPM trained with *P2 weighting* [Choi et al., 2022] on FFHQ [Karras et al., 2019], Flowers [Yu et al., 2015] and AFHQ [Choi et al., 2020]. We also use Stable Diffusion (SD) version 2.1 [Rombach et al., 2022] for the text-conditional diffusion model.

For image editing, we use the official codes and pre-trained checkpoints for all baselines and keep the parameters *frozen*. For analysis, we compare models with the same diffusion scheduling (linear schedule) and resolutions (256^2) to ensure a fair comparison, except Stable Diffusion.

Table B1 summarizes various hyperparameter settings in our experiments. Specific details not covered in the main text are discussed in the following paragraphs.

Table A1: Hyper-parameter settings.

model	t_{edit}	inversion step	γ	n	t_{boost}
Stable Diffusion	$0.7T$	100	1	50	$\times$
	$0.6T$	100	2	50	$\times$
Unconditional DMs	T	100	0.5	50	$0.2T$
	$0.8T$	100	1	50	$0.2T$
	$0.6T$	100	4	50	$0.2T$

Edit timestep (t_{edit}) For unconditional DMs, we show the editing results at $t_{edit} \in \{T, 0.8T, 0.6T\}$, while for Stable Diffusion, we show the editing results at $t_{edit} \in \{0.7T, 0.6T\}$. Note that our method allows manipulation at any timestep.

Inversion step We conduct real image editing with DDIM inversion [Song et al., 2020a]. We set the number of steps to 100 for obtaining the latent variable $\mathbf{x}_T$ and all experiments.

x-space guidance scale (γ) The value of γ determines the magnitude of a single editing step by **x**-space guidance. Fortunately, through experimentation, we observed that the value of γ does not have a significant impact on image quality unless it is excessively large.

Low-rank approximation (n) We employ a low-rank approximation of the tangent space using $n = 50$ for all settings.

Quality boosting (t_{boost}) While DDIM alone already generates high-quality images, Karras et al. [2022] showed that including stochasticity in the process improves image quality and Kwon et al. [2022] suggest similar technique: adding stochasticity at the end of the generative process. We employ this technique in our experiments on every experiment after $t = 0.2T$, except Stable Diffusion.

Computing resource For power-method approximation with $n = 50$, it spends about 3-4 minutes on a single NVIDIA RTX 3090 (24GB). As n specifies the number of bases, it can be as small as a user want to use for image editing. Reducing n provides faster runtime, e.g., 10 seconds for $n = 3$.

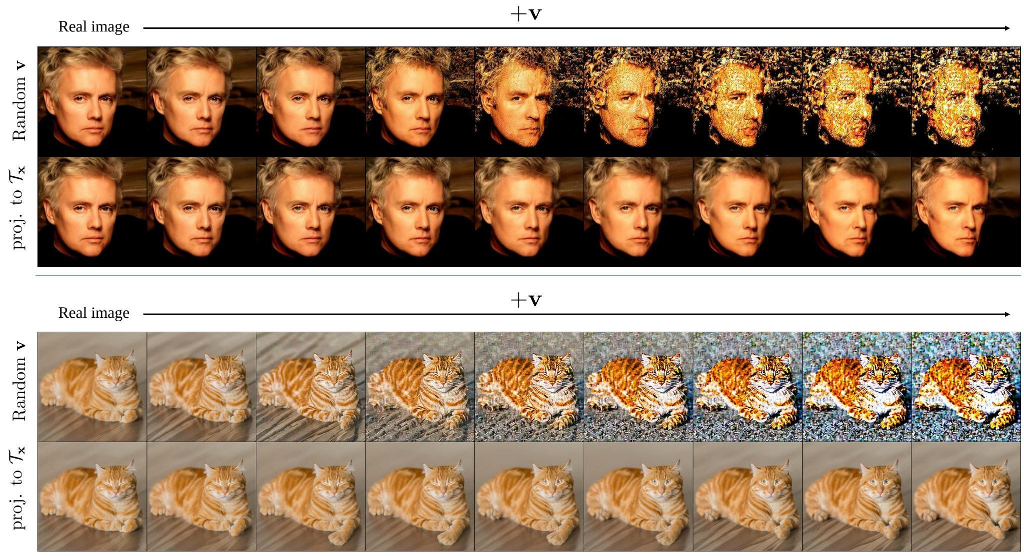


Figure A1: **Importance of the discovered latent directions.** Random direction experiments with CelebA-HQ pre-trained model (top) and Stable Diffusion (bottom). Adding random directions instead of latent directions severely distorts the resulting images. When we perform edits along the projection onto the latent subspace $\mathcal{T}_\mathbf{x}$, the generated image presents a semantically meaningful transformation.

C Ablation study

In this section, we validate our method with ablation study.

Random v To demonstrate the meaningfulness of the latent basis found by our method, we qualitatively compare its effect to naïve baseline: random directions. The first row in Figure A1 shows that manipulating the images with a random vector '$\mathbf{v}$' does not result in semantic editing but rather degrades images. The second row shows the results of projecting the random '$\mathbf{v}$' onto our obtained latent subspace. The projected results exhibit semantic manipulation such as pose changes without image distortion. It indicates that the found latent subspace captures semantics in the latent space effectively.

x-space guidance Figure A2 demonstrates the the effectiveness of $\mathbf{x}$-space guidance compared to a straightforward alternative: simple addition. First, $\mathbf{x}$-space guidance produces higher quality images with similar meaning. Especially Stable Diffusion apparently benefits from $\mathbf{x}$-space guidance regarding smoothness of the editing strength and artifacts. The difference is more significant at $t = 0.6T$. Note that the meaning of the same directions may slightly differ between the two settings due to non-linearity of the U-Net.

Currently, we do not have a deeper understanding of the underlying principles of $\mathbf{x}$-

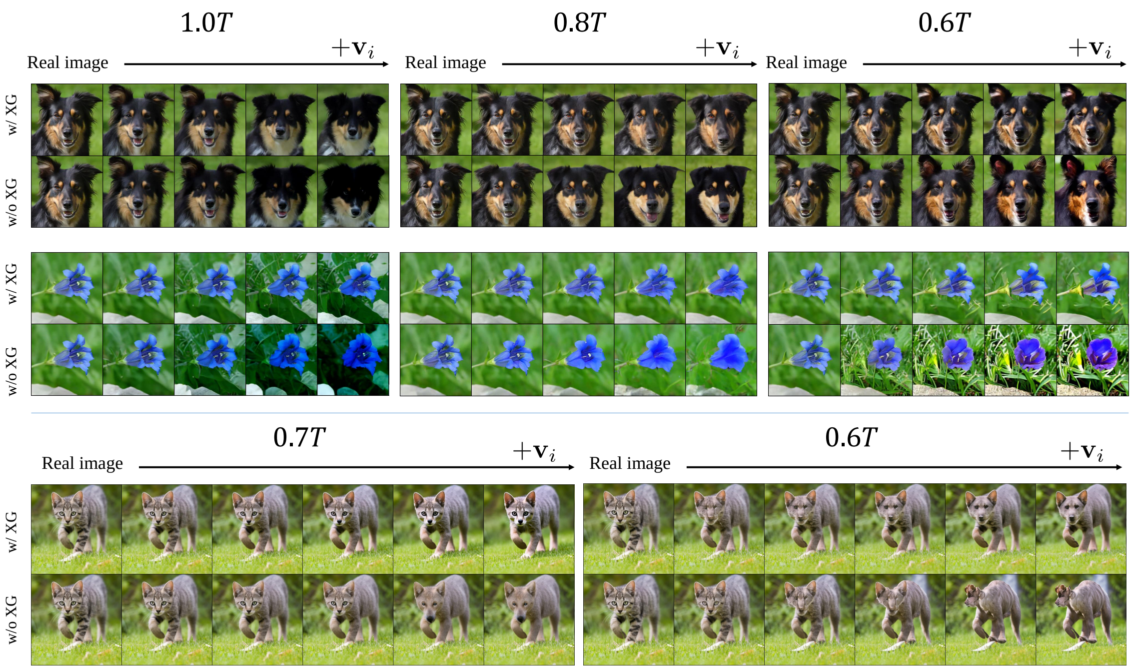


Figure A2: **Importance of the x-space guidance.** $\mathbf{x}$-space guidance experiments with AFHQ (top), Flowers pre-trained model (middle), and Stable Diffusion (bottom). $\mathbf{x}$-space guidance helps achieve qualitatively similar editing while preserving the content of the original image.

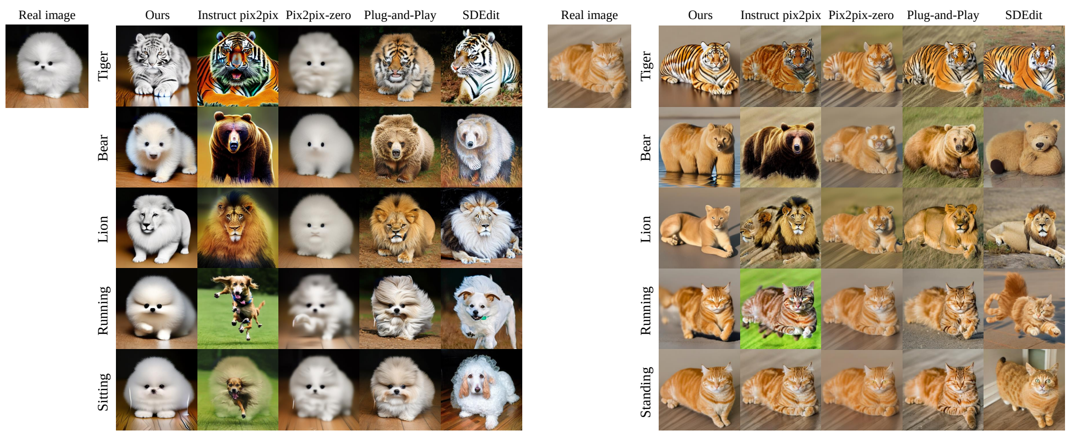


Figure A3: **Comparison with various image editing methods.** Our approach empowers image editing that aligns seamlessly with text conditions while upholding object identity. In contrast, alternative methods exhibit deficiencies such as: inadequate preservation of object structure (Instruct pix2pix), inefficacious manipulation (Pix2pix-zero), or challenges in maintaining identity fidelity (Plug-and-Play, SDEdit).

space guidance. Exploring the reasons behind its ability to improve manipulation quality would be an interesting direction for future work.

Table A1: **Comparisons of the time complexity of state-of-the-art editing methods**

Image Edit Method	Running time	Preprocessing
Ours	11 sec	N/A
SDEdit	4 sec	N/A
Pix2Pix-zero	25 sec	4 min
PnP	10 sec	40 sec
Instruct Pix2Pix	11 sec	N/A

D Comparative experiment to other state-of-the-art (SoTA) editing methods

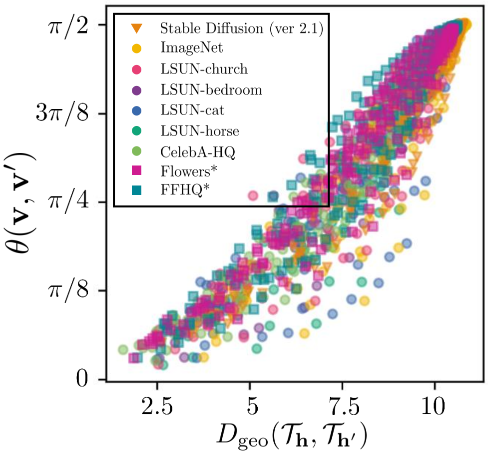
We conduct qualitative comparisons with text-guided image editing methods. Our SoTA baseline methods include: (i) SDEdit [Meng et al., 2021], (ii) Pix2Pix-zero [Parmar et al., 2023], (iii) PnP [Tumanyan et al., 2022], and (iv) Instruct Pix2Pix [Brooks et al., 2023]. All comparisons were performed using the official code. Please refer to Figure A3 for the qualitative results.

We also compare the time complexity of each method. For a fair comparison, we only identify the first singular vector $\mathbf{v}_1$, i.e., $n = 1$, and set the number of DDIM steps to 50. All experiments were conducted on an Nvidia RTX 3090. The runtime for each method is summarized in Table A1.

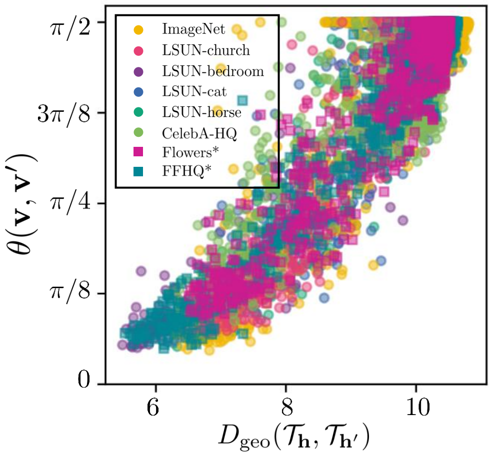
The computation cost of our method remains comparable to other approaches, although the Jacobian approximation takes around 2.5 seconds for $n = 1$. This is because we only need to identify the latent basis vector once at a specific timestep. Furthermore, our approach does not require additional preprocessing steps like generating 100 prompts with GPT and obtaining embedding vectors (as in Pix2Pix-zero), or storing feature vectors, queries, and key values (as in PnP). Our method also does not require finetuning (as in Instruct Pix2Pix). This leads to a significantly reduced total editing process time in comparison to other methods.

E More Discussions

Why do we measure the geodesic distance of the tangent spaces instead of the latent subspaces? The geodesic distance on the Grassmannian manifold between



(a) Parallel transport across timesteps



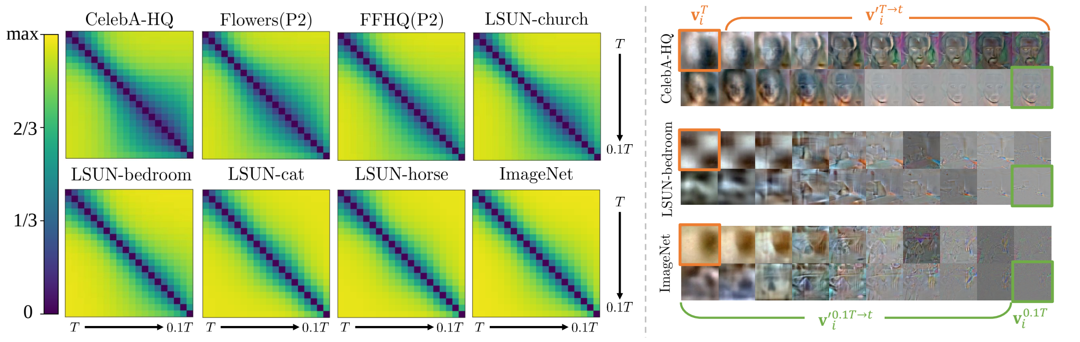
(b) Parallel transport across samples

Figure A1: **Parallel transport between similar tangent spaces creates similar latent directions.** The horizontal axis represents the geodesic distance between tangent spaces from (a) different timesteps (b) different samples at $t \in \{T, 0.9T, \cdots, 0.1T\}$. The vertical axis represents the angle between the original latent direction and transported latent direction. Different colors represent various datasets. A positive relationship is observed between tangent space distance and the distortion induced by parallel transport.

two subspaces is defined as the l_2-norm of principal angles. To define angles between different vector spaces, an inner product needs to be defined. In our work, we define the inner product in $\mathcal{T}_\mathbf{x}$ using the pullback metric. The issue is that the pullback metric is locally defined for each latent subspace $\mathcal{T}_\mathbf{x}$ (Eq. (3.1)). Therefore, measuring angles between distant latent subspaces becomes challenging. On the other hand, $\mathcal{H}$ follows the assumption of the Euclidean metric. Consequently, even for distant tangent spaces, angles can be easily computed using the dot product. In this regard, we measure the similarity between latent subspaces by exploiting the geodesic distance of their corresponding tangent spaces. Furthermore, when compared to $\mathcal{X}$, $\mathcal{H}$ offers the advantage of being a semantic space, making it more suitable for measuring semantic similarity.

Similar tangent space implies similar latent subspace In Figure A1, we calculated the geodesic distance of tangent spaces obtained at different timesteps (or different samples at same the timestep) and the angle between the original latent direction and parallel transported direction between them. It is evident that as the geodesic distance decreases, the amount of distortion during parallel transport also reduces.

Notice that the similarity between tangent spaces implies consistency of latent basis across timesteps. In Figure A2 (b), we parallel transport the latent vector $\mathbf{v}_i$ to various tangent spaces and visualize the outcomes. As expected, when the tangent spaces are



(a) Distance matrix of tangent space

(b) Parallel transport of latent direction across timesteps

Figure A2: **More examples of tangent spaces across diffusion timesteps.** (a) Distance matrix visualization of tangent space measured by geodesic metric across various timesteps. (b) Visualization of the result from parallel transport across timesteps. $\mathbf{v}'^{t_a \to t_b}_i$ denotes the latent vector transported from t_a to t_b. Transported vector significantly deviates from the original vector, as the tangent space grows further apart according to the distance matrix. For visualization purposes, $\mathbf{v}_i$ is min-max normalized.

similar, the transported vector $\mathbf{v}'_i$ retains the original signal. On the other hand, as we move to more distant timesteps, where the tangent space is farther apart, $\mathbf{v}'_i$ deviates from the original signal.

F Algorithms

In this section, for reproducibility purposes, we provide the code for two important algorithms. The code is implemented using PyTorch [Paszke et al., 2017].

Jacobian subspace iteration The diffusion model has dimensions that are too large in both $\mathcal{X}$ and $\mathcal{H}$, making the computation of the Jacobian infeasible. To overcome this challenge, we attempt the Jacobian subspace iteration algorithm to approximate the singular value of the Jacobian, as proposed in [Haas et al., 2023]. For details, please refer to Haas et al. [2023].

```
import torch # >= ver 2.0

def local_encoder_pullback(
        x, t, get_h, n=50, chunk_size=25, min_iter=10, max_iter=100,
    convergence_threshold=1e-4,
    ):
    '''
    Args
        - x : tensor ; latent variable
```

```
        - t : tensor ; diffusion timestep
        - get_h : function ; return h given x, t
        - n ; low-rank approximation dimension
        - chunk_size ; To avoid OOM error
        - min_iter (max_iter) ; minimum (maximum) number of iteration
        - convergence_threshold ; to check convergence of power-method
    '''
    # set number of chunk to avoid OOM
    num_chunk = n // chunk_size + 1

    # get dimensions of x space and h space
    h_shape = get_h(x, t).shape
    c_i, w_i, h_i = x.size(1), x.size(2), x.size(3)
    c_o, w_o, h_o = h_shape[1], h_shape[2], h_shape[3]

    # power-method
    a = torch.tensor(0., device=x.device, dtype=x.dtype)
    vT = torch.randn(c_i*w_i*h_i, n, device=x.device)
    vT, _ = torch.linalg.qr(vT)
    v = vT.T
    v = v.view(-1, c_i, w_i, h_i)

    for i in range(max_iter):
        v = v.to(device=x.device, dtype=x.dtype)
        v_prev = v.detach().cpu().clone()

        time_s = time.time()
        u = []
        v_buffer = list(v.chunk(num_chunk))
        for vi in v_buffer:
            g = lambda a : get_h(x + a*vi, t=t)
            ui = torch.func.jacfwd(
                g, argnums=0, has_aux=False, randomness='error'
            )(a)
            u.append(ui.detach().cpu().clone())
        u = torch.cat(u, dim=0)
        u = u.to(x.device, x.dtype)

        g = lambda x : torch.einsum(
```

```
                'b c w h, i c w h -> b', u, get_h(x, t=t)
            )
            v_ = torch.autograd.functional.jacobian(g, x)
            v_ = v_.view(-1, c_i*w_i*h_i)

            _, s, v = torch.linalg.svd(v_, full_matrices=False)
            v = v.view(-1, c_i, w_i, h_i)
            u = u.view(-1, c_o, w_o, h_o)

            convergence = torch.dist(v_prev, v.detach().cpu()).item()
            if torch.allclose(v_prev, v.detach().cpu(), atol=
    convergence_threshold) and (i > min_iter):
                break

        # reshape as a x space, h space vector
        u, s, vT = u.reshape(-1, c_o*w_o*h_o).T.detach(), s.sqrt().detach(), v.
    reshape(-1, c_i*w_i*h_i).detach()
        return u, s, vT
```

Listing 1: **Jacobian subspace iteration**

Geodesic metric For a detailed discussion on the geodesic metric, please refer to Choi et al. [2021a] for more information.

```
import torch

def geodesic_metric(U1, U2):
    _, S, _ = torch.linalg.svd(U1.T @ U2)
    th = torch.acos(S)
    return th.norm()
```

Listing 2: **Geodesic metric**

G Additional results

G.1 Latent basis

Unconditional latent basis We provide more examples of image editing using the latent basis. Figure A1, A2, A3, A4 and A5 show that every latent basis produces different results and editing at timestep T yields coarse changes while $0.6T$ leads to fine changes. Stable Diffusion shows a similar trend; $0.7T$ yields coarse changes while $0.6T$ leads to fine changes. The results of T in Stable Diffusion will be covered in the § G.3. Please zoom in for the best view.

Figure A1: **More examples of image editing using the latent basis in FFHQ.** The editing result using ten $\mathbf{v}_i$' in FFHQ. Each column represents edits made at different diffusion timesteps ($0.6T$, $0.8T$, and $1T$). Editing at timestep $1T$ yields coarse changes. On the other hand, editing at timestep $0.6T$ leads to fine changes. Please zoom in for the best view.

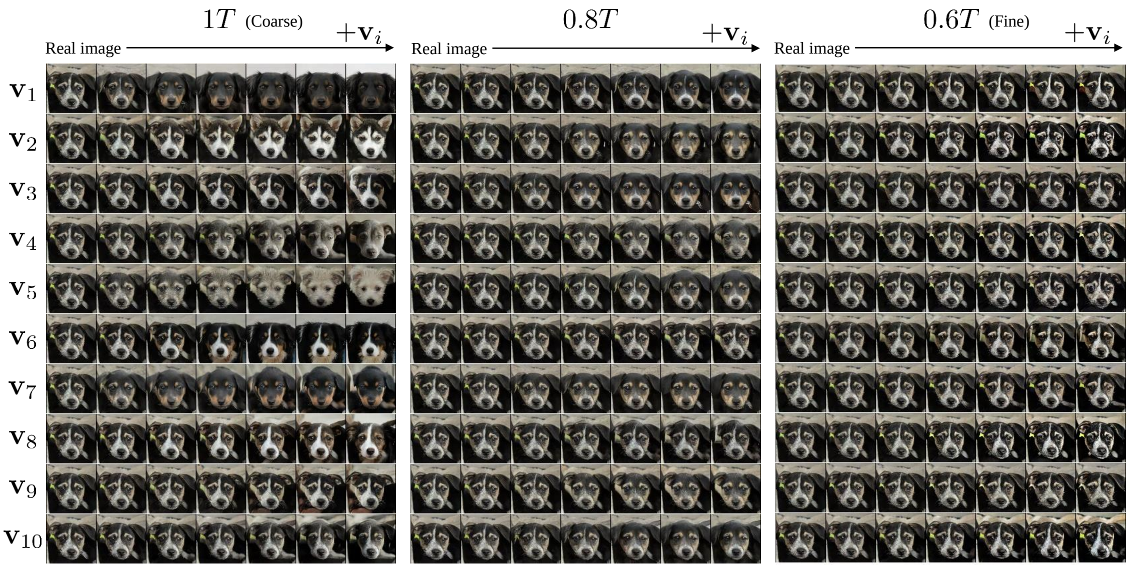


Figure A2: **More examples of image editing using the latent basis in AFHQ.** The editing result using ten $\mathbf{v}_i$' in AFHQ. Each column represents edits made at different diffusion timesteps ($0.6T$, $0.8T$, and $1T$). Editing at timestep $1T$ yields coarse changes. On the other hand, editing at timestep $0.6T$ leads to fine changes. Please zoom in for the best view.

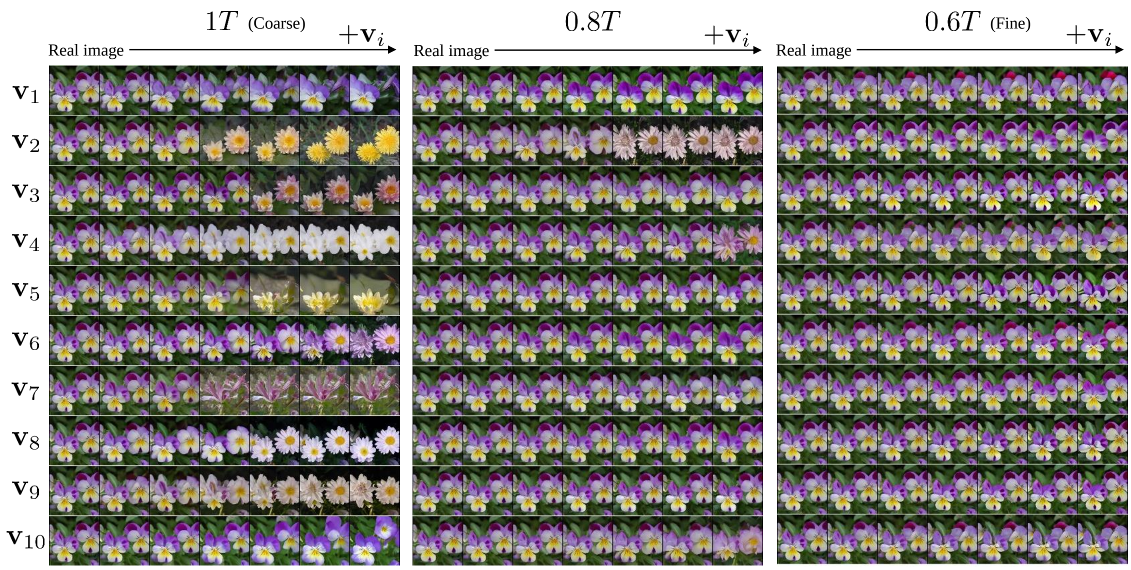


Figure A3: **More examples of image editing using the latent basis in Flowers.** The editing result using ten $\mathbf{v}_i$' in Flowers. Each column represents edits made at different diffusion timesteps ($0.6T$, $0.8T$, and $1T$). Editing at timestep $1T$ yields coarse changes. On the other hand, editing at timestep $0.6T$ leads to fine changes. Please zoom in for the best view.

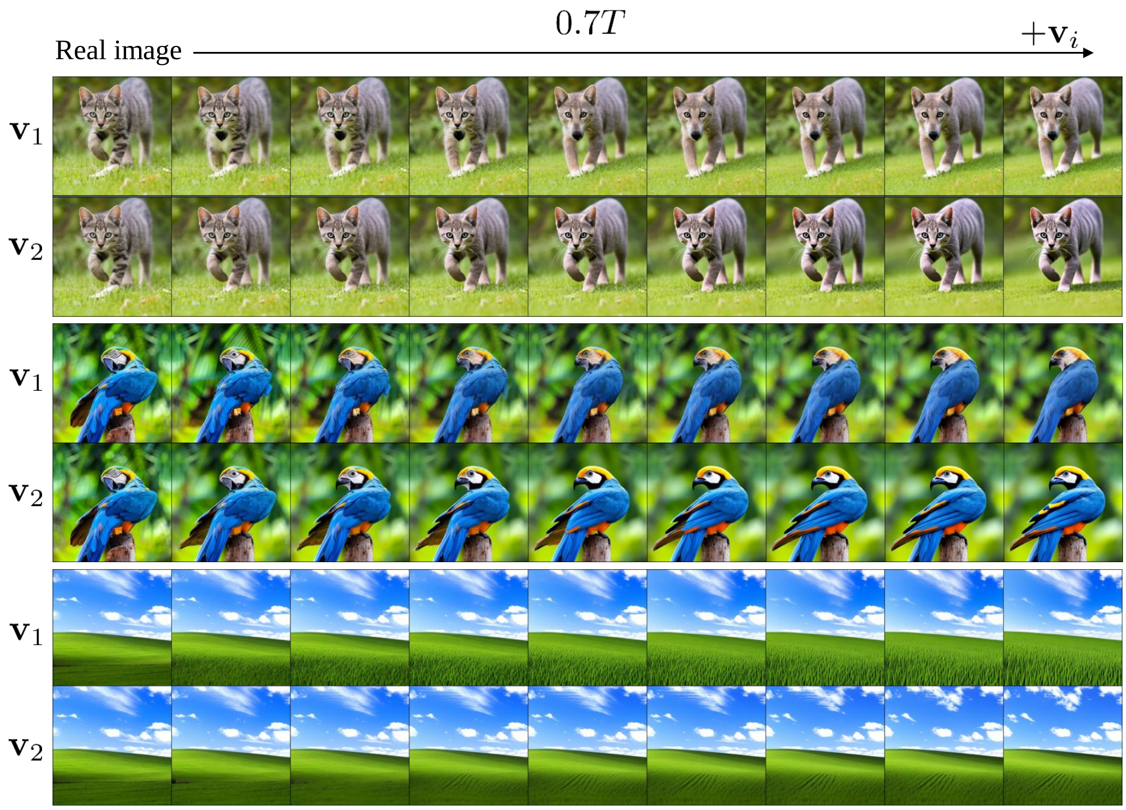


Figure A4: **More examples of image editing using the latent basis with Stable Diffusion.** The editing result using $\mathbf{v}_i$' in Stable diffusion at $0.7T$.

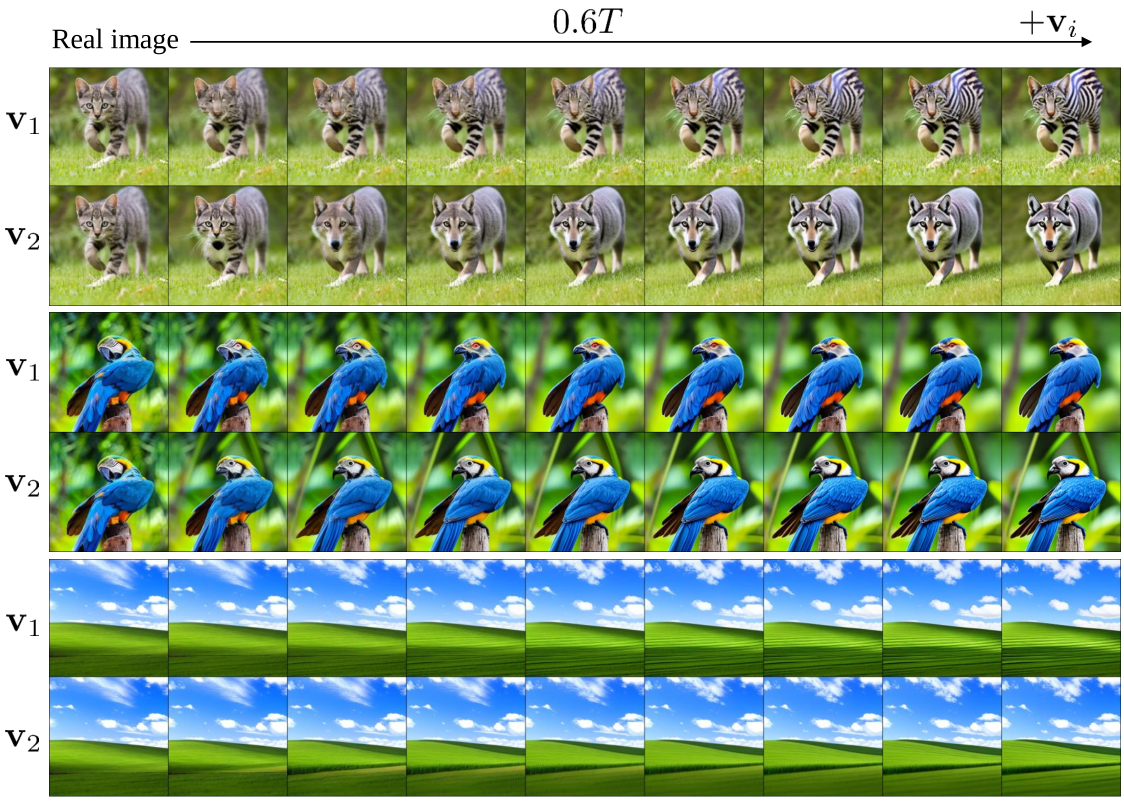


Figure A5: **More examples of image editing using the latent basis with Stable Diffusion.** The editing result using $\mathbf{v}_i$' in Stable diffusion at $0.6T$.

latent basis with given prompt As shown in Figure A6, when we condition a specific prompt, such as "Zebra" or "Chimpanzee", the entire latent basis corresponds to the prompt-related attributes. Notably, Changes to "zebra", which are clear, all show similar results, but "chimpanzee" show different results. Nevertheless, it is clear that they are all related to "chimpanzee".

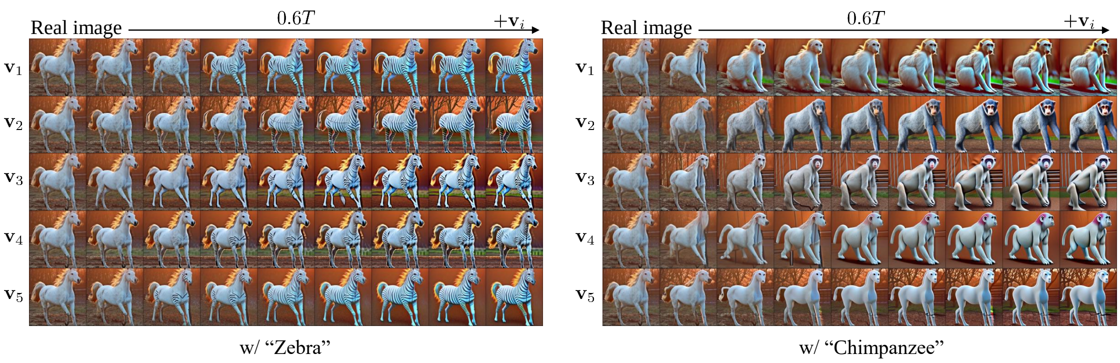


Figure A6: **More examples of image editing using top-5 latent basis vectors when given the prompt.** Notably, Changes to "zebras", which are clear, all show similar results, but "chimpanzees" show different results. Nevertheless, it is clear that it is all related to "chimpanzees".

G.2 Image editing using latent basis vectors discovered with various prompts

We provide additional examples of image editing using latent basis vectors discovered with various prompts. Figure A7, A8 show image editing with various pictures and various prompts. For brevity, we denote the prompt "a cat dressed as a witch wearing a wizard hat in a haunted house" by "[· · · cat · ·]" in Figure A7.

Figure A7: **More examples of image editing using latent basis vectors discovered with various prompts.**

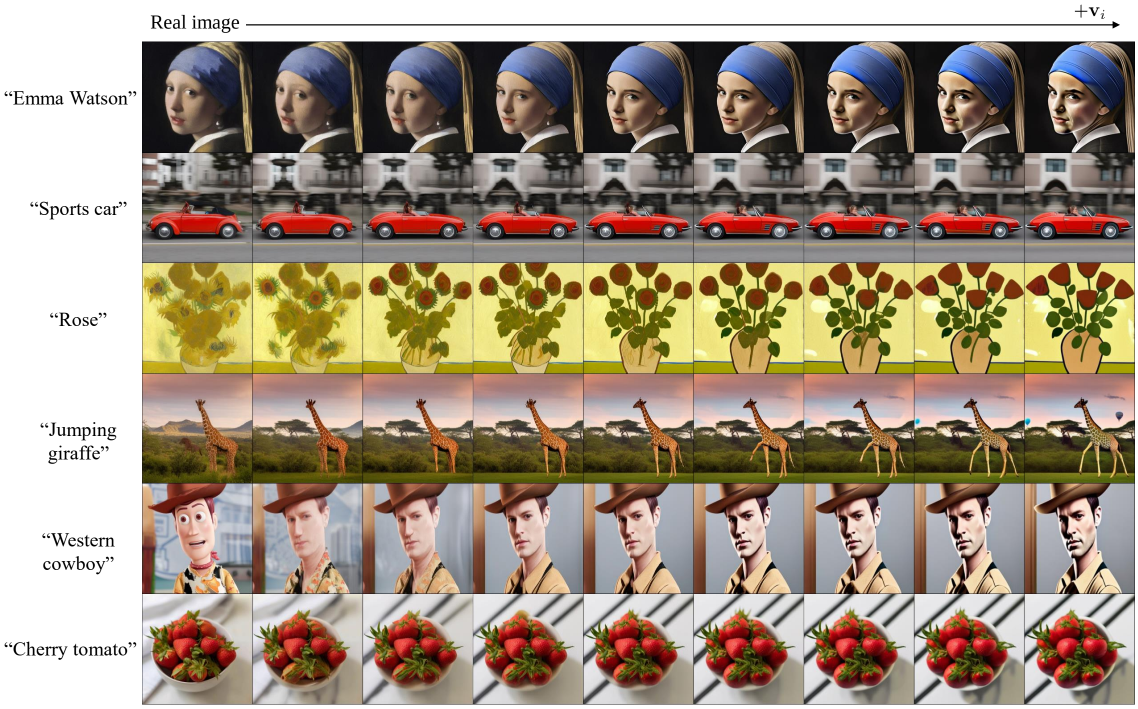


Figure A8: **More examples of image editing using latent basis vectors discovered with various prompts.**

G.3 More discussion on the editing capability of the latent basis discovered with text conditions

In this subsection, we provide a discussion based on the failure cases of our approach. Figure A9 shows the results of latent basis found at $t = T$ with Stable diffusion. Unlike unconditional models, the directions found at $t = T$ exhibit rapid and drastic unexpected changes. However, landscape photos, which do not contain a main object, exhibited desired editing effects at any timesteps. Moreover, in the case of landscapes, it is conjectured that the latent basis plays a significant role in representing patterns and textures. Analyzing the landscape images generated by Stable diffusion would be an interesting topic.

Figure A10 presents examples of failure cases in our image editing using latent basis vectors discovered with various prompts. (a) When using pose or action as a prompt, there are instances where the identity is not preserved. (b) When the shape of the target subject differs significantly from the source image, the results are often unsatisfactory. (c) There are cases where the preservation of the background is not achieved. (d) It is challenging to make significant changes to the entire image.

Regarding the reasons for these failure cases, we emphasize two factors. First, we manipulate in the $\mathcal{X}$. The result in Figure A10 (a) implies that $\mathcal{X}$ is not a space where disentanglement for identity is achieved effectively. On the other hand, in $\mathcal{H}$, there are results indicating successful preservation of identity [Kwon et al., 2022, Haas et al., 2023]. Investigating the disentanglement capability of $\mathcal{X}$ and any other distinguishing features it may have compared to $\mathcal{H}$ would be an interesting future research topic.

Secondly, we perform manipulation by adding and subtracting the "signal" that the model pays attention to in $\mathbf{x}_t$. Here, The signal is captured from the current input $\mathbf{x}_t$, which limits the deviation from the original form. Therefore, when there is a substantial difference in shape, such as transforming a giraffe into a tiger, the results may not be satisfactory. (Figure A10 (b)) When we utilize text conditions, the latent basis aligns with the text information. This leads to not capturing background information, resulting in changes in the background when manipulated. It is also an interesting research topic to capture signals related to the background. (Figure A10 (c)) Since the model focuses on finer features as t approaches 0, if broad changes are desired, manipulation should be performed at $t = T$. However, manipulation at $t = T$ is unstable. Deep analysis of $\mathbf{x}_t$ at $t = T$ in Stable diffusion is also an intriguing research topic. (Figure A10 (d))

Despite these limitations, we have successfully achieved direct manipulation in the latent space $\mathbf{x}_t$ at a single timestep, which, to our knowledge, is the first of its kind. Through this, we provide insights into the model and contribute to the understanding of the latent space, hopefully benefiting the diffusion community.

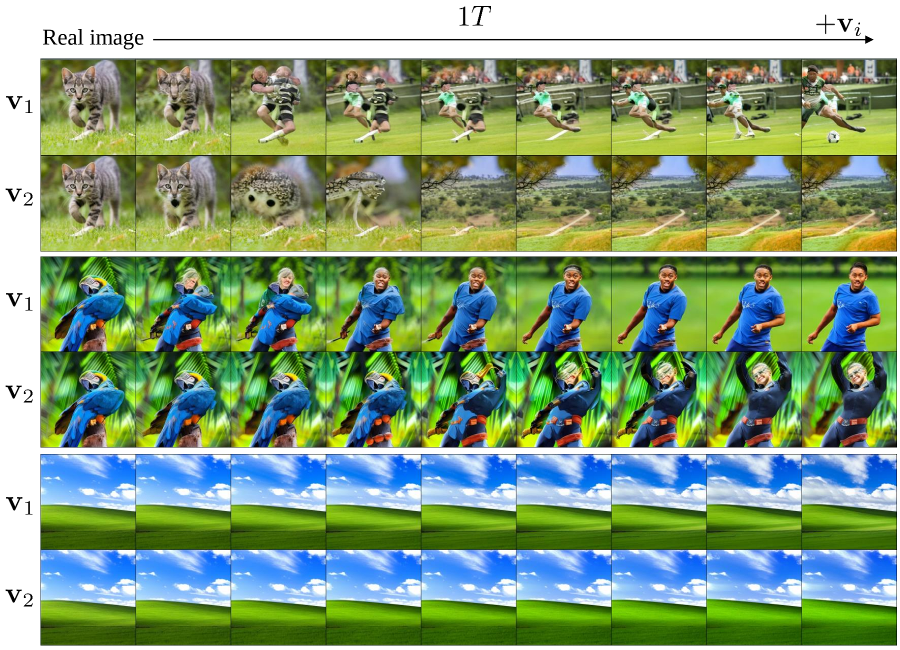


Figure A9: **Failure cases of image editing using the latent basis at $1T$.** The editing result using $\mathbf{v}_i$' in Stable diffusion at $1T$.

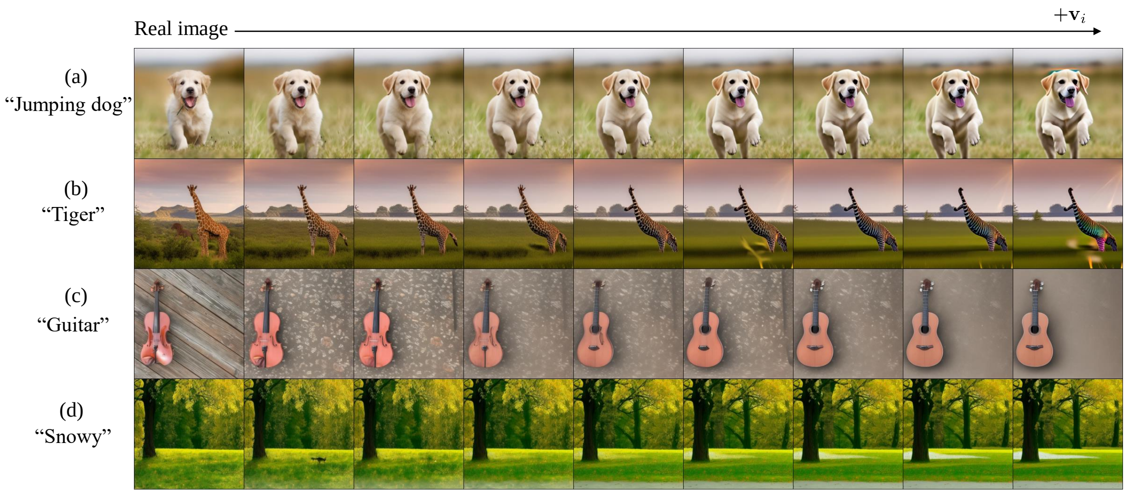


Figure A10: **Failure cases of using prompts.** (a) When using pose or action as a prompt, there are instances where the identity is not preserved. (b) When the shape of the target subject differs significantly from the source image, the results are often unsatisfactory. (c) There are cases where the preservation of the background is not achieved. (d) It is challenging to make significant changes to the entire image.

Bibliography

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

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

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

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

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.

Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022.

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

Hyungjin Chung, Byeongsu Sim, Dohoon Ryu, and Jong Chul Ye. Improving diffusion models for inverse problems using manifold constraints. *arXiv preprint arXiv:2206.00941*, 2022.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. *arXiv preprint arXiv:2211.12572*, 2022.

Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. *arXiv preprint arXiv:2302.03027*, 2023.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. *arXiv preprint arXiv:2211.09794*, 2022.

Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. *arXiv preprint arXiv:2210.10960*, 2022.

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. *arXiv preprint arXiv:2301.13826*, 2023.

Chaofan Ma, Yuhuan Yang, Chen Ju, Fei Zhang, Jinxiang Liu, Yu Wang, Ya Zhang, and Yanfeng Wang. Diffusionseg: Adapting diffusion towards unsupervised object discovery. *arXiv preprint arXiv:2303.09813*, 2023.

Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. *arXiv preprint arXiv:2303.04803*, 2023.

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.

Vikash Sehwag, Caner Hazirbas, Albert Gordo, Firat Ozgenel, and Cristian Canton. Generating high fidelity data from low-density regions using diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11492–11501, 2022.

Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218, 2022.

Xihui Liu, Dong Huk Park, Samaneh Azadi, Gong Zhang, Arman Chopikyan, Yuxiao Hu, Humphrey Shi, Anna Rohrbach, and Trevor Darrell. More control for free! image synthesis with semantic diffusion guidance. *arXiv preprint arXiv:2112.05744*, 2021.

Aditya Ramesh, Youngduck Choi, and Yann LeCun. A spectral regularizer for unsupervised disentanglement. *arXiv preprint arXiv:1812.01161*, 2018.

Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021.

Rameen Abdal, Peihao Zhu, Niloy J Mitra, and Peter Wonka. Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. *ACM Transactions on Graphics (ToG)*, 40(3):1–21, 2021.

Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable gan controls. *Advances in Neural Information Processing Systems*, 33:9841–9850, 2020.

Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1532–1540, 2021.

Oğuz Kaan Yüksel, Enis Simsar, Ezgi Gülperi Er, and Pinar Yanardag. Latentclr: A contrastive learning approach for unsupervised discovery of interpretable directions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14263–14272, 2021.

Xingang Pan, Ayush Tewari, Thomas Leimkühler, Lingjie Liu, Abhimitra Meka, and Christian Theobalt. Drag your gan: Interactive point-based manipulation on the generative image manifold. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023.

Jiapeng Zhu, Ruili Feng, Yujun Shen, Deli Zhao, Zheng-Jun Zha, Jingren Zhou, and Qifeng Chen. Low-rank subspaces in gans. *Advances in Neural Information Processing Systems*, 34:16648–16658, 2021.

Jaewoong Choi, Junho Lee, Changyeon Yoon, Jung Ho Park, Geonho Hwang, and Myungjoo Kang. Do not escape from the manifold: Discovering the local coordinates on the latent space of gans. *arXiv preprint arXiv:2106.06959*, 2021a.

Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*, 2021b.

Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

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

Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. *arXiv preprint arXiv:2112.03126*, 2021.

Georgios Arvanitidis, Lars Kai Hansen, and Søren Hauberg. Latent space oddity: on the curvature of deep generative models. *arXiv preprint arXiv:1710.11379*, 2017.

Hang Shao, Abhishek Kumar, and P Thomas Fletcher. The riemannian geometry of deep generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 315–323, 2018.

Nutan Chen, Alexej Klushyn, Richard Kurle, Xueyan Jiang, Justin Bayer, and Patrick Smagt. Metrics for deep generative models. In *International Conference on Artificial Intelligence and Statistics*, pages 1540–1550. PMLR, 2018.

Georgios Arvanitidis, Søren Hauberg, and Bernhard Schölkopf. Geometrically enriched latent spaces. *arXiv preprint arXiv:2008.00565*, 2020.

Yonghyeon Lee and Frank C Park. On explicit curvature regularization in deep generative models. 2023.

Yonghyeon Lee, Seungyeon Kim, Jinwon Choi, and Frank Park. A statistical manifold framework for point cloud data. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 12378–12402. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/lee22d.html.

LEE Yonghyeon, Sangwoong Yoon, MinJun Son, and Frank C Park. Regularized autoencoders for isometric representation learning. In *International Conference on Learning Representations*, 2021.

Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

René Haas, Inbar Huberman-Spiegelglas, Rotem Mulayoff, and Tomer Michaeli. Discovering interpretable directions in the semantic latent space of diffusion models. *arXiv preprint arXiv:2303.11073*, 2023.

Jooyoung Choi, Jungbeom Lee, Chaehun Shin, Sungwon Kim, Hyunwoo Kim, and Sungroh Yoon. Perception prioritized training of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11472–11481, 2022.

Giannis Daras and Alexandros G Dimakis. Multiresolution textual inversion. *arXiv preprint arXiv:2211.17115*, 2022.

Ke Ye and Lek-Heng Lim. Schubert varieties and distances between subspaces of different dimensions. *SIAM Journal on Matrix Analysis and Applications*, 37(3):1176–1197, 2016.

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 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

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

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8188–8197, 2020.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

감사의 글

본 논문이 완성되기까지 도움을 주신 많은 분들께 진심 어린 감사의 마음을 전합니다.

먼저, 연구 전 과정에 걸쳐 아낌없는 지도와 격려를 해주신 조정효 교수님께 깊은 감사를 드립니다. 교수님의 따뜻한 격려와 연구에 집중할 수 있는 환경을 조성해 주신 깊은 배려, 그리고 통찰력 있는 조언들이 이 논문의 완성을 가능케 했습니다. 부족한 저를 항상 이해해 주시고 인내심을 가지고 지도해 주신 점, 진심으로 감사드리며 죄송한 마음도 함께 전합니다.

공동 1저자로 연구에 함께해 주신 권민기님, 그리고 공저자이신 최재웅 박사님과 어영정 교수님께도 특별한 감사의 마음을 전합니다. 특히 외부인임에도 불구하고 귀중한 시간을 내어 열정적으로 지도해 주신 어영정 교수님의 헌신에 깊은 감사를 드립니다. 공저자 분들의 전문성과 협력이 이 연구의 질을 한층 높여주었습니다.

마지막으로, 항상 곁에서 변함없이 응원해 준 가족과 애인, 그리고 연구실 동료분들에게 진심 어린 고마움을 전합니다. 여러분의 끊임없는 지지와 이해, 그리고 때로는 힘들 때 건네준 따뜻한 위로의 말 한마디가 저에게 큰 힘이 되었고, 이 긴 여정을 완주할 수 있게 해주었습니다.

이 논문이 완성되기까지 도움을 주신 모든 분들께 다시 한 번 감사드립니다.

감사합니다.

2024년 8월

박용현 올림