SDDM: Score-Decomposed Diffusion Models on Manifolds for Unpaired Image-to-Image Translation

Shikun Sun [1 †] **Longhui Wei** [2] **Junliang Xing** [1] **Jia Jia** [1 3] **Qi Tian** [2]

Abstract

Recent score-based diffusion models (SBDMs) show promising results in unpaired image-to-image translation (I2I). However, existing methods, either energy-based or statistically-based, provide no explicit form of the interfered intermediate generative distributions. This work presents a new score-decomposed diffusion model (SDDM) on manifolds to explicitly optimize the tangled distributions during image generation. SDDM derives manifolds to make the distributions of adjacent time steps separable and decompose the score function or energy guidance into an image "denoising" part and a content "refinement" part. To refine the image in the same noise level, we equalize the refinement parts of the score function and energy guidance, which permits multi-objective optimization on the manifold. We also leverage the block adaptive instance normalization module to construct manifolds with lower dimensions but still concentrated with the perturbed reference image. SDDM outperforms existing SBDM-based methods with much fewer diffusion steps on several I2I benchmarks.

1. Introduction

Score-based diffusion models (Song & Ermon, 2019; Song et al., 2021; Ho et al., 2020; Nichol & Dhariwal, 2021; Bao et al., 2022a; Lu et al., 2022) (SBDMs) have recently made significant progress in a series of conditional image generation tasks. In particular, in the unpaired image-to-image translation (I2I) task (Pang et al., 2021), recent studies have shown that a pre-trained SBDM on the target image domain with energy guidance (Zhao et al., 2022) or statistical (Choi et al., 2021) guidance outperforms generative adversarial network (Goodfellow et al., 2014)(GAN)-based methods (Fu et al., 2019; Zhu et al., 2017; Yi et al., 2017; Park et al., 2020; Benaim & Wolf, 2017; Zheng et al., 2021; Shen et al., 2019; Huang et al., 2018; Jiang et al., 2020; Lee et al., 2018) and achieves the state-of-the-art performance.

SBDMs provide a diffusion model to guide how image-shaped data from a Gauss distribution is iterated step by step into an image of the target domain. In each step, SBDM gives score guidance which, from an engineering perspective, can be mixed with energy and statistical guidance to control the generation process. However, firstly due to the stochastic differential equations of the inverse diffusion process, the coefficient of the score guidance is not changeable. Secondly how energy guidances affect the intermediate distributions is still not clear. As a result, the I2I result is often unsatisfactory, especially when iterations are inadequate. Moreover, there has yet to be a method to ensure that the intermediate distributions are not negatively interfered with during the above guidance process.

To overcome these limitations, we propose to decompose the score function from a new manifold optimization perspective, thus better exerting the energy and statistical guidance. To this end, we present SDDM, a new score-decomposed diffusion model on manifolds to explicitly optimize the tangled distributions during the conditional image generation process. When generating an image from score guidance, an SBDM actually performs two distinct tasks, one is image "denoising", and the other is content "refinement" to bring the image-shaped data closer to the target domain distribution with the same noise level. Based on this new perspective, SDDM decomposes the score function into two different parts, one for image denoising and the other for content refinement. To realize this decomposition, we take statistical guidance as the manifold restriction to get an explicit division between the data distributions in neighboring time steps. We find that the tangent space of the manifold naturally separates the denoising part and the refinement part of the score function. In addition, the tangent space can also split out the denoising part of the energy guidance, thus achieving a more explanatory conditional generation.

[†]Work done when interning at Huawei Cloud. [1]Department of Computer Science and Technology, Tsinghua University, Beijing, China [2]Huawei Cloud. [3]Beijing National Research Center for Information Science and Technology. Correspondence to: Jia Jia <jiajia@tsinghua.edu.cn>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

Within the decomposed score functions, the content refinement part of the score function and energy functions are on an equal footing. Therefore we can treat the optimization on the manifold as a multi-objection optimization, thus avoiding the negative interference of other guidance on score guidance. To realize the score-decomposed diffusion model, we leverage the block adaptive instance normalization (BAdaIN) module to play the restriction function on the manifold, which is a stronger constraint than the widely used low-pass filter (Choi et al., 2021). With our carefully designed BAdaIN, the tangent space of the manifold provides a better division for the score and energy guidance. We also prove that our manifolds are equivalently concentrated with the perturbed reference image compared with those in (Choi et al., 2021).

To summarize, this work makes the following three main contributions:

- We present a new score-decomposed diffusion model on manifolds to explicitly optimize the tangled distributions during the conditional image generation process.
- We introduce a multi-objective optimization algorithm into the conditional generation of SBDMs, which permits not only many powerful gradient combination algorithms but also adjustment of the score factor.
- We design a BAdaIN module to construct a lower dimensional manifold compared with the low-pass filter and thus provide a concrete model implementation.

With the above contributions, we have obtained a high-performance conditional image generation model. Extensive experimental evaluations and analyses on two I2I benchmarks demonstrate the superior performance of the proposed model. Compared to other SBDM-based methods, SDDM generates better results with much fewer diffusion steps.

2. Background

2.1. Score-Based Diffusion Models (SBDMs)

SBDMs (Song et al., 2021; Ho et al., 2020; Dhariwal & Nichol, 2021; Zhao et al., 2022) first progressively perturb the training data via a forward diffusion process and then learn to reverse this process to form a generative model of the unknown data distribution. Denoting $q(\mathbf{x}_0)$ the training set with i.i.d. samples on $\mathbb{R}^d$ and $q(\mathbf{x}_t)$ the intermediate distribution at time t, the forward diffusion process $\{\mathbf{x}_t\}_{t\in[0,T]}$ follows the stochastic differential equation (SDE):

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}, \tag{1}$$

where $\mathbf{f}(\cdot, t) : \mathbb{R}^d \to \mathbb{R}^d$ is the drift coefficient, $\mathrm{d}t$ denotes an infinitesimal positive timestep, $g(t) \in \mathbb{R}$ is the diffusion coefficient, and $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, t\mathbf{I}_d)$ is a standard Wiener process. Denote $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$ the transition kernel from time 0 to t, which is decided by $\mathbf{f}(\mathbf{x}, t)$ and $g(t)$. In practice, $\mathbf{f}(\mathbf{x}, t)$ is usually an affine transformation *w.r.t.* $\mathbf{x}$ so that the $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$ is a linear Gaussian distribution and $\mathbf{x}_t$ can be sampled in one step (Zhao et al., 2022). In practice, the following VP-SDE is mostly used:

$$\mathrm{d}\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}\mathrm{d}t + \sqrt{\beta(t)}\mathrm{d}\mathbf{w}, \tag{2}$$

and DDPM (Ho et al., 2020; Dhariwal & Nichol, 2021) use the following discrete form of the above SDE:

$$\mathbf{x}_i = \sqrt{1-\beta_i}\mathbf{x}_{i-1} + \sqrt{\beta_i}\mathbf{z}_{i-1}, \quad i = 1, \cdots, N. \tag{3}$$

Normally an SDE is not time-reversible because the forward process loses information on the initial data distribution and converges to a terminal state distribution $q_T(\mathbf{x}_T)$. However, Song et al. (2021) find that the reverse process satisfies the following reverse-time SDE:

$$\mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - g(t)^2\nabla_{\mathbf{x}} \log q_t(\mathbf{x})\right] \mathrm{d}t + g(t)\mathrm{d}\overline{\mathbf{w}}, \tag{4}$$

where $\mathrm{d}t$ is an infinitesimal negative timestep and $\overline{\mathbf{w}}$ is a reverse-time standard Wiener process. Song et al. (2021) adopt a score-based model $\mathbf{s}(\mathbf{x}, t)$ to approximate $\nabla_{\mathbf{x}} \log q_t(\mathbf{x})$, *i.e.* $\mathbf{s}(\mathbf{x}, t) \doteq \nabla_{\mathbf{x}} \log q_t(\mathbf{x})$, obtaining the following reverse-time SDE:

$$\mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - g(t)^2\mathbf{s}(\mathbf{x}, t)\right] \mathrm{d}t + g(t)\mathrm{d}\overline{\mathbf{w}}. \tag{5}$$

In VP-SDE, $q_T(\mathbf{x}_T)$ is also a standard Gaussian distribution.

For a controllable generation, it is convenient to add some guidance function $\varepsilon(\mathbf{x}, t)$ to the score function and then get a new time-reverse SDE:

$$\mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - g(t)^2(\mathbf{s}(\mathbf{x}, t) + \nabla_{\mathbf{x}}\varepsilon(\mathbf{x}, t))\right]\mathrm{d}t + g(t)\mathrm{d}\overline{\mathbf{w}}. \tag{6}$$

2.2. SBDMs in Unpaired Image to Image Translation

Unpaired I2I aims to transfer an image from a source domain $\mathcal{Y} \subset \mathbb{R}^d$ to a different target domain $\mathcal{X} \subset \mathbb{R}^d$ as the training data. This translation process can be achieved by designing a distribution $p(\mathbf{x}_0|\mathbf{y}_0)$ on the target domain $\mathcal{X}$ conditioned on an image $\mathbf{y}_0 \in \mathcal{Y}$ to transfer.

In ILVR (Choi et al., 2021), given a reference image $\mathbf{y}_0$, they refine $\mathbf{x}_t$ after each denoising step with a low-pass filter $\mathbf{\Phi}$ for the faithfulness to the reference image:

$$\mathbf{x}'_t = \mathbf{x}_t - \mathbf{\Phi}(\mathbf{x}_t) + \mathbf{\Phi}(\mathbf{y}_t), \mathbf{y}_t \sim q_{t|0}(\mathbf{y}_t \mid \mathbf{y}_0). \tag{7}$$

In EGSDE (Zhao et al., 2022), they carefully designed two energy-based guidance functions and follow the conditional generation method in Song et al. (2021):

$$\mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - g(t)^2(\mathbf{s}(\mathbf{x}, t) - \nabla_{\mathbf{x}}\varepsilon(\mathbf{x}, \mathbf{y}_0, t))\right]\mathrm{d}t + g(t)\mathrm{d}\overline{\mathbf{w}}. \tag{8}$$

Notably, energy-based methods do not avoid the intermediate distribution being overly or negatively disturbed, and they both do not fully make use of the statistics of the reference image; thus the generation results may be suboptimal.

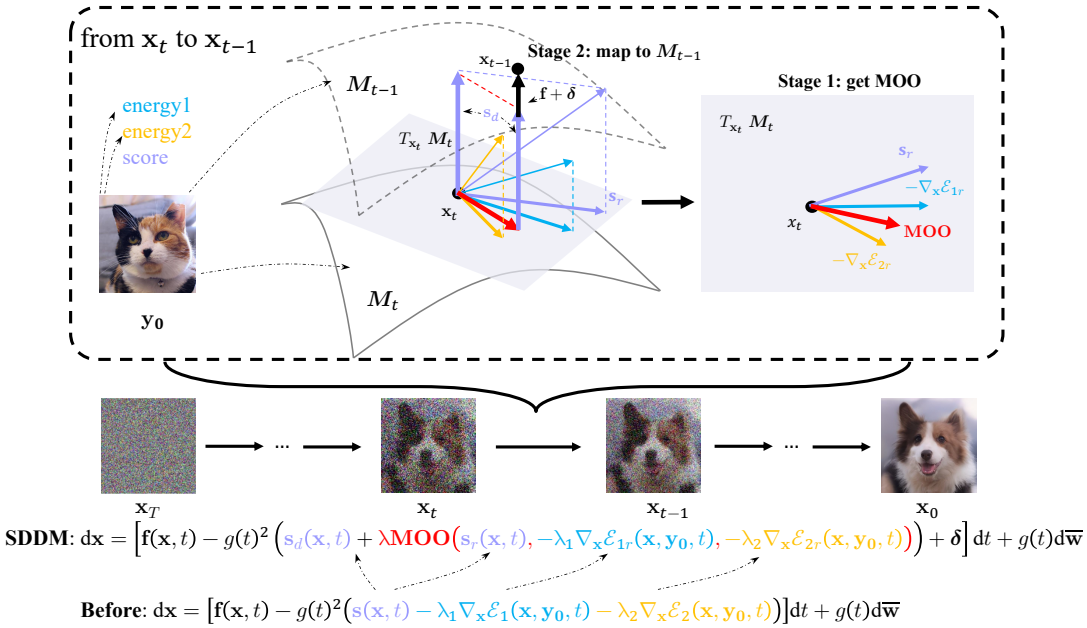

Figure 1. The overview of our SDDM. At each time step, compared with directly adding energy guidance to the score function, we firstly use the moments of the distribution $\mathbf{y}_t$ as constraints to get the manifolds $\mathcal{M}_t$ at each time step t. Then, we restrict potential energy of the score function $\mathbf{s}(\mathbf{x},t)$ and energy function ε_i on the manifold $\mathcal{M}_t$ at $\mathbf{x}_t$ to get the components of corresponding gradients $\mathbf{s}_r(\mathbf{x},t)$ and $\nabla_{\mathbf{x}_t}\varepsilon_{ir}(\mathbf{x}_t,\mathbf{y}_0,t)$ on the tangent space $T_{\mathbf{x}_t}\mathcal{M}_t$, and they are the "refinement" parts. Then we use multi-objective optimization viewpoint to get MOO, the optimal sum on the tangent space near $\mathbf{x}_t$, Finally, we restrict $\mathbf{f}(\mathbf{x},t), \mathbf{s}(\mathbf{x},t)$, and the noise on the $N_{\mathbf{x}_t}\mathcal{M}$ to get the components pointing to the next manifold $\mathcal{M}_{t-1}$. For clarity $g(t)\mathrm{d}\overline{\mathbf{w}}$ does not appear and the restriction on $\mathcal{M}_{t-1}$ is indicated as $\boldsymbol{\delta}$.

3. Score-Decomposed Diffusion Model

This section starts the elaboration of the proposed model from Eqn. (8). For the choice of the guidance function $\varepsilon(\mathbf{x},\mathbf{y}_0,t)$ in Eqn. (8), we set it to the following widely adopted form (Zhao et al., 2022; Bao et al., 2022b):

$$\varepsilon(\mathbf{x},\mathbf{y}_0,t) = \lambda_1\varepsilon_1(\mathbf{x},\mathbf{y}_0,t) + \lambda_2\varepsilon_2(\mathbf{x},\mathbf{y}_0,t), \quad (9)$$

where $\varepsilon_1(\cdot,\cdot,\cdot)$ and $\varepsilon_2(\cdot,\cdot,\cdot)$ denote two different energy guidance; λ_1 and λ_2 are two weighting coefficients.

3.1. Model Overview

Figure 1 overviews the main process of the proposed SDDM model. The second equation at the bottom is the equivalent SDE formulation from Eqns. (8) and (9). Starting with this equation, we have the first SDE in Figure 1 to indicate such a generation process. The illustration explains the two-stage optimization at time step t.

To explicitly optimize the tangled distributions during image generation, we use moments of the perturbed reference image $\mathbf{y}_0$ as constraints for constructing separable manifolds, thus disentangling the distributions of adjacent time steps. As shown in Figure 1, the manifolds of adjacent time steps t and $t-1$, $\mathcal{M}_t$ and $\mathcal{M}_{t-1}$ are separable, which indicates the conditional distributions of adjacent time steps $\mathbf{x}_t$ and $\mathbf{x}_{t-1}$ are also separable. Furthermore, at time step t, the manifold $\mathcal{M}_t$ decompose the score function $\mathbf{s}(\mathbf{x},t)$ into the content refinement part $\mathbf{s}_r(\mathbf{x},t)$ and the image denoising part $\mathbf{s}_d(\mathbf{x},t)$, and also separate out the content refinement parts $\nabla_{\mathbf{x}}\varepsilon_{1r}(\mathbf{x},\mathbf{y}_0,t), \nabla_{\mathbf{x}}\varepsilon_{2r}(\mathbf{x},\mathbf{y}_0,t)$ of $\nabla_{\mathbf{x}}\varepsilon_1(\mathbf{x},\mathbf{y}_0,t), \nabla_{\mathbf{x}}\varepsilon_2(\mathbf{x},\mathbf{y}_0,t)$ on the tangent space $T_{\mathbf{x}_t}\mathcal{M}_t$. Therefore, The entire optimization process at each time step is divided into two stages: one is to optimize on the manifold $\mathcal{M}_t$, and the other stage is to map to the next manifold $\mathcal{M}_{t-1}$ properly.

In the first stage, we optimize on the manifold $\mathcal{M}_t$. We apply a multi-objective optimization algorithm to get the red vector MOO, which is the optimal direction considering the score function and energy guidance on the tangent space $T_{\mathbf{x}_t}\mathcal{M}_t$. Then at the second stage, we use the rest of the first equation in Figure 1, which contains $\left[\mathbf{f}(\mathbf{x},t) - g(t)^2\mathbf{s}_d(\mathbf{x},t) + \boldsymbol{\delta}\right]\mathrm{d}t + g(t)\mathrm{d}\overline{\mathbf{w}}$ to map the $\mathbf{x}_t + \mathrm{MOO}$ to the next manifold $\mathcal{M}_{t-1}$ properly. Note that here we use $\boldsymbol{\delta}$ to indicate the restriction on $\mathcal{M}_{t-1}$ for the consistency of form.

3.2. Decomposition of the Score and Energy Guidance

Given a score function $\mathbf{s}(\mathbf{x}) = \nabla_{\mathbf{x}}\log p(\mathbf{x})$ on $\mathbb{R}^d$, suppose $\mathcal{M}$ is a smooth, compact submanifold of $\mathbb{R}^d$. We let $p_{\mathcal{M}}(\mathbf{x})$ is the corresponding probability distribution restricted on

$\mathcal{M}$. Then we have the following definitions:

Definition 1. The tangent score function $\mathbf{s}_r(\mathbf{x})$.

$$\mathbf{s}_r(\mathbf{x}) := \nabla_{\mathbf{x}} \log p_{\mathcal{M}}(\mathbf{x}),$$

which is the score function on the manifold. If there is a series of manifolds $\{\mathcal{M}_t\}$, and the original score function is denoted $\mathbf{s}(\mathbf{x}, t)$, we denote $\mathbf{s}_r(\mathbf{x}, t)$ the tangent score function on $\mathcal{M}_t$.

Definition 2. The normal score function $\mathbf{s}_d(\mathbf{x})$.

$$\mathbf{s}_d(\mathbf{x}) := \mathbf{s}(\mathbf{x})|_{N_{\mathbf{x}}\mathcal{M}},$$

which is the score function on the normal space of the manifold. We also denote $\mathbf{s}_d(\mathbf{x}, t)$ the normal score function on the manifold $\mathcal{M}_t$.

Then we have the following score function decomposition:

Lemma 1. $\mathbf{s}(\mathbf{x}) = \mathbf{s}_r(\mathbf{x}) + \mathbf{s}_d(\mathbf{x})$,

which can be derived when knowing $\mathbf{s}_r(\mathbf{x}) = \mathbf{s}(\mathbf{x})|_{T_{\mathbf{x}}\mathcal{M}}$.

Normally this division is meaningless because the manifolds of adjacent time steps are coupled with each other. Previous researchers usually treat the entire $\cup_t \mathbf{x}_t$ as an entire manifold (Liu et al., 2022), or use strong assumptions (Chung et al., 2022). However, in some conditional generation tasks, for example, the image-to-image transition task, a given reference image $\mathbf{y}_0$ can provide compact manifolds at different time steps, and manifolds of adjacent time steps can be well separated. In this situation, the tangent score function can be treated as a refinement part on the manifold. The normal score function is part of the mapping function between manifolds of adjacent time steps.

We have Proposition 1 to describe the manifolds.

Proposition 1. *At time step t, for any single reference image $\mathbf{y}_0$, the perturbed distribution $q_{t|0}(\mathbf{y}_t \mid \mathbf{y}_0)$ is concentrated on a compact manifold $\mathcal{M}_t \subset \mathbb{R}^d$ and the dimension of $\mathcal{M}_t \leq d - 2$ when d is large enough. Suppose the distributions of perturbed reference image $\mathbf{y_t} = \hat{\alpha}_t \mathbf{y}_0 + \hat{\beta}_t \mathbf{z}_t$, where $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The following statistical constraints define such (d-2)-dim $\mathcal{M}_t$.*

$$\begin{aligned} \mu[\mathbf{x}_t] &= \hat{\alpha}_t \mu[\mathbf{y}_0], \\ \mathrm{Var}[\mathbf{x}_t] &= \hat{\alpha}_t^2 \,\mathrm{Var}[\mathbf{y}_0] + \hat{\beta}_t^2. \end{aligned} \tag{10}$$

Proposition 1 shows that we can use statistical constraints to define concentrated manifolds with lower dimensions than $\mathbb{R}^d$. We can also use the chunking trick to lower the manifold dimensions, which will be introduced in Section 4. Therefore, we can use such manifolds to represent the maintenance of the statistics, which indicates that the tangent space $T_{\mathbf{x}_t}\mathcal{M}_t$ can separate the "refinement" part well.

We also have Lemma 2 to show that perturbed distributions of adjacent time steps, $\mathbf{y}_t$ and $\mathbf{y}_{t-1}$ can be well separated.

Lemma 2. *With the $\mathcal{M}_t$ defined in Proposition 1, assume $t \neq t'$, Then $\mathcal{M}_t$ and $\mathcal{M}_{t'}$ can be well separated. Rigorously, $\forall \varepsilon > 0, \exists \mathcal{M}_d$ divide the $\mathbb{R}^d$ into two disconnect spaces $\mathcal{A}, \mathcal{B}$, where $\mathcal{M}_t \in \mathcal{A}$,and $\mathcal{M}_{t'} \in \mathcal{B}$.*

Therefore, we can use $\mathcal{M}_t$ to decompose $\mathbf{s}$ into $\mathbf{s}_r$ and $\mathbf{s}_d$ approximately. More generally, we can decouple the optimization space with the tangent space $T_{\mathbf{x}_t}\mathcal{M}_t$. With $T_{\mathbf{x}_t}\mathcal{M}_t$, we can operate the score function of SBDM and energy more elaborately. We can also split the "refinement" part out, thus preventing the "denoising" part of the score function from being overly disturbed.

3.3. Stage 1: Optimization on Manifold

Firstly, we will give some main definitions about manifold optimization and muti-objective optimization in our task. We use restriction $\mathbf{R}_{\mathbf{x}_t}$ represent the function that maps the points on $T_{\mathbf{x}_t}\mathcal{M}_t$ near $\mathbf{x}_t$ to the manifold $\mathcal{M}_t$, which is normally an orthogonal projection onto $\mathcal{M}_t$.

Definition 3. Manifold optimization.

Manifold optimization (Hu et al., 2020) is a task to optimize a real-valued function $f(\mathbf{x})$ on a given Riemannian manifold $\mathcal{M}$. The optimized target is:

$$\min_{\mathbf{x} \in \mathcal{M}} f(\mathbf{x}). \tag{11}$$

Because that given t, the score function $\mathbf{s}(\mathbf{x}, t)$ is an estimation of $\nabla_{\mathbf{x}} \log q_t(\mathbf{x})$, and we can use $\log q_t(\mathbf{x})$ as the potential energy of $\mathbf{s}(\mathbf{x}, t)$, so are the guidance of energy funcitons. Then Stage 1 is a manifold optimization.

Definition 4. Pareto optimality on the manifold.

Consider $\mathbf{x}_t, \widehat{\mathbf{x}}_t \in \mathcal{M}_t$,

- $\mathbf{x}_t$ dominates $\widehat{\mathbf{x}}_t$ if $\mathbf{s}_r(\mathbf{x}_t, t) \geq \mathbf{s}_r(\widehat{\mathbf{x}}_t, t)$, $\varepsilon_{ir}(\mathbf{x}_t, \mathbf{y}_0, t) \leq \varepsilon_{ir}(\widehat{\mathbf{x}}_t, \mathbf{y}_0, t)$ for all i, and not all equal signs hold at the same time.
- A solution $\mathbf{x}_t$ is called Pareto optimal if there exists no solution $\widehat{\mathbf{x}}_t$ that dominates $\mathbf{x}_t$.

Then, the goal of multi-objective optimization is to find the Pareto optimal solution. The local Pareto optimality can also be reached via gradient descent like single-objective optimization. We just follow the multiple gradient descent algorithm (MGDA) (Désidéri, 2012). MGDA also leverages the Karush-Kuhn-Tucker (KKT) conditions for the multi-objective optimization, which in our task is that:

Theorem 1. ***K.K.T. conditions on a smooth manifold.***

At time step t on the tangent space $T_{\mathbf{x}_t}\mathcal{M}_t$, there $\exists \alpha, \beta^1, \beta^2, ..., \beta^m \geq 0$ such that $\alpha + \sum_{i=1}^m \beta^i = 1$ and $\alpha \mathbf{s}_r(\mathbf{x_t}, t) = \sum_{i=1}^m \beta^i \nabla_{\mathbf{x_t}} \varepsilon_{ir}(\mathbf{x}_t, \mathbf{y}_0, t)$, where $\mathbf{s}_r(\mathbf{x_t}, t)$ are the fractions of $\mathbf{s}(\mathbf{x_t}, t)$ on the tangent space and $\varepsilon_{ir}(\mathbf{x}_t, \mathbf{y}_0, t)$ are functions restricted on the manifold $\mathcal{M}_t$.

All points that satisfy the above conditions are called Pareto stationary points. Every Pareto optimal point is Pareto stationary point, while the reverse is not true. Désidéri (2012) showed that the optimization solution for the problem :

$$\min_{\substack{\alpha,\beta^1,\ldots,\beta^m\geq 0\\ \alpha+\beta^1+\ldots+\beta^m=1}} \left\{ \left\| \alpha \mathbf{s}_r(\mathbf{x}_t,t) - \sum_{i=1}^{m} \beta^i \nabla_{\mathbf{x_t}} \varepsilon_{ir}(\mathbf{x}_t,\mathbf{y}_0,t) \right\|_2^2 \right\} \tag{12}$$

gives a descent direction that improves all tasks or gives a Pareto stationary point. For a balanced result, we normalize all gradients first.

However, In our task, we can search Pareto stationary points in $\mathbf{B}_\epsilon(\mathbf{x}_t) \cap \mathcal{M}_t$ for a small ϵ because we have many time steps of different manifolds. $\mathbf{B}_\epsilon(\mathbf{x}_t)$ is an open ball with center $\mathbf{x}_t$, radius ϵ.

We have the following algorithm:

Algorithm 1 Multi-Objective Optimization on Manifold

Input: stepsize λ, current $\mathbf{x}_t$, refinement score s_r, energy funcitons ε_{ir} on $\mathcal{M}_t, i = 1,\ldots,m$, ϵ
Output: $\mathbf{x}_t^*$
Initialize $\mathbf{x}_t^* = \mathbf{x}_t$
repeat
  $\nabla_{\mathbf{x}_t^*}\varepsilon'_{ir} = \lambda_i \frac{\|\mathbf{s}_r(\mathbf{x}_t^*,t)\|}{\|\nabla_{\mathbf{x}_t^*}\varepsilon_{ir}(\mathbf{x}_t^*,\mathbf{y}_0,t)\|}\nabla_{\mathbf{x}_t^*}\varepsilon_{ir}(\mathbf{x}_t^*,\mathbf{y}_0,t)$
  Get the min value v of eq. (12) and corresponding $\alpha,\beta^1,\ldots,\beta^m$
  if $v == 0$ **then**
    return $\mathbf{x}_t^*$
  end if
  $\delta = \alpha\mathbf{s}_r(\mathbf{x}_t^*,t) - \sum_{i=1}^m \beta^i \nabla_{\mathbf{x}_t^*}\varepsilon'_{ir}$
  $\mathbf{x}'_t = \mathbf{x}_t^* + \lambda\delta$
  $\mathbf{x}_t^* = \mathbf{R}_{\mathbf{x}_t^*}(\mathbf{x}'_t)$
until $\mathbf{x}_t^* \notin \mathbf{B}_\epsilon(\mathbf{x}_t)$
return $\mathbf{x}_t^*$

Remark 1. We can use $\mathbf{f}(\mathbf{x}_t,t)$ and $\mathbf{s}_d(\mathbf{x}_t,t)$ to approximate $\mathbf{f}(\mathbf{x}_t^*,t)$ and $\mathbf{s}_d(\mathbf{x}_t^*,t)$ when ϵ is small.

Remark 2. Notably, EGSDE (Zhao et al., 2022) applies coefficients directly on the guidance vectors, and DVCE (Augustin et al., 2022) uses coefficients after the normalization on the guidance vectors. We can also provide coefficients λ_is for normalized energy vectors to change the impact of the vectors. A smaller norm means greater impact, as mentioned in (Désidéri, 2012).

3.4. Stage 2: Transformation between adjacent manifolds

After the optimization on the manifold $\mathcal{M}_t$, we get $\mathbf{x}_t^*$ that dominates $\mathbf{x}_t$, then we use $\mathbf{f}(\mathbf{x}_t^*,t)$, the "denoising" part score function $\mathbf{s}_d(\mathbf{x}^*,t)$, reverse-time noise and restriction function on $\mathcal{M}_{t-1}$ to map to the adjacent manifold $\mathcal{M}_{t-1}$.

Firstly, we have the following proposition to describe the properties of the adjacent map.

Proposition 2. *Suppose the $\mathbf{f}(\cdot,\cdot)$ is affine. Then the adjacent map has the following properties:*

- $\exists!\mathbf{v}_{\mathbf{x_t}} \in N_{\mathbf{x}_t}\mathcal{M}_t$ *that* $\mathbf{x}_t + \mathbf{v}_{\mathbf{x_t}} \in \mathcal{M}_{t-1}$.
- $N_{\mathbf{x}_t}\mathcal{M}_t = N_{\mathbf{x}_t+\mathbf{v}_{\mathbf{x_t}}}\mathcal{M}_{t-1}$.
- $\mathbf{v}_{\mathbf{x_t}}$ *is a transition map from* $T_{\mathbf{x}_t}\mathcal{M}_t$ *to* $T_{\mathbf{x}_t+\mathbf{v}}\mathcal{M}_{t-1}$.
- $\mathbf{v}_{\mathbf{x_t}}$ *is determined with* $\mathbf{f}(\cdot,\cdot),\mathbf{x_t},\mathbf{y}_0$ *and* $g(\cdot)$.

However, if we use $\mathbf{v}_{\mathbf{x_t}}$ as the adjacent map, we will lose the impact of $\mathbf{s}_d$ and the reverse-time noise. Therefore, we follow the reverse SDE, using the extra part of which on the normal space $N_{\mathbf{x}_t}\mathcal{M}_t$ and a restriction function on $\mathcal{M}_{t-1}$ as the adjacent map, we denote this part as $\mathbf{v}^*_{\mathbf{x_t}}$.

Finally, we have Algorithm 2 in the following to generate images with our proposed SDDM.

Algorithm 2 Generation with SDDM

Input: time steps T, milstone time step T_0, stepsize λ, score function $\mathbf{s}$, energy funcitons $\varepsilon_i, i = 1,\ldots,m$, small ϵ, reference image $\mathbf{y}_0$
Output: generated image $\mathbf{x}_0$
Initialize $\mathbf{x}_T \in \mathcal{M}_T$
for $t = T$ **to** T_0 **do**
  Divide the $\mathbb{R}^d$ into two orthogonal spaces $T_{\mathbf{x}_t}\mathcal{M}_t$ and $N_{\mathbf{x}_t}\mathcal{M}_t$.
  Calculate $\mathbf{s}_r(\cdot,\cdot)$ and $\varepsilon_{ir}(\cdot,\cdot,\cdot)$
  Optimize on manifold $\mathcal{M}_t$ with algorithm 1, and get the output $\mathbf{x}_t^*$
  Apply the time-reverse SDE on the $N_{\mathbf{x}_t^*}\mathcal{M}_t$ and then use the restriction function $\mathbf{R}$ on manifold $\mathcal{M}_{t-1}$ to map the $\mathbf{x}_t^*$ to $\mathbf{x}_{t-1} \in \mathcal{M}_{t-1}$
end for
for $t = T_0 - 1$ **to** 1 **do**
  Apply unconditional time-reverse SDE from $\mathbf{x_t}$ to $\mathbf{x}_{t-1}$
end for
return $\mathbf{x}_0$

Remark 3. If $\mathbf{f}(\mathbf{x}_t,t)$ is linear to $\mathbf{x}_t$, then $\mathbf{f}(\mathbf{x}_t,t) \in N_{\mathbf{x}_t}\mathcal{M}$.

Remark 4. When the ϵ is small, we can just use $N_{\mathbf{x}_t}\mathcal{M}_t$ to approximate $N_{\mathbf{x}_t^*}\mathcal{M}_t$.

Remark 5. Suppose $\|\mathbf{s}_r\|$ is $\mathbf{o}\left(\|\mathbf{v}^*_{\mathbf{x_t}}\|\right)$, and ϵ is $\mathbf{O}(\|\mathbf{s}_r\|)$. We can ignore the restriction step in algorithm 1.

Remark 6. At time step T_0, we can set larger ϵ for better results.

4. Implementations

Chunking Trick. The chunking trick is an easy but powerful trick to reduce the dimensions of the manifolds in

high-dimensional space problems, like the generation of images. We will divide the image shape $C \times H \times W$ into blocks $N \times N$, and the shape will be like $CN^2 \times \frac{H}{N} \times \frac{W}{N}$, and the manifold will be the direct product of CN^2 manifolds at each $\frac{H}{N} \times \frac{W}{N}$-sized block, we index them with (c, i, j). This trick has the following advantages:

- We can easily get the $T\mathcal{M}$ and the $N\mathcal{M}$, which are also the direct product of each block's $T\mathcal{M}$ and $N\mathcal{M}$.
- We can control the impact of the reference image on the generation process.
- We can optimize on block level and lower the impact of other distant blocks.

Manifold Details. For each chunked $\frac{H}{N} \times \frac{W}{N}$-sized block of the image, we use the first-order and second-order moments to restrict the statistics of the pixels of the block to get a $(\frac{H}{N} \times \frac{W}{N} - 2) - dim$ manifold. In particular, we denote the $\frac{H}{N} \times \frac{H}{N}$ as d. Suppose $\mathbf{y}_t = \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \sqrt{1-\bar{\alpha}_t}\mathbf{z_t}$, Then, $\mathbf{y}_t \sim \mathcal{N}(\sqrt{\bar{\alpha}_t}\mathbf{y}_0, (1-\bar{\alpha}_t)\mathbf{I})$, and the $\mathbf{y}_t^{c,i,j} \sim \mathcal{N}\left(\sqrt{\bar{\alpha}_t}\mathbf{y}_0^{c,i,j}, (1-\bar{\alpha}_t)\mathbf{I}\right)$. Then, the manifold $\mathcal{M}_t^{c,i,j}$ of block (c, i, j) is restricted with:

$$\begin{aligned} \mu[\mathbf{x}_t^{c,i,j}] &= \sqrt{\bar{\alpha}_t}\,\mu[\mathbf{y}_0^{c,i,j}], \\ \mathrm{Var}[\mathbf{x}_t^{c,i,j}] &= \bar{\alpha}_t\,\mathrm{Var}[\mathbf{y}_0^{c,i,j}] + (1-\bar{\alpha}_t). \end{aligned} \tag{13}$$

And the restrictions of Eqn. 13, $\mathcal{M}_t^{c,i,j}$ is a $(d-2)$ dimensional hypersphere. Then we can formulate $\mathcal{M}_t$ as:

$$\mathcal{M}_t = \otimes_{c,i,j}\mathcal{M}_t^{c,i,j}. \tag{14}$$

The $\otimes$ denotes the direct product. Huang & Belongie (2017) use the AdaIN module to transfer neural features as

$$\mathrm{AdaIN}(\mathbf{x}, \mathbf{y}) = \sigma(\mathbf{y})\left(\frac{\mathbf{x} - \mu(\mathbf{x})}{\sigma(\mathbf{x})}\right) + \mu(\mathbf{y}). \tag{15}$$

Based on that, we leverage a useful module of BAdaIN as the restriction function on any $T_{\mathbf{x}_t}\mathcal{M}_t$:

$$\begin{aligned} \mathrm{BAdaIN}(\mathbf{x}_t, \mathbf{y}_t) = \otimes_{c,i,j}\sigma(\mathbf{y}_t^{c,i,j})\left(\frac{\mathbf{x}_t^{c,i,j} - \mu(\mathbf{x}_t^{c,i,j})}{\sigma(\mathbf{x}_t^{c,i,j})}\right) \\ + \otimes_{c,i,j}\,\mu(\mathbf{y}_t^{c,i,j}) \end{aligned} \tag{16}$$

In practice, we use the distribution moments of the perturbed reference image to simplify the calculation and eliminate randomness after knowing the relationship between the perturbed and original reference images, as in Eqn. (10).

We have Lemma 3 to describe Proposition 1 in detail:

Lemma 3. *$\forall \epsilon, \xi > 0, \exists D > 0, \forall d > D$ we have:*

$$\mathcal{P}\left(d_2\left(\mathbf{y}_t^{c,i,j}, \mathcal{M}_t^{c,i,j}\right) < \epsilon\sqrt{d}\right) > 1 - \xi,$$

where d is the dimension of the Euclid space $\mathbf{y}_t^{c,i,j}$ in.

Remark 7. The ILVR (Choi et al., 2021) method employs the low-pass filter to transfer the reference image information in Eqn. 7, and the low-pass filter calculates the block means. We have the following relationship between our mean restriction and the low-pass filter:

$$\mathbb{E}[\mathbf{\Phi}(\mathbf{y}_t)] = \otimes_{c,i,j}\sqrt{\bar{\alpha}_t}\,\mu[\mathbf{y}_0^{c,i,j}] \tag{17}$$

Energy Function. We can also use the BAdaIN module for constructing weak energy functions. Firstly, we use the first several layers of VGG19 (Simonyan & Zisserman, 2014) net to extract neural features of $\mathbf{x}_t$ and $\mathbf{y}_t$ to get $\widehat{\mathbf{x}}_t$ and $\widehat{\mathbf{y}}_t$. Then we use the L_2 distance of $\mathrm{BAdaIN}(\widehat{\mathbf{x}}_t, \widehat{\mathbf{y}}_t)$ and $\hat{\mathbf{x}}_t$ as the energy function for faithfulness. To verify SDDM's advantage, we only use this weak energy function.

5. Experiments

Datasets. We evaluate our SDDM on the following datasets. All the images are resized to 256×256 pixels.

- AFHQ (Choi et al., 2020) contains high-resolution animal faces of three domains: cat, dog, and wild. Each domain has 500 testing images. we conduct Cat $\rightarrow$ Dog and Wild $\rightarrow$ Dog on this dataset, following the experiments of CUT (Park et al., 2020) and EGSDE (Zhao et al., 2022).
- CelebA-HQ (Karras et al., 2017) consists of high-quality human face images of two categories, male and female. Each category contains 1000 testing images. We conduct Male $\rightarrow$ Female on this dataset, following the experiments of EGSDE (Zhao et al., 2022).

Evaluation Metrics. We evaluate our translated images from two aspects. One is to assess the distance between the translated and the source images, and we report the SSIM between them. The other is to evaluate the distance of generated images and target domain images, and we calculate the widely-used Frechet Inception Score (FID) (Heusel et al., 2017) between the generated images and the target domain images. Details about the FID settings are in Appendix D.

5.1. Comparison with the State-of-the-arts

We compare our experiments with other GANs-based and SBDM-based methods, as shown in Table 1.

Compared with other SBDM-based methods, our SDDM improves on both metrics FID and SSIM, which indicates the effectiveness of the two-stage generation process of our SDDM via the decomposition of score function and energy guidance with manifolds. Especially compared with EGSDE*, which has strong pre-trained energy functions, in the Cat $\rightarrow$ Dog task, our SDDM improves the FID score by 3.53 and SSIM score by 0.007 with much lower time

Table 1. Quantitative comparisons. The results marked with * come from (Zhao et al., 2022) Our method and ILVR have 100 diffusion steps. SDEdit and EGSDE* have 1000 diffusion steps. For a fair comparison with our SDDM, we also report the results of EGSDE** with 200 diffusion steps. All SBDM-based methods are repeated 5 times to reduce the randomness. Details about SDDM and SDDM† are shown in Appendix C.2.

MODEL	FID↓	SSIM↑
CAT → DOG		
CYCLEGAN*	85.9	-
MUNIT*	104.4	-
DRIT*	123.4	-
DISTANCE*	155.3	-
SELFDISTANCE*	144.4	-
GCGAN*	96.6	-
LSESIM*	72.8	-
ITTR (CUT)*	68.6	-
STARGAN V2*	**54.88 ± 1.01**	0.27 ± 0.003
CUT*	76.21	**0.601**
SDEDIT*	74.17 ± 1.01	**0.423 ± 0.001**
ILVR*	74.37 ± 1.55	0.363 ± 0.001
EGSDE*	65.82 ± 0.77	0.415 ± 0.001
EGSDE**	70.16± 1.03	0.411 ± 0.001
SDDM(OURS)	**62.29 ± 0.63**	**0.422± 0.001**
SDDM† (OURS)	**49.43 ± 0.23**	0.361± 0.001
WILD → DOG		
SDEDIT*	68.51 ± 0.65	**0.343 ± 0.001**
ILVR*	75.33 ± 1.22	0.287 ± 0.001
EGSDE*	**59.75 ± 0.62**	**0.343 ± 0.001**
SDDM(OURS)	**57.38 ± 0.53**	0.328 ± 0.001
MALE → FEMALE		
SDEDIT*	49.43 ± 0.47	**0.572 ± 0.000**
ILVR*	46.12 ± 0.33	0.510 ± 0.001
EGSDE*	**41.93 ± 0.11**	**0.574 ± 0.000**
EGSDE**	45.12± 0.24	0.512 ± 0.001
SDDM(OURS)	**44.37± 0.23**	0.526 ± 0.001

steps, 1000 → 100. For the comparison with EGSDE** having 200 diffusion steps, SDDM improves the FID score by 7.87 and the SSIM score by 0.011 in the Cat → Dog task and improves the FID score by 0.75 and the SSIM score by 0.014 in the Male → Female task, which suggests the advantage of our SDDM in fewer diffusion steps. The visual comparison is in Appendix F.

5.2. Ablation Studies

Observations on Score Components. While performing the Cat → Dog experiment, we report the L_2 norms of the deterministic guidance values on $T_{\mathbf{x}_t}\mathcal{M}$ and $N_{\mathbf{x}_t}\mathcal{M}$. As shown in Figure 2, the component on the normal space has one in 128 dimensions but contains the most value of the deterministic guidance of diffusion models, while the component on the tangent space $\mathbf{s}_r(\cdot,\cdot)$ has 127 in 128 dimensions but contains a minimal value, which indicates we have relative large optimization space on the manifold which will not excessively interfere with the intermediate distributions.

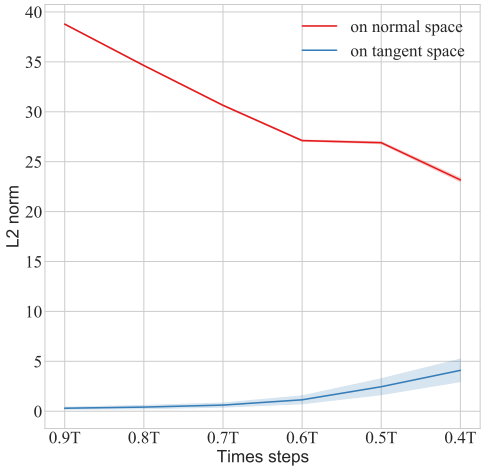

Figure 2. The mean and standard deviation of L_2 norms of $\mathbf{s}_r$ in $T_{\mathbf{x}_t}\mathcal{M}$ and other part in $N_{\mathbf{x}_t}\mathcal{M}$. We repeat 100 times of our SDDM with different reference images.

Comparison of Different Manifolds. We compare SDDM with different manifold methods and report the results in Table 2. Compared with the manifold restricted with a low-pass filter, the manifold restricted with our BAdaIN has better performance both on FID and SSIM, because our manifold separates the content refinement part and image denoising part better.

Table 2. Comparisons of different manifolds.

MODEL	FID↓	SSIM↑
SDDM(LOW-PASS FILTER)	67.56	0.411
SDDM(BADAIN)	**62.29**	**0.422**

Comparison of Different Coefficients. We have two coefficients at each iteration step, the coefficient of the step size λ of optimal multi-objection direction and the coefficient of the energy guidance λ_1. As in Table 3, the larger λ is, the better FID will be because, at each optimization on the manifold, it reaches a position with higher probability $p_{\mathcal{M}}(\mathbf{x})$, but when λ is too large, the FID score will be worse again. The λ_1 has a negative connection with the impact of energy guidance, which indicates that smaller λ_1 makes the energy guidance stronger and thus has a better SSIM score.

Comparison w./w.o. Multi-Objective Optimization. We compare SDDM with SDDM without the MOO method and report the FID, SSIM, and probability of negative impact (PNI), which indicates the situation that the total guidance

Table 3. Comparisons of different coefficients.

COEFFICIENTS	FID↓	SSIM↑
$\lambda = 2, \lambda_1 = 10$	65.09	**0.429**
$\lambda = 2, \lambda_1 = 40$	**62.02**	0.420
$\lambda = 1, \lambda_1 = 25$	66.04	0.428
$\lambda = 3, \lambda_1 = 25$	62.32	0.415
$\lambda = 2, \lambda_1 = 25$	62.29	0.422

including score and energy decreases the $p(\mathbf{x})$ in Table 4. The proposed SDDM method avoids such situations and reaches better performance.

Table 4. Comparisons of different manifold optimization policies.

MODEL	FID↓	SSIM↑	PNI↓
SDDM(W/O MOO)	64.93	0.421	0.024
SDDM	**62.29**	**0.422**	**0**

ϵ Policy in The Optimization on Manifolds. We mainly compare three different ϵ policies:

- Policy 1: The ϵ is very small such that in Algorithm 1, we iterate only once. With Remark 1, this method will introduce no additional calculations of scores.
- Policy 2: $\epsilon_t = \|\mathbf{s}_r(\mathbf{x}, t)\|$, which normally iterates twice in Algorithm 1. In practice, we iterate twice.
- Policy 3: At the time step T_0 in Algorithm 2, we set ϵ_t larger to iterate another 4 times. and other time steps are as same as Policy 1.

We report the FID and SSIM of different policies in Table 5. Policy 3 has the best performance, which reveals that iteration a little more at T_0 time step can balance different metrics better without introducing too much cost.

Table 5. Comparisons of the ϵ Policies.

POLICY	FID↓	SSIM↑
POLICY 1	**61.33**	0.413
POLICY 2	64.05	0.418
POLICY 3	62.29	**0.422**

The Choice of Middle-Time T_0 and Block Number. As shown in Figure 3, when we chunk more blocks or set the T_0 smaller, the generated image is more faithful to the reference image. But too many blocks will also introduce some bad details, like the mouth in the left bottom image.

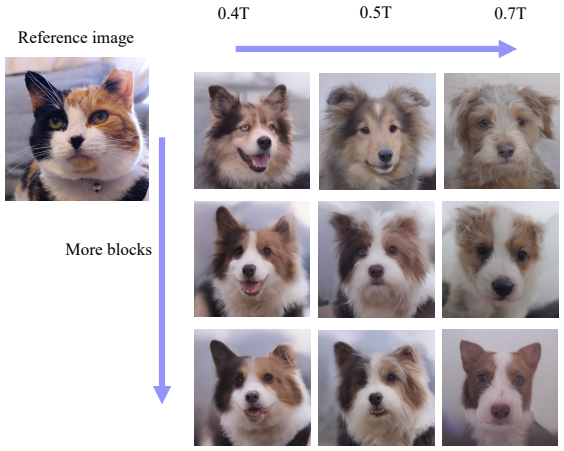

Figure 3. The comparison of different numbers of blocks and different middle time T_0s.

Table 6. Comparisons of different block numbers.

BLOCK NUMBER	FID↓	SSIM↑
8×8	**54.56**	0.359
16×16	62.29	0.422
32×32	68.03	**0.426**

6. Related Work

GAN-based Unpaired Image-to-Image Translation. There are mainly two categories of GANs-based methods in the unpaired I2I task. One is two-side way, while the other is one-side mapping. In the first category, the key idea is that the translated image could be translated back with another inverse mapping. CycleGAN (Zhu et al., 2017), DualGAN (Yi et al., 2017), DiscoGAN (Kim et al., 2017), SCAN (Van Gansbeke et al., 2020) and U-GAT-IT (Kim et al., 2019) are in this class. But translations usually lose information. Several new studies have started to map two domains to the same metric space and use the distance of this space as supervision.DistanceGAN (Benaim & Wolf, 2017), GCGAN (Fu et al., 2019), CUT (Park et al., 2020) and LSeSim (Zheng et al., 2021) are in this categoriy.

It is also noteworthy that other techniques have been proposed to tackle the problem of unpaired image-to-image translation. For instance, some studies (Xie et al., 2021; 2018) leverage cooperative learning, whereas others (Zhao et al., 2021) adopt an energy-based framework or a short-run MCMC like Langevin dynamics (Xie et al., 2016).

SBDMs-based Conditional Methods. There are mainly two classes of conditional generation with SBDMs. The first one is to empower SBDMs with the conditional generation ability during training with the classifier-free guidance trick (Ho & Salimans, 2022), which learns the score functions and conditional score functions via a single neural network. The other method is to train another classifier to lead the learned score functions for a conditional generation. EGSDE (Zhao et al., 2022) generalizes the classifiers to any energy-based functions. These methods cannot describe the intermediate distributions clearly, which is a hard problem because the distributions of adjacent time steps are deeply

coupled. However, when the conditions can give constraints to separate the adjacent distributions well, we can get better results, and this observation inspires our model.

7. Conclusions

In this work, we have presented a new score-decomposed diffusion model, SDDM, which leverages manifold analyses to decompose the score function and explicitly optimize the tangled distributions during image generation. SDDM derives manifolds to separate the distributions of adjacent time steps and decompose the score function or energy guidance into an image "denoising" part and a content "refinement" part. With the new multi-objective optimization algorithm and block adaptive instance normalization module, our realized SDDM method demonstrates promising results in unpaired image-to-image translation on two benchmarks. In future work, we plan to improve and apply the proposed SDDM model in more image translation tasks.

One limitation of our approach involves additional computations, although these computations are negligible compared to the inferences of neural networks. Additionally, we should prevent any misuse of generative algorithms for malicious purposes.

Acknowledgements

This work is supported by the National Key R&D Program of China under Grant No. 2021QY1500, and the State Key Program of the National Natural Science Foundation of China (NSFC) (No.61831022). It is also partly supported by the NSFC under Grant No. 62076238 and 62222606.

References

Augustin, M., Boreiko, V., Croce, F., and Hein, M. Diffusion visual counterfactual explanations. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022.

Bao, F., Li, C., Zhu, J., and Zhang, B. Analytic-DPM: An analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*, 2022a.

Bao, F., Zhao, M., Hao, Z., Li, P., Li, C., and Zhu, J. Equivariant energy-guided SDE for inverse molecular design. *arXiv preprint arXiv:2209.15408*, 2022b.

Benaim, S. and Wolf, L. One-sided unsupervised domain mapping. In *Advances in Neural Information Processing Systems*, pp. 752–762, 2017.

Choi, J., Kim, S., Jeong, Y., Gwon, Y., and Yoon, S. ILVR: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*, 2021.

Choi, Y., Uh, Y., Yoo, J., and Ha, J.-W. StarGAN v2: Diverse image synthesis for multiple domains. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8185–8194, 2020.

Chung, H., Sim, B., Ryu, D., and Ye, J. C. Improving diffusion models for inverse problems using manifold constraints. *arXiv preprint arXiv:2206.00941*, 2022.

Désidéri, J.-A. Multiple-gradient descent algorithm (MGDA) for multiobjective optimization. *Comptes Rendus Mathematique*, 350(5-6):313–318, 2012.

Dhariwal, P. and Nichol, A. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, pp. 8780–8794, 2021.

Fu, H., Gong, M., Wang, C., Batmanghelich, K., Zhang, K., and Tao, D. Geometry-consistent generative adversarial networks for one-sided unsupervised domain mapping. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2427–2436, 2019.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, pp. 6629–6640, 2017.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, pp. 6840–6851, 2020.

Hu, J., Liu, X., Wen, Z.-W., and Yuan, Y.-X. A brief introduction to manifold optimization. *Journal of the Operations Research Society of China*, 8(2):199–248, 2020.

Huang, X. and Belongie, S. Arbitrary style transfer in real-time with adaptive instance normalization. In *IEEE International Conference on Computer Vision*, pp. 1501–1510, 2017.

Huang, X., Liu, M.-Y., Belongie, S., and Kautz, J. Multimodal unsupervised image-to-image translation. In *European Conference on Computer Vision*, pp. 172–189, 2018.

Jiang, L., Zhang, C., Huang, M., Liu, C., Shi, J., and Loy, C. C. Tsit: A simple and versatile framework for image-to-image translation. In *European Conference on Computer Vision*, pp. 206–222, 2020.

Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of GANs for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

Kim, J., Kim, M., Kang, H., and Lee, K. U-gat-it: Unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. *arXiv preprint arXiv:1907.10830*, 2019.

Kim, T., Cha, M., Kim, H., Lee, J. K., and Kim, J. Learning to discover cross-domain relations with generative adversarial networks. In *International Conference on Machine Learning*, pp. 1857–1865, 2017.

Laurent, B. and Massart, P. Adaptive estimation of a quadratic functional by model selection. *Annals of Statistics*, pp. 1–18, 2000.

Lee, H.-Y., Tseng, H.-Y., Huang, J.-B., Singh, M., and Yang, M.-H. Diverse image-to-image translation via disentangled representations. In *European Conference on Computer Vision*, pp. 35–51, 2018.

Lee, J. *Introduction to topological manifolds*, volume 202. Springer Science & Business Media, 2010.

Lee, J. M. and Lee, J. M. *Smooth manifolds*. Springer, 2012.

Liu, L., Ren, Y., Lin, Z., and Zhao, Z. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022.

Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., and Zhu, J. DPM-Solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *arXiv preprint arXiv:2206.00927*, 2022.

Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.-Y., and Ermon, S. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, pp. 1–14, 2022.

Nichol, A. Q. and Dhariwal, P. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pp. 8162–8171, 2021.

Pang, Y., Lin, J., Qin, T., and Chen, Z. Image-to-image translation: Methods and applications. *IEEE Transactions on Multimedia*, 24:3859–3881, 2021.

Park, T., Efros, A. A., Zhang, R., and Zhu, J.-Y. Contrastive learning for unpaired image-to-image translation. In *European Conference on Computer Vision*, pp. 319–345, 2020.

Särkkä, S. and Solin, A. *Applied stochastic differential equations*, volume 10. Cambridge University Press, 2019.

Sener, O. and Koltun, V. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems*, pp. 1–12, 2018.

Shen, Z., Huang, M., Shi, J., Xue, X., and Huang, T. S. Towards instance-level image-to-image translation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3683–3692, 2019.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, pp. 11918–11930, 2019.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Tu, L. W. Manifolds. In *An Introduction to Manifolds*, pp. 47–83. Springer, 2011.

Van Gansbeke, W., Vandenhende, S., Georgoulis, S., Proesmans, M., and Van Gool, L. Scan: Learning to classify images without labels. In *European Conference on Computer Vision*, pp. 268–285, 2020.

Xie, J., Lu, Y., Zhu, S.-C., and Wu, Y. A theory of generative convnet. In *International Conference on Machine Learning*, pp. 2635–2644, 2016.

Xie, J., Lu, Y., Gao, R., Zhu, S.-C., and Wu, Y. N. Cooperative training of descriptor and generator networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(1):27–45, 2018.

Xie, J., Zheng, Z., Fang, X., Zhu, S.-C., and Wu, Y. N. Learning cycle-consistent cooperative networks via alternating mcmc teaching for unsupervised cross-domain translation. In *AAAI Conference on Artificial Intelligence*, pp. 10430–10440, 2021.

Yi, Z., Zhang, H., Tan, P., and Gong, M. DualGAN: Unsupervised dual learning for image-to-image translation. In *IEEE International Conference on Computer Vision*, pp. 2849–2857, 2017.

Zhao, M., Bao, F., Li, C., and Zhu, J. EGSDE: Unpaired image-to-image translation via energy-guided stochastic differential equations. *arXiv preprint arXiv:2207.06635*, 2022.

Zhao, Y., Xie, J., and Li, P. Learning energy-based generative models via coarse-to-fine expanding and sampling. In *International Conference on Learning Representations*, 2021.

Zheng, C., Cham, T.-J., and Cai, J. The spatially-correlative loss for various image translation tasks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16407–16417, 2021.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision*, pp. 2223–2232, 2017.

A. Basic Knowledge about Manifolds

These definitions come from (Lee, 2010; Tu, 2011; Lee & Lee, 2012).

Definition 5. Topological space.

A topological space $\mathcal{M}$ is locally Euclidean of dimension n if every point p in $\mathcal{M}$ has a neighborhood U such that there is a homeomorphism ϕ from U onto an open subset of $\mathbb{R}^n$. We call the pair $(U, \phi : U \to \mathbb{R}^n)$ a chart, U a coordinate neighborhood or an open coordinate set, and ϕ a coordinate map or a coordinate system on U. We say that a chart (U, ϕ) is centered at $p \in U$ if $\phi(p) = 0$. A chart (U, ϕ) about p simply means that (U, ϕ) is a chart and $p \in U$.

Definition 6. Locally Euclidean property.

The locally Euclidean property means that for each $p \in \mathcal{M}$, we can find the following:

- an open set $U \subset \mathcal{M}$ containing p;
- an open set $\widetilde{U} \subset \mathbb{R}^n$;and
- a homeomorphism $\phi : U \to \widetilde{U}$ (*i.e.*, a continuous bijective map with continuous inverse).

Definition 7. Topological manifold.

Suppose $\mathcal{M}$ is a topological space. We say $\mathcal{M}$ is a topological manifold of dimension n or a topological n-manifold if it has the following properties:

- M is a Hausdorff space: For every pair of points $p, q \in \mathcal{M}$, there are disjoint open subsets $U, V \subset \mathcal{M}$ such that $p \in U$and $q \in V$.
- $\mathcal{M}$ is second countable: There exists a countable basis for the topology of $\mathcal{M}$.
- $\mathcal{M}$ is locally Euclidean of dimension n: Every point has a neighborhood that is homeomorphic to an open subset of $\mathbb{R}^n$.

Definition 8. Tangent vector.

A tangent vector at a point p in a manifold $\mathcal{M}$ is a derivation at p.

Definition 9. Tangent space.

As for $\mathbb{R}^n$, the tangent vectors at p form a vector space $T_p(\mathcal{M})$, called the tangent space of $\mathcal{M}$ at p. We also write $T_p\mathcal{M}$ instead of $T_p(\mathcal{M})$.

Definition 10. Normal space.

the normal space to $\mathcal{M}$ at p to be the subspace $N_p\mathcal{M} \subset \mathbb{R}^m$ consisting of all vectors that are orthogonal to $T_p\mathcal{M}$ with respect to the Euclidean dot product. The normal bundle of M is the subset $N\mathcal{M} \subset \mathbb{R}^m \times \mathbb{R}^m$defined by

$$N\mathcal{M} = \coprod_{p \in M} N_p\mathcal{M} = \{(p, v) \in \mathbb{R}^m \times \mathbb{R}^m : p \in M \text{ and } v \in N_p\mathcal{M}\} \tag{18}$$

B. Proofs

B.1. Proof of Lemma 1

Proof. Consider the local coordinate system at $\mathbf{x}$. Suppose $\{x_i\}_{i=1,2,\ldots,d}$ are the orthonormal basis of $\mathbb{R}^d$ and $\{x_i\}_{i=1,2,\ldots,m}$ $(m < d)$ are in the tangent space $T_{\mathbf{x}}\mathcal{M}$ and the rest of them are in the normal space $N_{\mathbf{x}}\mathcal{M}$. Then:

$$\begin{aligned}
\mathbf{s}_r(\mathbf{x}) =& \nabla_{\mathbf{x}} \log p_{\mathcal{M}}(\mathbf{x}) = \sum_{i=1}^{d} \frac{\partial}{\partial x_i} \log p_{\mathcal{M}}(\mathbf{x}) \\
=& \sum_{i=1}^{m} \frac{\partial}{\partial x_i} \log p_{\mathcal{M}}(\mathbf{x}) = \sum_{i=1}^{m} \frac{\partial}{\partial x_i} \log Cp(\mathbf{x}) \\
=& \sum_{i=1}^{m} \frac{\partial}{\partial x_i} \log p(\mathbf{x}) = \mathbf{s}(\mathbf{x})|_{T_{\mathbf{x}}\mathcal{M}}.
\end{aligned} \tag{19}$$

Therefore, we have:

$$\begin{aligned}\mathbf{s}(\mathbf{x}) =&\mathbf{s}(\mathbf{x})|_{T_{\mathbf{x}}\mathcal{M}} + \mathbf{s}(\mathbf{x})|_{N_{\mathbf{x}}\mathcal{M}} \\ =&\mathbf{s}_r(\mathbf{x}) + \mathbf{s}_d(\mathbf{x})\end{aligned} \tag{20}$$

□

In the following sections, consider the distributions of perturbed reference image $\mathbf{y_t} = \hat{\alpha}_t \mathbf{y}_0 + \hat{\beta}_t \mathbf{z}_t$, where $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and the reference image $\mathbf{y}_0$ is fixed.

B.2. Proof of Proposition 1 and Lemma 3

Before proving the Proposition 1 and Lemma 3, we will prove another two relevant lemmas.

Lemma 4. *$\mathbf{y_t}$ is clustered on the $(d-1)-dim$ manifolds $\{\mathcal{M}_t\}$ restricted with the first-order moment constraints,*

$$\mu[\mathbf{x}_t] = \hat{\alpha}_t \, \mu[\mathbf{y}_0] \tag{21}$$

under the d_2 distance of $\mathbb{R}^d$.

Strictly speaking, in the original Cartesian coordinate system $\mathbb{R}^d$.

$\forall \epsilon, \xi > 0, \exists D > 0, \forall d > D$ we have:

$$\mathcal{P}\left(d_2\left(\mathbf{y}_t, \mathcal{M}_t\right) < \epsilon\sqrt{d}\right) > 1 - \xi$$

Proof. The manifold provided with restriction of Eqn. (21) is a hyperplane in $\mathbb{R}^d$, and the normal vector $\mathbf{n} = \frac{1}{\sqrt{d}}(1, 1, ..., 1)$. Then the L_2 distance from $\mathbf{y}_t$ to the manifold $\mathcal{M}_t$ is $|\hat{\beta}_t \mathbf{z}_t \cdot \mathbf{n}|$, where $\hat{\beta}_t \mathbf{z}_t \cdot \mathbf{n} \sim \mathcal{N}(0, \hat{\beta}_t^2)$. Therefore,

$$d_2\left(\mathbf{y}_t, \mathcal{M}_t\right) = \sqrt{d}|\frac{1}{\sqrt{d}}\hat{\beta}_t \mathbf{z}_t \cdot \mathbf{n}| \tag{22}$$

Thus, as $d \to +\infty$, the variance of $\frac{1}{\sqrt{d}}\hat{\beta}_t \mathbf{z}_t \cdot \mathbf{n} \to 0$.

Then strictly speaking, $\forall \epsilon, \xi > 0, \exists D > 0, \forall d > D$ we have:

$$\mathcal{P}\left(d_2\left(\mathbf{y}_t, \mathcal{M}_t\right) < \epsilon\sqrt{d}\right) > 1 - \xi$$

□

Lemma 5. *Suppose $\mathbf{y}_t$ shares the same bound A, which means $\|\mathbf{y}_t\|_\infty < A$. $\mathbf{y_t}$ is clustered on the $(d-1)-dim$ manifolds $\{\mathcal{M}_t\}$ restricted with the second-order moment constraints,*

$$\mu\left[\mathbf{x}_t - \hat{\alpha}_t \, \mu[\mathbf{y}_0]\right]^2 = \hat{\alpha}_t^2 \, \mathrm{Var}[\mathbf{y}_0] + \hat{\beta}_t^2 \tag{23}$$

under the metric of d_2 distance.

Strictly speaking,

$\forall \epsilon, \xi > 0, \exists D > 0, \forall d > D$ we have:

$$\mathcal{P}\left(d_2\left(\mathbf{y}_t, \mathcal{M}_t\right) < \epsilon\sqrt{d}\right) > 1 - \xi$$

Proof. The manifold provided with restriction of Eqn. (23) is a hypersphere on the $\mathbb{R}^d$. The center of the hypersphere is

$\hat{\alpha}_t\,\mu[\mathbf{y}_0](1,1,...,1)$ and the radius is $\sqrt{d}\sqrt{\hat{\alpha}_t^2\,\mathrm{Var}[\mathbf{y}_0]+\hat{\beta}_t^2}$. The square of the L_2 distance of $\mathbf{y}_t$ to the center is:

$$\begin{aligned}\left[\hat{\alpha}_t\mathbf{y}_0+\hat{\beta}_t\mathbf{z}_t-\hat{\alpha}_t\,\mu[\mathbf{y}_0]\right]^2&=\sum_{i=1}^{d}\left[(\hat{\alpha}_t\mathbf{y}_0^i-\hat{\alpha}_t\,\mu[\mathbf{y}_0])+\hat{\beta}_t\mathbf{z}_t^i\right]^2\\&=\sum_{i=1}^{d}\left[(\hat{\alpha}_t\mathbf{y}_0^i-\hat{\alpha}_t\,\mu[\mathbf{y}_0])^2+\hat{\beta}_t^2(\mathbf{z}_t^i)^2+2(\hat{\alpha}_t\mathbf{y}_0^i-\hat{\alpha}_t\,\mu[\mathbf{y}_0])\hat{\beta}_t\mathbf{z}_t^i\right]\\&=d\hat{\alpha}_t^2\,\mathrm{Var}[\mathbf{y}_0]+\hat{\beta}_t^2\mathbf{z}_t^2+2\hat{\alpha}_t\hat{\beta}_t(\mathbf{y_0}-\mu[\mathbf{y}_0])\cdot\mathbf{z}_t,\end{aligned}\tag{24}$$

Therefore,

$$\begin{aligned}d_2\left(\mathbf{y}_t,\mathcal{M}_t\right)=&\left|\sqrt{\left[\hat{\alpha}_t\mathbf{y}_0+\hat{\beta}_t\mathbf{z}_t-\hat{\alpha}_t\,\mu[\mathbf{y}_0]\right]^2}-\sqrt{d}\sqrt{\hat{\alpha}_t^2\,\mathrm{Var}[\mathbf{y}_0]+\hat{\beta}_t^2}\right|\\=&\frac{\left|\left[\hat{\alpha}_t\mathbf{y}_0+\hat{\beta}_t\mathbf{z}_t-\hat{\alpha}_t\,\mu[\mathbf{y}_0]\right]^2-d\hat{\alpha}_t^2\,\mathrm{Var}[\mathbf{y}_0]-d\hat{\beta}_t^2\right|}{\sqrt{\left[\hat{\alpha}_t\mathbf{y}_0+\hat{\beta}_t\mathbf{z}_t-\hat{\alpha}_t\,\mu[\mathbf{y}_0]\right]^2}+\sqrt{d}\sqrt{\hat{\alpha}_t^2\,\mathrm{Var}[\mathbf{y}_0]+\hat{\beta}_t^2}}\\\leq&\frac{1}{\sqrt{d}\hat{\beta}_t}\left|\left[\hat{\alpha}_t\mathbf{y}_0+\hat{\beta}_t\mathbf{z}_t-\hat{\alpha}_t\,\mu[\mathbf{y}_0]\right]^2-d\hat{\alpha}_t^2\,\mathrm{Var}[\mathbf{y}_0]-d\hat{\beta}_t^2\right|\\=&\frac{1}{\sqrt{d}\hat{\beta}_t}\left|d\hat{\alpha}_t^2\,\mathrm{Var}[\mathbf{y}_0]+\hat{\beta}_t^2\mathbf{z}_t^2+2\hat{\alpha}_t\hat{\beta}_t(\mathbf{y_0}-\mu[\mathbf{y}_0])\cdot\mathbf{z}_t-d\hat{\alpha}_t^2\,\mathrm{Var}[\mathbf{y}_0]-d\hat{\beta}_t^2\right|\\\leq&\frac{1}{\sqrt{d}}\left(\hat{\beta}_t^2\left|\mathbf{z}_t^2-d\right|+\left|2\hat{\alpha}_t\hat{\beta}_t(\mathbf{y_0}-\mu[\mathbf{y}_0])\cdot\mathbf{z}_t\right|\right)\\=&\sqrt{d}\left(\hat{\beta}_t\frac{\left|\mathbf{z}_t^2-d\right|}{d}+2\hat{\alpha}_t\frac{\left|(\mathbf{y_0}-\mu[\mathbf{y}_0])\cdot\mathbf{z}_t\right|}{d}\right),\end{aligned}\tag{25}$$

where the $\mathbf{z}_t^2$ is a stand chi-square distribution with d degrees of freedom. We apply the standard Laurent-Massart bound (Laurent & Massart, 2000) for it and get

$$\begin{aligned}\mathcal{P}[\mathbf{z}_t^2-d\geq 2\sqrt{dt}+2t]\leq e^{-t}\\\mathcal{P}[\mathbf{z}_t^2-d\leq -2\sqrt{dt}]\leq e^{-t},\end{aligned}\tag{26}$$

which holds for any $t>0$. We let $t=d\epsilon'$, where the $\epsilon'+\sqrt{\epsilon'}=\frac{\epsilon}{4\hat{\beta}_t}$ for any given ϵ, Then we have

$$\mathcal{P}\left[-2d\sqrt{\epsilon'}\leq\mathbf{z}_t^2-d\leq 2d(\epsilon'+\sqrt{\epsilon'})\right]\geq 1-2e^{-d\epsilon'}.\tag{27}$$

Therefore,

$$\mathcal{P}\left[\hat{\beta}_t\frac{\left|\mathbf{z}_t^2-d\right|}{d}<\frac{\epsilon}{2}\right]>1-2e^{-d\epsilon'}\tag{28}$$

and $\exists D_1,\forall d>D_1, 4e^{-d\epsilon'}\leq\xi$, thus

$$\mathcal{P}\left[\hat{\beta}_t\frac{\left|\mathbf{z}_t^2-d\right|}{d}<\frac{\epsilon}{2}\right]>1-\frac{\xi}{2}\tag{29}$$

Similar to Lemma 4, $2\hat{\alpha}_t\frac{(\mathbf{y_0}-\mu[\mathbf{y}_0])\cdot\mathbf{z}_t}{d}$ is a Gaussian distribution, and the mean is 0, variance is bounded with $\frac{(4\hat{\alpha}_tA)^2}{d}$. As $d\to+\infty$, the variance $\to 0$, thus $\exists D_2,\forall d>D_2$, we have

$$\mathcal{P}\left[2\hat{\alpha}_t\frac{\left|(\mathbf{y_0}-\mu[\mathbf{y}_0])\cdot\mathbf{z}_t\right|}{d}<\frac{\epsilon}{2}\right]>1-\frac{\xi}{2}\tag{30}$$

Finally, $\forall \epsilon, \xi > 0, \exists D = Max\{D_1, D_2\}, \forall d > D$,

$$\begin{aligned}
\mathcal{P}\left(d_2\left(\mathbf{y}_t, \mathcal{M}_t\right) < \epsilon\sqrt{d}\right) \geq & \mathcal{P}\left[\hat{\beta}_t \frac{\left|\mathbf{z}_t^2 - d\right|}{d} < \frac{\epsilon}{2}, 2\hat{\alpha}_t \frac{\left|(\mathbf{y_0} - \mu[\mathbf{y}_0]) \cdot \mathbf{z}_t\right|}{d} < \frac{\epsilon}{2}\right] \\
= & \mathcal{P}\left[\hat{\beta}_t \frac{\left|\mathbf{z}_t^2 - d\right|}{d} < \frac{\epsilon}{2}\right] + \mathcal{P}\left[2\hat{\alpha}_t \frac{\left|(\mathbf{y_0} - \mu[\mathbf{y}_0]) \cdot \mathbf{z}_t\right|}{d} < \frac{\epsilon}{2}\right] \\
& - \mathcal{P}\left[\hat{\beta}_t \frac{\left|\mathbf{z}_t^2 - d\right|}{d} < \frac{\epsilon}{2} \text{ or } 2\hat{\alpha}_t \frac{\left|(\mathbf{y_0} - \mu[\mathbf{y}_0]) \cdot \mathbf{z}_t\right|}{d} < \frac{\epsilon}{2}\right] \\
\geq & 1 - \frac{\xi}{2} + 1 - \frac{\xi}{2} - 1 \\
= & 1 - \xi
\end{aligned} \tag{31}$$

□

Then we will prove the Proposition 1.

Proof. Consider $\mathcal{M}_t$, which is restricted with:

$$\begin{aligned}
\mu[\mathbf{x}_t] &= \hat{\alpha}_t\, \mu[\mathbf{y}_0], \\
\mathrm{Var}[\mathbf{x}_t] &= \hat{\alpha}_t^2\, \mathrm{Var}[\mathbf{y}_0] + \hat{\beta}_t^2.
\end{aligned} \tag{32}$$

We can substitute the $\mu[\mathbf{x}_t]$ with $\sqrt{\bar{\alpha}_t}\, \mu[\mathbf{y}_0]$ in the calculation of the variance and get the following equivalent restrictions:

$$\begin{aligned}
\mu[\mathbf{x}_t] &= \hat{\alpha}_t\, \mu[\mathbf{y}_0], \\
\mu\left[\mathbf{x}_t - \hat{\alpha}_t\, \mu[\mathbf{y}_0]\right]^2 &= \hat{\alpha}_t^2\, \mathrm{Var}[\mathbf{y}_0] + \hat{\beta}_t^2.
\end{aligned} \tag{33}$$

These two constraints correspond to Lemma 4 and Lemma 5 respectively. We denote the manifold restricted with one of these constraints as $\mathcal{M}_{tA}$ and $\mathcal{M}_{tB}$. $\mathcal{M}_{tA} \cap \mathcal{M}_{tB} = \mathcal{M}_t$. Suppose the angle of $\mathcal{M}_{tA}$ and $\mathcal{M}_{tB}$ at the intersection is θ. Then locally the hypersphere can be treated as a hyperplane and the error is second-order. We have the following relationship when ϵ is small:

$$B_{(\frac{1}{\sin\frac{\theta}{2}}+1)\epsilon\sqrt{d}}(\mathcal{M}_t) \supset B_{\epsilon\sqrt{d}}(\mathcal{M}_{tA}) \cap B_{\epsilon\sqrt{d}}(\mathcal{M}_{tB}) \tag{34}$$

Then, according to Lemma 4 and Lemma 5:

$\forall \epsilon, \xi, \exists D, \forall d \geq D$, where ϵ is small,

$$\begin{aligned}
\mathcal{P}\left(d_2\left(\mathbf{x}_t, \mathcal{M}_{tA}\right) < \epsilon\sqrt{d}\right) &> 1 - \frac{\xi}{2} \\
\mathcal{P}\left(d_2\left(\mathbf{x}_t, \mathcal{M}_{tB}\right) < \epsilon\sqrt{d}\right) &> 1 - \frac{\xi}{2}.
\end{aligned} \tag{35}$$

Then,

$$\begin{aligned}
\mathcal{P}\left(d_2\left(\mathbf{x}_t, \mathcal{M}_t\right) < (\frac{1}{\sin\frac{\theta}{2}}+1)\epsilon\sqrt{d}\right) \geq & \mathcal{P}\left[d_2\left(\mathbf{x}_t, \mathcal{M}_{tA}\right) < \epsilon\sqrt{d}, d_2\left(\mathbf{x}_t, \mathcal{M}_{tB}\right) < \epsilon\sqrt{d}\right] \\
= & \mathcal{P}\left(d_2\left(\mathbf{x}_t, \mathcal{M}_{tA}\right) < \epsilon\sqrt{d}\right) + \mathcal{P}\left(d_2\left(\mathbf{x}_t, \mathcal{M}_{tB}\right) < \epsilon\sqrt{d}\right) \\
& - \mathcal{P}\left[d_2\left(\mathbf{x}_t, \mathcal{M}_{tA}\right) < \epsilon\sqrt{d} \text{ or } d_2\left(\mathbf{x}_t, \mathcal{M}_{tB}\right) < \epsilon\sqrt{d}\right] \\
> & 1 - \frac{\xi}{2} + 1 - \frac{\xi}{2} - 1 \\
= & 1 - \xi
\end{aligned} \tag{36}$$

□

Let $\hat{\alpha}_t^2 = \bar{\alpha}_t$ and $\hat{\beta}_t^2 = 1 - \bar{\alpha}_t$ and only consider the block(c, i, j), We can get the Lemma 3.

B.3. Proof of Lemma 2

Proof. We just use the following hyperplane:

$$\mu[\mathbf{x}] = \hat{\alpha}_{\frac{t+t'}{2}}\, \mu[\mathbf{y}_0] \tag{37}$$

Then, because$\hat{\alpha}_t$ monotonically decreasing with t in VP-SDE. Therefore, The hyperplanes $\mu[\mathbf{x}] = \hat{\alpha}_t\, \mu[\mathbf{y}_0]$ and $\mu[\mathbf{x}] = \hat{\alpha}_{t'}\, \mu[\mathbf{y}_0]$ are on different sides of the given hyperplane. As a consequence, $\mathcal{M}_t$ and $\mathcal{M}_{t'}$ are on different sides of the given hyperplane. □

B.4. Proof of Proposition 2

$\exists!\mathbf{v}_{\mathbf{x_t}} \in N_{\mathbf{x}_t}\mathcal{M}_t$ that $\mathbf{x}_t + \mathbf{v}_{\mathbf{x_t}} \in \mathcal{M}_{t-1}$.

Proof. We consider the 2-dim normal space $N_{\mathbf{x}_t}\mathcal{M}_t$. Easy to show that it has these two orthogonal basis vectors, $[\mathbf{x}_t - \mu[\mathbf{x}_t](1, 1, ..., 1)]$ and $\mu[\mathbf{x}_t](1, 1, ..., 1)$. There are only two points in 2-dim normal space $N_{\mathbf{x}_t}\mathcal{M}_t$ that are in $\mathcal{M}_{t-1}$ because there are only two points, in 2-dim normal space $N_{\mathbf{x}_t}\mathcal{M}_t$ meet the following conditions:

- The distance between this point to $\mu[\mathbf{x}_{t-1}](1, 1, ..., 1)$ is C_0.
- The line connecting this point to $\mu[\mathbf{x}_{t-1}](1, 1, ..., 1)$ is perpendicular to $(1, 1, ..., 1)$.

And there is only one of them near $\mathbf{x}_t$.

Thus, $\exists!\mathbf{v}_{\mathbf{x_t}} \in N_{\mathbf{x}_t}\mathcal{M}_t$ that $\mathbf{x}_t + \mathbf{v}_{\mathbf{x_t}} \in \mathcal{M}_{t-1}$.

□

$N_{\mathbf{x}_t}\mathcal{M}_t = N_{\mathbf{x}_t+\mathbf{v}_{\mathbf{x_t}}}\mathcal{M}_{t-1}$.

Proof. Because $\mathbf{x}_t + \mathbf{v}_{\mathbf{x_t}} \in N_{\mathbf{x}_t}\mathcal{M}_t$, $\mu[\mathbf{x}_t + \mathbf{v}_{\mathbf{x_t}}](1, 1, ..., 1)\ in N_{\mathbf{x}_t}\mathcal{M}_t$, and they are two orthogonal basis vectors of $N_{\mathbf{x}_t+\mathbf{v}_{\mathbf{x_t}}}\mathcal{M}_{t-1}$.

Therefore, $N_{\mathbf{x}_t}\mathcal{M}_t = N_{\mathbf{x}_t+\mathbf{v}_{\mathbf{x_t}}}\mathcal{M}_{t-1}$.

□

$\mathbf{v}_{\mathbf{x_t}}$ is a transition map from $T_{\mathbf{x}_t}\mathcal{M}_t$ to $T_{\mathbf{x}_t+\mathbf{v}_{\mathbf{x_t}}}\mathcal{M}_{t-1}$.

Proof. Because $N_{\mathbf{x}_t}\mathcal{M}_t = N_{\mathbf{x}_t+\mathbf{v}_{\mathbf{x_t}}}\mathcal{M}_{t-1}$, $T_{\mathbf{x}_t}\mathcal{M}_t$ and $T_{\mathbf{x}_t+\mathbf{v}_{\mathbf{x_t}}}\mathcal{M}_{t-1}$ are parallel, and $\mathbf{v}_{\mathbf{x_t}}$ maps $\mathbf{x}_t$ to $\mathbf{x}_t + \mathbf{v}_{\mathbf{x_t}}$.

Therefore, $\mathbf{v}_{\mathbf{x_t}}$ is a transition map from $T_{\mathbf{x}_t}\mathcal{M}_t$ to $T_{\mathbf{x}_t+\mathbf{v}_{\mathbf{x_t}}}\mathcal{M}_{t-1}$.

□

$\mathbf{v}_{\mathbf{x_t}}$ is determained with $\mathbf{f}(\cdot, \cdot), \mathbf{x_t}, \mathbf{y}_0$ and $g(\cdot)$.

Proof. As proved in (Särkkä & Solin, 2019), the means and covariances of linear SDEs can be transformed to corresponding ODEs. Therefore, suppose $\mathbf{y_t} = \hat{\alpha}_t\mathbf{y}_0 + \hat{\beta}_t\mathbf{z}_t$, $\mathbf{y_{t-1}} = \hat{\alpha}_{t-1}\mathbf{y}_0 + \hat{\beta}_{t-1}\mathbf{z}_{t-1}$, all the coeffients are determained by $\mathbf{f}(\cdot, \cdot), g(\cdot)$. For clarity, we will represent $\mathbf{v}_{\mathbf{x_t}}$ with $\mathbf{x}_t, \mathbf{y}_0$ and the coefficients above.

In fact, it is easy to show that

$$\sqrt{\hat{\alpha}_{t-1}^2 Var[\mathbf{y}_0] + \hat{\beta}_{t-1}^2}\left(\frac{\mathbf{x}_t - \hat{\alpha}_t\, \mu(\mathbf{y}_0)}{\sqrt{\hat{\alpha}_t^2 Var[\mathbf{y}_0] + \hat{\beta}_t^2}}\right) + \hat{\alpha}_{t-1}\, \mu(\mathbf{y}_0) \tag{38}$$

is in both $N_{\mathbf{x}_t}\mathcal{M}_t$ and $\mathcal{M}_{t-1}$, and is the one in $N_{\mathbf{x}_t}\mathcal{M}_t \cap \mathcal{M}_{t-1}$ near $\mathbf{x}_t$. Therefore,

$$\mathbf{v}_{\mathbf{x_t}} = \sqrt{\hat{\alpha}_{t-1}^2 Var[\mathbf{y}_0] + \hat{\beta}_{t-1}^2}\left(\frac{\mathbf{x}_t - \hat{\alpha}_t\, \mu(\mathbf{y}_0)}{\sqrt{\hat{\alpha}_t^2 Var[\mathbf{y}_0] + \hat{\beta}_t^2}}\right) + \hat{\alpha}_{t-1}\, \mu(\mathbf{y}_0) - \mathbf{x}_t \tag{39}$$

□

B.5. Proof of Remark 3

Proof. Equivalently, we prove that $\mathbf{x}_t \in N_{\mathbf{x}_t}\mathcal{M}_t$. Consider $\mathcal{M}_{tA}$ and $\mathcal{M}_{tB}$ restricted with one of the equations in Eqn. (33). We do the following decomposition of $\mathbf{x}_t$

$$\mathbf{x}_t = [\mathbf{x}_t - \mu[\mathbf{x}_t](1,1,...,1)] + \mu[\mathbf{x}_t](1,1,...,1) \tag{40}$$

Easy to know that $[\mathbf{x}_t - \mu[\mathbf{x}_t](1,1,...,1)] \in N_{\mathbf{x}_t}\mathcal{M}_{tB}$ and $\mu[\mathbf{x}_t](1,1,...,1) \in N_{\mathbf{x}_t}\mathcal{M}_{tA}$. Because $\mathcal{M}_t = \mathcal{M}_{tA} \cap \mathcal{M}_{tB}$, thus the two compoments are all $\in N_{\mathbf{x}_t}\mathcal{M}_t$ and then $\mathbf{x}_t \in N_{\mathbf{x}_t}\mathcal{M}_t$. □

B.6. Proof of Remark 7

Proof.

$$\begin{aligned}\mathbb{E}[\mathbf{\Phi}(\mathbf{y}_t)] &= \mathbb{E}[\otimes_{c,i,j}\mu[\mathbf{y}_t^{c,i,j}]] \\ &= \otimes_{c,i,j}\mathbb{E}[\mu[\mathbf{y}_t^{c,i,j}]] \\ &= \otimes_{c,i,j}\sqrt{\bar{\alpha}_t}\,\mu[\mathbf{y}_0^{c,i,j}]\end{aligned} \tag{41}$$

□

C. Details about SDDM

Assumption 1. Suppose $\mathbf{s}(\cdot,\cdot) : \mathbb{R}^D \times \mathbb{R} \to \mathbb{R}^D$ is the score-based model, $\mathbf{f}(\cdot,\cdot) : \mathbb{R}^D \times \mathbb{R} \to \mathbb{R}^D$ is the drift coefficient, $g(\cdot) : \mathbb{R} \to \mathbb{R}$ is the diffusion coefficient, and $\mathcal{E}(\cdot,\cdot,\cdot) : \mathbb{R}^D \times \mathbb{R}^D \times \mathbb{R} \to \mathbb{R}$ is the energy function. $\mathbf{y}_0$ is the given source image.

Like previous works (Zhao et al., 2022; Choi et al., 2021; Meng et al., 2022), we define a valid conditional distribution $p(\mathbf{x}_0 \mid \mathbf{y}_0)$ under following assumptions:

- $\exists C > 0, \forall t \in \mathbb{R}, \forall \mathbf{x}, \mathbf{y}_0 \in \mathbb{R}^D : \|f(\mathbf{x},t) - f(\mathbf{y}_0,t)\|_2 \le C\|\mathbf{x}-\mathbf{y}\|_2$.
- $\exists C > 0, \forall t,s \in \mathbb{R}, \forall \mathbf{x} \in \mathbb{R}^D : \|f(\mathbf{x},t) - f(\mathbf{x},s)\|_2 \le C|t-s|$.
- $\exists C > 0, \forall t \in \mathbb{R}, \forall \mathbf{x}, \mathbf{y}_0 \in \mathbb{R}^D : \|\mathbf{s}(\mathbf{x},t) - \mathbf{s}(\mathbf{y}_0,t)\|_2 \le C\|\mathbf{x}-\mathbf{y}\|_2$.
- $\exists C > 0, \forall t,s \in \mathbb{R}, \forall \mathbf{x} \in \mathbb{R}^D : \|\mathbf{s}(\mathbf{x},t) - \mathbf{s}(\mathbf{x},s)\|_2 \le C|t-s|$.
- $\exists C > 0, \forall t \in \mathbb{R}, \forall \mathbf{x}, \mathbf{y}_0 \in \mathbb{R}^D : \|\nabla_{\mathbf{x}}\mathcal{E}(\mathbf{x},\mathbf{y}_0,t) - \nabla_{\mathbf{y}}\mathcal{E}(\mathbf{y}_0,\mathbf{y}_0,t)\|_2 \le C\|\mathbf{x}-\mathbf{y}\|_2$.
- $\exists C > 0, \forall t,s \in \mathbb{R}, \forall \mathbf{x} \in \mathbb{R}^D : \|\nabla_{\mathbf{x}}\mathcal{E}(\mathbf{x},\mathbf{y}_0,t) - \nabla_{\mathbf{x}}\mathcal{E}(\mathbf{x},\mathbf{y}_0,s)\|_2 \le C|t-s|$.
- $\exists C > 0, \forall t,s \in \mathbb{R} : |g(t) - g(s)| \le C|t-s|$.

C.1. Details about Pre-Trained Diffusion Models

We use two pre-trained diffusion models and a VGG model.

In the Cat → Dog task and Wild → Dog task, we use the public pre-trained model provided in the official code https://github.com/jychoi118/ilvr_adm of ILVR (Choi et al., 2021).

In the Male → Female task, we use the public pre-trained model provided in the official code https://github.com/ML-GSAI/EGSDE of EGSDE (Zhao et al., 2022).

Our energy function uses the pre-trained VGG net provided in the unofficial open source code https://github.com/naoto0804/pytorch-AdaIN of AdaIN (Huang & Belongie, 2017).

C.2. Details about Our Default Model Settings

Our default SDDM settings:

- Using BAdaIN to construct manifolds.

- Using multi-optimization on manifolds.
- $\lambda = 2, \lambda_1 = 25$.
- Using ϵ Policy 3.
- Blocks are 16×16.
- $T_0 = 0.5T$.
- 100 diffusion steps.

SDDM† sets $T_0 = 0.6T$.

C.3. Implementation Details about Solving Problem (12)

To simplify the process, we denote all the vectors as $\{\mathbf{v}_i\}$, and coefficients as $\{\lambda_i\}$, and rewrite Problem (12) as

$$\min_{\substack{\lambda_1,\lambda_2,\ldots,\lambda_n\geq 0\\ \lambda_1+\lambda_2+\ldots+\lambda_n=1}} \left\{ \left\| \sum_{i=1}^{n} \lambda_i \mathbf{v}_i \right\|_2^2 \right\}. \tag{42}$$

When there are only two vectors (in our situation) and no restriction $\lambda_1, \lambda_2, \ldots, \lambda_n \geq 0$, we can get the following analytical solution:

$$\hat{\lambda}_1^* = \frac{(\mathbf{v}_2 - \mathbf{v}_1)^T \mathbf{v}_2}{\|\mathbf{v}_2 - \mathbf{v}_1\|_2^2}. \tag{43}$$

Therefore, easy to prove that when there are only two vectors, the analytical solution is:

$$\lambda_1^* = \min\left(1, \max\left(\frac{(\mathbf{v}_2 - \mathbf{v}_1)^T \mathbf{v}_2}{\|\mathbf{v}_2 - \mathbf{v}_1\|_2^2}, 0\right)\right). \tag{44}$$

For general situations, we can apply Frank–Wolfe algorithm on this problem as in (Sener & Koltun, 2018).

D. Details about FID calculation

The FID is calculated between 500 generated images and the target validation dataset containing 500 images in the Cat $\rightarrow$ Dog and Wild $\rightarrow$ Dog task. The number is 1000 in the Male $\rightarrow$ Female task. All experiments are repeated 5 times to eliminate the randomness.

E. FID on the Male→Female task

It is true that EGSDE with sufficient diffusion steps outperforms our SDDM on the Male→Female task, it is important to note that the energy function used in EGSDE is strongly pretrained on related datasets and contains significant domain-specific information. In contrast, to demonstrate the effectiveness and versatility of our framework, we intentionally chose to use a weak energy function consisting of only one layer of convolution without any further pretraining. After incorporating the strong guidance function from EGSDE, our method outperforms EGSDE in the FID score, as shown in the following table.

Table 7. The FID comparison between EGSDE and our SDDM with the same energy guidance function on the Male→Female task.

MODEL	FID↓
EGSDE	41.93 ± 0.11
SDDM(OURS)	**40.08 ± 0.13**

F. Samples

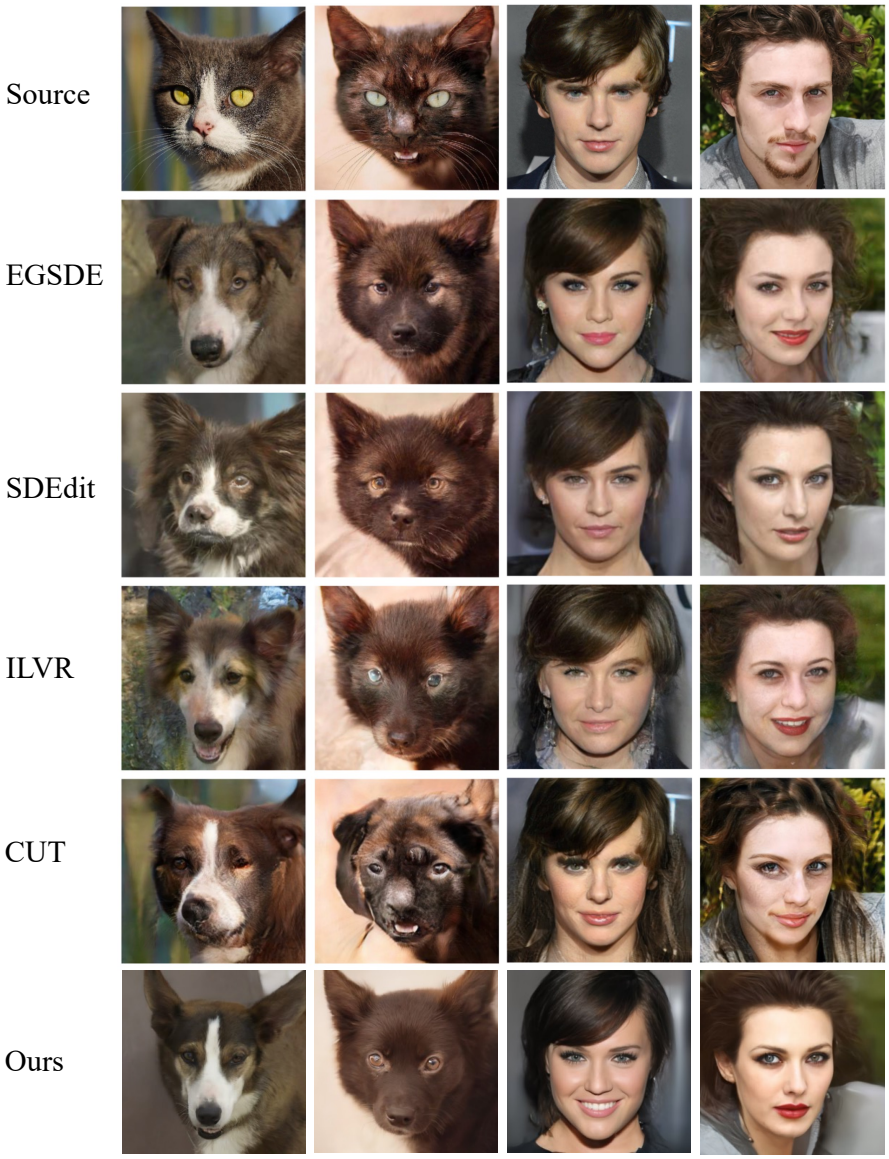

Figure 4. The visual comparison of different methods.

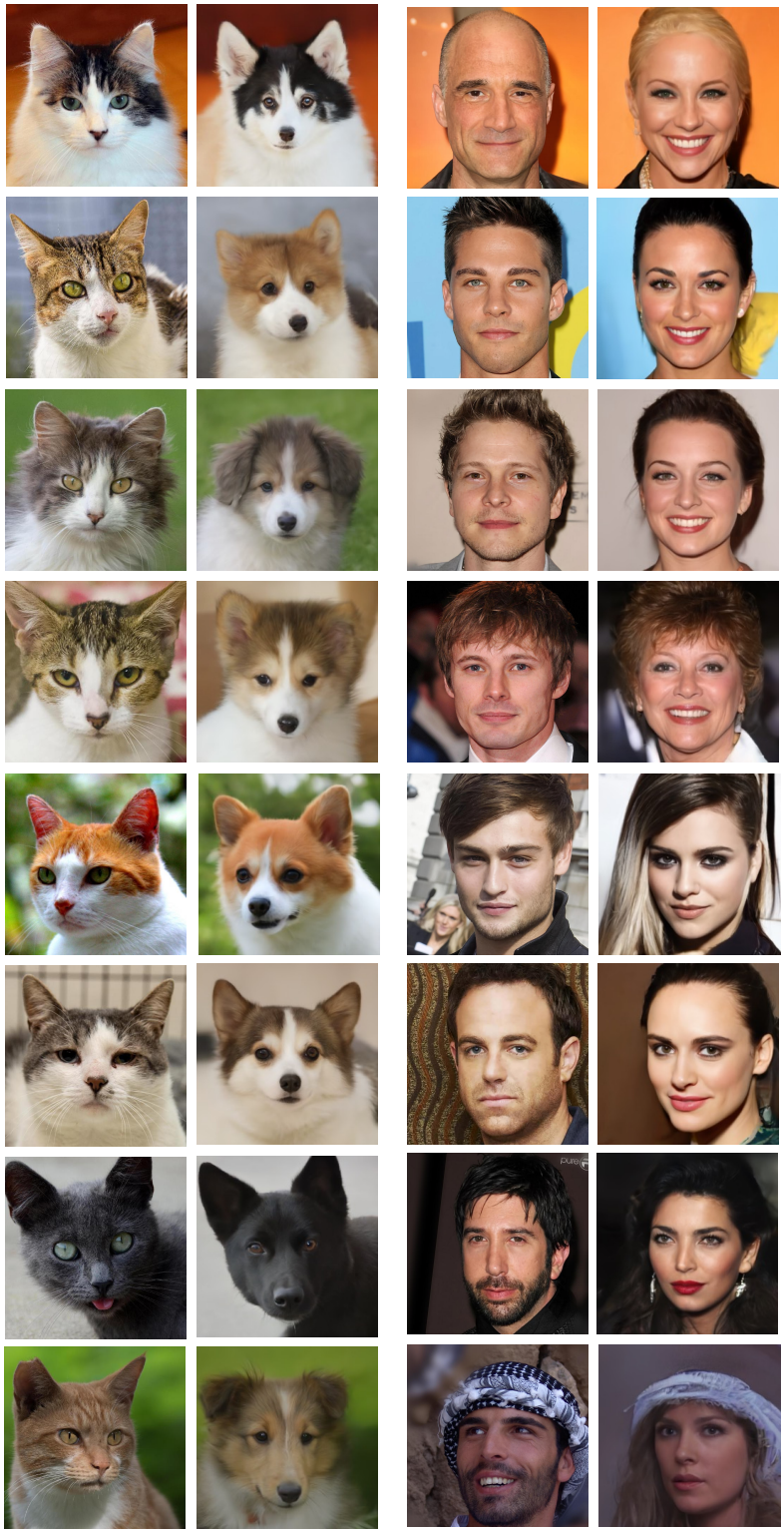

Figure 5. More random samples of our methods.