ILVR: Conditioning Method for Denoising Diffusion Probabilistic Models

Jooyoung Choi[1] Sungwon Kim[1] Yonghyun Jeong[3] Youngjune Gwon[3] Sungroh Yoon[1,2,*]
[1] Data Science and AI Laboratory, Seoul National University, Korea
[2] ASRI, INMC, and Interdisciplinary Program in AI, Seoul National University, Korea
[3] AI Research Team, Samsung SDS
{jy_choi, ksw0306, sryoon}@snu.ac.kr {yhyun.jeong, gyj.gwon}@samsung.com

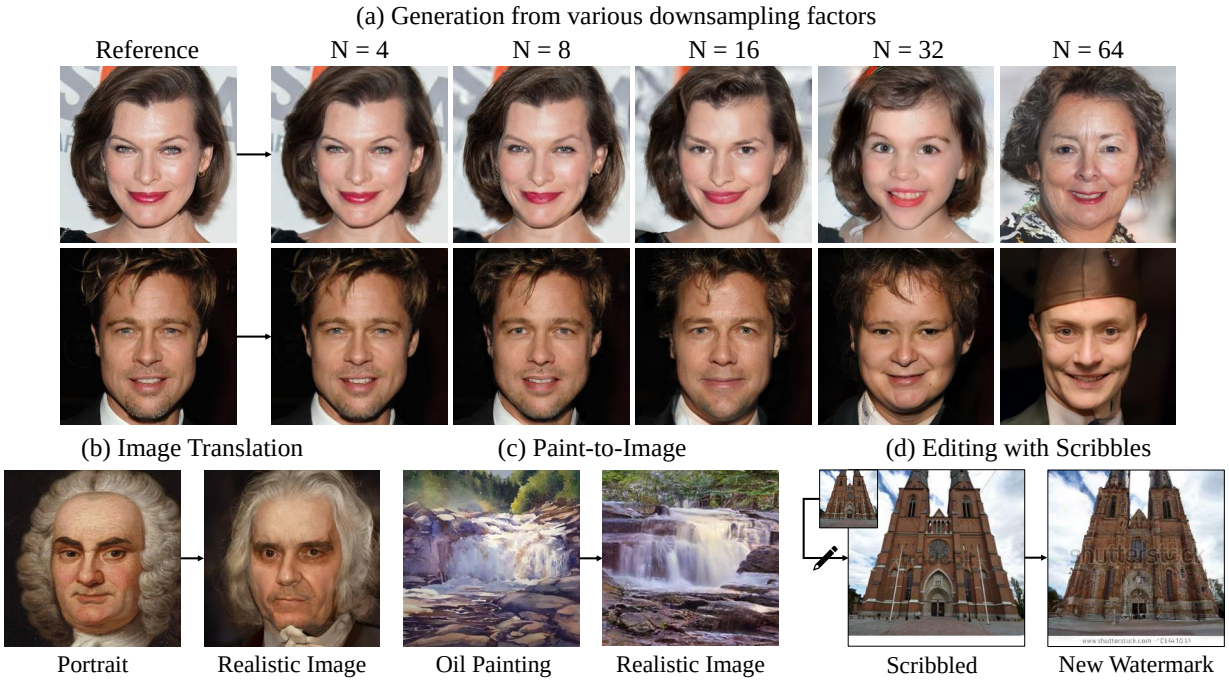

Figure 1: **Iterative Latent Variable Refinement for DDPM.** Our method of controlling Denoising Diffusion Probabilistic Model (DDPM) motivates various image generation tasks such as: (a) Generating from various downsampling factors; (b) Image translation; (c) Paint-to-image; and (d) Editing with scribbles.

Abstract

Denoising diffusion probabilistic models (DDPM) have shown remarkable performance in unconditional image generation. However, due to the stochasticity of the generative process in DDPM, it is challenging to generate images with the desired semantics. In this work, we propose Iterative Latent Variable Refinement (ILVR), a method to guide the generative process in DDPM to generate high-quality images based on a given reference image. Here, the refinement of the generative process in DDPM enables a single DDPM to sample images from various sets directed by the reference image. The proposed ILVR method generates high-quality images while controlling the generation. The controllability of our method allows adaptation of a single DDPM without any additional learning in various image generation tasks, such as generation from various downsampling factors, multi-domain image translation, paint-to-image, and editing with scribbles. Our code is available at: https://github.com/jychoi118/ilvr_adm.

*Correspondence to: Sungroh Yoon (sryoon@snu.ac.kr)

1. Introduction

Generative models, such as generative adversarial networks (GAN) [3, 10, 19], normalizing flows [21], and variational autoencoders [42], have shown remarkable quality in image generation, and have been applied to numerous purposes such as image-to-image translation [7, 11, 31, 32, 35, 47] and image editing [1, 12, 36].

There are mainly two approaches to control generative models to generate images as desired: one is by designing the conditional generative models for the desired purpose, and the other is by leveraging well-performed unconditional

generative models.

The first approach learns to control by providing the desired condition in training procedure and has shown remarkable performance on various tasks, such as segmentation mask conditioned generation [31, 59], style transfer [9, 50], and inpainting [23, 52]. The second approach utilizes high-quality generative models, such as StyleGAN [19, 20] or BigGAN [3]. Shen *et al*. [36] and Härkönen *et al*. [12] manipulate semantic attributes of images by analyzing latent space of pre-trained generative models, while Huh *et al*. [16] and Zhu *et al*. [57] perform image editing by projecting image into the latent space.

Denoising diffusion probabilistic models (DDPM) [14, 39], an iterative generative model, has shown comparable performance to the state-of-the-art models in unconditional image generation. DDPM learns to model the Markov transition from simple distribution to data distribution and generates diverse samples through sequential stochastic transitions. Samples obtained from the DDPM depend on the initial state of the simple distribution and each transition. However, it is challenging to control DDPM to generate images with desired semantics, since the stochasticity of transitions generates images with inconsistent high-level semantics, even from the same initial state.

In this work, we propose a learning-free method, iterative latent variable refinement (ILVR), to condition the generation process in well-performing unconditional DDPM. Each transition in the generation process is refined utilizing a given reference image. By matching each latent variable, ILVR ensures the given condition in each transition thus enables sampling from a conditional distribution. Thus, ILVR generates high-quality images sharing desired semantics.

We describe user controllability of our method, which enables control on semantic similarity of generated images to the refenence. Fig. 1(a) and Fig. 4 show samples sharing semantics ranging from coarse to fine information. Besides, reference images can be selected from unseen data domains. From these properties, we were motivated to leverage unconditional DDPM learned on single data domain to multi-domain image translation; a challenging task where existing works had to learn on multiple data domains. Furthermore, we extend our method to paint-to-image and editing with scribbles (Fig. 1(c) and (d)). We demonstrate that our ILVR enables leveraging a single unconditional DDPM model on these various tasks without any additional learning or models. Measuring Fréchet Inception Distance (FID) and Learned Perceptual Image Patch Similarity (LPIPS), we confirm that our generation method from various downsampling factors provides control over diversity while maintaining visual quality.

Our paper makes the following contributions:

- We propose ILVR, a method of refining each transition in the generative process by matching each latent variable with given reference image.
- We investigate several properties that allows user controllability on semantic similarity to the reference.
- We demonstrate that our ILVR enables leveraging unconditional DDPM in various image generation tasks including multi-domain image translation, paint-to-image, and editing with scribbles.

2. Background

Denoising diffusion probabilistic models (DDPM) [14, 39] is a class of generative models that show superior performance [14] in unconditional image generation. It learns a Markov Chain which gradually converts a simple distribution such as isotropic Gaussian, into a data distribution. Generative process learns the reverse of the DDPM's forward (diffusion) process, a fixed Markov Chain that gradually adds noise to data when sequentially sampling latent variables $x_1, ..., x_T$ of the same dimensionality. Here, each step in the forward process is a Gaussian translation.

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

where $\beta_1, ..., \beta_T$ is a fixed variance schedule rather than learned parameters [14]. Eq. 1 is a process finding x_t by adding a small Gaussian noise to the latent variable. Given clean data x_0, sampling of x_t is expressed in a closed form:

$$q(x_t|x_0) := N(x_t; \sqrt{\overline{\alpha}_t}x_0, (1-\overline{\alpha}_t)\mathbf{I}), \tag{2}$$

where $\alpha_t := 1 - \beta_t$ and $\overline{\alpha}_t := \prod_{s=1}^{t} \alpha_s$. Therefore, x_t can be expressed as a linear combination of x_0 and ϵ:

$$x_t = \sqrt{\overline{\alpha}_t}x_0 + \sqrt{1-\overline{\alpha}_t}\epsilon, \tag{3}$$

where $\epsilon \sim N(0, \mathbf{I})$ has the same dimensionality as data x_0 and latent variables $x_1, ..., x_T$.

Since the reverse of the forward process $q(x_{t-1}|x_t)$ is intractable, DDPM learns parameterized Gaussian transitions $p_\theta(x_{t-1}|x_t)$. The generative (or reverse) process has the same functional form [39] as the forward process, and it is expressed as a Gaussian transition with learned mean and fixed variance [14]:

$$p_\theta(x_{t-1}|x_t) = N(x_{t-1}; \mu_\theta(x_t, t), \sigma_t^2\mathbf{I}). \tag{4}$$

Further, by decomposing μ_θ into a linear combination of x_t and the noise approximator ϵ_θ, the generative process is expressed as:

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\overline{\alpha}_t}}\epsilon_\theta(x_t, t)) + \sigma_t\mathbf{z}, \tag{5}$$

where $\mathbf{z} \sim N(0, \mathbf{I})$, which suggests that each generation step is stochastic. Multiple stochastic process steps result in

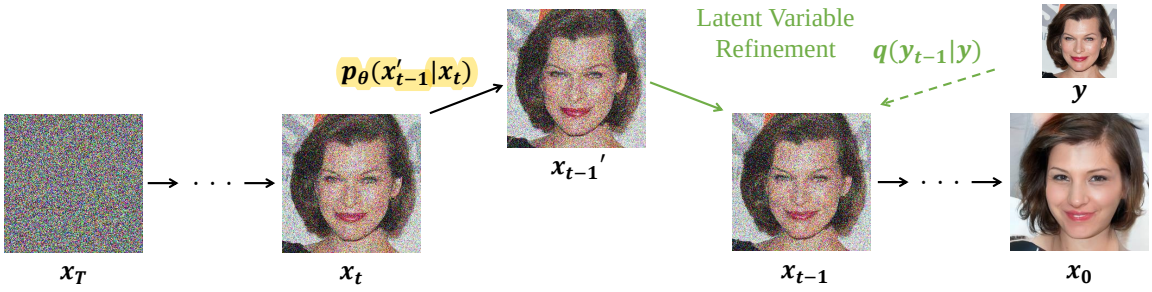

Figure 2: **Graphical model of Iterative Latent Variable Refinement.** From state x_t, we first sample unconditional proposal x_{t-1} according to Eq. 5. Then, we match latent variable with encoded condition y_{t-1} according to Eq. 8.

a difficulty in controlling the DDPM generative process. ϵ_θ represents a neural network with the same input and output dimensions and the noise predicted by the neural network ϵ_θ in each step is used for the denoising process in Eq. 5.

3. Method

Leveraging the capabilities of DDPM, we propose a method of controlling unconditional DDPM. We introduce our method, Iterative Latent Variable Refinement (ILVR), in Section. 3.1. Section. 3.2 investigates several properties of ILVR, which motivate control of two factors: downsampling factors and conditioning range.

3.1. Iterative Latent Variable Refinement

In this section, we introduce Iterative Latent Variable Refinement (ILVR), a method of conditioning the generative process of the unconditional DDPM model to generate images that share high-level semantics from given reference images. For this purpose, we sample images from the conditional distribution $p(x_0|c)$ with the condition c:

$$p_\theta(x_0|c) = \int p_\theta(x_{0:T}|c)dx_{1:T},$$
$$p_\theta(x_{0:T}|c) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t, c). \tag{6}$$

Each transition $p_\theta(x_{t-1}|x_t, c)$ of the generative process depends on the condition c. However, the unconditionally trained DDPM represents unconditional transition $p_\theta(x_{t-1}|x_t)$ of Eq. 4. Our ILVR provides condition c to unconditional transition $p_\theta(x_{t-1}|x_t)$ without additional learning or models. Specifically, we refine each unconditional transition with a downsampled reference image.

Let $\phi_N(\cdot)$ denote a linear low-pass filtering operation, a sequence of downsampling and upsampling by a factor of N, therefore maintaining dimensionality of the image. Given a reference image y, the condition c is to ensure the downsampled image $\phi_N(x_0)$ of the generated image x_0 to be equal to $\phi_N(y)$.

Algorithm 1 Iterative Latent Variable Refinement

Input: Reference image y
Output: Generated image x
$\phi_N(\cdot)$: low-pass filter with scale N
Sample $x_T \sim N(\mathbf{0}, \mathbf{I})$
for $t = T, ..., 1$ **do**
  $\mathbf{z} \sim N(\mathbf{0}, \mathbf{I})$
  $x'_{t-1} \sim p_\theta(x'_{t-1}|x_t)$ ▷ unconditional proposal
  $y_{t-1} \sim q(y_{t-1}|y)$ ▷ condition encoding
  $x_{t-1} \leftarrow \phi_N(y_{t-1}) + x'_{t-1} - \phi_N(x'_{t-1})$
end for
return x_0

Utilizing the forward process $q(x_t|x_0)$ of Eq. 3 and the linear property of ϕ_N, each Markov transition under the condition c is approximated as follows:

$$p_\theta(x_{t-1}|x_t,\ c) \approx p_\theta(x_{t-1}|x_t, \phi_N(x_{t-1}) = \phi_N(y_{t-1})), \tag{7}$$

where y_t can be sampled following Eq. 3. The condition c in each transition from x_t to x_{t-1} can be replaced with a local condition, wherein latent variable x_{t-1} and corrupted reference y_{t-1} share low-frequency contents. To ensure the local condition in each transition, we first use DDPM to compute the unconditional proposal distribution of x'_{t-1} from x_t. Then, since operation ϕ maintains dimensionality, we refine the proposal distribution by matching $\phi(x'_{t-1})$ of the proposal x'_{t-1} with that of y_{t-1} as follows:

$$x'_{t-1} \sim p_\theta(x'_{t-1}|x_t),$$
$$x_{t-1} = \phi(y_{t-1}) + (I - \phi)(x'_{t-1}). \tag{8}$$

By matching latent variables following Eq. 8, ILVR ensures local condition in Eq. 7, thus enables conditional generation with unconditional DDPM. Fig. 2 and Algorithm 1 illustrate our ILVR. Although we approximate the conditional transition with a simple modification of the unconditional proposal distribution, Fig. 1(a) and Fig. 4 show di-

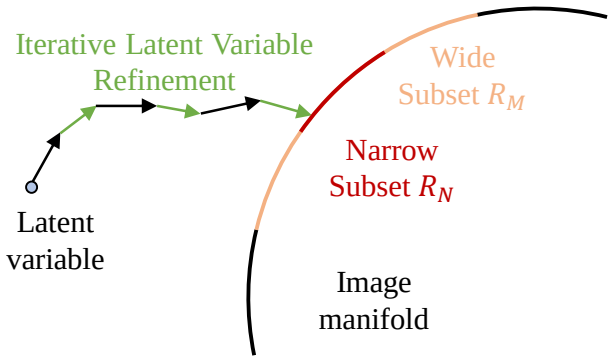

Figure 3: Guiding generation process toward reference directed subset, where $N \leq M$.

verse, high-quality samples sharing semantics of the references.

3.2. Reference selection and user controllability

Let μ be the set of images that an unconditional DDPM can generate. Our method enables sampling from a conditional distribution with a given reference image y. In other words, we sample images from a subset of μ, which is directed by the reference image.

To extend our method to various applications, we investigate 1) minimum requirement on reference image selection and 2) user controllability on reference directed subset, which defines semantic similarity to the reference. To provide an intuition for reference selection and control, we investigate several properties. Fig. 3 visualizes ILVR in each generation step to guide toward the subset directed by the reference.

We denote the directed subset as:

$$R_N(y) = \{x : \phi_N(x) = \phi_N(y)\}, \tag{9}$$

representing the set of images $x \in \mathbb{R}^{H \times H}$ which are equivalent to the downsampled reference image y.

We consider a range of conditioning steps by extending the above notation:

$$R_{N,\,(a,\,b)}(y) = \{x : \phi_N(x_t) = \phi_N(y_t),\, t \in [b,\, a]\}, \tag{10}$$

where $R_{N,\,(a,\,b)}(y)$ represents the distribution of images matching latent variables (line 9 of Alg. 1) in steps b to a. We will now discuss several properties on the reference selection and subset control.

Property 1. Reference image can be any image selected from the set:

$$Y = \{y : \phi_N(y) = \phi_N(x),\, x \in \mu\}, \tag{11}$$

the reference image only needs to match the low-resolution space of learned data distribution. Even reference images from unseen data domains are possible. Thus, we can select a reference from unseen data domains and perform multi-domain image translation, as demonstrated in Section. 4.2.

Property 2. Considering downsampling factors N and M where $N \leq M$,

$$R_N \subset R_M \subset \mu, \tag{12}$$

which suggests that higher factors correspond to broader image subsets.

As higher factor N enables sampling from broader set of images, sampled images are more diverse and exhibit lower semantic similarity to the reference. In Fig. 4, perceptual similarity to the reference image is controlled by the downsampling factors. Samples obtained from higher factor N share coarse features of the reference, while samples from lower N share also finer features. Note that since R_N is a subset of μ, our sampling method maintains the sample quality of unconditional DDPM.

Property 3. Limiting the range of conditioning steps enables sampling from a broader subset, while sampling from learned image distribution is still guaranteed.

$$R_N \subset R_{N,(T,k)} \subset \mu. \tag{13}$$

Fig. 5 shows the tendency of generated images when gradually limiting the range of conditioned steps. Compared to changing downsampling factors, changing conditioning range has a fine-grained influence on sample diversity.

4. Experiments and Applications

As discussed previously, ILVR generates high-quality images and allows control on semantic similarity to the reference. We first show qualitative results of controlled generation in Section. 4.1. Then we demonstrate ILVR on various image generation tasks in Sections 4.2, 4.3, and 4.4. Quantitative evaluations on the visual quality and diversity of ILVR are presented in Section. 4.5.

We trained the DDPM model on FFHQ [19], MetFaces [18], AFHQ [7], LSUN-Church [51], and Places365 datasets [56], to exemplify its applicability in various tasks. We used correctly implemented resizing library [37] for the operation ϕ_N. Reference face images are from the web, those unseen during training. See supplementary materials for details on implementation and evaluations.

4.1. Qualitative Results on User Controllability

Semantic similarity to the reference vary based on the downsampling factor N and the conditioning step range $[b,\, a]$. In Fig. 4, images are generated from the reference

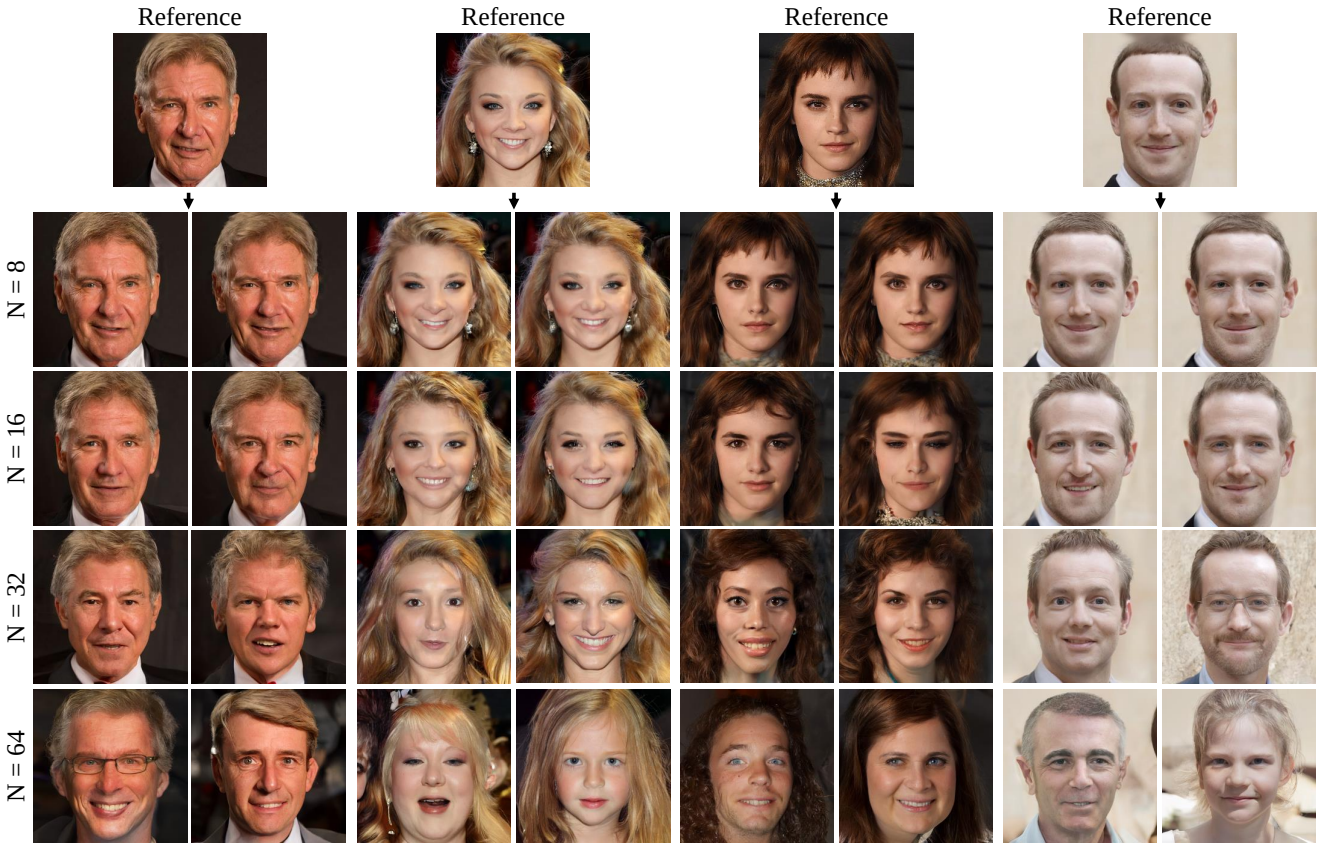

Figure 4: **Generating from various downsampling factors.** Samples obtained from high factor (N=64) are diverse, only bringing coarse information (color scheme) from reference images. Samples from middle factor (N=32) brought middle level semantics (facial features) while samples from low factors (N=16,8) are highly similar to the reference.

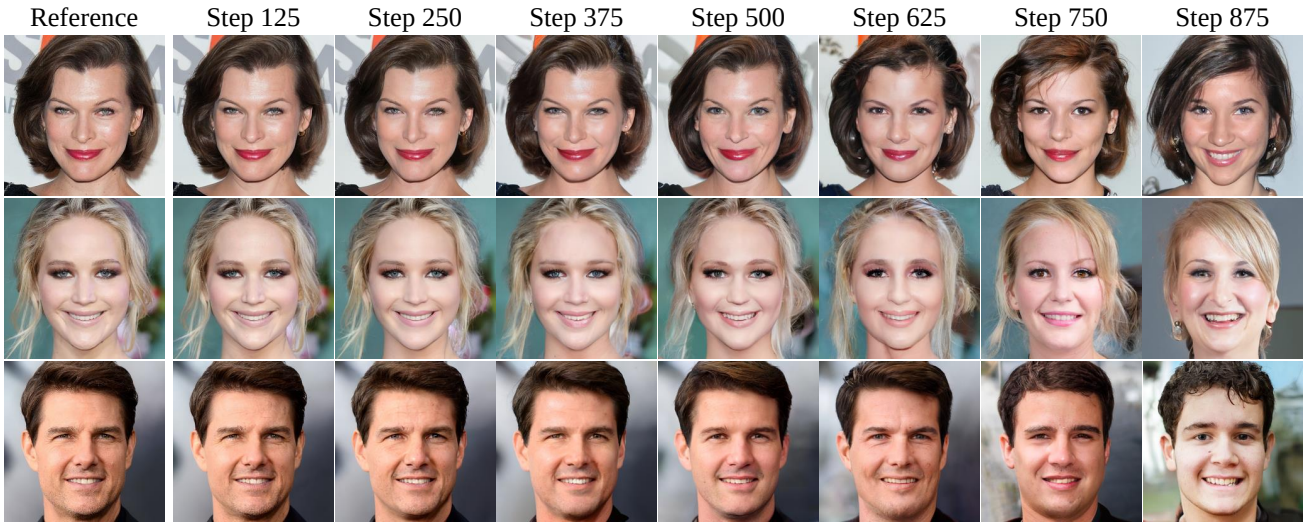

Figure 5: **Generating from various conditioning ranges.** Samples are generated with ILVR on steps from 1000 to 125, from 1000 to 250, and so on. Samples start to deviate from the reference images with range narrower than step 1000-500.

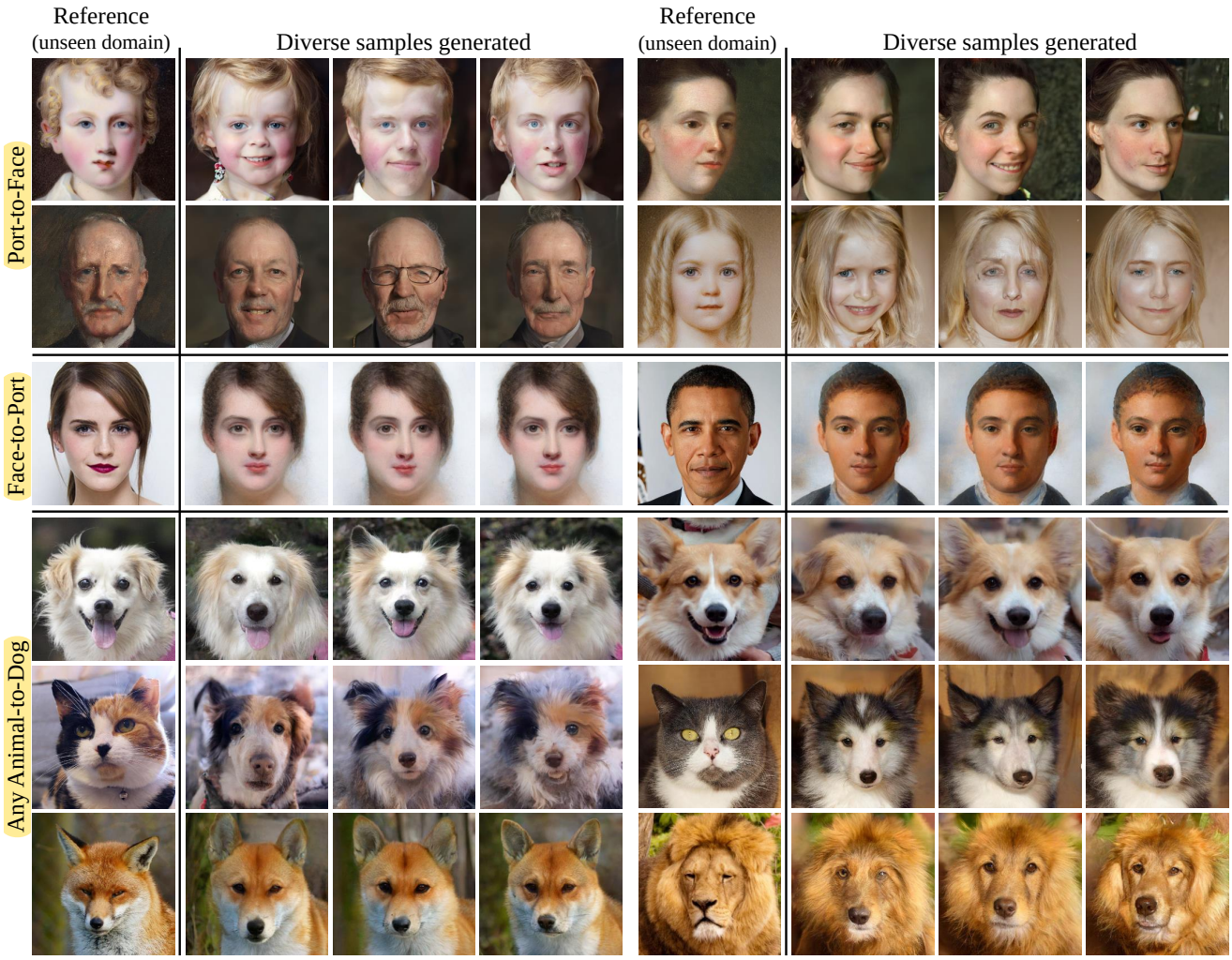

Figure 6: **Image translation from various source domains.** Row1-2: portrait to face. Row3: face to portrait. Row4-6: any animal (dog, cat, wildlife) to dog. Our method enables translation from any source domains, unseen in the training phase. Moreover, our method generate diverse samples.

image downsampled by various factors. As the factor N increase, samples are more diverse and perceptually less similar to the reference, as stated in Eq. 12. For example, samples obtained from N=8 differ with references in fine details (e.g., hair curls, eye color, earring) while samples from N=64 share only coarse features (e.g., color scheme) with the reference. This user controllability on similarity to the reference supports learning-free adaptation of a single pretrained model to various tasks, as described subsequently.

In addition to models we reproduced, we also utilize publicly available guided-diffusion [8], recent state-of-the-art DDPM. Fig. 9 shows samples generated with unconditional models trained on LSUN [51] datasets. Samples share either coarse (N=64) or fine (N=16) features from the references. Such results suggest that our method can be applied to any unconditional DDPMs without retraining.

Fig. 5 shows samples generated from a varying the range of conditioning steps. Here, a narrower range allows image sampling from a broader subset following Eq. 13, resulting in diverse images. Conditioning in less than 500 steps, facial features differ from the references. The downsampling factor and conditioning range provide user controllability, where the later has a finer control on sample diversity.

4.2. Multi-Domain Image Translation

Image-to-Image translation aims to learn the mapping between two visual domains. More specifically, generated images need to take the texture of the target domain while preserving the structure of the input images [30]. ILVR performs this task by matching the coarse information in reference images. We chose N=32 to preserve the coarse structure of the reference.

Figure 7: **Paint-to-Image.** Photo-realistic images generated from unnatural images (oil painting, water color, clip art)

The first two rows in Fig. 6 show samples generated with DDPM model trained on the FFHQ [19] dataset, which contains high-quality photos of human faces. Samples from portrait [18] references show successful translation into photo-realistic faces. We also generated portraits from photos, with DDPM trained on METFACES [18], the dataset of face portraits. Here, diverse samples are generated, however, some existing image translation models fail [17, 30] to produce stochastic samples.

Generally, image translation models [7, 17, 24, 58], including multi-domain translation models [7, 15], learn translation between different domains. Thus they can only translate from domains learned in the training phase. However, ILVR requires only a single model trained on the target domain. Therefore ILVR enables image translation from unseen source domains, with reference images from the low-resolution space of learned dataset as suggested in Eq. 11. Quantitative comparison to existing translation models is presented in the supplementary materials.

With a DDPM model trained on AFHQ-dog [7], we translated images of dogs, cats, and wildlife animals from the validation set. The fourth to sixth row of Fig. 6 show the results. DDPM model trained only on dog images translates unseen cat and wildlife images well into dog images.

4.3. Paint-to-Image

Paint-to-image is the task of transferring unnatural paintings into photo-realistic images. We validate our extension on this task using a model trained on the waterfall category from Places365 [56].

As shown in Fig 7, clip art, oil painting, and water-color are well translated into photo-realistic images. Paintings and photo-realistic images differ in detailed texture. We chose a factor of N=64 to preserve only the coarse aspect (e.g., color scheme) of the reference. From Eq. 11, we can infer that the given paintings share coarse features of the learned dataset.

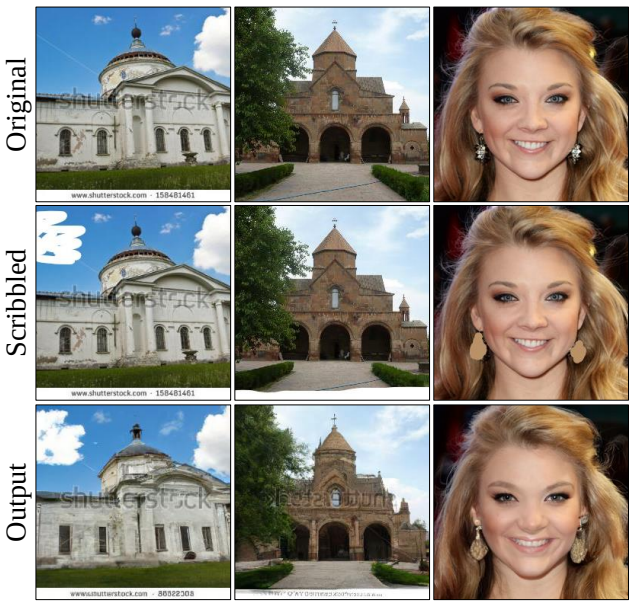

Figure 8: **Editing with scribbles.** Row1: cloud generated from white scribble at top-left corner. Row2: watermark generated from white scribble at the bottom. Row3: beige earring generated from scribble at top of silver earring.

4.4. Editing with Scribbles

We extend our method to application of performing editions with user scribbles, which was also presented in Image2StyleGAN++ [2]. We generated samples with DDPM trained on LSUN-Church [51] and FFHQ [19]. On reference images from the validation set, we added scribbles. Then, scribbled images are provided as references in factor N=8 on time steps from 1000 to 200, in order to both maintain details of original images and harmonize the scribbles. Interesting samples are shown in Fig. 8. In the second row, DDPM generated the "Shutterstock" watermark in the middle and the article number at the bottom. Since these pair of watermark and article number is common in the dataset, DDPM generated such features from a white scribble at the bottom. See supplementary for more samples.

4.5. Quantitative Evaluation

We evaluated the quality and diversity of our generated images with widely used FID [13] and LPIPS [55]. The FID score evaluates the visual quality and distance between real and generated image distributions. LPIPS measures the perceptual similarity between two images.

Table 1 reports FID scores measured from each downsampling factor N and unconditional generation with models trained on FFHQ [19] and METFACES [18] datasets. Scores (lower is better) are mostly comparable to the unconditional models, suggesting that our conditioning method does not harm the generation quality of unconditional

	FFHQ	METFACES
baseline	11.38	37.39
4	4.62	11.85
8	7.24	16.94
16	10.35	22.68
32	12.05	32.77

Table 1: FIDs of various downsampling factors and unconditional models. Models are trained on FFHQ and METFACES. Comparing to unconditional models (baseline), our conditioning maintains visual quality.

N	1	2	4	8	16	32
LPIPS	0.011	0.039	0.101	0.185	0.299	0.439

Table 2: LPIPS distances computed on various downsampling factors. Higher N results in higher diversity.

model. In addition, FID scores of lower downsampling factors are better, as generated images from lower factors align almost perfectly with reference images.

To evaluate the diversity among samples generated from the same reference, we generated 10 images for each reference image and calculated average pairwise (45 pairs) LPIPS distance, following StarGAN2 [7]. Table 2 shows that the higher the factor N, the higher the LPIPS, thus more diverse samples are generated as suggested in Eq. 12. In contrary, samples from lower N share more amount of contents from the references, therefore less diverse.

5. Related Work

5.1. Iterative generative models

Successful iterative generative models gradually add noise to the data and learn to reverse this process. Score-based models [40, 41] estimate a score (gradient of log-likelihood), and sample images with Langevin dynamics. A denoising score matching [45] is utilized to learn the scores in a scalable manner. DDPM [14, 39] learns to reverse the diffusion process that corrupts data, and utilizes the same functional form of the diffusion and reverse process. Ho *et al*. [14] show superior performance in image generation, by achieving exceptionally low FID. Diffusion models also show superior performance in other domains such as speech synthesis [5, 22] and point cloud generation [25]. Our conditioning method allows this powerful DDPM to be utilized for a variety of purposes.

5.2. Conditional generative models

Depending on the input type, such as class-label [3, 43, 54], segmentation mask [31, 47], feature from classifier [28, 38], and image [17, 26], various conditional generative models are available. The studies employing images as a condition began with Isola *et al*. [17], and extended to unsupervised [15, 58], few-shot [24], and multi-domain image translations [7]. Concurrent to our work, SR3 [34] trained conditional DDPM for super-resolution. These models show remarkable performances, however, only in the desired setting. In contrast, we demonstrate adaptation of a single unconditional model to various applications.

5.3. Leveraging unconditional models

Researches on leveraging pre-trained unconditional generators for various purposes, such as image editing [2, 36], style transfer [1], and super-resolution [11, 26] are being conducted. Specifically, by projecting given images into the latent vectors [1, 2, 4, 49, 57] and manipulating them [6, 11, 12, 26, 36], images are easily edited. Leveraging capability of the unconditional models, these works exhibit high-quality images. GAN [10] forms a cornerstone of these works. However, we utilized the iterative generative model, DDPM, which has not been explored in this context.

5.4. High-level semantics

Image semantics contained in CNN features [38], segmentation masks [31], and low-resolution images [26, 46] are actively used as conditions in generative models. From our derivation in Eq. 6, various kind of semantic conditions, such as features or segmentations, can provide high-level semantics. However, they require additional models (classifier or segmentation models). Since we are interested in controlling DDPM without any additional models, we provided semantics with a low-resolution image by utilizing iterative nature of DDPM.

6. Conclusion

We proposed a learning-free method of conditioning the generation process of unconditional DDPM. By refining each transition with given reference, we enable sampling from the space of plausible images. Further, downsampling factors and the conditioning range provide user controllability over this method. We demonstrated that a single unconditional DDPM can be leveraged to various tasks without any additional learning and models.

Acknowledgements: This work was supported by Samsung Research Funding & Incubation Center of Samsung Electronics under Project Number SRFC-IT1901-12, the National Research Foundation of Korea (NRF) grant funded by the Korea government (Ministry of Science and ICT)[2018R1A2B3001628], AIRS Company in Hyundai Motor and Kia through HMC/KIA-SNU AI Consortium Fund, and the BK21 FOUR program of the Education and Research Program for Future ICT Pioneers, Seoul National University in 2021.

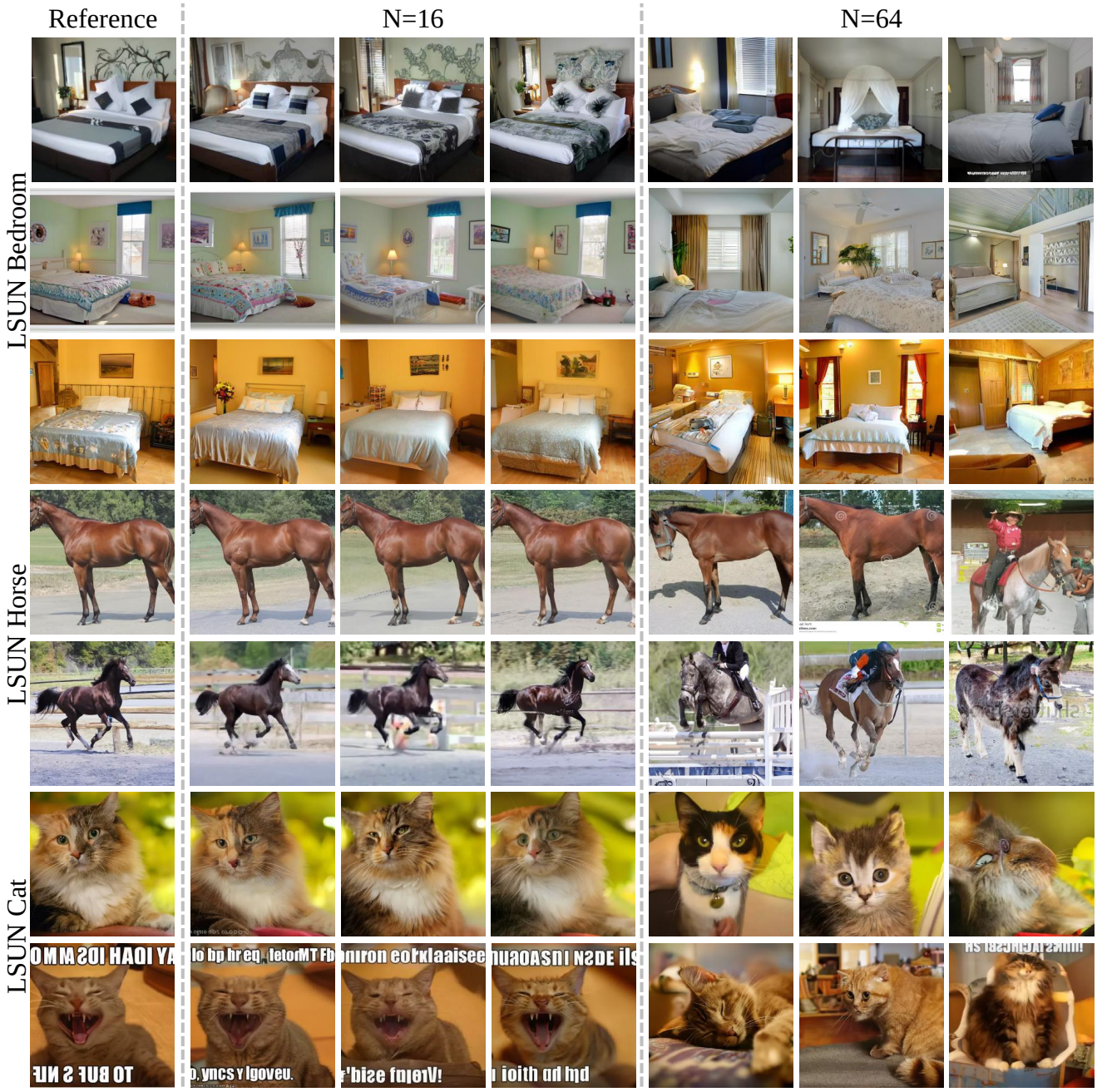

Figure 9: **ILVR samples with guided-diffusion [8].** Publicly available guided-diffusion trained on LSUN Bedroom, Horse, and Cat datasets. For efficiency, samples are generated with 250 steps using uniform stride, following IDDPM [29]. Conditions are given in factor N=16,64 from time step 250 to 100. Samples share either coarse or fine semantics from the references.

A. Derivation of approximation

In the main paper, we proposed iterative latent variable refinement (ILVR), where each transition of the generative process is matched with a given reference image. Condition in each transition was replaced with a local condition based on our approximation, as suggested in Eq.7 of the main text.

Before detailed derivations of the approximation (Eq.7), we review notations used in the main text. With pre-defined hyperparameter $\overline{\alpha}_t$, latent variable x_t can be sampled in closed-form: $x_t \sim q(x_t|x_0)$ (Eq.2). Trained model $\epsilon_\theta(x_t, t)$ predicts noise added in x_t, conditioned with time step t.

From the property of the forward process that latent variable x_t can be sampled from x_0 in closed-form, denoised data x_0 can be approximated with model prediction $\epsilon_\theta(x_t, t)$:

$$x_0 \approx f_\theta(x_t,\ t) = (x_t - \sqrt{1-\overline{\alpha}_t}\,\epsilon_\theta(x_t, t))/\sqrt{\overline{\alpha}_t} \quad \text{(A)}$$

Below is a derivation of Eq.7, where we approximated each conditioned Markov transition. We denote ϕ_N as ϕ and $f_\theta(x_t,\ t)$ as $f(x_t)$ for brevity. From Eq. A, each conditional Markov transition with given reference image y can be approximated as follows:

$$\begin{aligned}
& p_\theta(x_{t-1}|x_t, \phi(x_0) = \phi(y)) \\
\approx\ & p_\theta(x_{t-1}|x_t, \phi(f(x_{t-1})) = \phi(y)) \\
\approx\ & \mathbb{E}_{q(y_{t-1}|y)}[p_\theta(x_{t-1}|x_t, \phi(f(x_{t-1})) = \phi(f(y_{t-1})))].
\end{aligned}$$

With linear property of operation ϕ and Eq. A, we have

$$\begin{aligned}
& \mathbb{E}_{q(y_{t-1}|y)}[p_\theta(x_{t-1}|x_t, \phi(f(x_{t-1})) = \phi(f(y_{t-1})))] \\
=\ & \mathbb{E}_{q(y_{t-1}|y)}[p_\theta(x_{t-1}|x_t, \phi(x_{t-1}) = \phi(y_{t-1}), \\
& \phi(\epsilon_\theta(x_{t-1})) = \phi(\epsilon_\theta(y_{t-1})))] \\
\approx\ & \mathbb{E}_{q(y_{t-1}|y)}[p_\theta(x_{t-1}|x_t, \phi(x_{t-1}) = \phi(y_{t-1}))].
\end{aligned}$$

As shown in Eq.8 and Algorithm 1 of the main text, we first compute unconditional proposal x'_{t-1}, then refine it by ensuring $\phi(x_{t-1}) = \phi(y_{t-1})$. Therefore,

$$\begin{aligned}
& \mathbb{E}_{q(y_{t-1}|y)}[p_\theta(x_{t-1}|x_t, \phi(x_{t-1}) = \phi(y_{t-1}))] \\
=\ & \mathbb{E}_{q(y_{t-1}|y)}[p_\theta(\phi(y_{t-1}) + (I - \phi)(x'_{t-1}) \\
& |x_t, \phi(x_{t-1}) = \phi(y_{t-1}))] \\
=\ & \mathbb{E}_{q(y_{t-1}|y)}[p_\theta(x'_{t-1}|x_t)] \\
=\ & p_\theta(x'_{t-1}|x_t) \\
=\ & p_\theta(x_{t-1}|x_t, \phi(x_{t-1}) = \phi(y_{t-1})).
\end{aligned}$$

N	HR	Nearest	Bicubic	PULSE	ILVR
16 ↓	5.25	17.56	8.09	4.34	**4.06**
64 ↓	5.25	14.15	12.45	4.10	**4.02**

Table A: **NIQE comparison on generation quality.** Lower is better. Scores measured with generated images from reference images downscaled by a factor of 16 and 64. ILVR exhibits the highest perceptual quality.

CycleGAN [58]	MUNIT [15]	CUT [30]	Ours
85.9	104.4	**76.2**	**79.8**

Table B: **FID comparison on image translation.** FID measured with images translated from test set of AFHQ-dog. ILVR is comparable to a state-of-the-art model.

B. Additional evaluations

B.1. Generation quality

We provide additional qualitative and quantitative evaluations on the generation quality of ILVR. We evaluate images generated from low-resolution (LR) images downsampled by a factor of 16 and 64. Here, we compare ILVR with bicubic interpolation and PULSE [26], a super-resolution study that leverages pre-trained StyleGAN [19]. PULSE finds a latent vector that generates an image that downscales to the given LR image. We used publicly available StyleGAN2 [20] model[1] trained at 256×256. Combining loss function from PULSE and StyleGAN2, we search for latent vectors with a loss as follows:

$$\begin{aligned}
L_{total} =& ||\phi(G(z)) - \phi(y)||_2^2 \\
& + GEOCROSS(v_1, ..., v_{14}) + \alpha L_{noise}, \quad \text{(B)}
\end{aligned}$$

where each term refers to mean square error (MSE), geodesic cross loss [26], and noise regularization [20], respectively. MSE ensures generated image $G(z)$ and reference image y to match at low-resolution space. The geodesic cross loss ensures the latent vectors $v_1, ..., v_{14}$ remain in the learned latent space. Noise regularization L_{noise} discourages signal sneaking into the noise maps of StyleGAN2. We chose $\alpha = 5e^3$. Refer to StyleGAN2 literature for details on the noise regularization. We inherited initialization and learning rate schedule from StyleGAN2.

Fig. A presents additional qualitative results. ILVR and PULSE both show high-quality images generated from extremely downscaled images. Table. A shows NIQE [27] score, which is a no-reference metric that measures the perceptual quality of an image. ILVR shows higher perceptual quality, even better than the original 256^2 reference images (HR). We measured NIQE with reference images in Fig. B.

[1] https://github.com/rosinality/stylegan2-pytorch

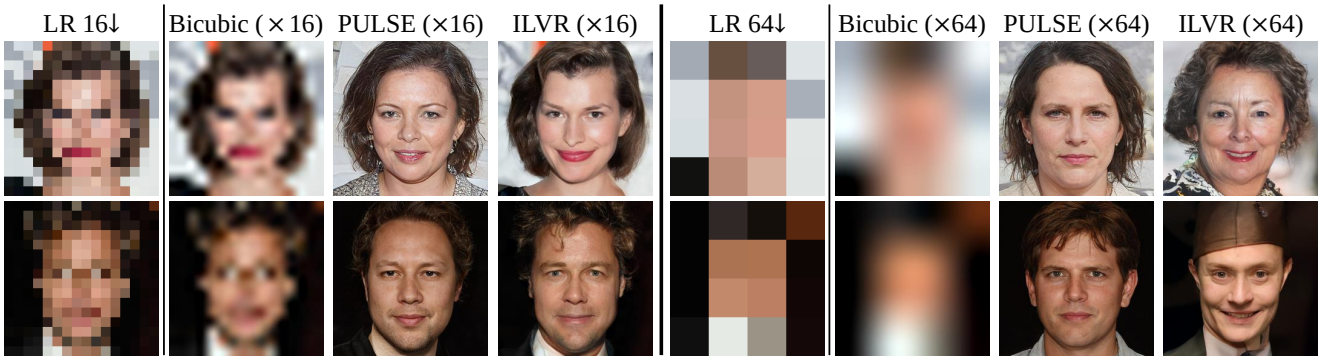

Figure A: **Qualitative comparison on generation quality.** Images generated from reference images downsampled by a factor of 16 and 64. From LR images, ILVR generates faces with detailed features.

Figure B: Images used for NIQE score.

B.2. Image translation

We compare Frechét inception distance (FID) [13] with image translation models on cat-to-dog (AFHQ [7] dataset) translation. Table. B shows the results. FID scores are calculated with the test set from AFHQ [7]. ILVR presents comparable performance to CUT [30], which is a state-of-the-art on cat-to-dog translation. Note that ILVR requires a model trained only on dog images, unlike the other models trained on both cat and dog images. We expect our result to broaden the applicability of DDPM to such image translation tasks.

B.3. Additional samples

Fig. 9 shows samples generated with publicly available guided-diffusion [8] trained on LSUN [51] datasets. We present additional editing with scribbles in Fig. C.

C. Implementation details

We trained unconditional DDPM with publicly available PyTorch implementation.[2]

C.1. Low-pass filters

We used bicubic downsampling and upsampling with correctly implemented function [37]. In Fig. D, we compare generated samples where the same noises were added through the generative process, only differing resizing kernels. Among kernels, images are almost identical, suggesting that our method is robust to kernel choice.

C.2. Datasets and training

Here we describe datasets and training details. For all datasets, we trained at 256^2 resolution with a batch size 8.

FFHQ [19] consists of 70,000 high-resolution face images. We trained a model for 1.2M steps.

METFACES [18] consists of 1,000 high-resolution portrait images. To avoid overfitting, we fine-tuned a model pre-trained on FFHQ [19], for 20k steps.

AFHQ [7] consists of 15,000 high-resolution animal face images, which are equally split into three categories: dog, cat, and wild. We trained on the train set of dog category, then used test sets of three categories as reference images to demonstrate multi-domain image translation.

Places365 [56] consists of 10M images of over 400 scene categories. We trained a model on a waterfall category, which consists of 5,000 images. We used this model to paint-to-image task.

LSUN Church [51] consists of 126,227 images of churches. We trained a model for 1M steps.

Paintings used for paint-to-image task are collected from the web.

C.3. Architecture

We trained the same neural network architecture as Ho *et al*. [14], which is U-Net [33] based on Wide ResNet [53]. Details include group normalization [48], self-attention blocks at 16×16 resolution, sinusoidal positional embedding [44], and a fixed linear variance schedule $\beta_1, ..., \beta_T$.

C.4. Evaluation

In Table 1 of the main text, we calculated FID with 50k real and 50k generated images using code[3] of PyTorch framework.

[2] https://github.com/rosinality/denoising-diffusion-pytorch

[3] https://github.com/mseitzer/pytorch-fid

Figure C: **Additional editing with scribbles.** Faces generated with our reproduced model trained on FFHQ [19]. Bedrooms generated with publicly available model [8] trained on LSUN Bedroom [51].

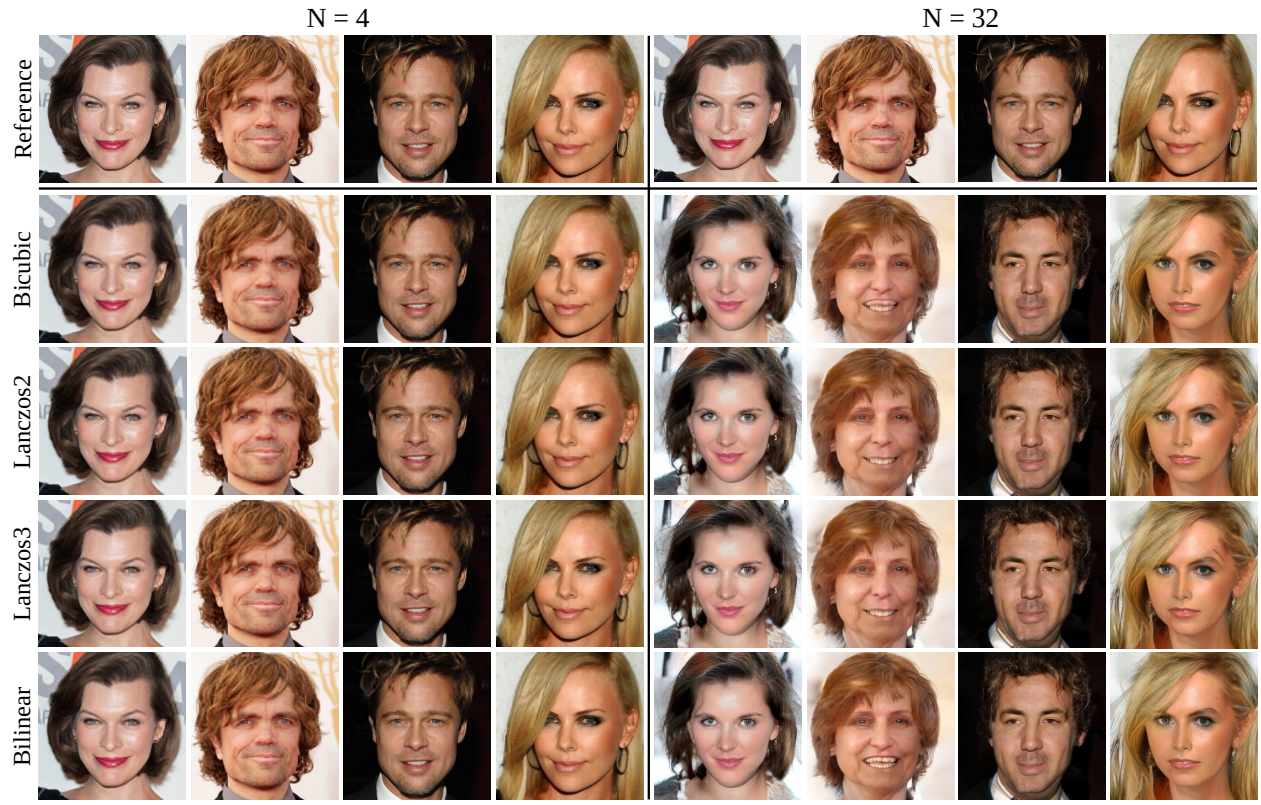

Figure D: **Ablation on low-pass filters.** First column set: samples from downsampling factor N=4; Second column set: samples from downsampling factor N=32. Samples are generated with bicubic, lanczos2, lanczos3, bilinear interpolation for downsampling and upsampling. There is only a minor difference among filters, such as the exact position of teeth and hair.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *ICCV*, 2019. 1, 8

[2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan++: How to edit the embedded images? In *CVPR*, 2020. 7, 8

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. 1, 2, 8

[4] Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Neural photo editing with introspective adversarial networks. *arXiv preprint arXiv:1609.07093*, 2016. 8

[5] Nanxin Chen, Yu Zhang, Heiga Zen, Ron J Weiss, Mohammad Norouzi, and William Chan. Wavegrad: Estimating gradients for waveform generation. *arXiv preprint arXiv:2009.00713*, 2020. 8

[6] Jooyoung Choi, Jungbeom Lee, Yonghyun Jeong, and Sungroh Yoon. Toward spatially unbiased generative models. *arXiv preprint arXiv:2108.01285*, 2021. 8

[7] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *CVPR*, 2020. 1, 4, 7, 8, 11

[8] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *arXiv preprint arXiv:2105.05233*, 2021. 6, 9, 11, 12

[9] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *CVPR*, 2016. 2

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 1, 8

[11] Jinjin Gu, Yujun Shen, and Bolei Zhou. Image processing using multi-code gan prior. In *CVPR*, 2020. 1, 8

[12] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable gan controls. *arXiv preprint arXiv:2004.02546*, 2020. 1, 2, 8

[13] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017. 7, 11

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020. 2, 8, 11

[15] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 172–189, 2018. 7, 8, 10

[16] Minyoung Huh, Richard Zhang, Jun-Yan Zhu, Sylvain Paris, and Aaron Hertzmann. Transforming and projecting images into class-conditional generative networks. *arXiv preprint arXiv:2005.01703*, 2020. 2

[17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 7, 8

[18] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *NeurIPS*, 2020. 4, 7, 11

[19] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 1, 2, 4, 7, 10, 11, 12

[20] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *CVPR*, 2020. 2, 10

[21] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *NeurIPS*, 2018. 1

[22] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761*, 2020. 8

[23] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *ECCV*, 2018. 2

[24] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. In *ICCV*, 2019. 7, 8

[25] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. *arXiv preprint arXiv:2103.01458*, 2021. 8

[26] Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In *CVPR*, 2020. 8, 10

[27] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal processing letters*, 20(3):209–212, 2012. 10

[28] Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *CVPR*, 2017. 8

[29] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. *arXiv preprint arXiv:2102.09672*, 2021. 9

[30] Taesung Park, Alexei A Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for unpaired image-to-image translation. In *ECCV*, 2020. 6, 7, 10, 11

[31] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, 2019. 1, 2, 8

[32] Taesung Park, Jun-Yan Zhu, Oliver Wang, Jingwan Lu, Eli Shechtman, Alexei A Efros, and Richard Zhang. Swapping autoencoder for deep image manipulation. *arXiv preprint arXiv:2007.00653*, 2020. 1

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 2015. 11

[34] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *arXiv preprint arXiv:2104.07636*, 2021. 8

[35] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. Singan: Learning a generative model from a single natural image. In *ICCV*, 2019. 1

[36] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *CVPR*, 2020. 1, 2, 8

[37] Assaf Shocher. Resizeright. https://github.com/

assafshocher/ResizeRight, 2018. 4, 11

[38] Assaf Shocher, Yossi Gandelsman, Inbar Mosseri, Michal Yarom, Michal Irani, William T Freeman, and Tali Dekel. Semantic pyramid for image generation. In *CVPR*, 2020. 8

[39] Jascha Sohl-Dickstein, Eric A Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *arXiv preprint arXiv:1503.03585*, 2015. 2, 8

[40] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, 2019. 8

[41] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *arXiv preprint arXiv:2006.09011*, 2020. 8

[42] Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. *NeurIPS*, 2020. 1

[43] Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. In *NeurIPS*, 2016. 8

[44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017. 11

[45] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 2011. 8

[46] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P Xing. High-frequency component helps explain the generalization of convolutional neural networks. In *CVPR*, 2020. 8

[47] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018. 1, 8

[48] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 11

[49] Raymond A Yeh, Chen Chen, Teck Yian Lim, Alexander G Schwing, Mark Hasegawa-Johnson, and Minh N Do. Semantic image inpainting with deep generative models. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5485–5493, 2017. 8

[50] Jaejun Yoo, Youngjung Uh, Sanghyuk Chun, Byeongkyu Kang, and Jung-Woo Ha. Photorealistic style transfer via wavelet transforms. In *ICCV*, 2019. 2

[51] 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. 4, 6, 7, 11, 12

[52] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *ICCV*, 2019. 2

[53] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016. 11

[54] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *ICML*, 2019. 8

[55] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 7

[56] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2017. 4, 7, 11

[57] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain gan inversion for real image editing. *arXiv preprint arXiv:2004.00049*, 2020. 2, 8

[58] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 7, 8, 10

[59] Peihao Zhu, Rameen Abdal, Yipeng Qin, and Peter Wonka. Sean: Image synthesis with semantic region-adaptive normalization. In *CVPR*, 2020. 2