EFFICIENT AND SCALABLE GENERATIVE MODEL CONTROL FOR HIGH-QUALITY MULTIMODAL SYNTHESIS

by
Kangfu Mei

A dissertation submitted to The Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy

Baltimore, Maryland
01 2025

Abstract

Generative models, such as diffusion models and generative adversarial networks, have recently transformed foundational vision tasks, including generating images from noise. However, challenges remain in designing generative models that are best suited for real-world applications and in improving their generalization capabilities across downstream tasks. This thesis addresses these challenges through comprehensive theoretical analyses and empirical experiments, with a focus on practical visual perception rectification tasks.

First, the thesis investigates the scaling properties of latent diffusion models, widely used in text-to-image generation and its downstream applications. It introduces inference scaling laws that reveal the surprising superiority of small models over large models when leveraging increased inference compute. Next, it presents a novel scalable video diffusion model capable of generating continuous scenes from pure noise, extending generative capabilities from static images to dynamic, expressive videos. To further reduce the complexity of designing modality-specific models, the thesis proposes a versatile field diffusion model that can seamlessly handle various modalities, including image, video, 3D, and game environments. Additionally, the thesis introduces an efficient diffusion distillation technique that achieves comparable visual quality while reducing computational cost by 99%, significantly enhancing the sampling efficiency of generative models.

Building on these advancements, the thesis applies realism priors derived from genera-

tive models to three real-world image processing tasks: turbulence removal, shadow removal, and single-image super-resolution. These approaches consistently achieve state-of-the-art performance, surpassing traditional regression-based methods and producing results with enhanced realism. Finally, the thesis shows the applications of rectification on high-level vision tasks, including denoised image segmentation and super-resolved image re-id.

Keywords: Diffusion Models, GAN Inversion, Scaling Laws, Diffusion Distillation, Multi-modal Vision-Language Model, Video Generation, Image Processing

Primary reader and thesis advisor

Dr. Vishal M. Patel
Professor
Department of Electrical and Computer Engineering
Johns Hopkins University, Baltimore MD

Secondary readers

Dr. Rama Chellappa
Professor
Department of Electrical and Computer Engineering
Johns Hopkins University, Baltimore, MD

Dr. Alan Yuille
Professor
Department of Computer Science
Johns Hopkins University, Baltimore, MD

This thesis is dedicated to my grandfather who told me I should go into doctor degrees. This wasn't what we were expecting, but it's what worked!

Acknowledgement

First and foremost, I extend my deepest gratitude to my PhD advisor, Prof. Vishal M. Patel, for graciously accepting me as his PhD student at Johns Hopkins University. His unwavering inspiration and encouragement have been pivotal in shaping my academic journey and nurturing my aspirations.

I am equally grateful to my committee members, Prof. Rama Chellappa and Prof. Alan Yuille, whose extraordinary contributions to the field have profoundly influenced and guided my own research endeavors.

I also wish to extend my sincere thanks to Mauricio Delbracio, Hossein Talebi, and Peyman Milanfar for hosting me at Google Research in Mountain View during 2023 and 2024. Collaborating on the paper exploring diffusion distillation, diffusion scaling properties, and multimodal super-resolution was a true privilege.

Additionally, my heartfelt appreciation goes to my family, collaborators, labmates, and friends, whose unwavering support has made this journey possible.

Lastly, I want to acknowledge my younger self for choosing to embark on this challenging yet rewarding life experience.

Table of Contents

List of Tables

List of Figures

Chapter 1
Introduction and Background

This chapter provides an introduction to generative models, focusing on the background and principles of diffusion models as a key representative of this family.

1.1 Generative Models

Generative models learn the probability distribution of natural data, such as a set of dog images. By accurately representing this distribution, they can generate new, realistic dog images that resemble the training data but are entirely novel. Specifically, we denote the observed images as $\mathbf{x}$, drawn from the natural image distribution $p^*(\mathbf{x})$. A generative model aims to approximate $p^*(\mathbf{x})$ using a network with parameters θ as

$$\mathbf{x} \sim p_\theta(\mathbf{x}). \tag{1.1}$$

By optimizing the parameters θ to minimize the difference between the model distribution $p_\theta(\mathbf{x})$ and the true data distribution $p^*(\mathbf{x})$, the generative model can accurately capture the characteristics of natural images.

1.2 Diffusion Models

Diffusion Models. A diffusion model [1, 2] has latent variables $\{\mathbf{z}_t | t \in [0, T]\}$ specified by a noise schedule comprising differentiable functions $\{\alpha_t, \sigma_t\}$ with $\sigma_t^2 =$

$1 - \alpha_t^2$. The clean data $\mathbf{x} \sim p_{\text{data}}$ is progressively perturbed in a (forward) Gaussian process as in the following Markovian structure:

$$q(\mathbf{z}_t|\mathbf{x}) = \mathcal{N}(\mathbf{z}_t; \alpha_t \mathbf{x}, \sigma_t^2 \mathbf{I}), \tag{1.2}$$

$$q(\mathbf{z}_t|\mathbf{z}_s) = \mathcal{N}(\mathbf{z}_t; \alpha_{t|s} \mathbf{z}_s, \sigma_{t|s}^2 \mathbf{I}), \tag{1.3}$$

where $0 \leq s < t \leq 1$ and $\alpha_{t|s}^2 = \alpha_t / \alpha_s$. Here the latent $\mathbf{z}_t$ is sampled from the combination of the clean data and random noise by using the reparameterization trick [3], which has $\mathbf{z}_t = \alpha_t \mathbf{x} + \sigma_t \epsilon$.

Deterministic sampling. The aforementioned diffusion process that starts from $\mathbf{z}_0 \sim p_{\text{data}}(\mathbf{x})$ and ends at $\mathbf{z}_T \sim \mathcal{N}(0, \mathbf{I})$ can be modeled as the solution of an stochastic differential equation (SDE) [1]. The SDE is formed by a vector-value function $f(\cdot, \cdot) : \mathbb{R}^d \to \mathbb{R}^d$, a scalar function $g(\cdot) : \mathbb{R} \to \mathbb{R}$, and the standard Wiener process $\mathbf{w}$ as:

$$\mathrm{d}\mathbf{z}_t = f(\mathbf{z}_t, t)\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}. \tag{1.4}$$

The overall idea is that the reverse-time SDE that runs backwards in time, can generate samples of p_{data} from the prior distribution $\mathcal{N}(0, \mathbf{I})$. This reverse SDE is given by

$$\mathrm{d}\mathbf{z}_t = [f(\mathbf{z}_t, t) - g(t)^2 \nabla_{\mathbf{z}} \log p_t(\mathbf{z}_t)]\mathrm{d}t + g(t)\mathrm{d}\bar{\mathbf{w}}, \tag{1.5}$$

where the $\bar{\mathbf{w}}$ is a also standard Wiener process in reversed time, and $\nabla_{\mathbf{z}} \log p_t(\mathbf{z}_t)$ is the score of the marginal distribution at time t. The score function can be estimated by training a score-based model $s_\theta(\mathbf{z}_t, t) \approx \nabla_z \log p_t(\mathbf{z}_t)$ with score-matching [4] or a

denoising network $\hat{\mathbf{x}}_\theta(\mathbf{z}_t, t)$ [2]:

$$s_\theta(\mathbf{z}_t, t) := (\alpha_t \hat{\mathbf{x}}_\theta(\mathbf{z}_t, t) - \mathbf{z}_t)/\sigma_t^2. \tag{1.6}$$

Such backward SDE satisfies a special ordinary differential equation (ODE) that allows deterministic sampling given $\mathbf{z}_T \sim \mathcal{N}(0, \mathbf{I})$. This is known as the *probability flow* (PF) ODE [1] and is given by

$$\mathrm{d}\mathbf{z}_t = [f(\mathbf{z}_t, t) - \frac{1}{2} g^2(t) s_\theta(\mathbf{z}_t, t)]\mathrm{d}t, \tag{1.7}$$

where $f(\mathbf{z}_t, t) = \frac{\mathrm{d}\log\alpha_t}{\mathrm{d}t}\mathbf{z}_t$, $g^2(t) = \frac{\mathrm{d}\sigma_t^2}{\mathrm{d}t} - 2\frac{\mathrm{d}\log\alpha_t}{\mathrm{d}t}\sigma_t^2$ with respect to $\{\alpha_t, \sigma_t\}$ and t according to [5]. This ODE can be solved numerically with diffusion samplers like DDIM [6], where starting from $\hat{\mathbf{z}}_T \sim \mathcal{N}(0, \mathbf{I})$, we update for $s = t - \Delta t$:

$$\hat{\mathbf{z}}_s := \alpha_s \hat{\mathbf{x}}_\theta(\hat{\mathbf{z}}_t, t) + \sigma_s(\hat{\mathbf{z}}_t - \alpha_t \hat{\mathbf{x}}_\theta(\hat{\mathbf{z}}_t, t))/\sigma_t, \tag{1.8}$$

till we reach $\hat{\mathbf{z}}_0$.

Diffusion models parametrizations. Leaving aside the aforementioned way of parametrizing diffusion models with a denoising network (signal prediction) or a score model (noise prediction equation 1.6), in this work, we adopt a parameterization that mixes both the score (or noise) and the signal prediction. Existing methods include either predicting the noise $\hat{\epsilon}_\theta(\mathbf{x}_t, t)$ and the signal $\hat{\mathbf{x}}_\theta(\mathbf{z}_t, t)$ separately using a single network [7], or predicting a combination of noise and signal by expressing them in a new term, like the velocity model $\hat{\mathbf{v}}_\theta(\mathbf{z}_t, t) \approx \alpha_t \epsilon - \sigma_t \mathbf{x}$ [8]. Note that one can derive

an estimation of the signal and the noise from the velocity one,

$$\hat{\mathbf{x}} = \alpha_t \mathbf{z}_t - \sigma_t \hat{\mathbf{v}}_\theta(\mathbf{z}_t, t), \text{ and } \hat{\epsilon} = \alpha_t \hat{\mathbf{v}}_\theta(\mathbf{z}_t, t) + \sigma_t \mathbf{z}_t. \tag{1.9}$$

Similarly, DDIM update rule (equation 1.8) can be rewritten in terms of the velocity parametrization:

$$\hat{\mathbf{z}}_s := \alpha_s(\alpha_t \hat{\mathbf{z}}_t - \sigma_t \hat{\mathbf{v}}_\theta(\hat{\mathbf{z}}_t, t)) + \sigma_s(\alpha_t \hat{\mathbf{v}}_\theta(\hat{\mathbf{z}}_t, t) + \sigma_t \hat{\mathbf{z}}_t). \tag{1.10}$$

Chapter 2

Scaling Properties of Latent Diffusion Models

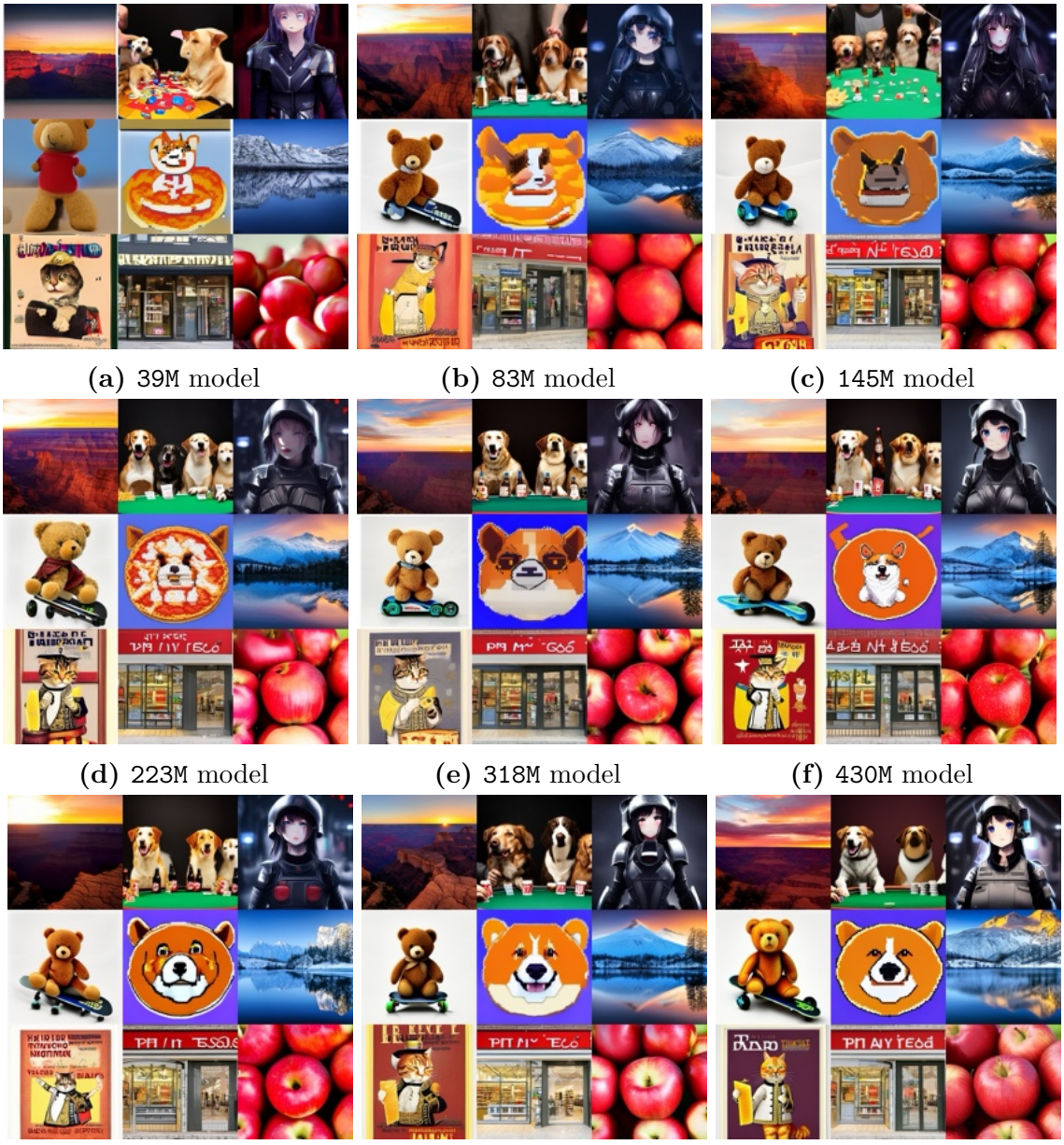
2.1 Introduction

Latent diffusion models (LDMs) [9], and diffusion models in general, trained on large-scale, high-quality data [10, 11] have emerged as a powerful and robust framework for generating impressive results in a variety of tasks, including image synthesis and editing [9, 12–15], video creation [16–19], audio production [20], and 3D synthesis [21, 22]. Despite their versatility, the major barrier against wide deployment in real-world applications [23, 24] comes from their low *sampling efficiency*. The essence of this challenge lies in the inherent reliance of LDMs on multi-step sampling [1, 2] to produce high-quality outputs, where the total cost of sampling is the product of sampling steps and the cost of each step. Specifically, the go-to approach involves using the 50-step DDIM sampling [6, 9], a process that, despite ensuring output quality, still requires a relatively long latency for completion on modern mobile devices with post-quantization. In contrast to single shot generative models (e.g., generative-adversarial networks (GANs) [25]) which bypass the need for iterative refinement [25, 26], the operational latency of LDMs calls for a pressing need for efficiency optimization to further facilitate their practical applications.

Recent advancements in this field [24, 27–31] have primarily focused on developing faster network architectures with comparable model size to reduce the inference time per step, along with innovations in improving sampling algorithms that allow

for using less sampling steps [6, 32–36]. Further progress has been made through diffusion-distillation techniques [8, 37–41], which simplifies the process by learning multi-step sampling results in a single forward pass, and then broadcasts this single-step prediction multiple times. These distillation techniques leverage the redundant learning capability in LDMs, enabling the distilled models to assimilate additional distillation knowledge. Despite these efforts being made to improve diffusion models, the sampling efficiency of smaller, less redundant models has not received adequate attention. A significant barrier to this area of research is the scarcity of available modern accelerator clusters [42], as training high-quality text-to-image (T2I) LDMs from scratch is both time-consuming and expensive—often requiring several weeks and hundreds of thousands of dollars.

In this chapter, we empirically investigate the scaling properties of LDMs, with a particular focus on understanding how their scaling properties impact the sampling efficiency across various model sizes. We trained a suite of 12 text-to-image LDMs from scratch, ranging from 39 million to 5 billion parameters, under a constrained budget. Example results are depicted in Fig. 2.1. All models were trained on TPUv5 using internal data sources with about 600 million aesthetically-filtered text-to-image pairs. Our study reveals that there exist a scaling trend within LDMs, notably that smaller models may have the capability to surpass larger models under an equivalent sampling budget. Furthermore, we investigate how the size of pre-trained text-to-image LDMs affects their sampling efficiency across diverse downstream tasks, such as real-world super-resolution [43, 44] and subject-driven text-to-image synthesis (i.e., Dreambooth) [45].



(a) 39M model **(b)** 83M model **(c)** 145M model

(d) 223M model **(e)** 318M model **(f)** 430M model

(g) 558M model **(h)** 704M model **(i)** 2B model

Figure 2.1: Text-to-image results from our scaled LDMs (39M - 2B), highlighting the improvement in visual quality with increased model size (note: 39M model is the exception). All images generated using 50-step DDIM sampling and CFG rate of 7.5.

2.1.1 Summary

Our key findings for scaling latent diffusion models in text-to-image generation and various downstream tasks are as follows:

Pretraining performance scales with training compute. We demonstrate a clear link between compute resources and LDM performance by scaling models from 39 million to 5 billion parameters. This suggests potential for further improvement with increased scaling. See Section 2.3.1 for details.

Downstream performance scales with pretraining. We demonstrate a strong correlation between pretraining performance and success in downstream tasks. Smaller models, even with extra training, cannot fully bridge the gap created by the pretraining quality of larger models. This is explored in detail in Section 2.3.2.

Smaller models sample more efficient. Smaller models initially outperform larger models in image quality for a given sampling budget, but larger models surpass them in detail generation when computational constraints are relaxed. This is further elaborated in Section 2.3.3.

Sampler does not change the scaling efficiency. Smaller models consistently demonstrate superior sampling efficiency, regardless of the diffusion sampler used. This holds true for deterministic DDIM [6], stochastic DDPM [2], and higher-order DPM-Solver++ [46]. For more details, see Section 2.3.3.

Smaller models sample more efficient on the downstream tasks with fewer steps. The advantage of smaller models in terms of sampling efficiency extends to the

downstream tasks when using less than 20 sampling steps. This is further elaborated in Section 2.3.4.

Diffusion distillation does not change scaling trends. Even with diffusion distillation, smaller models maintain competitive performance against larger distilled models when sampling budgets are constrained. This suggests distillation does not fundamentally alter scaling trends. See Section 2.3.5 for in-depth analysis.

2.2 Related Work

Scaling laws. Recent Large Language Models (LLMs) including GPT [47], PaLM [48], and LLaMa [49] have dominated language generative modeling tasks. The foundational works [47, 50, 51] for investigating their scaling behavior have shown the capability of predicting the performance from the model size. They also investigated the factors that affect the scaling properties of language models, including training compute, dataset size and quality, learning rate schedule, etc. Those experimental clues have effectively guided the later language model development, which have led to the emergence of several parameter-efficient LLMs [49, 51–53]. However, scaling generative text-to-image models are relatively unexplored, and existing efforts have only investigated the scaling properties on small datasets or small models, like scaling UNet [54] to 270 million parameters and DiT [29] on ImageNet (14 million), or less-efficient autoregressive models [55]. Different from these attempts, our work investigates the scaling properties by scaling down the efficient and capable diffusion models, *i.e.* LDMs [9], on internal data sources that have about 600 million aesthetics-filtered text-to-image pairs for featuring the sampling efficiency of scaled LDMs. We also scale

LDMs on various scenarios such as finetuning LDMs on downstream tasks [45, 56] and distilling LDMs [41] for faster sampling to demonstrate the generalizability of the scaled sampling-efficiency.

Efficient diffusion models. Nichol et al. [54] show that the generative performance of diffusion models improves as the model size increases. Based on this preliminary observation, the model size of widely used LDMs, *e.g.*, Stable Diffusion [9], has been empirically increased to billions of parameters [12, 57]. However, such a large model makes it impossible to fit into the common inference budget of practical scenarios. Recent work on improving the sampling efficiency focus on improving network architectures [24, 27–31, 58] or the sampling procedures [6, 32–36, 59]. We explore sampling efficiency by training smaller, more compact LDMs. Our analysis involves scaling down the model size, training from scratch, and comparing performance at equivalent inference cost.

Efficient non-diffusion generative models. Compared to diffusion models, other generative models such as, Variational Autoencoders (VAEs) [3, 60–62], Generative Adversarial Networks (GANs) [25, 26, 63–65], and Masked Models [66–70], are more efficient, as they rely less on an iterative refinement process. Sauer et al. [71] recently scaled up StyleGAN [26] into 1 billion parameters and demonstrated the single-step GANs' effectiveness in modeling text-to-image generation. Chang et al. [70] scaled up masked transformer models for text-to-image generation. These non-diffusion generative models can generate high-quality images with less inference cost, which require fewer sampling steps than diffusion models and autoregressive models, but they need more parameters, *i.e.*, 4 billion parameters.

2.3 Scaling LDMs

We developed a family of powerful Latent Diffusion Models (LDMs) built upon the widely-used `866M` Stable Diffusion v1.5 standard [9]. The denoising UNet of our models offers a flexible range of sizes, with parameters spanning from `39M` to `5B`. We incrementally increase the number of filters in the residual blocks while maintaining other architecture elements the same, enabling a predictably controlled scaling. Table 2.1 shows the architectural differences among our scaled models. We also provide the relative cost of each model against the baseline model. Fig. 2.2 shows the architectural differences during scaling. Models were trained using the web-scale aesthetically filtered text-to-image dataset, *i.e.*, WebLI [72]. All the models are trained for 500K steps, batch size 2048, and learning rate 1e-4. This allows for all the models to have reached a point where we observe diminishing returns. Fig. 2.1 demonstrates the consistent generation capabilities across our scaled models. We used the common practice of 50 sampling steps with the DDIM sampler, 7.5 classifier-free guidance rate, for text-to-image generation. The visual quality of the results exhibits a clear improvement as model size increases.

In order to evaluate the performance of the scaled models, we test the text-to-image performance of scaled models on the validation set of COCO 2014 [10] with 30k samples. For downstream performance, specifically real-world super-resolution, we test the performance of scaled models on the validation of DIV2K with 3k randomly cropped patches, which are degraded with the RealESRGAN degradation [56].

Params	39M	83M	145M	223M	318M	430M	558M	704M	866M	2B	5B
Filters (c)	64	96	128	160	192	224	256	288	320	512	768
GFLOPS	25.3	102.7	161.5	233.5	318.5	416.6	527.8	652.0	789.3	1887.5	4082.6
Norm. Cost	0.07	0.13	0.20	0.30	0.40	0.53	0.67	0.83	1.00	2.39	5.17
FID ↓	25.30	24.30	24.18	23.76	22.83	22.35	22.15	21.82	21.55	20.98	20.14
CLIP ↑	0.305	0.308	0.310	0.310	0.311	0.312	0.312	0.312	0.312	0.312	0.314

Table 2.1: We scale the baseline LDM (*i.e.*, 866M Stable Diffusion v1.5) by changing the base number of channels c that controls the rest of the U-Net architecture as $[c, 2c, 4c, 4c]$ (See Fig. 2.2). GFLOPS are measured for an input latent of shape $64 \times 64 \times 4$ with FP32. We also show a normalized running cost with respect to the baseline model. The text-to-image performance (FID and CLIP scores) for all scaled LDMs is evaluated on the COCO-2014 validation set with 30k samples, using 50-step DDIM sampling and Classifier-free Guidance (CFG) with a rate of 7.5. It is worth noting that all the model sizes, and the training and the inference costs reported in this work only refer to the denoising UNet in the latent space, and do not include the 1.4B text encoder and the 250M latent encoder and decoder.

2.3.1 Training compute scales text-to-image performance

We find that our scaled LDMs, across various model sizes, exhibit similar trends in generative performance relative to training compute cost, especially after training stabilizes, which typically occurs after 200K iterations. These trends demonstrate a smooth scaling in learning capability between different model sizes. To elaborate, Fig. 2.3 illustrates a series of training runs with models varying in size from 39 million to 5 billion parameters, where the training compute cost is quantified as the product of relative cost shown in Table 2.1 and training iterations. Model performance is evaluated by using the same sampling steps and sampling parameters. In scenarios with moderate training compute (i.e., $< 1G$, see Fig. 2.3), the generative performance of T2I models scales well with additional compute resources.

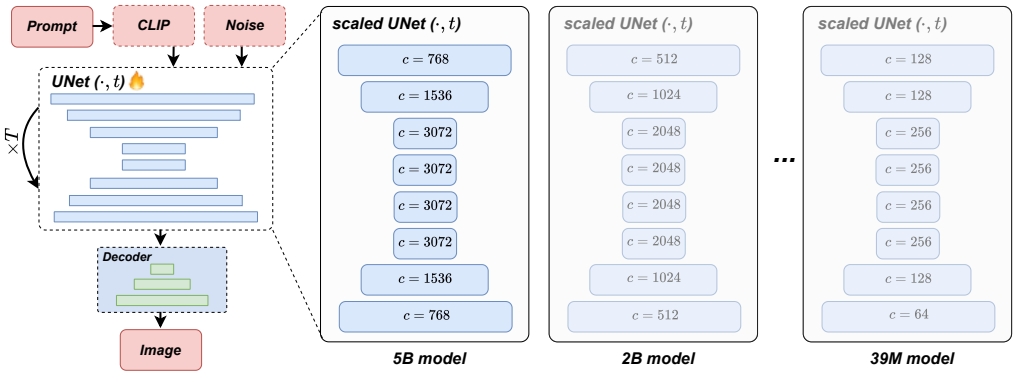


Figure 2.2: Our scaled latent diffusion models vary in the number of filters within the denoising U-Net. Other modules remain consistent. Smooth channel scaling (64 to 768) within residual blocks yields models ranging from `39M` to `5B` parameters. For downstream tasks requiring image input, we use an encoder to generate a latent code; this code is then concatenated with the noise vector in the denoising U-Net.

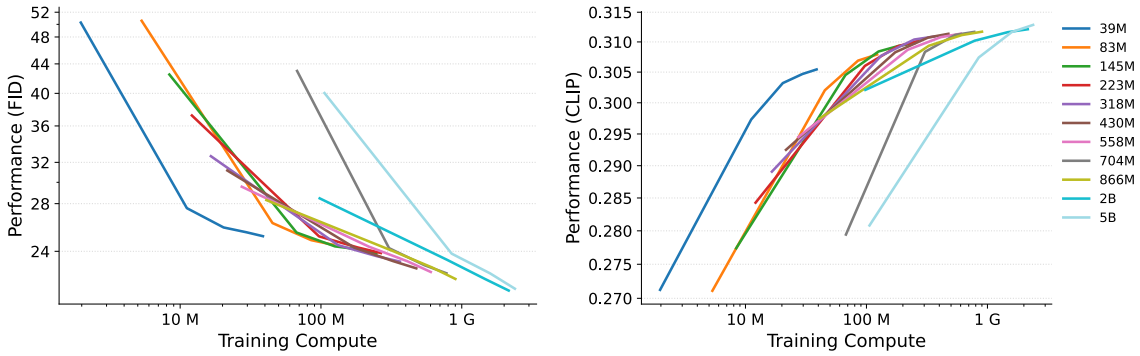


Figure 2.3: In text-to-image generation using 50-step DDIM sampling and CFG rate of 7.5, we observe consistent trends across various model sizes in how quality metrics (FID and CLIP scores) relate to training compute (*i.e.*, the total GFLOPS spend on training). Under moderate training resources, training compute is the most relevant factor dominating quality.

2.3.2 Pretraining scales downstream performance

Using scaled models based on their pretraining on text-to-image data, we finetune these models on the downstream tasks of real-world super-resolution [43, 44] and DreamBooth [45]. The performance of these pretrained models is shown in Table. 2.1.

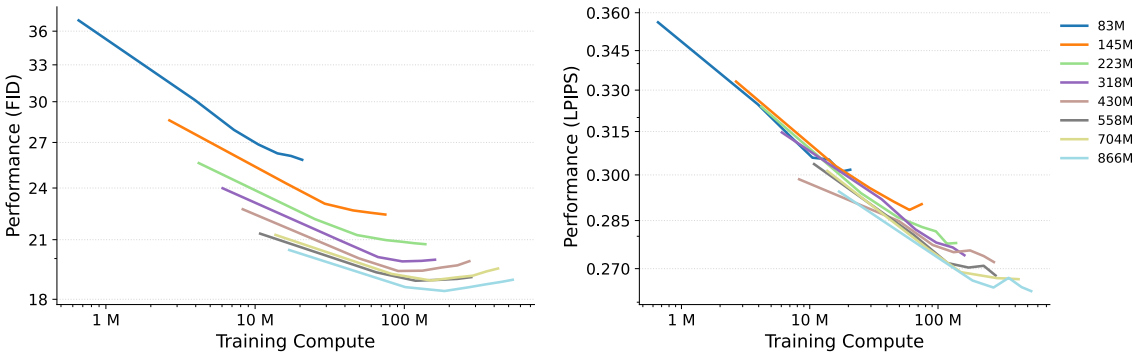


Figure 2.4: In 4× real image super-resolution using 50-step DDIM sampling, FID and LPIPS scores reveal an interesting divergence. Model size drives FID score improvement, while training compute most impacts LPIPS score. Despite this, visual assessment (Fig. 8.8) confirms the importance of model size for superior detail recovery (similarly as observed in the text-to-image pretraining).

In the left panel of Fig. 2.4, we present the generative performance FID versus training compute on the super-resolution (SR) task. It can be seen that the performance of SR models is more dependent on the model size than training compute. Our results demonstrate a clear limitation of smaller models: they cannot reach the same performance levels as larger models, regardless of training compute.

While the distortion metric LPIPS shows some inconsistencies compared to the generative metric FID (Fig. 2.4), Fig. 8.8 clearly demonstrates that larger models excel in recovering fine-grained details compared to smaller models.

The key takeaway from Fig. 2.4 is that large super-resolution models achieve superior results even after short finetuning periods compared to smaller models. This suggests that pretraining performance (dominated by the pretraining model sizes) has a greater influence on the super-resolution FID scores than the duration of finetuning (*i.e.*, training compute for finetuning). Furthermore, we compare the visual results of the

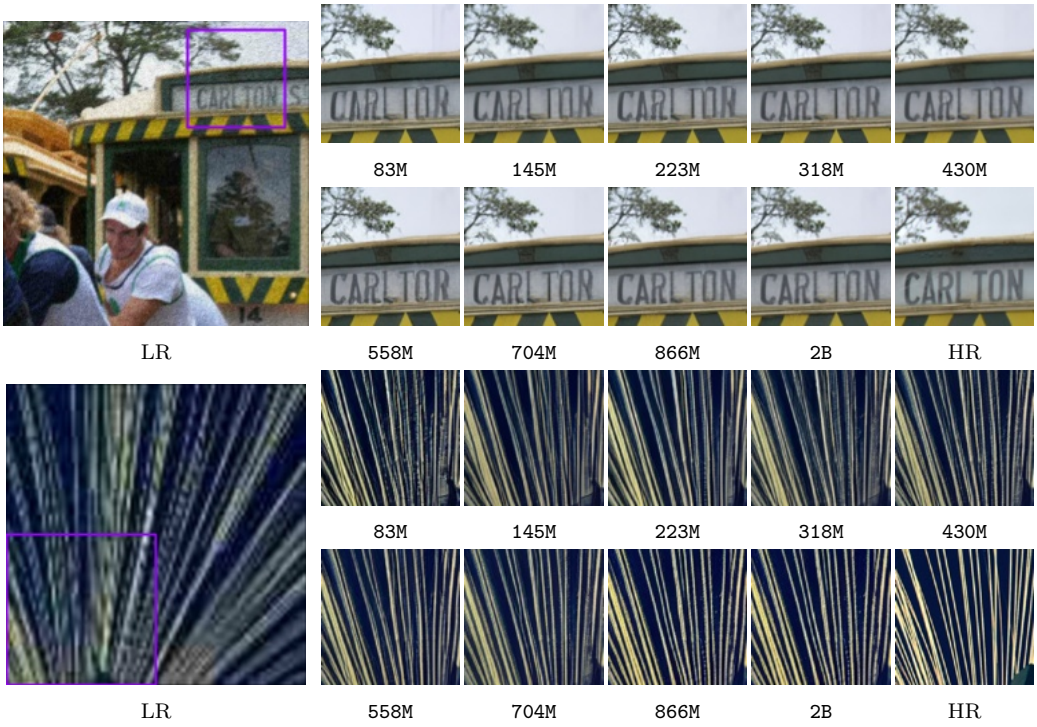


Figure 2.5: In 4× super-resolution using 50-step DDIM sampling, visual quality directly improves with increased model size. As these scaled models vary in pretraining performance, the results clearly demonstrate that pretraining boosts super-resolution capabilities in both quantitative (Fig 2.4) and qualitative ways.

DreamBooth finetuning on the different models in Fig. 2.6. We observe a similar trend between visual quality and model size.

2.3.3 Scaling sampling-efficiency

Analyzing the effect of CFG rate. Text-to-image generative models require nuanced evaluation beyond single metrics. Sampling parameters are vital for customization, with the Classifier-Free Guidance (CFG) rate [73] directly influencing the balance between visual fidelity and semantic alignment with text prompt. Rombach et al. [9] experimentally demonstrate that different CFG rates result in different CLIP

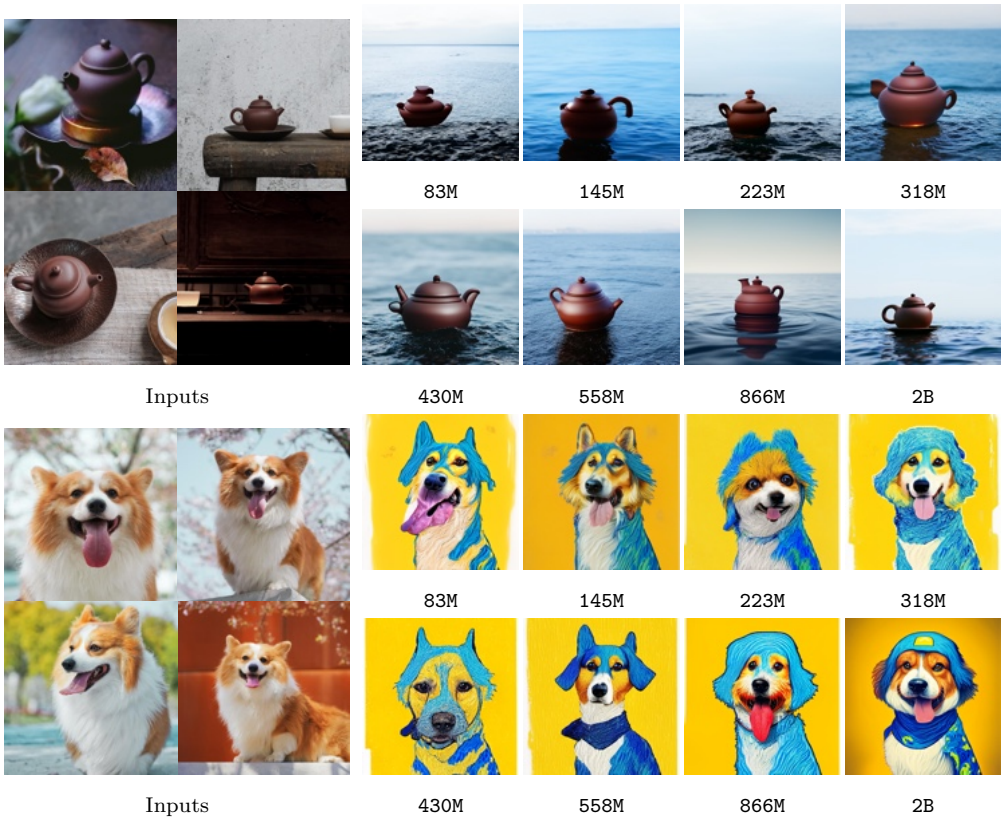


Figure 2.6: Visualization of the Dreambooth results (using 50-step DDIM sampling and CFG rate of 7.5) shows two distinct tiers based on model size. Smaller models (`83M-223M`) perform similarly, as do larger ones (`318M-2B`), with a clear quality advantage for the larger group.

and FID scores.

In this study, we find that CFG rate as a sampling parameter yields inconsistent results across different model sizes. Hence, it is interesting to quantitatively determine the *optimal* CFG rate for each model size and sampling steps using either FID or CLIP score. We demonstrate this by sampling the scaled models using different CFG rates, *i.e.*, $(1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0)$ and comparing their quantitative and qualitative results. In Fig. 2.7, we present visual results of two models under varying CFG rates,

(a) 50-step sampling results of the 145M model

(b) 50-step sampling results of the 866M model

Figure 2.7: Visualization of text-to-image results with 50-step DDIM sampling and different CFG rates (from left to right in each row: $(1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0)$). The prompt used is "*A raccoon wearing formal clothes, wearing a top hat and holding a cane. Oil painting in the style of Rembrandt.*". We observe that changes in CFG rates impact visual quality more significantly than the prompt semantic accuracy. We use the FID score for quantitative determination of optimal sampling performance (Fig. 2.8) because it directly measures visual quality, unlike the CLIP score, which focuses on semantic similarity.

highlighting the impact on the visual quality. We observed that changes in CFG rates impact visual quality more significantly than prompt semantic accuracy and therefore opted to use the FID score for quantitative determination of the optimal CFG rate. performance. Fig. 2.8 shows how different classifier-free guidance rates affect the FID scores in text-to-image generation (see figure caption for more details).

Scaling efficiency trends. Using the optimal CFG rates established for each model at various number of sampling steps, we analyze the optimal performance to understand the sampling efficiency of different LDM sizes. Specifically, in Fig. 2.9, we present a comparison between different models and their optimal performance given the sampling cost (normalized cost $\times$ sampling steps). By tracing the points of optimal performance across various sampling cost—represented by the dashed

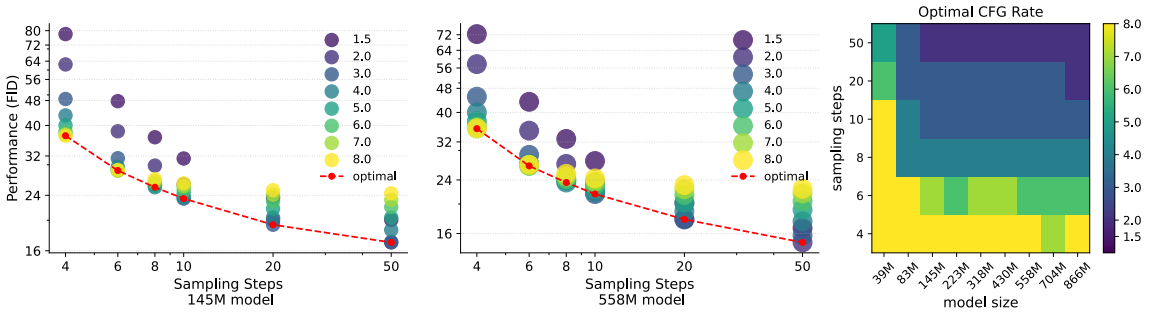


Figure 2.8: The impact of the CFG rate on text-to-image generation depends on the model size and sampling steps. As demonstrated in the left and center panels, the optimal CFG rate changes as the sampling steps increased. To determine the optimal performance (according to the FID score) of each model and each sampling steps, we systematically sample the model at various CFG rates and identify the best one. As a reference of the optimal performance, the right panel shows the CFG rate corresponding to the optimal performance of each model for a given number of sampling steps.

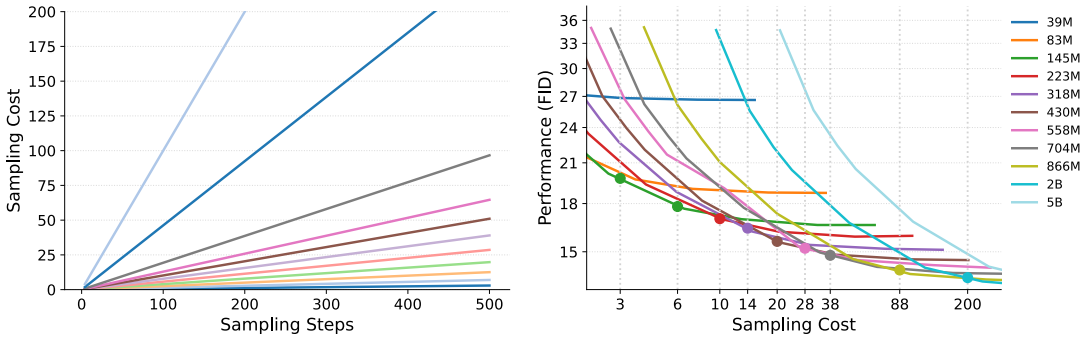


Figure 2.9: Comparison of text-to-image performance of models with varying sizes. The left figure shows the relationship between sampling cost (normalized cost $\times$ sampling steps) and sampling steps for different model sizes. The right figure plots the optimal text-to-image FID score among CFG rates of $(1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0)$ as a function of the sampling cost for the same models. Key Observation: Smaller models achieve better FID scores than larger models for a fixed sampling cost. For instance, at a cost of 3, the 83M model achieves the best FID compared to the larger models. This suggests that smaller models can be more efficient in achieving good results with lower costs.

(a) Prompt: *"A corgi's head depicted as a nebula."*. Sampling Cost ≈ 6.

(b) Prompt: *"A pineapple surfing on a wave."*. Sampling Cost ≈ 12.

Figure 2.10: Text-to-image results of the scaled LDMs under approximately the same inference cost (normalized cost $\times$ sampling steps). Smaller models can produce comparable or even better visual results than larger models under similar sampling cost.

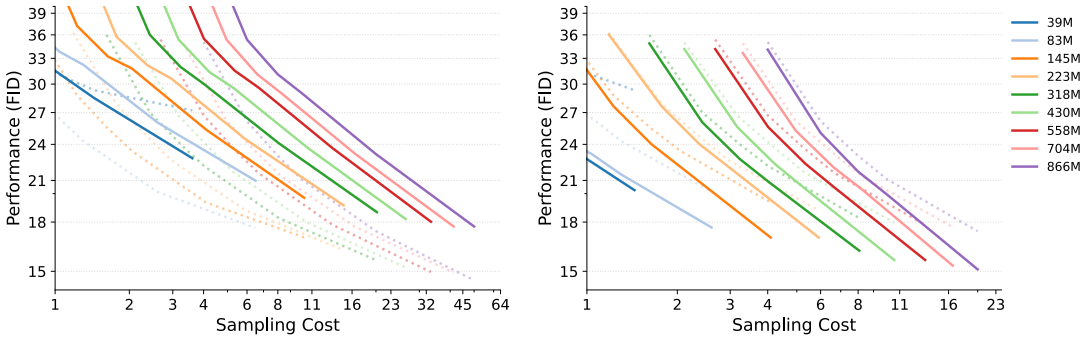


Figure 2.11: *Left*: Text-to-image performance FID as a function of the sampling cost (normalized cost × sampling steps) for the DDPM sampler (solid curves) and the DDIM sampler (dashed curves). *Right*: Text-to-image performance FID as a function of the sampling cost for the second-order DPM-Solver++ sampler (solid curves) and the DDIM sampler (dashed curves). Suggested by the trends shown in Fig. 2.9, we only show the sampling steps ≤ 50 as using more steps does not improve the performance.

vertical line—we observe a consistent trend: smaller models frequently outperform larger models across a range of sampling cost in terms of FID scores. Furthermore, to visually substantiate better-quality results generated by smaller models against larger ones, Fig. 2.10 compares the results of different scaled models, which highlights that the performance of smaller models can indeed match their larger counterparts under similar sampling cost conditions.

Scaling sampling-efficiency in different samplers To assess the generalizability of observed scaling trends in sampling efficiency, we compared scaled LDM performance using different diffusion samplers. In addition to the default DDIM sampler, we employed two representative alternatives: the stochastic DDPM sampler [2] and the high-order DPM-Solver++ [46].

Experiments illustrated in Fig. 2.11 reveal that the DDPM sampler typically produces

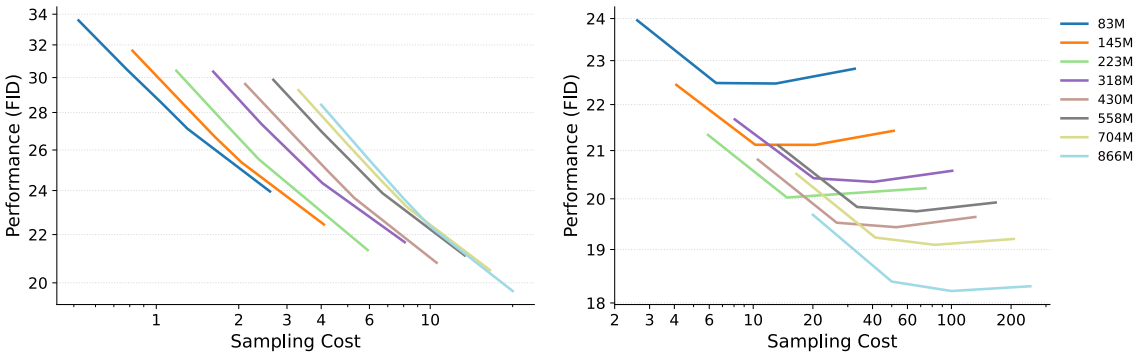


Figure 2.12: Super-resolution performance vs. sampling cost for different model sizes. *Left:* FID scores of super-resolution models under limited sampling steps (less than or equal to 20). Smaller models tend to achieve lower (better) FID scores within this range. *Right:* FID scores of super-resolution models under a larger number of sampling steps (greater than 20). Performance differences between models become less pronounced as sampling steps increase.

lower-quality results than DDIM with fewer sampling steps, while the DPM-Solver++ sampler generally outperforms DDIM in image quality (see the figure caption for details). Importantly, we observe consistent sampling-efficiency trends with the DDPM and DPM-Solver++ sampler as seen with the default DDIM: smaller models tend to achieve better performance than larger models under the same sampling cost. Since the DPM-Solver++ sampler is not designed for use beyond 20 steps, we focused our testing within this range. This finding demonstrates that the scaling properties of LDMs remain consistent regardless of the diffusion sampler used.

2.3.4 Scaling downstream sampling-efficiency

Here, we investigate the scaling sampling-efficiency of LDMs on downstream tasks, specifically focusing on the super-resolution task. Unlike our earlier discussions on optimal sampling performance, there is limited literature demonstrating the positive

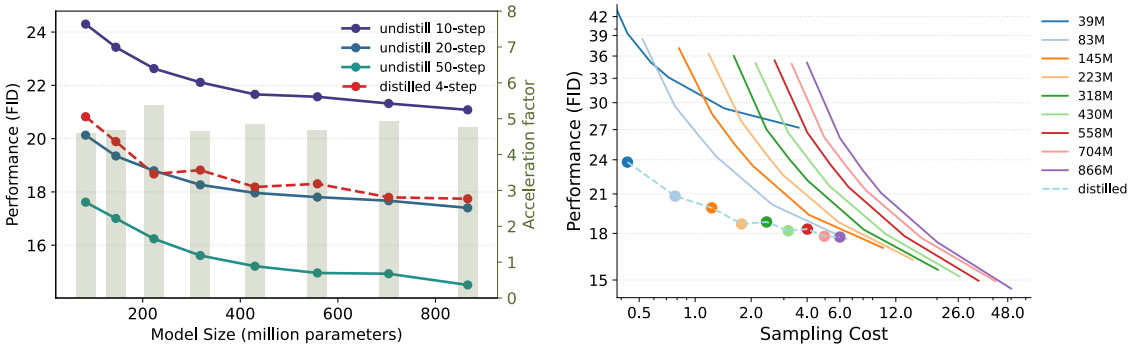


Figure 2.13: Distillation improves text-to-image performance and scalability. *Left:* Distilled Latent Diffusion Models (LDMs) consistently exhibit lower (better) FID scores compared to their undistilled counterparts across varying model sizes. The consistent acceleration factor (approx. 5×) indicates that the benefits of distillation scale well with model size. *Right:* Distilled models using only 4 sampling steps achieve FID scores comparable to undistilled models using significantly more steps. Interestingly, at a sampling cost of 7, the distilled `866M` model performs similarly to the smaller, undistilled `83M` model, suggesting improved efficiency.

impacts of SR performance without using classifier-free guidance. Thus, our approach directly uses the SR sampling result without applying classifier-free guidance. Inspired from Fig. 2.4, where the scaled downstream LDMs have significant performance difference in 50-step sampling, we investigate sampling efficiency from two different aspects, *i.e.*, fewer sampling steps $[4, 20]$ and more sampling steps $(20, 250]$. As shown in the left part of Fig. 2.12, the scaling sampling-efficiency still holds in the SR tasks when the number of sampling steps is less than or equal to 20 steps. Beyond this threshold, however, larger models demonstrate greater sampling-efficiency than smaller models, as illustrated in the right part of Fig. 2.12. This observation suggests the consistent sampling efficiency of scaled models on fewer sampling steps from text-to-image generation to super-resolution tasks.

2.3.5 Scaling sampling-efficiency in distilled LDMs.

We have featured the scaling sampling-efficiency of latent diffusion models, which demonstrates that smaller model sizes exhibit higher sampling efficiency. A notable caveat, however, is that smaller models typically imply reduced modeling capability. This poses a challenge for recent diffusion distillation methods [8, 37–41, 74, 75] that heavily depend on modeling capability. One might expect a contradictory conclusion and believe the distilled large models sample faster than distilled small models. In order to demonstrate the sampling efficiency of scaled models after distillation, we distill our previously scaled models with conditional consistency distillation [38, 41] on text-to-image data and compare those distilled models on their optimal performance.

To elaborate, we test all distilled models with the same 4-step sampling, which is shown to be able to achieve the best sampling performance; we then compare each distilled model with the undistilled one on the normalized sampling cost. We follow the same practice discussed before for selecting the optimal CFG rate and compare them under the same relative inference cost. The results shown in the left part of Fig. 2.13 demonstrate that distillation significantly improves the generative performance for all models in 4-step sampling, with FID improvements across the board. By comparing these distilled models with the undistilled models in the right part of Fig. 2.13, we demonstrate that distilled models outperform undistilled models at the same sampling cost. However, at the specific sampling cost, *i.e.*, sampling cost ≈ 8, the smaller undistilled 83M model still achieves similar performance to the larger distilled 866M model. The observation further supports our proposed scaling sampling-efficiency after diffusion distillation.

Chapter 3
Video Implicit Diffusion Models

3.1 Introduction

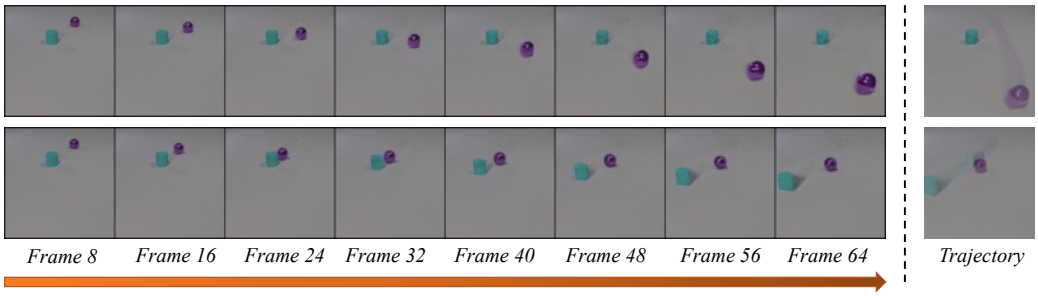
Image generation has gained significant traction since the introduction of Generative Adversarial Networks (GANs) [76]. In these methods, the idea is to generate new images that conform to the training data distribution. Following the success of image synthesis, video generation has also gained significant attention. Various video generation methods have been proposed in the literature including GAN-based methods [77–80], Autoregressive models [81], and Time-series models [82, 83]. An advantage of some of these generative models is that they can learn to synthesize high-quality videos without requiring any labels. These generative models have been shown to be beneficial in various high-level recognition tasks [77, 84]. A GAN-based video generation model was proposed by [77], which makes use of a spatio-temporal convolutional architecture and untangles the scene's foreground from the background. Another work proposed by [79] is a continuous-time video generator. In this method, a video is decomposed into the content and motion vectors at generation and discriminated coherently by the discriminators. While these GAN-based methods can model plausible moving objects and scenes, a better video generation model should be able to model the distribution of internal spatial and temporal changes with regards to the video content.

Different from GANs, diffusion models [2, 85] that model the probability directly have emerged as the new state-of-the-art generative models, and they have been

shown to outperform GANs in various generation tasks [7]. By learning to reverse the diffusion process that adds noise to data in finite successive steps, diffusion models can gradually map a Gaussian distribution to the probability distribution corresponding to a real complex high-dimensional dataset. In its denoising process, conditional features like class labels of data can be applied to the network for specializing its sampling process. By appropriately using conditional features, diffusion models have shown impressive performance in various applications, *e.g.*, image debluring [86] that conditions on the image residual, high-resolution image generation [87] that conditions on the low-resolution images, and image editing [88] that conditions on the style. With more expressive conditional features like CLIP embeddings, diffusion models like DALLE-2 [57] are capable of generating highly creative images with impressive photorealism. But the condition mechanism in diffusion models is non-trivial and requires careful design to improve the quality of the generated images.

We assume that the subspace of a real video can be represented as a subspace of the video content, and the video motion is then generated by traversing point on the video content subspace. Accurately modeling the content subspace increases the realism of frames, while accurately modeling the subspace of the trajectory regarding the video content can produce continuous and smooth video. Thus a better video generation model should own delicate modulation capability for simulating both the trajectory and realistic content.

Following this idea, we propose to model the video content and motion with two diffusion models separately. The first video frame is generated by the content generator. Subsequently, the motion generator generates the next video frame based on the latent



(a) Generated two videos with the same content on CLEVRER for collision reasoning

(b) Generated extra long SKY Time-lapse video with 128 frames (4 frames skipped)

(c) Generated high-resolution 256x256 TaiChi videos with 16 frames

Figure 3.1: Sample results corresponding to our method on multiple video datasets.

map of the first frame and the latest frame, i.e., an optical-flow like feature between the first and the latest frame estimated by an additional network. This enables implicit modeling of dynamics by conditioning on the latent features. After training, the optimized condition can best represent the spatial and temporal changes for generating the next frame. By iteratively running the motion generator, the final video is generated in an autoregressive manner. We experimentally find that the estimated condition significantly enhances the modeling capability of diffusion models, Such an expressive model is capable of simulating the trajectory of videos according to the conditional latent.

The major idea of our video implicit diffusion models is:

- *Content Generator:* We propose to learn video content separately with an introduced diffusion model on video frames. It simplifies video generation modeling and provides easy scalability of complex models. Two heuristic mechanisms, including constant truncation and robustness penalty, are proposed for further improving its performance.

- *Motion Generator:* We propose a motion generator for modeling spatial and temporal changes. It can generate future frames according to the generated content in an autoregressive way. The generator is implicitly conditioned on the latent code predicted by a module similar to an optical-flow network. Furthermore, the coherency of spatial and temporal changes is regularized with an introduced positional group normalization, and the learning is simplified with our proposed adaptive feature residual.

3.2 Related Work

Generative Models. Existing generative models can be categorized into likelihood-based and implicit models, based on the way of representing probability distribution. Among them, Variational Auto-encoders (VAEs) [3], Autoregressive models [89], Normalizing Flow [90], and Diffusion models [2, 85] directly model the probability distribution of data via maximum likelihood. In contrast, GANs [76] implicitly represent the probability distribution via their sampled results. Though the idea of GANs is simple, the boundary has been significantly pushed by GANs and their representative variants, including StyleGAN [26, 91, 92] and BigGAN [93]. Moreover, many general techniques based on GANs have emerged, including R_1 regularization [94],

path length regularization [91], truncation trick [26], spectral normalization [65], image inversion [95], and adaptive discriminator [92]. However, we find that such techniques are rarely explored in diffusion models.

Conditional Generative Models. Modeling the probability distribution of complex datasets such as ImageNet [96] can face potential training instability and mode collapse issues. Therefore, the way of leveraging additional conditions as a guidance is explored and becoming the most promising way of mitigating the issues. For GANs, class information can be fed into the generator [93] and the discriminator [26] for fascinating class-conditional sampling. For diffusion models, the class condition shows a better performance boost as the class embeddings used in DDPM [2]. Furthermore, resulting from the iterative denoising process of diffusion models, which enables hierarchical conditional features, utilizing the class feature of noisy images of different time steps can help diffusion models achieve the new art [7]. Different from class conditions, the modality of conditions could be images [97] and even texts like DALLE [98] for different aims.

Video Generation. Video generation has been dominated by 3D CNNs [99] for a long time until the recent emergence of Implicit Neural Representation (INR) [100]. The early 3D CNNs based video generation works take all frames of the video as a single point on the video subspace. They then generate a cuboid as the result of each sampling process [77, 78], and such a manner has been extended into the recent diffusion fashion [101, 102]. However, this line can hardly achieve desired results due to the difficulty of modeling spatial-temporal changing, and their scalability is significantly limited according to the cubic complexity [103]. Later work decomposes the generation

process into content and motion separately [79, 104–106], which simplifies the learning but still requires the discriminator to apply 3D CNNs on extracting temporal features. The other line of video generation [80, 83] based on INRs is similar to the image generation applications work [107], which adds additional temporal dimension at the coordinates and thus can process each frame separately. However, such an INR protocol can hardly be applied to diffusion models. Therefore, our method incorporates the coordinate embeddings of INRs as the normalization and conditions on the implicit latent. We experimentally find that the new paradigm benefits the continuous of the generated complex videos.

3.3 Methods

Our proposed video generation method consists of two streams for content and motion generation, respectively. The two streams share a similar network architecture but different in learning objectives. In addition, they have different conditions which helps to keep the design redundancy minimal and reduces the optimization cost. We denote the n-th frame of the N-frame video as $\mathbf{x}(n)_0$. The noisy frame at the tth timestep is denoted as $\mathbf{x}(n)_t$.

Content Generator models the distribution of random video frames $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ with a network $\boldsymbol{\epsilon}_\theta(\cdot)$ and is truncated by constant tensor c. The frame $\mathbf{x}_0$ is randomly selected from videos without specification. The network $\boldsymbol{\epsilon}_\theta(\cdot)$ is the modified U-Net proposed by [7] with *Multi-Head Attention* [108] and utilizes *GroupNorm* [109].

Motion Generator models the distribution of motion from the first frame to the

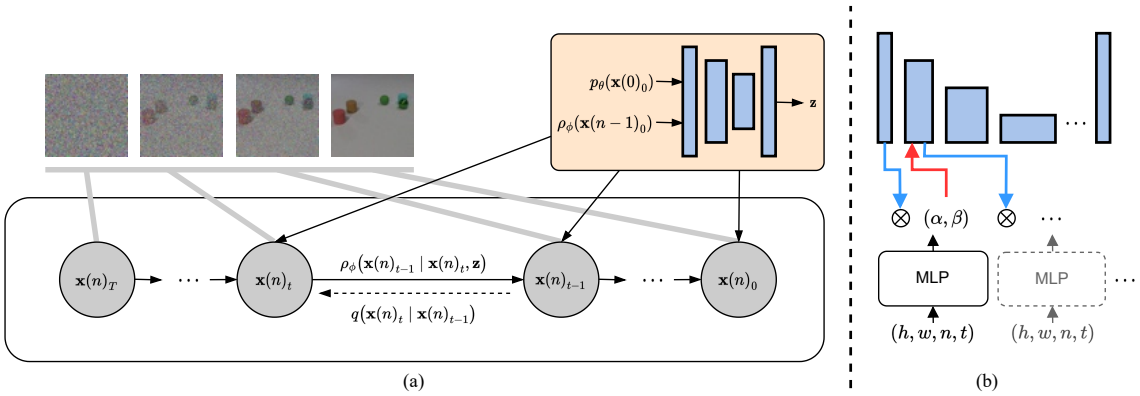


Figure 3.2: (a) Illustration of our graphical model at the n-th video frame sampling process. (b) The proposed positional group normalization concept when it is applied to the diffusion network.

random n-th frame, and it is implemented with another network $\rho_\phi(\cdot)$ with parameters ϕ. Therefore, the learning process minimizes the difference between $\rho_\phi(\mathbf{x}(0)_0, n)$ and $\mathbf{x}(n)_0$ as Figure 3.2 shows, which is similar to recent implicit neural function methods [80, 83].

By experimentally combining the two streams, one can model video data. However, due to the complexity of video data, we observed that the basic implementation does not converge. In addition, it losses significant generation quality and generates discontinuous motions. Therefore, we propose to extend the aforementioned video generation process with the following improvements.

3.3.1 Positional Group Normalization

Our first key idea for improving the diffusion network is to incorporate the spatial and temporal positional encoding of 4D coordinates (h, w, n, t) between each U-Net blocks, for modeling continuous changes in both the space h, w and time n with different

diffusion timesteps t. The correlation between spatial and temporal features crucially affects the continuity of video data but is conventionally ignored due to its complexity. Empirically, such complexity can be decomposed for modeling in an iterative denoising process. We propose to directly incorporate the correlation into networks in a feature modulation manner, similar to AdaIN [26] and FiLM [110].

The concept is illustrated in the right part of Figure 3.2. Specifically, the positional encoding mapped from 4D coordinates is extracted through an MLP (fully-connected neural network) with sinusoidal activation [100] after its first layer. Recent studies on implicit neural representations (INRs) [100, 111] have shown that periodic activation is capable of modeling high dimensional space with coordinates. Inspired by it, our introduced Positional Group Normalization (PosGN) based on group-norm [109] is defined as

$$\alpha, \beta = \mathrm{MLP}(h, w, n, t) \tag{3.1}$$

$$\mathrm{PosGN}(x, \alpha, \beta) = \alpha \cdot \mathrm{GroupNorm}(x) + \beta, \tag{3.2}$$

where x is the obtained feature from the U-Net blocks, (α, β) is a pair of affine transformation parameters extracted from the MLP, and it then scales and shifts feature x using parameters (α, β). PosGN is based on the empirical superiority of adaptive group normalization (AdaGN) [54], which has been shown to benefit diffusion models, and the difference between them are the introduced periodic activated MLP and the additional spatial and temporal dimensions. Compared with the recent INR-based work, PosGN is particularly suited for diffusion models. It is because the noisy images are essential conditions that cannot be replaced by coordinates as INRs

have done. Besides, PosGN provides a hierarchical feature modulation when it is incorporated into the applied diffusion networks.

As a result, our proposed VIDM benefits from the capability of modeling spatial and temporal changes led by PosGN. Based on the new paradigm, the learning objective of our motion generation extended from the content modeling

$$\mathcal{L}_\theta(\boldsymbol{\epsilon}, \mathbf{x}_t, t) = \sqrt{(\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, c, t) - \boldsymbol{\epsilon})^2 + \eta^2}, \tag{3.3}$$

for an arbitrary n-th frame is formulated as

$$\mathcal{L}_\phi(\boldsymbol{\epsilon}, \mathbf{x}(n)_t, t, n) = \sqrt{(\rho_\phi(\mathbf{x}(n)_t, t, n) - \boldsymbol{\epsilon})^2 + \eta^2}. \tag{3.4}$$

Coordinates (h, w) are derived from features on-the-fly and thus are not treated as the network input. For convenience and efficiency, coordinates (h, w, n, t) are only generated at the first time and then cached for the next running. Therefore, PosGN shares a very similar computational cost as the vanilla AdaGN when the running times are large, which is natural to diffusion models. In the rest of this paper, we use PosGN as our default settings and denote $\rho_\phi(\cdot, t, n)$ as $\rho_\phi(\cdot)$ for simplification.

3.3.2 Implicit Motion Condition

Modeling long continuous video data has been a long-standing problem, even though we have seen the exploration in INRs and our proposed PosGN with positional encoding, the intermediate information between long video frames cannot be accurately represented. Furthermore, from the results in the literature [80, 83], we find that the

intermediate information plays a crucial role in the video continuation, otherwise, the generated long videos only contain nearly meaningless motions.

Our second idea is extended from the proposed PosGN, based on the time condition, instead of explicit coordinates, we propose to condition on the latent code of the latest frame and the first frame at the denoising process. The latent code is an optical flow [112] like feature estimated by an additional network $\mathbf{v}(\cdot)$, implemented as SpyNet [113], which has been demonstrated in motion extraction for video enhancement and interpolation. To elaborate, a pretrained optical flow estimation network $\mathbf{v}(\cdot)$ is applied to estimate the latent $\mathbf{z}$ between frames $\mathbf{v}(\mathbf{x}(0)_0, \mathbf{x}(n-1)_0)$ for the n-th frame $\mathbf{x}(n)_0$ generation, which is performed in an autoregressive manner. Since the latent code is capable of ensembling the continuous motion feature that consistently exist in the denoising process, it can ensure that the intermediate information is implicitly incorporated into learning. Therefore, the learning process with the implicit latent condition is

$$\begin{aligned} \mathbf{z} &= \mathbf{v}(\mathbf{x}(0)_0, \mathbf{x}(n-1)_0) \\ \mathcal{L}_\phi(\boldsymbol{\epsilon}, \mathbf{x}(n)_t, \mathbf{z}) &= \sqrt{(\rho_\phi(\mathbf{x}(n)_t, \mathbf{z}) - \boldsymbol{\epsilon})^2 + \eta^2}. \end{aligned} \tag{3.5}$$

The parameters of $\mathbf{v}(\cdot)$ are updated with the diffusion networks together without specification. The cost of conditioning on the latent at each denoising process is only increased at the first timesteps and can be then cached. As will be shown in the ablation study, implicit learning is crucial for modeling long video data and can significantly improve the ultimate performance.

3.3.3 Adaptive Feature Residual

To further simplify the motion modeling complexity, we propose to model the residual of content features at each denoising timestep adaptively. An additional encoder that shares the similar architecture of the diffusion network is utilized, and it conditions on the first frame $\mathbf{x}(0)_0$ and timesteps t. We denote the encoding as $\hat{\rho}_\phi(\cdot)$ and the residual feature as r, and thus network $\rho_\phi(\cdot)$ is actually learning to synthesize the residual, which significantly simplifies the learning at each timestep and enables better implicit motion learning. Remark that content generation learning is kept the same as DDPM except for the truncation trick and robustness penalty is applied for enhancing the generation capability.

3.4 Experiments

Datasets and settings. Most datasets follow the protocols of their original papers except where specified. To compare the visual quality of the results, we use the I3D network trained on Kinetics-400 [114] for reporting the Fréchet video distance (FVD) [115] performance, which measures the probability distribution difference between two groups of video results and is recognized by the other prior arts [80, 83]. For reference, we also report the Inception score (IS) [116] performance and Fréchet inception distance (FID) [117] following the evaluation procedure of DIGAN [80]. All evaluation is conducted on 2048 randomly selected real and generated videos for reducing variance. The experiments are conducted on *UCF-101* [118], *Tai-Chi-HD* [119], *Sky Time-lapse* [120], and *CLEVRER* [121].

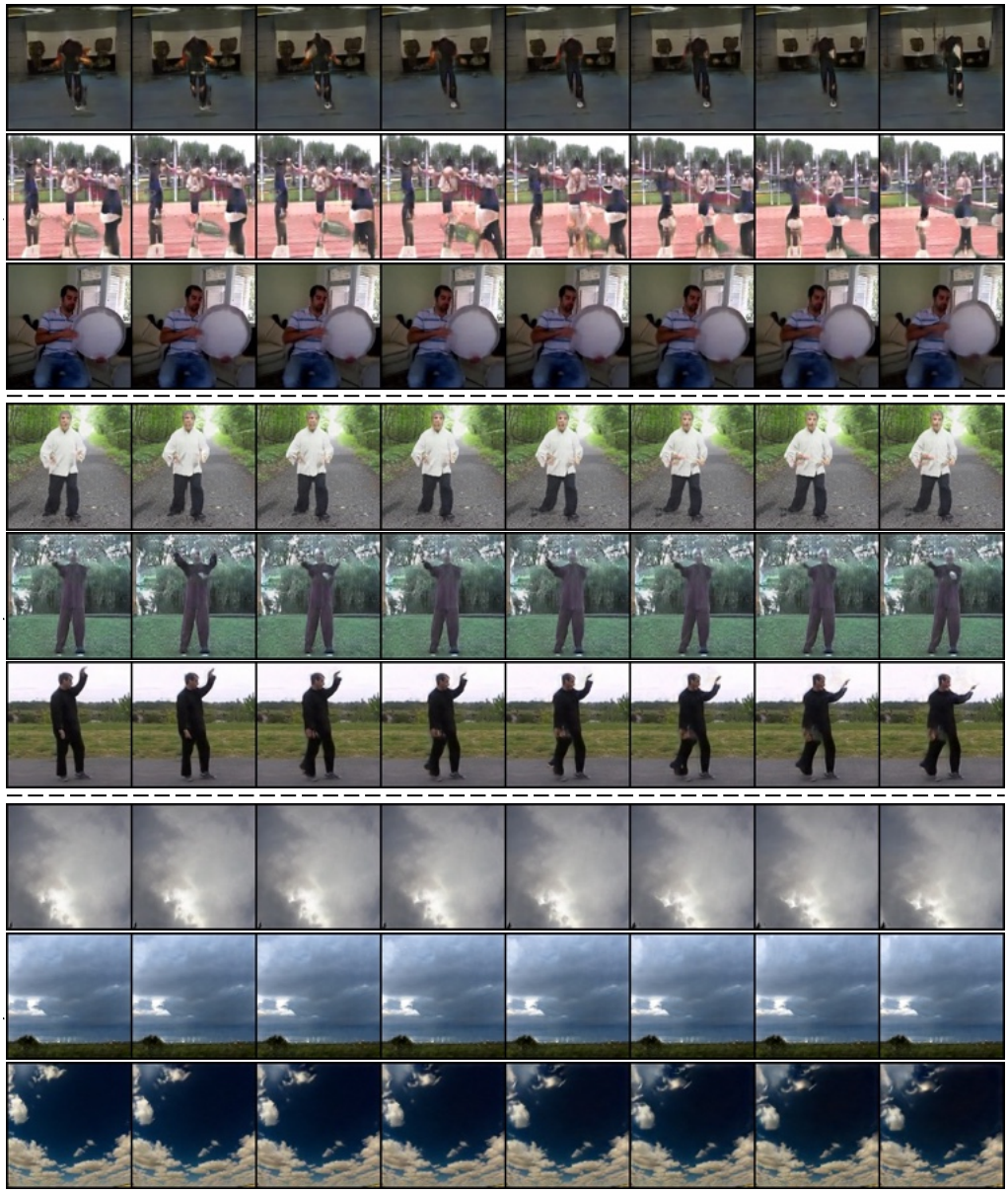


Figure 3.3: Sample result comparisons on the *256-UCF101*[16], *128-TaiChi*[16], and *256-SkyTimelapse*[16] datasets. Each presented frame is selected with 2 frames interval. Each group consists of three row results, which are DIGAN, StyleGAN-V, and ours, from top to bottom.

Baselines. The major baseline for comparison is DIGAN [80], which is the current state-of-the-art in video generation and is the first work that incorporates INRs. We also compare the performance of our method with that of VGAN [77], TAGN [78], MoCoGAN [79], ProgressiveVGAN [122], DVD-GAN [104], LDVD-GAN [123], TGANv2 [103], MoCoGAN-HD [82], VideoGPT [124], StyleGAN-V [83], VDM [101], and TATS [125]. We collect the performance score from the references or re-implemented results from DIGAN and StyleGAN-V if available. For the CLEVRER performance, we train DIGAN and StyleGAN-V with their official code and our implementation with the same settings.

Diffusion Network. The diffusion network architecture of our method is an autoencoder network that follows the design of PixelCNN++ [126]. We apply multiple multi-head attention modules [108] at features in a resolution of 16×16 for capturing long-range dependence that benefits the perceptual quality. It has been verified by DDPM [2] and its variants [7, 54], and we keep minimal changes.

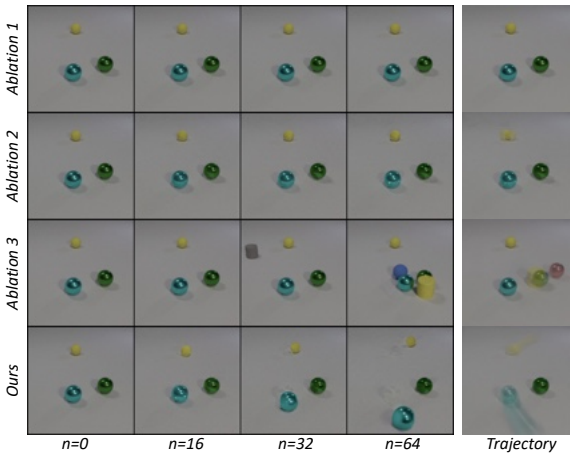


Figure 3.4: Ablation results in different settings.

	MoCoGAN† *CVPR18*	MoCoGAN-HD *ICLR21*	VideoGPT *arXiV21*	DIGAN *ICLR22*	DIGAN‡ *ICLR22*	StyleGAN-V *CVPR22*	TATS *ECCV22*	VIDM (ours)
256-UCF101^{16}	1821.4	1729.6	2880.6	1630.2	471.9	1431.0	332	**294.7**
256-UCF101^{128}	2311.3	2606.5	*N/A*	2293.7	*N/A*	1773.4	-	**1531.9**
256-SkyTimelapse^{16}	85.9	164.1	222.7	83.1	83.1	79.5	132	**57.4**
256-SkyTimelapse^{128}	272.8	878.1	*N/A*	196.7	196.7	197.0	-	**140.9**

	DIGAN	StyleGAN-V	VIDM (ours)		DIGAN	StyleGAN-V	VDM	VIDM (ours)
256-CLEVRER^{16}	112.5	106.1	**87.4**		*128-TaiChi*^{16}	128.1	143.5	**121.9**
256-CLEVRER^{128}	531.7	493.3	**426.5**		*128-TaiChi*^{128}	748.0	691.1	**563.6**

Table 3.1: Fréchet video distance [115] comparison. The compared methods are re-trained on the CLEVRER dataset by us, and by [83] and [80] on the other datasets with their official implementation. MoCoGAN† is implemented with StyleGAN2 as its backbone. DIGAN‡ is class conditional.

Main Results. We present the main quantitative results comparison in Table 3.1, and the main qualitative results comparison is Figure 3.3. We remark that our performance significantly outperforms the very recent state-of-the-art DIGAN and StyleGAN-V in all of the video data as can be seen from the two tables. Among them, *128-TaiChi* and *256-UCF101* is the hardest video data since their movement is minimal and Frames Per Second (FPS) is varying between videos, but our method can still achieve comparable performance and even better without discriminators.

Ablations. Multiple potential design choices are available in our final method, and most of them affect the results to some degree. We ablate the core components and show the details in Table 3.2 for content generator ablations and motion generator ablations. As the results are shown in Table 3.2, the removed sampling space truncation and robustness penalty hurt the performance of content modeling. These results also verify that removing the robustness penalty decreases both the content modeling

ability and motion modeling ability.

For the motion generator, we measure the ablation effects by comparing generated videos in a varying number of frames, which is the most representative score for measuring continuous and smoothness differences. In Table 3.2, we remove the positional group normalization and implicit motion conditions to see the difference. It is surprising that the modeling capability severely depends on the two proposed components, especially for long video generation. From the results visualized in Figure 3.4, we can notice that simply applying diffusion models (i.e., *Ablation1*) without modification can only generate static images. Applying implicit conditions without PosGN (i.e., *Ablation2*) faces the same issue since they cannot model the spatial and temporal changes. In contrast, even though applying PosGN without implicit conditions (i.e., *Ablation3*) can help the network generates different frames, its results are still noncontinuous.

Method	FID	IS	FVD^{16}	Method	FVD^{16}	FVD^{64}	FVD^{128}
vanilla one	23.0	3.04	115.4	vanilla one	603.7	610.0	648.7
w/o *sampling space truncation*	21.1	3.07	107.9	w/o *PosGN*	532.1	581.3	604.5
w/o *robustness penalty*	19.4	3.07	95.5	w/o *Implicit Conditions*	584.8	552.1	614.1
default VIDM	18.4	3.07	87.4	default VIDM	87.4	286.6	426.5

Table 3.2: Ablation study regarding content generator and motion generator.

Chapter 4

Diffusion Transformer on Unified Field Generation

4.1 Introduction

Generative tasks [57, 127] are overwhelmed by diffusion probabilistic models that hold state-of-the-art results on most modalities like audio, images, videos, and 3D geometry. Take image generation as an example, a typical diffusion model [2] consists of a forward process for sequentially corrupting an image into standard noise, a backward process for sequentially denoising a noisy image into a clear image, and a score network that learns to denoise the noisy image.

The forward and backward processes are agnostic to different data modalities; however, the architectures of the existing score networks are not. The existing score networks are highly customized towards a single type of modality, which is challenging to adapt to a different modality. For example, a recently proposed multi-frame video generation

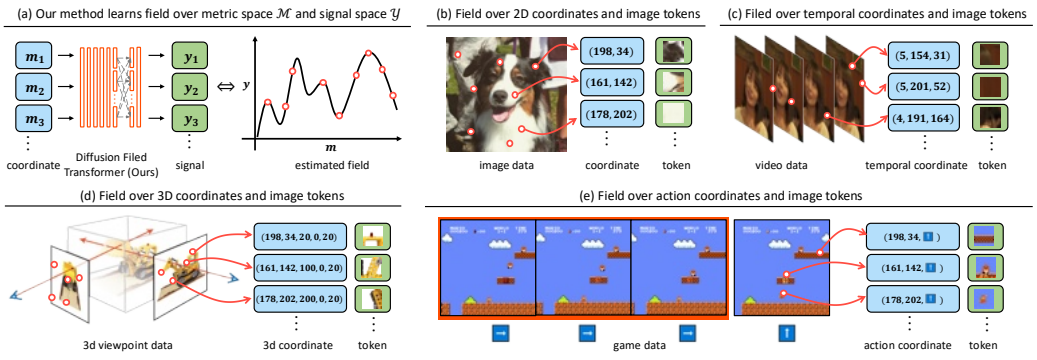


Figure 4.1: Illustration of the field model's capability to model visual content. The model learns the distribution through attention between coordinate-signal pairs, which is modality-agnostic.

network [128, 129] adapting single-frame image generation networks involves significant designs and efforts in modifying the score networks. Therefore, it is important to develop a single, versatile architecture that can be applied across different modalities without modification. Such a unified architecture simplifies the development process, reduces the complexity associated with designing modality-specific models, and enables knowledge transfer between modalities.

Field model [100, 111, 130, 131] is a promising unified score network architecture for different modalities. It learns the distribution over the functional view of data. Specifically, the field f maps the observation from the *metric* space $\mathcal{M}$ (*e.g.*, coordinate or camera pose) into the *signal* space $\mathcal{Y}$ (*e.g.*, RGB pixel) as $f : \mathcal{M} \mapsto \mathcal{Y}$. For instance, an image is represented as $f : \mathbb{R}^2 \mapsto \mathbb{R}^3$ that maps the spatial coordinates (*i.e.*, height and width) into RGB values at the corresponding location, while a video is represented as $f : \mathbb{R}^3 \mapsto \mathbb{R}^3$ that maps the spatial and temporal coordinates (*i.e.*, frame, height, and width) into RGB values. Different modalities usually use different weights to ensure the best performance. Recently, diffusion models are leveraged to characterize the field distributions over the functional view of data [131] for field generation. Given a set of coordinate-signal pairs $\{(\boldsymbol{m}_i, \boldsymbol{y}_i)\}$, the field f is regarded as the score network for the backward process, which turns a noisy signal into a clear signal $\boldsymbol{y}_i$ in a sequential process with $\boldsymbol{m}_i$ being fixed all the time. The visual content is then composed of the clear signal generated on a grid in the metric space.

Nevertheless, diffusion-based field models for generation still lag behind the modality-specific approaches [7, 128, 132] for learning from dynamic data in high resolution [133, 134]. For example, a 240p video lasting 5 seconds is comprised of up to 10 million

coordinate-signal pairs. Due to the memory bottleneck in existing GPU-accelerated computing systems, recent field models [131] are limited to observe merely a small portion of these pairs (*e.g.*, 1%) that are uniformly sampled during training. This limitation significantly hampers the field models in approximating distributions from such sparse observations [135]. Consequently, diffusion-based field models often struggle to capture the fine-grained local structure of the data, leading to, *e.g.*, unsatisfactory blurry results.

While it is possible to change the pair sampling algorithm to sample densely from local areas instead of uniformly, the global geometry is weakened. To alleviate this issue, it is desirable to introduce some complementary guidance on the global geometry in addition to local sampling.

In this paper, we propose a new diffusion field transformers, called **DiFT**. In contrast to previous methods, DiFT preserves both the local structure and the global geometry of the fields during learning by employing a new view-wise sampling algorithm in the coordinate space, and incorporates additional inductive biases from the text descriptions and autoregressive generation. By combining these advancements with our simplified transformer architecture, we demonstrate that modeling capability of our model surpasses previous methods, achieving improved generated results under the same memory constraints. We empirically validate its superiority against previous domain-agnostic methods across three different tasks, including text-to-video generation, 3D novel-view generation, and game generation.

Various experiments show that our method achieves compelling performance even when

compared to the state-of-the-art domain-specific methods, underlining its potential as a scalable and architecture-unified visual content generation model across various modalities.

Our contributions are summarized as follows:

- We propose a new transformer-based diffusion field model for long-context modeling, which comprises of a view-wise sampling algorithm and autoregressive generation for local structure and global geometry model respectively.
- We demonstrate the effectiveness and efficiency of a simple 675M model on different modalities generation including video, 3D, and game in a unified-architecture, which largely closes the performance gap with modality-specific models with different weights.
- We show the potential of action game generation using diffusion models, and we release the benchmarks including both training and testing data for replication and comparisons.

4.2 Related Work

Generation Models. In recent years, generative models have shown impressive performance in visual content generation. The major families are generative adversarial networks [25, 26, 63, 93], variational autoencoders [3, 62], auto-aggressive networks [55, 136], and diffusion models [1, 2]. Recent diffusion models have obtained significant advancement with stronger network architectures [7], additional text conditions [57], and pretrained latent space [132]. Our method built upon these successes and targets

at scaling domain-agnostic models.

Field Models. Field models like SIREN [100] excel at effectively handling diverse data types, such as images, videos, 3D shapes, and audio, without requiring extensive customization. Compared with the modality-specific models, field models enable scalability by allowing advancements in one domain (e.g., images) to directly enhance others (e.g., 3D modeling and video synthesis), streamlining research and development. In order to model complex field distributions, representative methods like Functa [130] and GEM [137] adopt a two-stage modeling paradigm: first parameterizing fields, then learning distributions over the parameterized latent space. However, the learning efficiency of the two-stage methods hinders scaling the models, as their first stage incurs substantial computational costs to compress fields into latent codes. Building on recent exploration [131] into the use of diffusion models, which are more powerful for directly modeling complex data distributions without additional parametrization, we propose to model field distributions using explicit coordinate-signal pairs. Nevertheless, field models struggle with very large or highly diverse datasets, such as high-resolution videos. This is due to the complexity of preserving both local structures and global geometry. In contrast, our method leverages the benefits of a single-stage modeling approach, improving accuracy in preserving both local structures and global geometry.

Long-context Modeling. Our method also differs from the recently proposed domain-specific works for high-resolution, dynamic data, which models specific modalities in a dedicated latent space, including Spatial Functa [138] and PVDM [139]. These methods typically compress the high-dimensional data into a low-dimensional latent space. However, the compression is usually specific to a center modality and lacks the

flexibility in dealing with different modalities. For instances, PVDM compresses videos into three latent codes that represent spatial and temporal dimensions separately. However, such a compressor cannot be adopted into the other similar modalities like 3D scenes. In contrast, our method owns the unification flexibility and the achieved advancement can be easily transferred into different modalities.

4.3 Method

Definition. Conceptually, the diffusion-based field models sample from field distributions by reversing a gradual noising process. As shown in Fig. 4.1, in contrast to the data formulation of the conventional diffusion models [2] applied to the complete data like a whole image, diffusion-based field models apply the noising process to the sparse observation of the field, which is a kind of parametrized functional representation of data consisting of coordinate-signal pairs, *i.e.*, $f : \mathcal{M} \mapsto \mathcal{Y}$. Specifically, the sampling process begins with a coordinate-signal pair $(\mathbf{m}_i, \mathbf{y}_{(i,T)})$, where the coordinate comes from a field and the signal is a standard noise, and less-noisy signals $\mathbf{y}_{(i,T-1)}, \mathbf{y}_{(i,T-2)}, \ldots,$ are progressively generated until reaching the final clear signal $\mathbf{y}_{(i,0)}$, with $\mathbf{m}_i$ being constant. Diffusion Probabilistic Field (DPF) [131] is one of the recent representative diffusion-based field models. It parameterizes the denoising process with a transformer-based network $\epsilon_\theta(\cdot)$, which takes noisy coordinate-signal pairs as input and predicts the noise component ϵ of $\mathbf{y}_{(i,t)}$. The less-noisy signal $\mathbf{y}_{(i,t-1)}$ is then sampled from the noise component ϵ using a denoising process [2].

In practice, when handling low-resolution data consisting of N coordinate-signal pairs with DPF, the scoring network $\epsilon_\theta(\cdot)$ takes all pairs $\{(\mathbf{m}_i, \mathbf{y}_{(i,T)})\}$ as input at once. For

high-resolution data with a large number of coordinate-signal pairs that greatly exceed the modern GPU capacity, [131] uniformly sample a subset of pairs from the data as input. They subsequently condition the diffusion model on the other non-overlapping subset, referred to as *context pairs*. Specifically, the sampled pairs interact with the query pairs through cross-attention blocks. [131] show that the ratio between the context pairs and the sampling pairs is strongly related to the quality of the generated fields, and the quality decreases as the context pair ratio decreases. Due to the practical memory bottleneck, DPF can only support a maximum 64×64 resolution, let alone being extended to long context such as multi-frame video generation.

4.3.1 Diffusion Field Transformer

In order to scale diffusion-based field models for high-resolution, dynamic data generation, we build upon the recent DPF model [131] and address its limitations in preserving the local structure of fields, as it can hardly be captured when the uniformly sampled coordinate-signal pairs are too sparse. Specially, our method not only can preserve the local structure, but also introduce additional inductive biases for capturing the global geometry, such as text descriptions and past frames in autoregressive generation.

In order to preserve the local structure of fields, we propose a new view-wise sampling algorithm that samples local coordinate-signal pairs for better representing the local structure of fields. For instance, the algorithm samples the coordinate-signal pairs belonging to a single or several ($n \geqslant 1$; n denotes the number of views) views for video data, where a view corresponds to a single frame. It samples pairs belonging to a

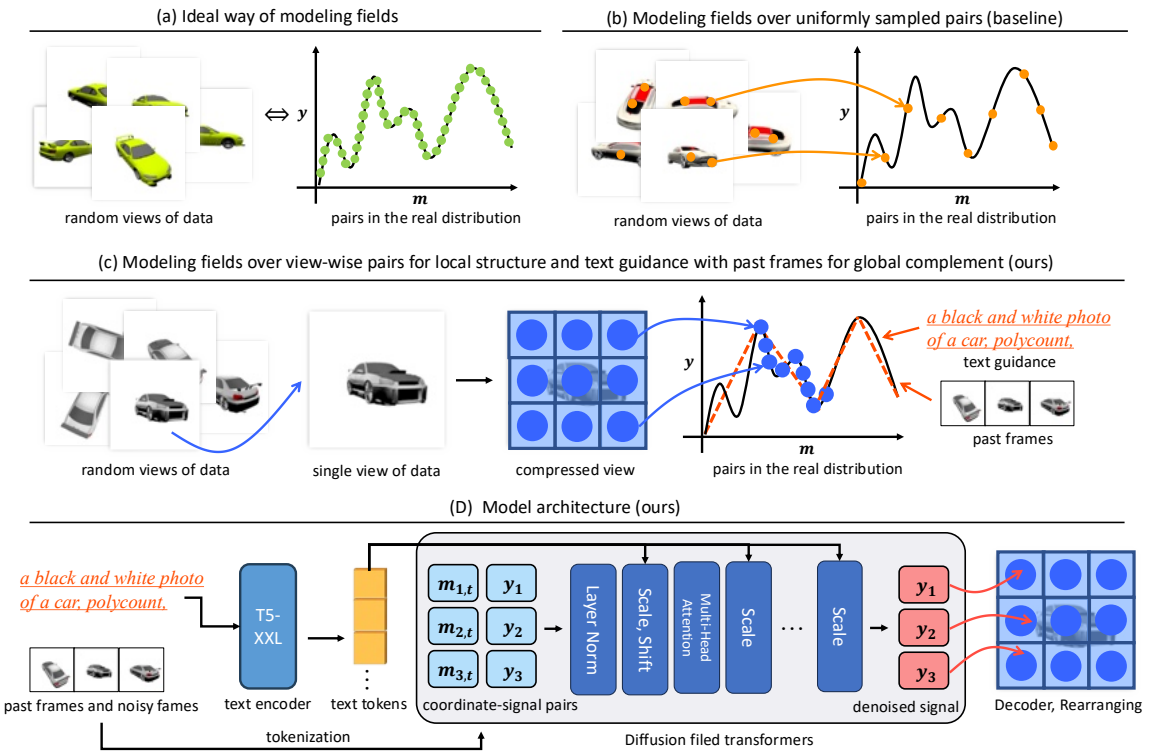


Figure 4.2: (a) Ideally, all pairs within a field (green points) should be used for training, but this is impractical due to memory limitations. (b) Previous methods uniformly sample a sparse set of pairs (orange points) to represent the field to mitigate memory limitations. (c) Compared to uniform sampling, our local sampling extracts high-fidelity pairs (blue points), better covering the local structure. The text prompt and past frames serve as an approximation to complement the global geometry. (d) Visualization of our sampling pipeline. Note that the input coordinates include the diffusion timesteps of each input frames.

single or several rendered images for 3D viewpoints, where a view corresponds to an image rendered at a specific camera pose. A view of an image is the image itself.

This approach restricts the number of interactions among pairs to be modeled and reduces the learning difficulty on high-resolution, dynamic data. Nevertheless, even a single high-resolution view , *e.g.*, in merely 128×128 resolution can still consist of 10K pairs, which in practice will very easily reach the memory bottleneck if we leverage a large portion of them at one time, and hence hinder scaling the model for generating high-resolution dynamic data.

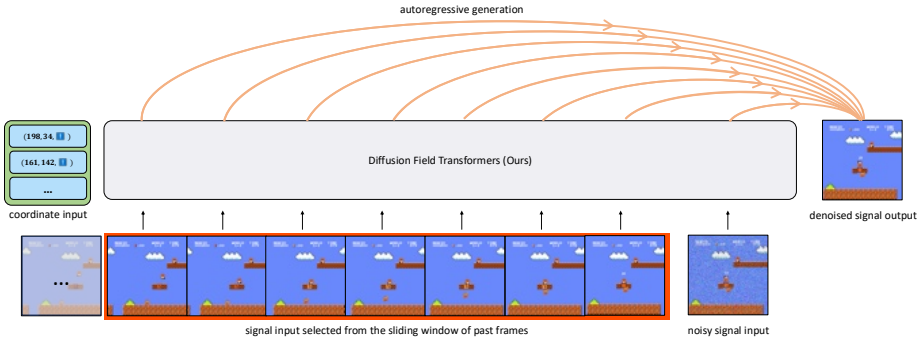


Figure 4.3: Autoregressive next-frame prediction. Our model takes past frames selected from a sliding window and next action coordinates, such as actions like jump or move, as input. It then generates the next frame, reflecting both the action and the long context of the past frames.

To address this issue, our method begins with increasing the signal resolution of coordinate-signal pairs and hence reducing memory usage in the score network. Specifically, we replace the signal space with a compressed latent space, and employ a more efficient network architecture that only contains decoders. This improvement in efficiency allows the modeling of interactions among pairs representing higher-resolution data while keeping the memory usage constrained. Based on this, one can then model the interactions of pairs within a single or several views of high-resolution data. The

overall diagram of the proposed sampling method can be found in Fig. 4.2.

View-wise Sampling. Based on the high-resolution signal and decoder-only network architecture, our method represents field distributions by using view-consistent coordinate-signal pairs, *i.e.*, collections of pairs that belong to a single or several ($n \geqslant 1$) views of the data, such as one or several frames in a video, and one or several viewpoints of a 3D geometry. In particular, take the spatial and temporal coordinates of a video in $H \times W$ resolution lasting for T frames as an example, for all coordinates $\{\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_i, \ldots, \mathbf{m}_{H\times W\times T}\}$, we randomly sample a consecutive sequence of length $H \times W$ that correspond to a single frame, *i.e.*, $\{\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_i, \ldots, \mathbf{m}_{H\times W}\}$. For data consisting of a large amount of views (*e.g.* $T >> 16$), we randomly sample n views (sequences of length $H \times W$), resulting in an $H \times W \times n$ sequence set. Accordingly, different from the transformers in previous works [131] that model interaction among all pairs across all views, ours only models the interaction among pairs that belongs to the same view, which reduces the complexity of field model by limiting the number of interactions to be learned.

4.3.2 Long-context Conditioning

To complement our effort in preserving local structures that may weaken global geometry learning, since the network only models the interaction of coordinate-signal pairs in the same view, we propose to supplement the learning with a long-context conditioning of the field, avoiding issues in cross-view consistency like worse spatial-temporal consistency between frames in video generation.

In particular, we propose to condition diffusion models on long-context such as

text-prompt and past frames related to the fields. Text-prompt can represent data in compact but highly expressive features [47, 66, 67], and serve as a low-rank approximation of data [140]. Past frames are especially useful in autoregressive generation, such as in game data. By conditioning diffusion models on long-context, we demonstrate that our method can capture the global geometry for generating long videos and game sequences.

Text-prompt for Cross-view Condition Consistency. In order to model the dependency variation between views belonging to the same field, *i.e.*, the global geometry of the field, we condition the diffusion model on the text embeddings of the field description or equivalent embeddings (*i.e.*, the language embedding of a single view in the CLIP latent space [140]). Our approach leverages the adaptive layer normalization layers in GANs [26, 93], and adapts them by modeling the statistics from the text embeddings of shape $Z \times D$. For pairs that make up a single view, we condition on their represented tokens $Z \times D$, (Z tokens of size D), by modulating them with the scale and shift parameters regressed from the text embeddings. For pairs $(T \times Z) \times D$ that make up multiple views, we condition on the view-level pairs by modulating feature in $Z \times D$ for each of the T views with the same scale and shift parameters. Specifically, each transformer blocks of our score network learns to predict statistic features β_c and γ_c from the text embeddings per channel. These statistic features then modulate the transformer features F_c as: $\text{adLNorm}(F_c|\beta_c, \gamma_c) = \text{Norm}(F_c) \cdot \beta_c + \beta_c$.

Past frames for Autoregressive Generation. Generating long videos and games can be formulated as autoregressive generation, where each frame depends only on past frames and current actions. In Figure. 4.3, we illustrate the input and output of

our model, where it conditions on the past T frames and generates the next $T+1$ frame. Additional inputs include coordinates consisting of spatiotemporal coordinates and one-hot encoded actions, such as *jump* and *move*, from the last frame. Due to memory constraints, the input past frames are limited to a fixed number of n past frames, acting as a sliding window for long-context modeling. The generation of the next n frames on the diffusion field can be simplified as

$$p\left(\mathbf{y}_1, \mathbf{y}_2 \ldots, \mathbf{y}_{n-1}, \mathbf{y}_{(n,t-1)}\right) = \prod_{i=1}^{n} p\left(\mathbf{y}_{(n,t-1)} \mid \mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{n-1}\right), \tag{4.1}$$

where $p(\cdot)$ represents the modeled signal probability conditioned on the past frames. Additional conditions also include coordinate inputs $(\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_n)$ and the diffusion timestep t. Empirically, we use the last 16 frames as the context length for game generation, and the last 8 frames as the context length for text-to-video generation.

The proposed autoregressive generation not only preserves global geometry of the data but also significantly improves efficiency in long-context generation. Typical autoregressive transformer models like GPT [141] depend on the number of generated tokens, as each new token is conditioned on all previously generated tokens. In contrast to GPT, our method achieves linear complexity with respect to the number of generated frames, similar to the parallel generation efficiency. Each new frame depends only on a fixed number of the most recently generated frames, where conditioning frames are updated in the sliding window. Our game generation maximizes this efficiency, enabling the stable generation of games with an infinite number of frames.

4.4 Experimental Results

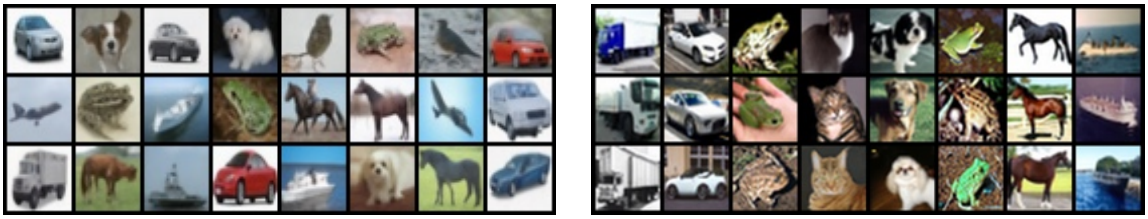
We demonstrate the effectiveness of our method on multiple modalities, including 2D image data on a spatial metric space $\mathbb{R}^2$, 3D video data on a spatial-temporal metric space $\mathbb{R}^3$, and 3D viewpoint data on a camera pose and intrinsic parameter metric space $\mathbb{R}^6$, game data on a action and spatial-temporal metric space $\mathbb{R}^4$, while the score network implementation remains identical across different modalities, except for the embedding size.

Experimental Details. In the interest of maintaining simplicity, we adhere to the methodology outlined by Dhariwal et al. [7] and utilize a 256-dimensional frequency embedding to encapsulate input denoising timesteps. This embedding is then refined through a two-layer Multilayer Perceptron (MLP) with Swish (SiLU) activation functions. Our model aligns with the size configuration of DiT-XL [29], which includes retaining the number of transformer blocks (*i.e.* 28), the hidden dimension size of each transformer block (*i.e.*, 1152), and the number of attention heads (*i.e.*, 16). Our model derives text embeddings employing T5-XXL [67], culminating in a fixed length token sequence (*i.e.*, 256) which matches the length of the noisy tokens. To further process each text embedding token, our model compresses them via a single layer MLP, which has a hidden dimension size identical to that of the transformer block. Our model uses classifier-free guidance in the backward process with a fixed scale of 8.5. To keep consistency with DiT-XL [29], we only applied guidance to the first three channels of each denoised token.

Model	CIFAR10 64×64		CelebV-Text 256×256×128			ShapeNet-Cars 128×128×251			
	FID (↓)	IS (↑)	FVD (↓)	FID (↓)	CLIPSIM (↑)	FID (↓)	LPIPS (↓)	PSNR (↑)	SSIM (↑)
Functa [142]	31.56	8.12	✗	✗	✗	80.30	0.183	N/A	N/A
GEM [137]	23.83	8.36	✗	✗	✗	✗	✗	✗	✗
DPF [131]	15.10	8.43	✗	✗	✗	43.83	0.158	18.6	0.81
DiT [29]	7.53	8.97	✗	✗	✗		✗	✗	✗
TFGAN [143]	✗	✗	571.34	784.93	0.154	✗	✗	✗	✗
MMVID [144]	✗	✗	109.25	82.55	0.174	✗	✗	✗	✗
MMVID-interp [144]	✗	✗	80.81	70.88	0.176	✗	✗	✗	✗
VDM [128]	✗	✗	81.44	90.28	0.162	✗	✗	✗	✗
CogVideo [145]	✗	✗	99.28	54.05	0.186	✗	✗	✗	✗
Latte [146]	✗	✗	67.97	39.69	0.201	✗	✗	✗	✗
EG3D-PTI [147]	✗	✗	✗	✗	✗	20.82	0.146	19.0	0.85
ViewFormer [148]	✗	✗	✗	✗	✗	27.23	0.150	19.0	0.83
pixelNeRF [149]	✗	✗	✗	✗	✗	65.83	0.146	23.2	0.90
Zero-1-to-3 [22]	✗	✗	✗	✗	✗	**17.901**	**0.093**	23.1	0.80
DiFT (Ours)	**7.29**	**9.31**	**42.03**	**24.33**	**0.220**	24.36	0.118	**23.9**	**0.90**

Table 4.1: Sample quality comparison with state-of-the-art field models and representative modality-specific models for each task. "✗" denotes that the method cannot be applied to the modality due to its design or impractical computational costs.

Generative Metrics. In video generation, we use FVD [115] to evaluate the video spatial-temporal coherency, FID [117] to evaluate the frame quality, and CLIPSIM [140] to evaluate relevance between the generated video and input text. As all metrics are sensitive to data scale during testing, we randomly select 2,048 videos from the test data and generate results as the "real" and "fake" part in our metric experiments. For FID, we uniformly sample 4 frames from each video and use a total of 8,192 images. For CLIPSIM, we calculate the average score across all frames. We use the "openai/clip-vit-large-patch14" model for extracting features in CLIPSIM calculation.



(a) DPF [131] **(b)** **DiFT** (ours)

Figure 4.4: Qualitative comparisons of domain-agnostic methods and ours on CIFAR-10. Our results show better visual quality with more details than the others, while being domain-agnostic as well.

Images. For image generation, we use the standard benchmark dataset, *i.e.*, CIFAR10 64×64 [150] as a sanity test, in order to compare with other domain-agnostic and domain-specific methods. For the low-resolution CIFAR10 dataset, we compare our method with the previous domain-agnostic methods including DPF [131] and GEM [137]. We report Fréchet Inception Distance (FID) [117] and Inception Score (IS) [116] or quantitative comparisons.

The experimental results can be found in Tab. 4.1. Specifically, DiFT outperforms all domain-agnostic models in the FID and IS metrics. The qualitative comparisons in Fig. 4.4 further demonstrate our method's superiority in images. Note that our method does not use text descriptions for ensuring a fair comparison. It simply learns to predict all coordinate-signal pairs of a single image during training without using additional text descriptions or embeddings.

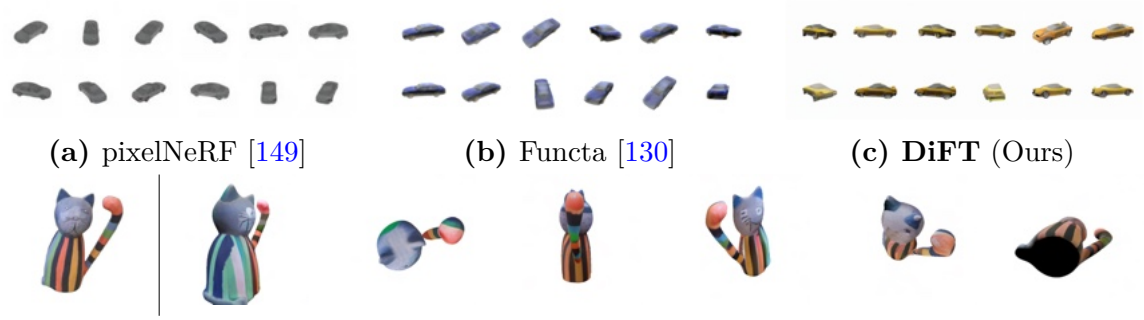
Videos. To show our model's capacity for more complex data, *i.e.*, high-resolution, dynamic video, we conduct experiments on the recent text-to-video benchmark: CelebV-Text 256×256×128 [151] (128 frames). As additional spatial and temporal coherence is enforced compared to images, video generation is relatively underexplored

(a) VDM [128] **(b)** CogVideo [145] **(c)** **DiFT** (Ours)

Figure 4.5: Qualitative comparisons between domain-specific text-to-video models and ours. Compared to VDM [128], our results are more continuous. Compared to CogVideo [145], our results feature more realistic textures. Please see https://transdif-web.pages.dev for the input prompt and video results.

by domain-agnostic methods. We compare our method with the representative domain-specific methods including TFGAN [143], MMVID [152], CogVideo [145], VDM [128], and Latte [146]. We report Fréchet Video Distance (FVD) [115], FID, and CLIPSIM [153], *i.e.*, the cosine similarity between the CLIP embeddings [140] of the generated images and the corresponding texts.

Our method achieves the comparable performance in both the video quality (FVD) and signal frame quality (FID) in Tab. 4.1, compared with the recent domain-specific text-to-video models. Moreover, our model learns more semantics as suggested by the CLIPSIM scores. The results show that our model, as a domain-*agnostic* method, can achieve a performance on par with domain-*specific* methods in modeling long-context. The qualitative comparisons in Fig. 4.5 further support our model in text-to-video generation compared with the recent state-of-the-art methods.



(a) pixelNeRF [149] **(b)** Functa [130] **(c)** **DiFT** (Ours)

(d) **DiFT** (our novel view generation results on the complex objects)

Figure 4.6: Qualitative comparisons between representative 3D novel view generation methods and ours. Our results demonstrate competitive quality without explicitly using 3D modeling.

3D novel views. We also evaluate our method on 3D novel view generation with the ShapeNet dataset [154]. Specifically, we use the "car" class of ShapeNet which involves 3514 different cars. Each car object has 50 random viewpoints, where each viewpoint is in 128 $\times$ 128 resolution. Unlike previous domain-agnostic methods [131, 137] that model 3D geometry over voxel grids at 64^3 resolution, we model over rendered camera views based on their corresponding camera poses and intrinsic parameters, similar to recent domain-specific methods [149, 155]. This approach allows us to extract more view-wise coordinate-signal pairs while voxel grids only have 6 views. We report our results in comparison with the state-of-the-art view-synthesis algorithms including pixelNeRF [149], viewFormer [148], EG3D-PTI [147], and Zero-1-to-3 [22]. Among the compared methods, only Zero-1-to-3 is a zero-shot generation approach, while the others are trained on the ShapeNet dataset, either by their authors or by us. Zero-1-to-3 is pretrained on large-scale data, making it robust to out-of-domain data. Note that our model performs one-shot novel view synthesis by conditioning on the text embedding of a random view. Compared to recent methods specifically

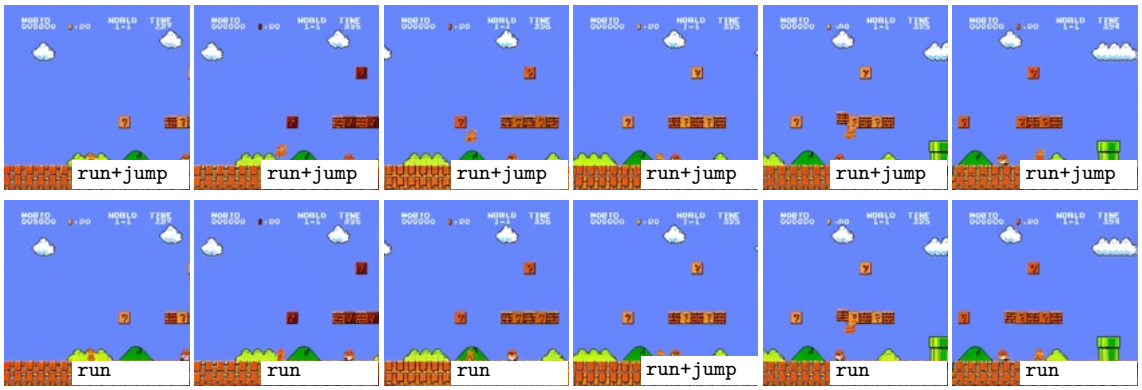


Figure 4.7: Visualization of our generated game (1/8 sampling rate at 50 frames), showcasing how our method generalizes to different actions within the same context. Each frame's action is labeled in the top-left corner. Please see https://transdif-web.pages.dev for videos.

designed for 3D modalities, our approach achieves higher fidelity metrics, such as PSNR and SSIM, while producing comparable scores in LPIPS. Although methods like EG3D-PTI and Zero-1-to-3, which directly fine-tune pretrained 2D image generation models like StyleGAN and Stable-Diffusion, achieve better FID scores, this metric prioritizes 2D visual quality. However, it does not strictly reflect 3D consistency, which limits its relevance for 3D evaluation. Concurrent 3D novel view generation methods, such as Eschernet [156], Wonder3D [157], and SV3D [158], have demonstrated their effectiveness in modeling sparse views of 3D objects. While our method may underperform in perceptual quality due to its modality-unified design, its unique ability to generate continuous, long-context 3D views offers a fresh perspective to the 3D modeling community.

Games. Game generation is an under-explored area and lacks data and benchmarks. We demonstrate the game generation capability of our method by showing the accuracy

PSNR (dB)	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
DPF [131]	24.00	21.97	20.87	20.66	✗	✗	✗	✗	✗	✗
DiFT (Ours)	44.30	43.96	43.87	44.16	42.92	42.20	42.42	42.51	42.07	42.22

Table 4.2: We demonstrate the long-context modeling capability of our model by showing its next-frame generation accuracy on game data, where a total of 100 frames are evaluated. ✗ denotes out-of-memory results when the model cannot handle such a long context.

of predicted frames compared with the frame of the real game when using the same action. Specially, we model the World 1-1 of Super Mario Bros (NES version) with a sliding window size of 16, and we test it with new actions for next-frame generation. Fig. 4.7 shows the visual results generated from two different actions starting from the same scene. Tab. 4.2 demonstrates our long-context modeling capability compared with the DPF, where ours performance loss is minor compared with DPF.

4.4.1 Ablations and Discussions

In this section, we demonstrate the effectiveness of each of our proposed components and analyze their contributions to the quality of the final result, as well as the computation cost. The quantitative text-to-video generation results under various settings are shown in Table 5.3.

Effect of text condition. To verify the effectiveness of the text condition for capturing the global geometry of the data, we use two additional settings. **(1)** The performance of our model when the text condition is removed is shown in the first row of Tab. 5.3. The worse FVD means that the text condition play a crucial role in preserving the global geometry, specifically the spatial-temporal coherence in videos.

Text	Cross-view consistent noise	Resolution	Training Views n	FVD (↓)	FID (↓)	CLIPSIM (↑)	MACs	Mems
✗	N/A	16.0	8	608.27	34.10	-	113.31G	15.34Gb
✓	✗	16.0	8	401.64	75.81	0.198	117.06G	15.34Gb
✓	✓	1.0	8	115.20	40.34	0.187	7.314T	22.99Gb
✓	✓	16.0	1	320.02	21.27	0.194	117.06G	15.34Gb
✓	✓	16.0	4	89.83	23.69	0.194	117.06G	15.34Gb
✓	✓	16.0	8	42.03	24.33	0.220	117.06G	15.34Gb

Table 4.3: Ablation analysis of the text-to-video results of our proposed method under different settings. All computation costs (MACs) and GPU memory usage (Mems) are estimated for generating a single view, regardless of the resolution, to ensure a fair comparison. The mark in the text column indicates whether a text prompt is used. The number in the resolution column denotes the usage of a latent encoder, where a resolution equal to 1 means the model is directly trained in pixel space.

(2) When the text condition is added, but not the cross-view consistent noise, the results can be found in the second row of Tab. 5.3. The FVD is slightly improved compared to the previous setting, but the FID is weakened due to underfitting against cross-view inconsistent noises. In contrast to our default setting, these results demonstrate the effectiveness of the view-consistent noise. Furthermore, we note that more detailed text descriptions can significantly improve the generated video quality.

Effect of number of views. We investigate the model performance change with varying number of views (n) for representing fields, as shown in the 2nd and 3rd rows of Tab. 5.3. Compared to the default setting of $n = 8$, reducing n to 1 leads to non-continuous frames and abrupt identity changes, as indicated by the low FVD. When n is increased to 4, the continuity between frames is improved, but still worse than $n = 8$ for the dynamics between frames. Thus, we can conclude that a larger number of views leads to a higher performance, along with a higher computation cost.

Comparison with Context Query Pairs. Even though context query pairs introduced by DPF [131] has been justified to achieve better performance than using latent space (which needs reconstruction training) in small models and low-resolution modalities, it is shown [131] to be impossible to largely reduce the memory footprint (by sampling less context pairs) while preserving its original modeling capability and performance. To scale up our model, we replace the context query pairs with latent space in our method. It can significantly reduce memory usage (*e.g.* using less than 2% pairs while maintaining a competitive performance) so that handling a larger model size becomes possible with high-resolution, long views. Based on these, the benefit of scaling using the latent space outweighs the potential performance loss led by the latent space, as backed by Tab. 4.1.

Comparison with Modality Unified Models. Our method shares the motivation of modality-unified models like SIREN and Functa for handling diverse data modalities but differs in complexity and scope. SIREN uses sinusoidal activations in MLPs to represent continuous signals, excelling in modeling structured data and solving mathematical problems like PDEs with high fidelity but is limited to simpler datasets due to its MLP architecture. In contrast, our diffusion transformer framework handles more diverse and complex data, integrating view-wise sampling for local structure and autoregressive generation for global consistency. Additionally, text and past frame conditioning enable DiFT to scale effectively to complex multi-modal tasks, making it more versatile for dynamic and high-dimensional datasets compared to SIREN's structured focus.

4.5 Limitations.

(1) Our method only applies to visual modalities interpretable by views. For modalities such as temperature manifold [159] where there is no "views" of such field, our method does not apply. As long as the data in the new domain (e.g., 3D dynamic scene and MRI) can be interpreted by views, our method can reuse the same latent autoencoder [127] without switching to domain-specific autoencoders. (2) Our method aligns with the standard practice outlined in DPF, using comparison methods with weights trained separately for each modality. While it performs exceptionally well on individual modalities, achieving strong performance across multiple modalities simultaneously is hindered by the inherent challenges of the multi-task problem. We believe our approach provides a solid foundation for the future versatile generative models.

Chapter 5
Conditional Diffusion Distillation

5.1 Introduction

Text-to-image diffusion models [9, 57, 160] trained on large-scale data [10, 11] have significantly dominated generative tasks by delivering impressive high-quality and diverse results. A newly emerging trend is to use the prior of pre-trained text-to-image models such latent diffusion models (LDMs) [9] to guide the generated results with external image conditions for image-to-image transformation tasks such as image manipulation, enhancement, or super-resolution [13, 14, 58, 161–163]. Among these transformation processes, the diffusion prior introduced by pre-trained models is shown to be capable of greatly promoting the visual quality of the conditional image generation results [22, 45, 164, 165].

However, diffusion models heavily rely on an iterative refinement process [1, 13, 43, 163, 166] that often demands a substantial number of iterations, which can be challenging to accomplish efficiently. Their reliance on the number of iterations further increases for high-resolution image synthesis. For instance, in state-of-the-art text-to-image latent diffusion models [9], achieving optimal visual quality typically requires $20-200$ sampling steps (function evaluations), even with advanced sampling methods [33, 34]. The slow sampling time significantly impedes practical applications of the aforementioned conditional diffusion models.

Recent efforts to accelerate diffusion sampling predominantly employ distillation methods [8, 37, 38]. These methods achieve significantly faster sampling, completing the process in just 4−8 steps, with only a marginal decrease in generative performance. Very recent works [27, 167] show that these strategies are even applicable for distilling pre-trained large-scale text-to-image diffusion models.

A very common application scenario is to incorporate new conditions into these distilled diffusion models, such as using low-resolution images for super-resoltion [43], or instruction-tuning for image editing [164], where the most straightforward way is to directly finetune the distilled text-to-image pre-trained model with new conditional data. An alternative common approach [167] is to first finetune the diffusion model with the new conditional data, then conducting distillation on the already-finetuned conditional model. While these two methods have been demonstrated to accelerate sampling, each has distinct disadvantages in terms of result quality and cross-task flexibility, as discussed below.

In this paper, we introduce a new algorithm for **Co**nditional **Di**stillation which we call **CoDi** for efficiently adding new controls into distilled models. Unlike previous distillation methods that rely on finetuning, our method directly distills a diffusion model from a text-to-image pretraining (*e.g.*, StableDiffusion) and ends with a fully distilled conditional diffusion model. As depicted in Figure 9.2, our distilled model is capable of predicting high-quality results in just $1-4$ sampling steps.

By design, our method eliminates the need for the original text-to-image data [11, 168], a requirement in previous distillation methods (*i.e.*, those that first distill

the unconditional text-to-image model), thereby making our method more practical. Additionally, our formulation avoids sacrificing the diffusion prior in the pre-trained model during finetuning, a common drawback in the first stage of the finetuning-first procedure. Our extensive experiments show that our CoDi outperforms previous distillation methods in both visual quality and quantitative metrics, particularly when operating under the same sampling time.

Parameter-efficient distillation methods are a relatively understudied area. We demonstrate that our method also enables a new **P**arameter-**E**fficient distillation paradigm (**PE-CoDi**). It can transform an unconditional diffusion model to conditional tasks by incorporating a small number of additional learnable parameters. Specifically, our formulation allows for integration with various existing parameter-efficient tuning algorithms, *e.g.*, ControlNet [162]. We show that our distillation process that integrates the ControlNet adapter can efficiently preserve the generative prior in pretraining while adapting the model to new conditioned data. This new paradigm significantly improves the practicality of different conditional tasks.

Our contributions are summarized as follows:

- We propose a new method for image and image-text conditioned generation. It can derive a conditional diffusion model from pretrained text-to-image LDMs for generating high-quality results in only a few sampling steps.
- The proposed method's efficiency and effectiveness arise from a non-trivial consistency between the model's predictions at different time steps. Enforcing this consistency through learning enables the simultaneous reduction of required

sampling steps and the integration of new conditions into the model.

- We introduce the first parameter-efficient distillation mechanism that can produce compelling results in just a few steps, while requiring only a small number of additional parameters compared with the pretrained LDMs.

5.2 Related Work

Diffusion Distillation. To reduce the sampling time of diffusion models, Luhman et al. [37] proposed to learn a single-step student model from the output of the original (teacher) model using multiple sampling steps. However, this method requires to run the full inference with many sampling steps during training which make it poorly scalable. Inspired by this, Progressive Distillation [8] and its variants, including Guided Distillation [167] and SnapFusion [27], use a progressive learning scheme for improving the learning efficiency. A student model learns to predict the output of two steps of the teacher model in one step. Then, the teacher model is replaced by the student model, and the procedure is repeated to progressively distill the mode by halving the number of required steps. We demonstrate our method by comparing these methods on the conditional generation tasks. We note that strategies like classifier-free guidance distillation [27, 167], or the different adopted sampling techniques [169, 170], are orthogonal to our method, and they could be incorporated in our formulation. Even though some concurrent works [171, 172] find that tasks like super-resolution requires less sampling steps, we later show that distilling pre-trained diffusion models can still improve the performance in such restoration tasks.

Consistency Distillation. A Consistency Model is a single-step generative approach

distilled from a pre-trained diffusion model [38]. The learning is achieved by enforcing a self-consistency in the predicted signal space. Based on this idea, following work [40, 173–175] have focus on improving the training techniques. However, learning consistency models for conditional generation has yet to be thoroughly studied. In this paper, we compare our method against a baseline approach that enforces self-consistency in an already fine-tuned conditional diffusion model. Our results demonstrate that our conditional distilled model outperforms the baseline approach, indicating the effectiveness of our proposed distillation strategy.

Diffusion Models Adaptations. Leveraging the knowledge of pre-trained models for new tasks, known as model adaptation, has gained significant traction in NLP and computer vision domains. This approach utilizes model adapters [16, 176–180] and HyperNetworks [181, 182] to effectively adapt pre-trained models to new domains and tasks. In the context of diffusion models, model adapters have been successfully employed to incorporate new conditions into pre-trained models [162, 183]. Our proposed method draws inspiration from these approaches and introduces a novel application of model adapters: distilling the sampling steps of diffusion models. Compared to fine-tuning the entire model [8], our method offers enhanced efficiency and flexibility. It enables the adaptation of multiple tasks using the same backbone model.

5.3 Methods

5.3.1 Consistency Models

To accelerate inference, [38] introduced the idea of consistency models. Let $s_\theta(\cdot, t)$ be a pre-trained diffusion model trained on data $\mathbf{x} \sim \mathcal{O}_{data}$. Then, a consistency function $f_\phi(\mathbf{z}_t, t)$ should satisfy that [38] where $f_\phi(\mathbf{x}, 0) = \mathbf{x}$ and

$$f_\phi(\mathbf{z}_t, t) = f_\phi(\mathbf{z}_{t'}, t'), \ \forall t, t' \in [0, T], \tag{5.1}$$

where $\{\mathbf{z}_t\}_{t\in[0,T]}$ is the solution trajectory of the probability flow ODE (PF-ODE) (equation 1.7). A boundary condition, *i.e.*, $f_\phi(\mathbf{x}, 0) = \mathbf{x}$ is parameterized with skip connections for ensuring continuous properties similar as done in previous works [33, 38, 184]:

$$F_\phi(\mathbf{z}_t, t) = c_{\text{skip}}(t)\mathbf{x} + c_{\text{out}}(t) f_\phi(\mathbf{z}_t, t), \tag{5.2}$$

where $c_{\text{skip}}(0) = 1$, $c_{\text{out}}(0) = 0$. In practice, $f_\phi(\mathbf{z}_t, t)$ is usually a denoising network that is distilled from a pre-trained diffusion model. We later show that we can replace the frozen PF-ODE with the distillation network and thus fit the PF-ODE for new conditional data during distillation.

5.3.2 From Unconditional to Conditional

In order to utilize the image generation prior encapsulated by the pre-trained unconditional[1] diffusion model, we first propose to adapt the unconditional diffusion model

[1]The discussed unconditional models include text-conditioned image generation models, *e.g.*, StableDiffusion [9] and Imagen [160], which are only conditioned on text prompts.

into a conditional version for the conditional data $(\mathbf{x}, c) \sim p_{\text{data}}$. Similar to the zero initialization technique used by controllable generation [54, 162], our method adapts the unconditional pre-trained architecture by using an additional conditional encoder.

To elaborate, we take the widely used U-Net as the diffusion network. Let us introduce the conditional-module by duplicating the encoder layers of the pretrained network. Then, let $\boldsymbol{h}_\theta(\cdot)$ be the encoder features of the pretrained network, and $\boldsymbol{h}_\eta(\cdot)$ be the features on the additional conditional encoder. We define the new encoder features of the adapted model by

$$\boldsymbol{h}_\theta(\mathbf{z}_t)' = (1-\mu)\boldsymbol{h}_\theta(\mathbf{z}_t) + \mu\boldsymbol{h}_\eta(c), \tag{5.3}$$

where μ is a learnable scalar parameter, initialized to $\mu = 0$. Starting from this zero initialization, we can adapt the unconditional architecture into a conditional one. Thus, our conditional diffusion model $\hat{\mathbf{w}}_\theta(\mathbf{z}_t, c, t)$ is the result of adapting the pre-trained unconditional diffusion model $\hat{\mathbf{v}}_\theta(\mathbf{z}_t, t)$ with the conditional features $\boldsymbol{h}_\eta(c)$.

5.3.3 A New Conditional Diffusion Consistency

Our core idea is to optimize the adapted conditional diffusion model $\hat{\mathbf{w}}_\theta(\mathbf{z}_t, c, t)$ from $\hat{\mathbf{v}}_\theta(\mathbf{z}_t, t)$, so it satisfies a conditional diffusion consistency property:

$$\hat{\mathbf{w}}_\theta(\mathbf{z}_t, c, t) = \hat{\mathbf{w}}_\theta(\hat{\mathbf{z}}_s, c, s), \ \forall t, s \in [0, T], \tag{5.4}$$

where the $\hat{\mathbf{z}}_s$ belong to the probability flow ODE (equation 1.7) of the adapted model. Note that this consistency property differs from the one in consistency models [38] in

the probability flow ODE model used for sampling $\hat{\mathbf{z}}_s$ and the consistency loss space. To motivate this formulation, let us introduce the following general remark.

Remark 5.1. If a diffusion model, parameterized by $\hat{\mathbf{v}}_\theta(\mathbf{z}_t, t)$, satisfies the self-consistency property (equation 5.1) on the noise prediction $\hat{\epsilon}_\theta(\mathbf{z}_t, t) = \alpha_t \hat{\mathbf{v}}_\theta(\mathbf{z}_t, t) + \sigma_t \mathbf{z}_t$, then it also satisfies the self-consistency property on the signal prediction $\hat{\mathbf{x}}_\theta(\mathbf{z}_t, t) = \alpha_t \mathbf{z}_t - \sigma_t \hat{\mathbf{v}}_\theta(\mathbf{z}_t, t)$.

The proof is a direct consequence of change of variables from noise into signal and is given in Appendix. Based on this general remark, we claim that we can optimize the conditional diffusion model $\hat{\mathbf{w}}_\theta(\mathbf{z}_t, c, t)$ to jointly learn to enforce the self-consistency property on the noise prediction $\hat{\epsilon}_\theta(\mathbf{z}_t, c, t)$ and the new conditional generation $(\mathbf{x}, c) \sim p_{\text{data}}$ with the signal prediction $\hat{\mathbf{x}}_\theta(\mathbf{z}_t, c, t)$. We then impose the boundary condition for consistency distillation by parameterizing the noise prediction $\hat{\epsilon}_\theta(\mathbf{z}_t, c, t)$ with the same skip connections of equation 5.2.

Prediction of $\hat{\mathbf{z}}_s$. In the distillation process given by equation 7.11, the latent variable $\hat{\mathbf{z}}_s$ is achieved by running one step of a numerical ODE solver. Consistency models [38] solve the ODE using the Euler solver, while progressive distillation [8] and guided distillation [167] run two steps using the DDIM sampler (equation 1.8).

We propose an alternative prediction for $\hat{\mathbf{z}}_s$ that leverages the adapted diffusion model, $\hat{\mathbf{x}}_\theta(\mathbf{z}_t, c, t)$, as opposed to the conventional frozen pretraining one. We then sample $\hat{\mathbf{z}}_s$ in the adapted diffusion model PF-ODE by

$$\hat{\mathbf{z}}_s = \alpha_s \hat{\mathbf{x}}_\theta(\mathbf{z}_t, c, t) + \sigma_s \epsilon, \text{ with } \mathbf{z}_t = \alpha_t \mathbf{x} + \sigma_t \epsilon, \tag{5.5}$$

and $\epsilon \sim \mathcal{N}(0, \mathbf{I})$. This novel formulation effectively harmonizes the conflicting optimization directions between consistency distillation from pretrained data and conditional guidance from conditional data.

Training scheme. Inspired by consistency models [38], we use the exponential moving averaged parameters θ^- as the target network for stabilize training. Then, we can minimize the following training loss for conditional distillation:

$$\mathcal{L}(\theta) := \mathbb{E}[d_\epsilon(\hat{\epsilon}_{\theta^-}(\hat{\mathbf{z}}_s, s, c), \hat{\epsilon}_\theta(\mathbf{z}_t, t, c))) + d_{\mathbf{x}}(\mathbf{x}, \hat{\mathbf{x}}_\theta(\mathbf{z}_t, t, c)]. \tag{5.6}$$

where $d_\epsilon(\cdot, \cdot)$ and $d_{\mathbf{x}}(\cdot, \cdot)$ are two distance functions to measure difference in the noise space and in the signal space respectively. Note that the total loss is a balance between the conditional guidance given by $d_{\mathbf{x}}$, and the noise self-consistency property given by d_ϵ.

The overall conditional distillation algorithm is presented in Appendix. In the following, we will detail how we sample $\hat{\mathbf{z}}_s$ and discuss other relevant hyperparameters in our method (e.g., $d_{\mathbf{x}}$).

5.3.4 Effects of Different Conditional Guidance

To finetune the adapted diffusion model with the new conditional data, our conditional diffusion distillation loss in equation 7.11 penalizes the difference between the predicted signal $\hat{\mathbf{x}}_\theta(\mathbf{z}_t, c, t)$ and the corresponding image $\mathbf{x}$ with a distance function $d_{\mathbf{x}}(\cdot, \cdot)$ for distillation learning.

Figure 5.1: Sampled results between distilled models learned with alternative conditional guidance. Left curves shows the quantitative performance between the LPIPS and FID in $\{1, 2, 4, 8\}$ steps. Right part show the visual results where each result comes from the 1 sampling step (top) or 4 sampling steps (bottom). The distance function from the left to right is $\|\mathbf{x} - \mathbb{E}(\mathbb{D}(\hat{\mathbf{x}}_\theta(\mathbf{z}_t, c)))\|_2^2$, $\|\mathbb{D}(\mathbf{x}) - \mathbb{D}(\hat{\mathbf{x}}_\theta(\mathbf{z}_t, c))\|_2^2$, $F_{\text{lpips}}(\mathbb{D}(\mathbf{x}), \mathbb{D}(\hat{\mathbf{x}}_\theta(\mathbf{z}_t, c))$, and our default $\|\mathbf{x} - \hat{\mathbf{x}}_\theta(\mathbf{z}_t)\|_2^2$, respectively.

Here we investigate the impact of the distance function $d_{\mathbf{x}}(\cdot, \cdot)$ in the conditional guidance. According to both qualitative and quantitative results, shown in Figure 5.1, different distance functions lead to different behaviours when doing multi-step sampling (inference). If $d_{\mathbf{x}} = \|\cdot\|^2$ in the pixel space or the encoded space, *i.e.*, $\|\mathbf{x} - \mathbb{E}(\mathbb{D}(\hat{\mathbf{x}}_\theta(\mathbf{z}_t, c, t)))\|_2^2$ and $\|\mathbb{D}(\mathbf{x}) - \mathbb{D}(\hat{\mathbf{x}}_\theta(\mathbf{z}_t, c, t))\|_2^2$, multi-step sampling leads to more smooth and blurry results. If instead we adopt a perceptual distance in the pixel space, *i.e.*, $\mathcal{F}_{\text{lpips}}(\mathbb{D}(\mathbf{x}), \mathbb{D}(\hat{\mathbf{x}}_\theta(\mathbf{z}_t, c, t)))$, the iterative refinement in the multi-step sampling leads to over-saturated results. Overall, by default we adopted the ℓ_2 distance in the latent space since it leads to better visual quality and achieve the optimal FID with 4 sampling steps in Figure 5.1.

5.3.5 Parameter-Efficient Conditional Distillation

Our method offers the flexibility to selectively update parameters pertinent to distillation and conditional finetuning, leaving the remaining parameters frozen. This leads us to introduce a new fashion of parameter-efficient conditional distillation, aiming at

unifying the distillation process across commonly-used parameter-efficient diffusion model finetuning, including ControlNet [162], T2I-Adapter [183], etc. We highlight the ControlNet architecture illustrated in Figure 5.2 as an example. This model duplicates the encoder part of the denoising network, highlighted in the green blocks, as the condition-related parameters. Our method can then optimizes the conditional guidance and the consistency by only updating the duplicated encoder.

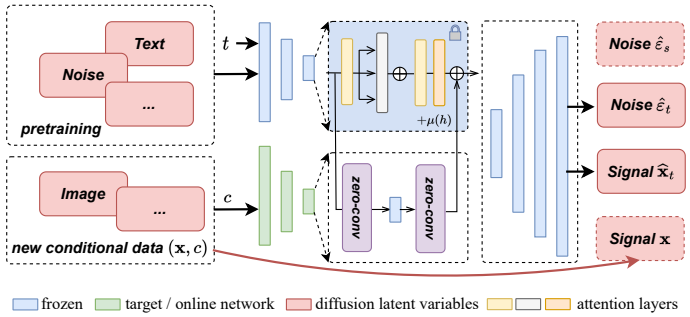


Figure 5.2: Network architecture illustration of our parameter-efficient conditional distillation framework.

5.4 Experiments

We demonstrate the efficacy of our method on representative conditional generation tasks, including, real-world super-resolution [185], depth-to-image generation [162], and instructed image editing [164]. We utilize a pre-trained text-to-image latent diffusion models[2] and conduct conditional distillation directly from the model. Each of the compared methods, including the text-to-image pretraining, was independently trained for 8 days on 64 TPU-v4 pods.

[2]We base our work on a version of Latent Diffusion Model trained on internal text-to-image data. It is comparable with StableDiffusion v1.4.

	CM-I	CM-II	GD-I	GD-II	Ours
stage-1	distill	finetune	distill	finetune	conditional distill
stage-2	finetune	distill	finetune	distill	n.a.
	✗	✓	✓	✓	✓

Table 5.1: We compare previous distillation methods by applying them to a T2I LDMs and then finetuning the distilled models (CM-X), and also distillation methods by directly applying them into the finetuned LDMs (GD-X). Since fine-tuning a distilled consistency model within the existing diffusion loss framework is not feasible, we excluded it from our comparison.

5.4.1 Results

Baselines. We compare our method with two previous SOTA diffusion distillation methods, *i.e.*, consistency models (CM) [38] and guided-distillation (GD) [167]. We implement CM with ControlNet without freezing denoising U-Net, which leads to the same network architecture and the same number of parameters as ours. For completeness, we consider two different ways of applying the tested distillation techniques, by first making the model conditional (fine-tuning first), or by first distilling the model and then making it conditional (distill first). A summary of the tested configurations is shown in Table 5.1. Additionally, we compare our method to recently introduced fast ODE solvers, including DPM-Solver [34] and DPM-Solver++ [46].

Real-world super-resolution. We evaluate our method on the challenging real-world super-resolution task, where the degradation is simulated using Real-ESRGAN pipeline [56]. Following StablSR [186], we compare all methods on 3,000 randomly degraded image pairs. The quantitative performance is shown in Table 5.2. The results demonstrate that our distilled method leads to a significant better performance than other distillation techniques. Our method achieves better results than fine-tuned

Super-resolution (DF2K)

Sampling Steps	Methods	FID ↓	LPIPS ↓
1 step	RealESRGAN [185]	37.640	0.3112
200 steps	StableSR [186]	24.440	0.3114
4 steps	DiffIR [171]	31.719	0.3088
4 steps	ControlNet [162]	34.56	0.3381
250 steps	LDMs [9]	19.200	0.2639
50 steps	LDMs [9]	19.231	0.2603
20 steps	LDMs [9]	20.510	0.2627
8 steps	LDMs [9]	24.493	0.2789
6 steps	LDMs [9]	26.338	0.2873
4 steps	LDMs [9]	29.266	0.3014
4 steps	+ DPM Solve [34]	28.936	0.3077
4 steps	+ DPM Solver++ [46]	28.937	0.3073
	GD-I [167]	27.806	0.3202
	GD-II [167]	23.675	0.2796
	CM-II (frozen) [38]	28.088	0.3192
	CM-II [38]	27.810	0.3172
4 steps	**PE-CoDi** (Ours)	25.214	0.2941
	CoDi (Ours)	19.637	0.2656

Inpainting (ImageNet)

Sampling Steps	Methods	FID	LPIPS
1000 steps	Palette [166]	13.151	-
250 steps	Repaint [187]	-	0.2827
50 steps	ControlNet [162]	14.895	0.2260
4 steps	ControlNet [9]	20.205	0.2635
	+ DPM Solver++ [46]	19.941	0.2644
	CM-II [38]	17.710	0.2580
	GD-II [167]	15.950	0.2452
4 steps	**PE-CoDi** (Ours)	14.700	0.2231

Text-guided Depth-to-image (WebLI)

Sampling Steps	Methods	FID	CLIP
250 steps	ControlNet [162]	20.884	0.2910
4 steps	ControlNet [162]	29.780	0.2854
	+ DPM Solver++ [46]	32.208	0.2834
	CM-II [38]	27.640	0.2869
	GD-II [167]	26.513	0.2870
4 steps	**PE-CoDi** (Ours)	23.047	0.2874

Table 5.2: Quantitative performance comparisons between the baselines and our methods. Our model can achieve comparable performance in 4 steps than models sampled in 250 steps. The 4-step sampling results of our parameters-efficient distillation (PE-CoDi) is comparable with the original 8-step sampling results, while PE-CoDi doesn't sacrifice the original generative performance with frozen backbone.

diffusion models that requires 50× more sampling setps. Compared with the distilled model by applying the guided-distillation, our model outperforms it both quantitatively and qualitatively. The visual comparison presented in Figure. 5.3 also demonstrates the superiority of our method.

Inpainting. Similar to the above super-resolution comparisons, we demonstrate our method on the inpainting task that conditioned on the masked image, as the quantitative performance shown in Table 5.2. Similar to Palette [166], we apply random masks into ImageNet data [188] for both training and testing. Note that we conduct experiments on the up-scaled images in a 512×512 resolution, which is different than Palette in 256×256 resolution. Even though we evaluate their results in the same resoltuion, their number can only be used for reference.

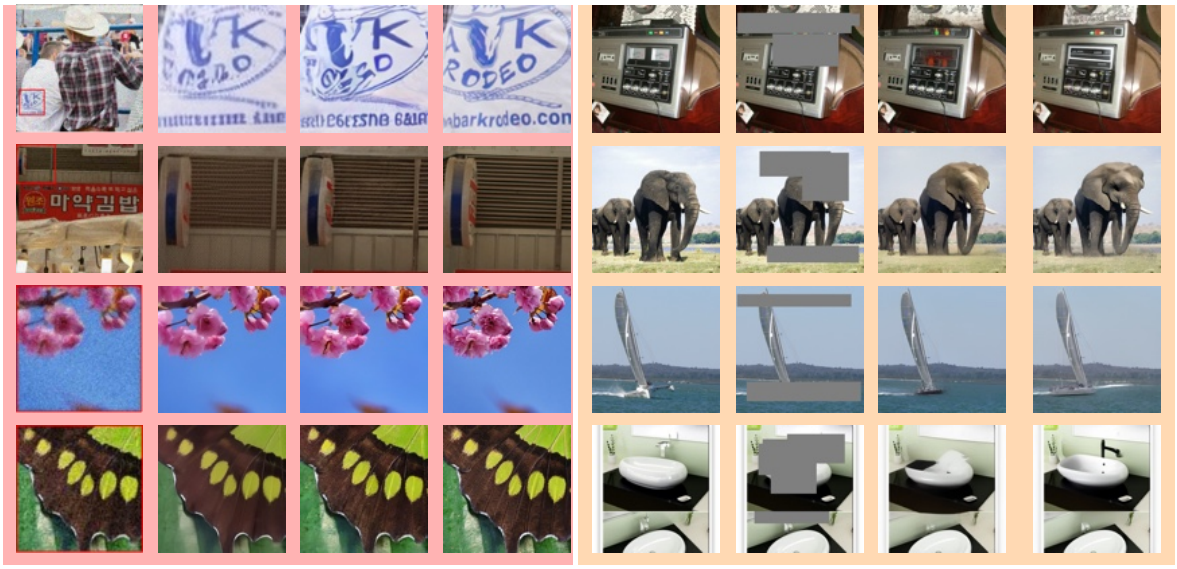


Figure 5.3: We show the results sampled in 4 steps by different models. Samples generated according to the low-resolution images (left) and masks (right) respectively. Please see our supplement for many more examples such as visual comparisons with the other methods.

Depth-to-image generation. In order to demonstrate the generality of our method on less informative conditions, we apply our method in depth-to-image generation. The task is usually conducted in parameter-efficient diffusion model finetuning [162, 183], which can demonstrate the capability of utilizing text-to-image generation priors. As Figure 5.4 illustrated, our distilled model from the unconditional pretraining can effectively utilize the less informative conditions and generate matched images with more details.

Instructed image editing. To demonstrate our conditional distillation capability on text-to-image generation, here we apply our method on text-instructed image editing data [164] and compare our conditional distilled model with the InstructPix2Pix (IP2P)

(a) Depth **(b)** ControlNet **(c) CoDi (Ours)**

Figure 5.4: Samples generated according to the depth image (left) from ControlNet sampled in 4 steps (middle), and ours from the unconditional pretraining sampled in 4 steps (right). Please see our supplement for many more examples.

model. As the results shown in Figure 5.5, our single-step sampling result can achieve comparable visual quality to 200 steps of the IP2P model. We experimentally find only small visual difference between the results from our single-step sampling and the 200 steps sampling. We believe this suggests that the effect of the conditional guidance on distillation correlates with the similarity between the conditions and the target data, further demonstrating the effectiveness of our method.

5.4.2 Ablations

Here we compare the performance of the aforementioned designs in our conditional distillation framework. Specifically we focus on the representative conditional generation task *i.e.*, real-world super-resolution [185] that conditions on the low-resolution, noisy, blurry images.

Network architecture and distillation process. To eliminate the impact of the architecture change, we compare our method with a baseline given by adding a ControlNet module trained on super-resolution without freezing the UNet. As Table 5.3 shows, simply adopting a ControlNet module for super-resolution has negligible impact

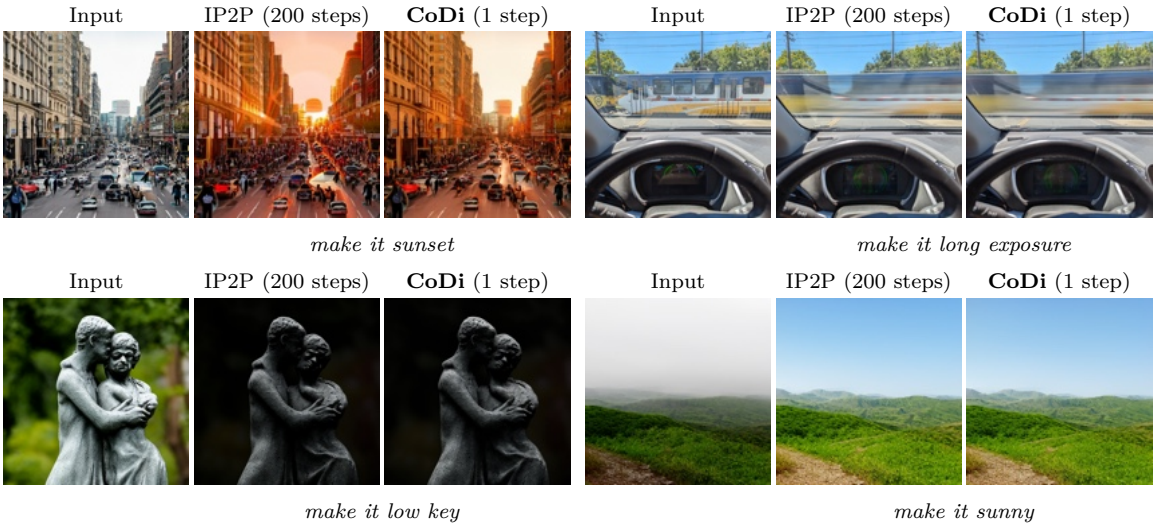


Figure 5.5: Generated edited image according to the input image and the instruction (bottom) from Instructed Pix2Pix (IP2P) sampled in 200 steps and ours sampled in 1 step. Please see our supplement for many more examples.

Methods	Params	FID	LPIPS
LDMs	865M	29.266	0.3014
+ ControlNet	1.22B	28.951	0.3049
PE-CoDi (Ours)	364M	25.214	0.2941
CoDi (Ours)	1.22B	19.637	0.2656
- distilling PF-ODE	1.22B	20.307	0.2733
- noise-consistency	1.22B	25.728	0.3252

Table 5.3: Impact of the network architecture and conditional distillation.

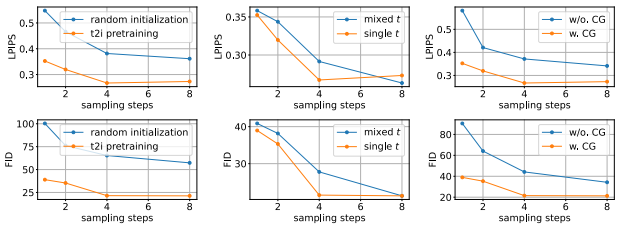


Figure 5.6: Ablations between alternative settings of our method.

on the performance. To evaluate the proposed conditional diffusion consistency, we removed the noise consistency term (equation 7.11) and employed the training model in the PF-ODE instead of the frozen one as used in [38] formulation. As shown in Table 5.3, adopting the distillation model PF-ODE and noise-space consistency have positive effects on the final results. These comparisons demonstrate the superiority of our method without network architecture effects.

Pretraining. To validate the effectiveness of leveraging pretraining in our model, we

compare the results of random initialization with initialization from the pre-trained text-to-image model. As shown in Figure 5.6, our method outperforms the random initialized counterpart by a large margin, thereby confirming that our strategy indeed utilizes the advantages of pretraining during distillation instead of simply learning from scratch.

Sampling of $\mathbf{z}_t$. We empirically show that the way of sampling $\mathbf{z}_t$ plays a crucial role in the distillation learning process. Compared with the previous protocol [8, 167] that samples $\mathbf{z}_t$ in different time t in a single batch, we show that using a consistent time t across different samples in a single batch leads to a better performance in our targeted 1-4 steps. As the comparisons shown in Figure 5.6, the model trained with a single time t (in a single batch) achieves better performance in both the visual quality (*i.e.*, FID) and the accuracy (*i.e.*, LPIPS) when the number of evaluations is increasing during inference.

Conditional guidance. In order to demonstrate the importance of our proposed conditional guidance (CG) for distillation, which is claimed to be capable of regularizing the distillation process during training, we conduct comparisons between the setting of using the conditional guidance as $r = \|\mathbf{x} - \hat{\mathbf{x}}_\theta(\mathbf{z}_t, c)\|_2^2$ and not using as $r = 0$. As the result shown in Figure 5.6, the conditional guidance improves both the fidelity of the generated results and visual quality. We further observed that the distillation process will converge toward over-saturated direction without CG, which thus lower the FID metric. In contrast, our model avoids such a local minimum by using the proposed guidance loss.

Chapter 6
Looking Through Turbulence by Inverting GANs

6.1 Introduction

Images collected by long range imaging systems often suffer from atmospheric turbulence that introduces blur and geometric deformation on the collected imagery (see Fig. 6.1). The degraded images under atmospheric turbulence can significantly degrade the performance of computer vision systems, including surveillance, detection, and recognition. Mitigating the effect of turbulence with modern deep learning techniques can be difficult, because collecting turbulence-degraded data is labor-intensive and expensive. One can rely on various turbulence simulation methods to synthesize degraded data for training deep networks, but as was shown in [189] existing turbulence simulation methods do not produce realistic looking degraded data. Therefore, a large gap still exists in the quality of restored images between turbulence mitigation and general image restoration.

Compared with the aforementioned data-driven methods, GAN inversion relies on generative priors and has shown to produce excellent results in various applications, including *image super-resolution* [190, 191], *image colorization* [192], *blind face restoration* [193, 194], and *general tasks* [195, 196]. Based on a well-trained GAN, GAN inversion first inverts the degraded image in its learned data distribution. Clear images with a similar fidelity as the degraded image can then be sampled from the distribution.

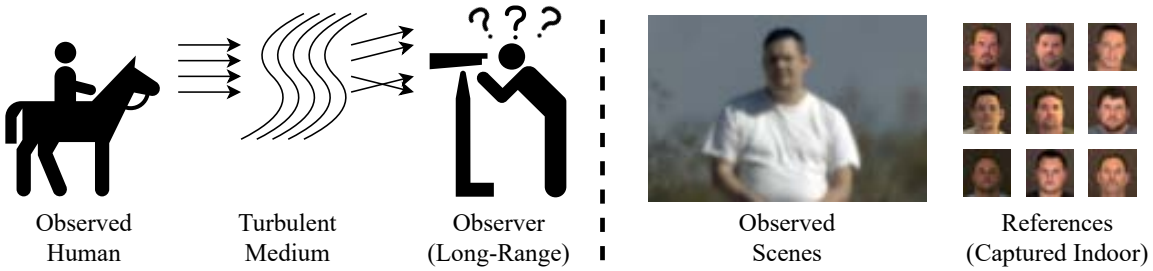


Figure 6.1: Left: Images collected by long-range imaging systems are degraded by atmospheric turbulence. In long-range surveillance applications one has to match an image collected in such turbulent medium and compare it with images collected in short-range and indoor conditions.

Despite photo-realistic results that can be acquired from well-trained GANs, inverting degraded images is nontrivial. Among two major GAN inversion ways, i.e., the iteration-based method [190, 195–198], and learning-based method [191, 193, 194, 199–201], the latter can achieve the most compelling efficiency in a single forward pass, and it does not require the physical degradation model. However, as the degradation complexity increases, unfaithful realism and unnatural details often appear in its inverted results. We show that these issues become more fatal in atmospheric turbulence degradation, and the inverted facial images can hardly be recognized by modern biometric systems [202].

In the GAN inversion literature, many methods consist of a learnable latent code encoder and a well-trained GAN [93, 203] as *Generative Embedding Network (GEN)*. A GEN first predicts the latent code of the degraded image with the encoder, and it then projects the latent code to the clear image with a well-trained GAN. During training, GEN learns with the GAN loss and an additional pixel-wise loss between the restored images and the target images or between the restored image features and

the target image features. The two loss functions aim to preserve the realism of the restored results and their identity. However, balancing the effects between adversarial learning and identity preservation is not easy, and the used pixel-wise loss like $\mathcal{L}_2$ is not expressive enough at capturing facial identity differences [204]. Under such a scenario, the motivation of our method comes from the following aspects.

A coherent probability-based loss function benefits GEN learning. Inspired by the contextual loss [205], which minimizes the probability distribution difference like GANs, we show that the unfaithful issue can be addressed by probability-based loss functions. The contextual loss is derived from the Jensen–Shannon (JS) divergence between two images [206]. Similarly, GANs have been shown to optimize the JS divergence in its original formulation [207]. We show that replacing the pixel-wise loss with the contextual loss during learning achieves around 1.77% performance improvement in the identity preservation. Therefore, we believe that the difference in the loss functions leads to unfaithful results. In contrast, to a coherence loss function, the probability-based loss function can perform better in addressing the unfaithful reconstruction issue.

The fine details (i.e., eyes, nose, mouth) related to the facial identity are difficult to restore. On the other hand coarse details (i.e. color, pose) can be recovered reasonabally well. Based on the hierarchical generation process used by GANs (e.g. StyleGAN [91, 203]), the fine details of the generated images are estimated as the combination of interpolated multi-scale coarse GAN features. We find that the identity related to the fine details cannot be easily preserved once the coarse features are not correctly generated. The incorrect coarse features will result in poor quality fine details.

Inspired by the modern biometric system that runs in low-resolution images with high accuracy [202, 208], we show that the redundant identity of each high-resolution image is able to guide GEN learning. Specifically, we delicately extract [209] sub-images from the original one without changing its identity, and we then estimate the contextual distance of these sub-images in the identity feature space. Using low-resolution sub-images in the new contextual space not only can better consider the identity, but also focuses more on coarse features. We show that a new contextual distance leads to a 2.40% improvement in the identity preservation.

In addition we show that gradually changing coarse features (in the shallow layer of a GAN) produces results with different fine details and similar appearances. These results can boost GEN learning by further enlarging the new contextual space. A similar way has been used in the recent generative models [2, 210] for increasing the diversity of results. Specifically, we connect the hierarchical layers of the embedded StyleGAN with multiple modulation features. By repeatedly running the hierarchical layers with different modulation features [203], the embedded StyleGAN can produce multiple similar results with different fine details in a single forward pass. We then extract sub-images from these results for estimating the new contextual loss. The enlarged new contextual distance further leads to around 3.77% improvement in the identity preservation, and the final results significantly outperform the previous best turbulence removal methods.

Overall, our GAN inversion mechanism based on a new contextual learning way can better preserve the identity of the restored results. The effectiveness is shown on the person recognition problem, where the face images are degraded by strong turbulence.

Note that this is the first GAN method that can produce sharp-looking faces from turbulence degraded images captured in the range of 300 meters. It substantially improves upon the previous SOTA with an FID of 4.05 and recognition accuracy of 11.24% in the synthesized and real-world turbulence images.

6.2 Related Work

Turbulence Image Simulation and Mitigation Turbulence mitigation algorithms have achieved remarkable progress due to the emergence of deep neural networks. Deep networks often required synthetic turbulence-degraded images to train networks. For more details regarding the effect of turbulence on images and videos, readers are referred to [189, 211, 212]. The most widely applied turbulence simulation method for turbulence mitigation learning comes from Chak et al. [213] and its extended version by Lau et al. [214], which implements random distortion and blur based on handcrafted rules. Based on the simulation, Yasarla et al. [215] proposes a single image turbulence mitigation network, for learning mappings from simulated turbulence images and corresponding clear images. Later turbulence mitigation networks [97, 216–218] follow a similar simulation method and make various improvements to the network architectures and objectives. However, compared with the delicately handcrafted simulation method, the real-world turbulence is more complicated and the simulation results remain far from realistic turbulence [189]. Typical image restoration methods including denoising [204, 219] and deblurring [220] have shown to achieve impressive performance in such challenge degradation scenarios. However, recent performance comparisons [97] on these methods validate that their superiority in terms of PSNR

and SSIM can hardly help to better preserve identity in terms of Top-K accuracy. In this paper, the synthesized validation images are based on the recent physical simulation method proposed by Chimitt et al. [221] and its improved version by Mao et al. [222]. The simulation follows the split-step propagation [223, 224] that models the turbulence caused by wavefront distortion, and it statistically fits the theoretical predictions of turbulence. Another relevant area is blind face restoration [225–227], which consists of diverse unknown and complex degradation in the wild, and it shows similar effects in the degraded images.

GAN Inversion for Realistic Restoration According to the recent survey of GAN inversion [228], there exist two major approaches for embedding a well-trained GAN into restoration learning, and they share the same aim i.e., to find the optimal code in the generative latent space, which can be mapped to an image that lies in the natural image manifold via a well-trained GAN. Typical methods [190, 195, 229, 230] belong to the first approach retrieve the optimal latent code with some iterative optimization algorithms, with different modifications on the latent space definition or optimization procedure. Typical methods [191, 193, 197, 231] in the second approach, which employ an additional encoder to predict the optimal point, usually require pair-wise data to train the additional encoder. Due to the iterative optimization procedure of the first approach, it usually costs more time than vanilla neural networks during inference. The second approach avoids the time-consuming step in an end-to-end manner, but it will achieve some relatively low recognition performance, especially in images that are out-of-domain from the training image pairs. Our proposed method, instead, explicitly ensures the identity consistency during the training stage, and it

further addresses the intrinsic uncertainty issue inside the learning procedure, without additional parameters.

6.3 Method

6.3.1 Preliminaries

Following the common definition used in recent works [216–218], the generation procedure of turbulence images from the clear image I can be denoted as

$$\tilde{I}_k = D_k(H_k(I)) + n_k, \tag{6.1}$$

where $\tilde{I}_k$ is the k-th single turbulence image, $D_k(*)$ and $H_k(*)$ are the k-th deformation function and turbulence inspired point spread function respectively, and n_k is the optical noise. In practice, the turbulence degraded image is usually collected under long exposure, and it implies that we can approximate the degradation model into a single non-linear transformation $T(*)$ that combines all possible deformation and point spread functions $D \circ H$, as well as the noise n_k as

$$\tilde{I} = T(I). \tag{6.2}$$

Under such a scenario, turbulence mitigation is to find a general function $G(*)$ that can map arbitrary $\tilde{I}$ into the $G(\tilde{I})$ that is the most similar image to I.

However, the problem of turbulence mitigation is highly ill-posed and has multiple possible solutions for each turbulence degraded image. Here we define these possible

solutions as an image set $\mathbf{I} = \{I_1, I_2, \ldots, I_n\}$ corresponding to $G(\tilde{I})$. Conventional methods for learning turbulence mitigation simply average the solution set and optimize $G(*)$ with the pixel-wise losses, i.e., $\mathcal{L}_2$ as

$$\mathcal{L}_{MSE}(G) = \frac{1}{n}\sum_{i}^{n} \|G(\tilde{I}) - I_i\|^2, \ I_i \in \mathbf{I}. \tag{6.3}$$

Minimizing equation equation 6.3 equivalently minimizes the maximum likelihood estimation of the conditional empirical distribution $\mathbb{C}_g$ of $G(\tilde{I})|\tilde{I}$ and the average of conditional distribution $\mathbb{C}_{\mathbf{I}}$ of $\mathbf{I}|\tilde{I}$ in n possible solutions as

$$\mathcal{L}(G) = \mathbb{E}_{x\sim\mathbb{C}_{\mathbf{I}}}\left[\log \frac{\prod_i^n \{\mathbb{C}_{I_i|\tilde{I}}(x)\}}{\mathbb{C}_g(x)}\right]. \tag{6.4}$$

Such a term of averaged loss is slightly different from the original formula of $\mathcal{L}_2$, but with the uncertainty of degradation increasing, the difference increases at the same time.

Different from the above pixel-wise loss function, the superiority of GANs comes from the availed adversarial objective, i.e., the logistic loss function with a discriminator $D(*)$. Under the scenario of learning a generative embedding network, which fine-tunes the network with a well-trained GAN, the applied well-trained discriminator is optimal, and thus the optimization of $G(*)$ with the logistic loss function is equivalent to minimizing the JS divergence between two image distributions according to Arjovsky

et al. [207] as

$$\begin{aligned} \mathcal{L}(D,G) &= 2\mathrm{JSD}(\mathbb{P}_{\mathbf{I}} \| \mathbb{P}_g) - 2\log 2, \\ \mathrm{JSD}(\mathbb{P}_{\mathbf{I}} \| \mathbb{P}_g) &= \mathbb{E}_{x\sim\mathbb{P}_{\mathbf{I}}}\left[\log \frac{\mathbb{P}_{\mathbf{I}}(x)}{\mathbb{P}_g(x)}\right] \\ &\quad + \mathbb{E}_{x\sim\mathbb{P}_{\mathbf{g}}}\left[\log \frac{\mathbb{P}_g(x)}{\mathbb{P}_{\mathbf{I}}(x)}\right], \end{aligned} \tag{6.5}$$

where $\mathbb{P}_{\mathbf{I}}$ and $\mathbb{P}_{\mathbf{I}}$ are the probability distribution of target images and generated images, respectively. Possibly resulting from the term difference of two loss functions, we can observe that the learning of generative embedding network that combines these loss functions usually does not converge well in both the visual quality and identity, especially in degradation tasks with large uncertainty. Existing solutions for alleviating such issues include using the perceptual loss [232], which is shown to be able to measure the maximum mean discrepancy [233] of two image distributions [234]. However, the identity information is often not maintained in the results obtained by these methods.

To address the issue, in Section 6.3.3, we propose a novel contextual distance to replace the pixel-wise losses for identity preservation, in Section 6.3.4 we further modify the vanilla feature flow of $G(*)$ so that we can leverage multiple results into consideration. The overall pipeline is illustrated in Figure 6.2. Moreover, comprehensive experiments and ablation studies conducted in Section 9.4.1 demonstrate our effectiveness.

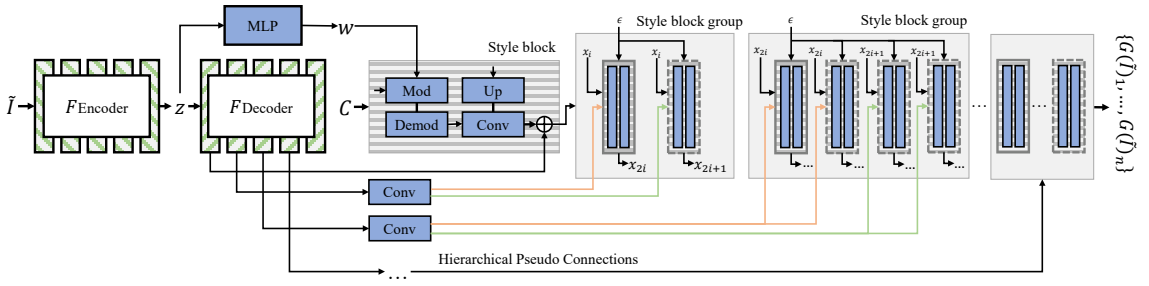


Figure 6.2: An overview of our method. F_{Encoder} and F_{Decoder} take the turbulence degraded image $\tilde{I}$ as input, then they generate the latent code z and modulation features for embedded StyleGAN, respectively. An additional MLP is utilized for embedding the latent code z into the StyeleGAN latent space w. We follow the StyleGAN convention which starts with a random initialized constant C and generates images in a hierarchical fashion with a sequence of intermediate features $\{x_i, \ldots, x_{2i}, x_{2i+1}\}$ and the noise ϵ. Here hierarchical pseudo connections boost the StyleGAN with multiple pseudo-style blocks, and thus they enable the network to generate multiple results $\{G(I)_1, \ldots, G(I)_n\}$.

6.3.2 Network Architecture

Our network architecture is illustrated in Figure 6.2 based on StyleGAN2, except additional encoder $F_{Encoder}$ and $F_{Decoder}$. We employ the inverse layer settings of StyleGAN2 but start with an image $\tilde{I}$ instead of a constant $\mathcal{C}$. The remaining part, i.e., the style block, is the same as StyleGAN2 and uses its elementary layers, including the modulation layer, up-sampling layer, de-modulation layer, and the convolution layer that take hierarchical feature x_i as input. The connection between the added decoder and the style blocks is by replacing the modulation parameters with the decoder outputs. The latent space $\mathcal{W}$ is replaced with the encoder output.

6.3.3 Spatial Periodic Contextual Distance

As discussed before, ameliorating the complex degradation while considering the identity difference is one of the major goals of our learning objective. The adversarial loss bypasses the uncertainty issue by statistically matching the difference between two image distributions, but it ignores the identity of each image. Another contextual approach was proposed by Mechrez et al. [205] which can tolerate the spatial uncertainty in pixel space, i.e., misalignment, but it fails to preserve the identity information as we expect.

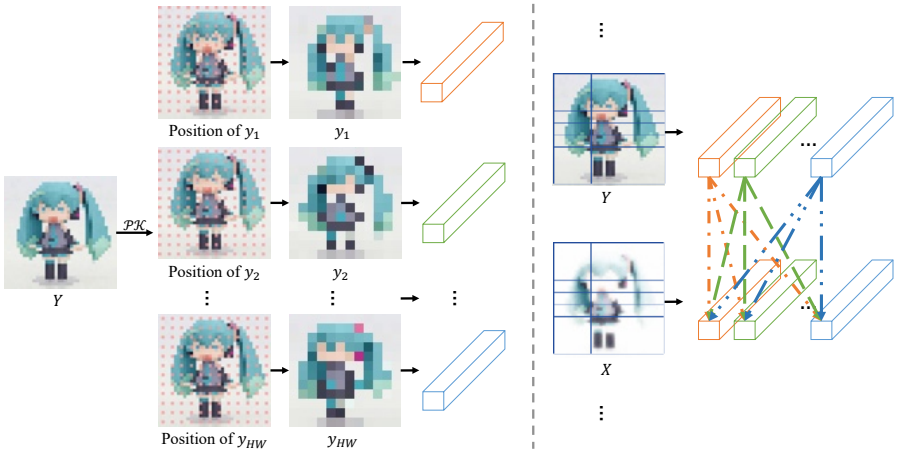


Figure 6.3: The procedure of $\mathcal{PK}$ takes an image Y as input and produces sub-images. The sampled positions of each sub-image x_i are highlighted in Y. These sub-images are then flattened into samples for contextual distance.

Our method treats each image as the spatial periodic combination of multiple sub-images, and it then considers the contextual distance of two sub-image collections as the final distance of two images. As shown in Figure 6.3, each sub-image $\{y_1, y_2, \ldots, y_{HW}\}$ in the dimension of $r \times r$ contains different parts of the original image Y in the dimension of $rH \times rW$. Though they are significantly different in appearance (no

overlapping pixels), they share the same identity, i.e., each sub-image can be recognized as the same class as the original image. We empirically find that considering the contextual distance built upon them improves identity preservation, compared with the original pixel-wise losses.

To effectively extract sub-images, past works utilized strategies like the internal patch [235] with sliding windows, or feature points [206] with pre-trained convolution layers. In contrast, the sub-image in our distance is extracted with a new packing operation $\mathcal{PK}$. For an image with dimension of $rH \times rW \times C$, $\mathcal{PK}$ unshuffles (i.e., an inverse operation of pixel shuffling [236]) it into $H * W$ number of images in the dimension of $r \times r \times C$ with rate r, which can be mathematically denoted as

$$\mathcal{PK}(I)_{x*y,r_1,r_2,c} = I_{\lfloor x*r_1 \rfloor, \lfloor y*r_2 \rfloor, c}. \tag{6.6}$$

Each sub-image is then flattened into a point (features to represent the original image in the high-dimensional identity feature space). We then estimate the contextual distance between the two features as the final loss for network learning.

Regarding the distance function $\mathrm{CX}(*)$, we follow Mechrez et al. [205] and define it on two collections of sub-images $\mathcal{PK}(X) = \{x_i\}$ and $\mathcal{PK}(Y) = \{y_j\}$ of two images instead of the perceptual features in Mechrez et al. [205], formulated as

$$\mathrm{CX}(\mathcal{PK}(X), \mathcal{PK}(Y)) = -\frac{1}{N} \sum_i \log \sum_j A_{ij}. \tag{6.7}$$

Here A_{ij} can be seen as the second log term of a multivariate kernel-density-estimation kernel. In practice, A_{ij} is implemented to be close to a delta function with normalized

consine distance $\tilde{d}(x_i, y_i)$ and bandwidth h as

$$A_{ij} = \frac{\exp\left(1 - \tilde{d}_{ij}/h\right)}{\sum_l \exp\left(1 - \tilde{d}_{il}/h\right)} = \begin{cases} \approx 1 & \text{if } \tilde{d}_{ij} \ll \tilde{d}_{il} \ \forall l \neq j \\ \approx 0 & \text{otherwise.} \end{cases} \tag{6.8}$$

Minimizing such a distance approximates the minimization of the divergence $KL(\mathbb{P}_X \| \mathbb{P}_Y)$ [206], which is similar to the term of adversarial loss that defines the statistical divergence of two images in Equation equation 6.5. In practical implementation, for face images with the dimension of 512×512, we empirically find that $r = 16$ leads to the best performance.

6.3.4 Hierarchical Pseudo Connections

In Section 6.3.3 we have discussed how the spatial periodic contextual distance helps to maintain the identity information. However, the existing issue mentioned in Equation 6.4 still affects the identity preserving capability of networks during learning. Therefore, based on the hierarchical generation pipeline of GANs that transforms coarse feature combinations into fine features, we show that multiple solutions can be acquired in a single forward pass by simply employing different modulations. Similar to StyleGAN, the modulation is carried out by scaling and shifting the GAN features, and thus different modulation parameters result in different outputs. Correspondingly, each layer of $F_{Decoder}$ generates the modulation parameters by using an additional spatial feature transformation [194]. Based on them, increasing the channel numbers of each $F_{Decoder}$ transformation layer can produce multiple modulation parameters and finally result in multiple solutions. The correlation example of the generated results

with their appearance and identity distance is shown in Figure 6.4 for demonstrating the idea. As Figure 6.2 illustrates, the major modification comes from the proposed *Hierarchical Pseudo Connections (**HPC**)*, which connects the degradation removal part ($F_{Encoder}$ and $F_{Decoder}$) with the style block group, in which the actual real style block is illustrated with solid line and the pseudo style block is illustrated with dotted line. The pseudo style block is defined as the real style block which processes feature with different modulation instead of the original modulation.

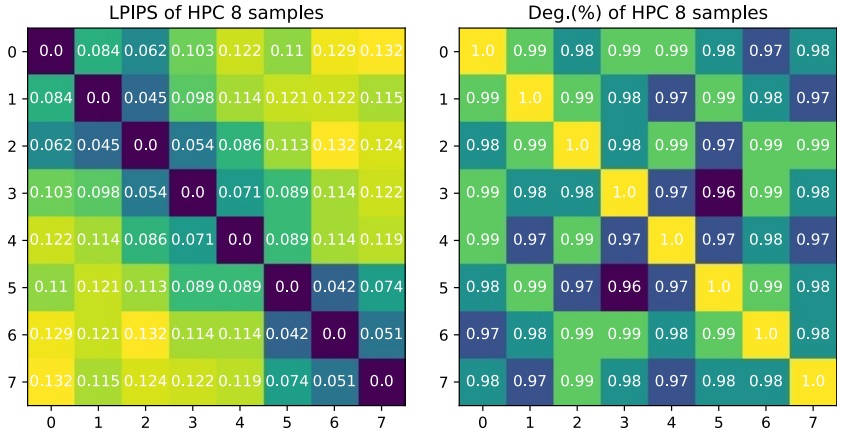


Figure 6.4: Visualization of the correlation of produced 8 sub-images in terms of both the appearance (LPIPS) and identity (Deg.(%)) metrics. We rescale their values according to the order for better visual comparison.

Similar to the hierarchical generation property of GANs, HPC allows a well-trained GAN to generate multiple possible results with similar coarse appearance but different fine details. Hence, the applied HPC with g number of style block groups in the final layer transforms the $G(*)$ as

$$\hat{\mathbf{I}} = \{\hat{I}_1, \hat{I}_2, \ldots, \hat{I}_{2^g}\} = G(\tilde{I}). \tag{6.9}$$

Even though many possible solutions are generated due to the ill-posed nature of

the problem, we can still significantly reduce their difference with the transformed Equation equation 6.7 in GANs as

$$\mathcal{L}(G) = CX(\mathcal{PK}(\mathbf{I})), \mathcal{PK}(\hat{\mathbf{I}}). \tag{6.10}$$

For instance, the generated result of the first style block will be availed as the coarse feature in the second style block. The second style block group generates two separate results with two different modulations connected by two pseudo connections, while the third style block group will generate four separate results under four different modulations and two separate coarse features connected by four pseudo connections. The number of generated results will increase until the final layer generates 2^g final results. Since these generated pseudo results share the same identity, their extracted sub-images can significantly enlarge the new contextual space for identity preservation learning.

6.3.5 Model Objective

The final learning objective for training our method combines both the proposed spatial periodic contextual loss, adversarial loss, perceptual loss, and identity preserving loss, which mostly follows GEN methods [193, 194].

Reconstruction Related Loss. For a network $G(*)$ with g number of hierarchical pseudo connections where it outputs 2^g number of results $G(\tilde{I})$ and I denotes the

target image, we use the following loss function to train the network

$$\mathcal{L}_{rec}(G) = \mathcal{L}_{CX} + \mathcal{L}_{adv} + \mathcal{L}_{per} + \mathcal{L}_{id}, \tag{6.11}$$

where the proposed contextual loss $\mathcal{L}_{CX}(G)$ is defined as

$$\mathcal{L}_{CX}(G) = \frac{1}{2^g} \sum_{i}^{2^g} CX(\mathcal{PK}(I), \mathcal{PK}(G(\tilde{I})_i)). \tag{6.12}$$

The adversarial loss $\mathcal{L}_{adv}(G)$ is defined as

$$\mathcal{L}_{adv}(G) = -\lambda_{adv} \mathbb{E}_{G(\tilde{I})_i} \text{softplus}(D(G(\tilde{I})_i)). \tag{6.13}$$

The perceptual loss $\mathcal{L}_{per}(G)$ is defined as

$$\mathcal{L}_{per}(G) = \lambda_{per} \|\phi(I) - \phi(G(\tilde{I})_i)\|, \tag{6.14}$$

where $\phi(*)$ denotes the pre-trained VGG-19 network [237] that is applied with its $\{\text{conv1}, \ldots, \text{conv5}\}$ layers before activation [238]. Finally, the identity loss $\mathcal{L}_{id}(G)$ is defined as

$$\mathcal{L}_{id}(G) = \lambda_{id} \|\eta(I) - \eta(G(\tilde{I})_i)\|, \tag{6.15}$$

where $\eta(*)$ corresponds to a pre-trained ArcFace network [202] without the final logistic layer. Note that λ_{adv}, λ_{per}, and λ_{id} are the weights corresponding to the adversarial loss, perceptual loss and identity preserving loss, respectively.

Adversarial Related Loss. The discriminator $D(*)$ is trained in the same manner as StyleGAN2 [239] except that multiple results are applied and the loss values are averaged.

6.4 Experiments

In this section, we present the experimental details for evaluating our method and its settings, as well as the comparison results with the state-of-the-art methods. The evaluation is conducted on both the synthesized and real-world turbulence degraded images introduced below.

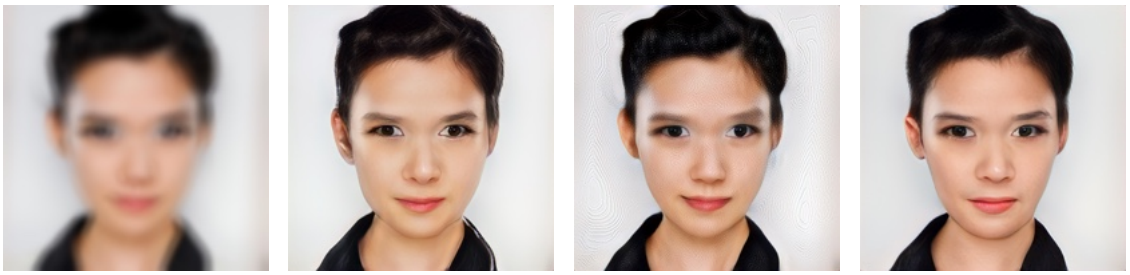
6.4.1 Training and Testing Settings

Synthesized Testing Benchmark Regarding the reference-based evaluation, which can best capture the turbulence mitigation capability of various methods, we synthesize turbulence degraded image pairs with TurbulenceSim_P2S [222], one of the state-of-the-art turbulence simulations works. The synthesis is conducted on the selected first 100 images of CelebAHQ [240], named **CelebAHQ100**, and the parameters of TurbulenceSim_P2S are carefully selected to match the real-world turbulence images as similar as possible. Specifically, we set D, $r0$, and $corr$ as $\{5, 1.25, -0.01\}$ for the CelebAHQ100 simulation.

Real-world Testing Benchmark Regarding the real-world turbulence images, which do not have pixel-wise corresponding clear images, we evaluate their face recognition accuracy with indoor reference clear images. We use the high-quality real-world turbulence degraded faces collected at 300 meters. These images were collected by

Left Pose | Front Pose | Right Pose | Indoor Reference

Figure 6.5: Visualization of used real-world turbulence images. These images are captured by cameras put in left, center, and right positions, and the last image is captured indoors without turbulence. Pixel-wise ground-truth is not available in the real-world data.



Synthesized Input | BFR | TurbulenceSim_P2S | ElasticAug (Ours)

Figure 6.6: Contour and ring artifacts can be seen in the restored results of networks trained on simulated images from TurbulenceSim and BFR. In contrast, the results from the network trained using our ElasticAug shows sharper edges and more accurate facial details. **(200% Zoom is recommended to see their difference.)**

the US Army Night Vision Lab under different atmospheric conditions. Face images without turbulence in indoor conditions were also collected. We carefully process these images to avoid possible data compression artifacts, and then crop and wrap faces with the pre-trained RetinaFace network [208]. The final dataset contains images from 89 individuals, each containing 3 images in different poses, named **TubFace89**.

Real-world Testing Benchmark Regarding the real-world turbulence images, which do not have pixel-wise corresponding clear images, we evaluate their face recognition

accuracy with indoor reference clear images. We use the high-quality real-world turbulence degraded faces collected at 300 meters. These images were collected by the US Army Night Vision Lab under different atmospheric conditions. Face images without turbulence in indoor conditions were also collected. We carefully process these images to avoid possible data compression artifacts, and then crop and wrap faces with the pre-trained RetinaFace network [208]. The final dataset contains images from 89 individuals, each containing 3 images in different poses, named **TubFace89**.

Another real-world turbulence removal benchmark is the LRFID dataset [241]. We demonstrate our performance superiority by evaluating on the LRFID dataset that follows the protocol of Nair et al. [97].

Table 6.1: Performance comparisons against the state-of-the-art methods and ours on the synthesized turbulence degraded face images. Our method achieves the best performance on visual quality, perceptual metrics, and identity metrics, i.e., FID, LPIPS, and Deg. (We highlight the best, the second best, and the third best performance with the colors red, orange, and yellow.)

	BFR	LPIPS ↓	FID ↓	NIQE ↓	Deg. (%) ↑	PSNR ↑	SSIM ↑	Images/Sec	Parameters (M)
Turbulence Img	-	0.6490	274.84	17.48	8.91	20.33	0.6455	-	-
TDRN [217] [TBIOM22]	-	0.5869	190.17	13.55	18.54	21.63	0.6611	3.41	8
ATFaceGAN [216] [TBIOM21]	-	0.5868	181.10	12.71	15.55	21.77	0.6633	7.32	68.70
PSFRGAN [242] [CVPR21]	✓	0.4070	122.39	4.051	22.30	19.96	0.5451	3.70	184.2
GPEN [193] [CVPR21]	✓	0.3757	89.77	4.299	32.06	19.82	0.5651	14.40	284.1
GFPGAN [194] [CVPR21]	✓	0.3800	113.42	5.515	26.55	20.31	0.5797	20.14	615.4
LTT-GAN (Ours)	✓	0.2906	85.72	4.285	49.88	20.96	0.6042	14.12	741.4

Training Dataset and Augmentation We use FFHQ [203] with 70,000 high-quality images in the resolution of 512×512 as the training dataset. The performance of the generative embedding networks highly depends on the degradation similarity of the training set and testing set. To find the best way to boost the turbulence

mitigation capability of networks, we conduct experiments with three existing data augmentation methods, i.e., Blind Face Restoration (BFR) [193, 194, 225, 226], TurbulenceSim_P2S [222], and our new method.

TurbulenceSim_P2S learns the basis functions for spatially varying convolutions from known turbulence models for turbulence simulation. Since the real-world turbulence degraded images usually suffer from strong blur, we empirically find that the degradation procedure used in BFR can also produce similar degraded results to real-world turbulence degradation. Moreover, in this paper, we introduce a new simulation method, called **ElasticAug**, which is based on BFR augmentation, and it combines the blur augmentation with Elastic transformation [243] that randomly moves pixels using local displacement fields.

Here we directly apply GFPGAN [194] as the baseline and train the network on three data augmentation methods. In Figure 6.6 we show their results. One can find that the network trained on ElasticAug achieves the best visual quality. Although BFR produces a similar degradation procedure of turbulence, it cannot estimate the spatial deformation, and hence its result contains strong ringing artifacts. Although TurbulenceSim_P2S produces comparable clear results, the network trained on its simulated images generates results with strange contour artifacts, and thus we use it for evaluation only. The proposed ElasticAug is availed as the default data augmentation method in the following experiments.

Implementation Our implementation follows the settings of conventional generative embedding networks [193, 194], i.e., the learning rate was set to 2×10^{-3} and decayed

Table 6.2: Face verification accuracy comparison against the state-of-the-art methods and ours on the real-world turbulence degraded front pose face images.

	Top1 ↑	Top3 ↑	Top5 ↑	Deg. (%) ↑
Turbulence Img	14.61	26.97	32.58	13.59
TDRN [217] [TBIOM21]	13.37	28.99	30.11	10.41
ATFaceGAN [216] [TBIOM21]	23.60	38.20	51.69	17.26
PSFRGAN [242] [CVPR21]	39.33	56.18	61.80	27.71
GPEN [193] [CVPR21]	43.82	66.29	73.03	32.46
GFPGAN [194] [CVPR21]	49.44	68.54	79.78	32.08
LTT-GAN (Ours)	57.30	71.91	82.02	35.27

Table 6.3: Face verification accuracy comparison against the state-of-the-art methods and ours on the real-world turbulence degraded all pose face images.

	Top1 ↑	Top3 ↑	Top5 ↑	Deg. (%) ↑
Turbulence Img	14.61	44.94	50.56	11.11
TDRN [217] [TBIOM22]	19.10	42.11	53.68	12.46
ATFaceGAN [216] [TBIOM21]	23.60	64.04	69.66	13.48
PSFRGAN [242] [CVPR21]	41.57	73.03	84.27	20.30
GPEN [193] [CVPR21]	48.31	79.78	85.39	24.47
GFPGAN [194] [CVPR21]	48.31	85.39	95.51	24.57
LTT-GAN (Ours)	59.55	87.64	93.26	26.57

at 600k and 700k iterations by a rate of 2, with a totally of 800k iterations. The whole training is conducted in a mini-batch size of 12 using 4 NVIDIA A100 GPUs with the PyTorch framework. For the $F_{Encoder}$ and $F_{Decoder}$ module, we initialized their parameters with Xavier initialization. For the GAN module, we apply StyleGAN2 [203] with its well-trained parameters on the FFHQ generation, and its parameters are frozen during network training. For the discriminator module, we apply the discriminator of a well-trained StyleGAN2 on the FFHQ generation, and its parameters are optimized with the same learning rate of $F_{Encoder}$ and $F_{Decoder}$.

Table 6.4: Performance comparisons against the state-of-the-art methods and ours on the real-world LRFID dataset [241]

	LPIPS↓	FID↓	Top-1↑	Top-3↑	Top-5↑
Turbulence Img	0.6293	195.71	35.3	62.2	71.2
MPRNET [244]	0.5755	176.41	34.1	64.6	74.4
ATNet [245]	0.6128	202.45	36.5	64.6	74.4
ATFaceGAN [246]	0.6300	169.60	47.5	65.8	82.3
GFPGAN [194]	0.5587	124.55	57.3	79.2	85.3
ILVR [247]	0.5661	161.38	31.7	59.7	67.0
LTT-GAN (Ours)	0.4803	119.23	58.5	81.7	85.3

6.4.2 Comparisons with State-of-the-art Methods

We conduct comparisons with several state-of-the-art methods in turbulence mitigation and blind face restoration tasks, i.e., TDRN [215], ATFaceGAN [216], PSFRGAN [242], GPEN [193], and GFPGAN [194]. For fair comparisons, we finetune all of them on FFHQ with their official released codes.

Synthesized Turbulence In Figure 6.7, we compare the visual results corresponding to various methods on the synthesized turbulence face images. One can find that our method can best restore the facial details from the severely degraded images. Among the compared methods, although the results from GPEN seem to contain more sharp details, the plausible details are less similar to the corresponding ground truth images, instead, our results best preserve the identify related details that can be found in the ground truth images. In Table 6.1, we further compare the quantitative performance using widely used visual quality metrics, i.e., LPIPS [248], FID [249], NIQE [250], and pixel-wise metrics, i.e., PSNR and SSIM. Since the turbulence mitigation task is highly correlated to face recognition, we further employ the face identification metric,

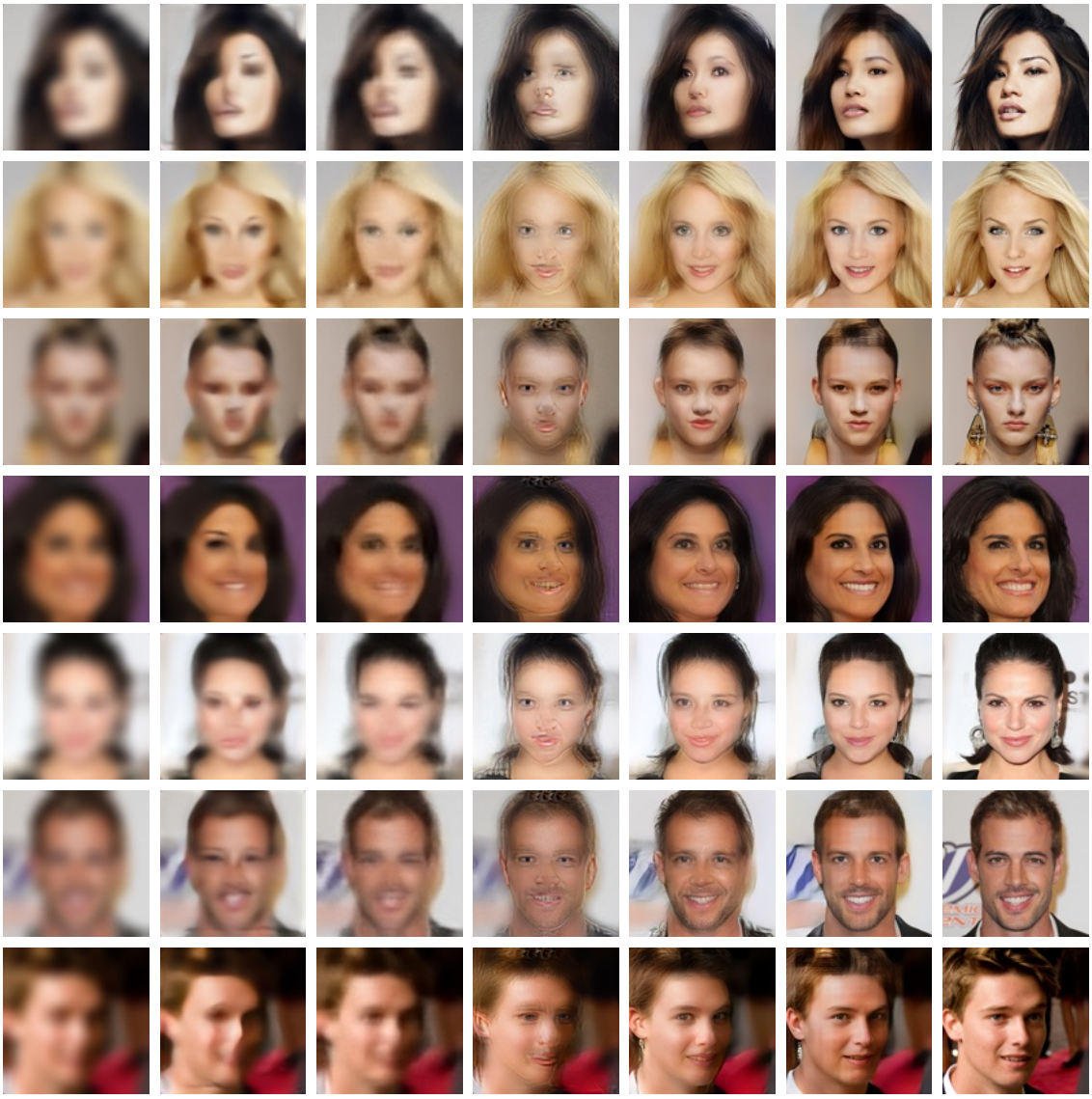


Figure 6.7: Visual comparisons of various methods on the synthesized turbulence face images. Compared with the results from other methods, ours achieves the best visual quality and similarity between the restored results and the ground truth.

i.e., Deg.(%) that is measured on facial features. From the results, we can notice that our method achieves the best performance in the LPIPS and FID metrics, and it

achieves comparable performance in NIQE. Regarding the facial identify preservation, our method outperforms the second best method by 17.82%.

Recent works [251, 252] have shown that the pixel-wise metrics, i.e., PSNR and SSIM, are not correlated well with the visual quality. Therefore, a slight performance drop in terms of PSNR and SSIM doesn't denote worse visual quality. Moreover, we want to point the reviewer to the visual comparison results of ATFaceGAN. Even though ATFaceGAN has achieved the best performance in PSNR, its results tend to be significantly blurry compared to ours. The observation is consistent with the conclusion proposed by Blau et al. [251], which finds that a blurry result tends to achieve higher PSNR performance than a less blurry one. Therefore, we adopt other more fair metrics, including LPIPS, FID, and identity-related metric Deg. (%). In these metrics, our method has achieved a consistent performance improvement compared with the others.

In Figure 6.8, we further present the turbulence mitigation results of the compared methods in real-world conditions. These images contain heavy turbulence effects and different poses, which is extremely difficult. However, our method can still achieve the superiority in visual quality comparisons, with fewer artifacts compared with the recent BFR methods. In the images captured in left and right poses, our method can still ensure identity consistency, while the other methods tend to fail in such cases. We argue that such superiority comes from the learning objective differences.

In Table 6.2 and Table 6.3, we show the face recognition accuracy of the restored results, in the center pose, and center, left, and right poses, respectively. From the comparisons, one can easily find that our method significantly outperforms the other

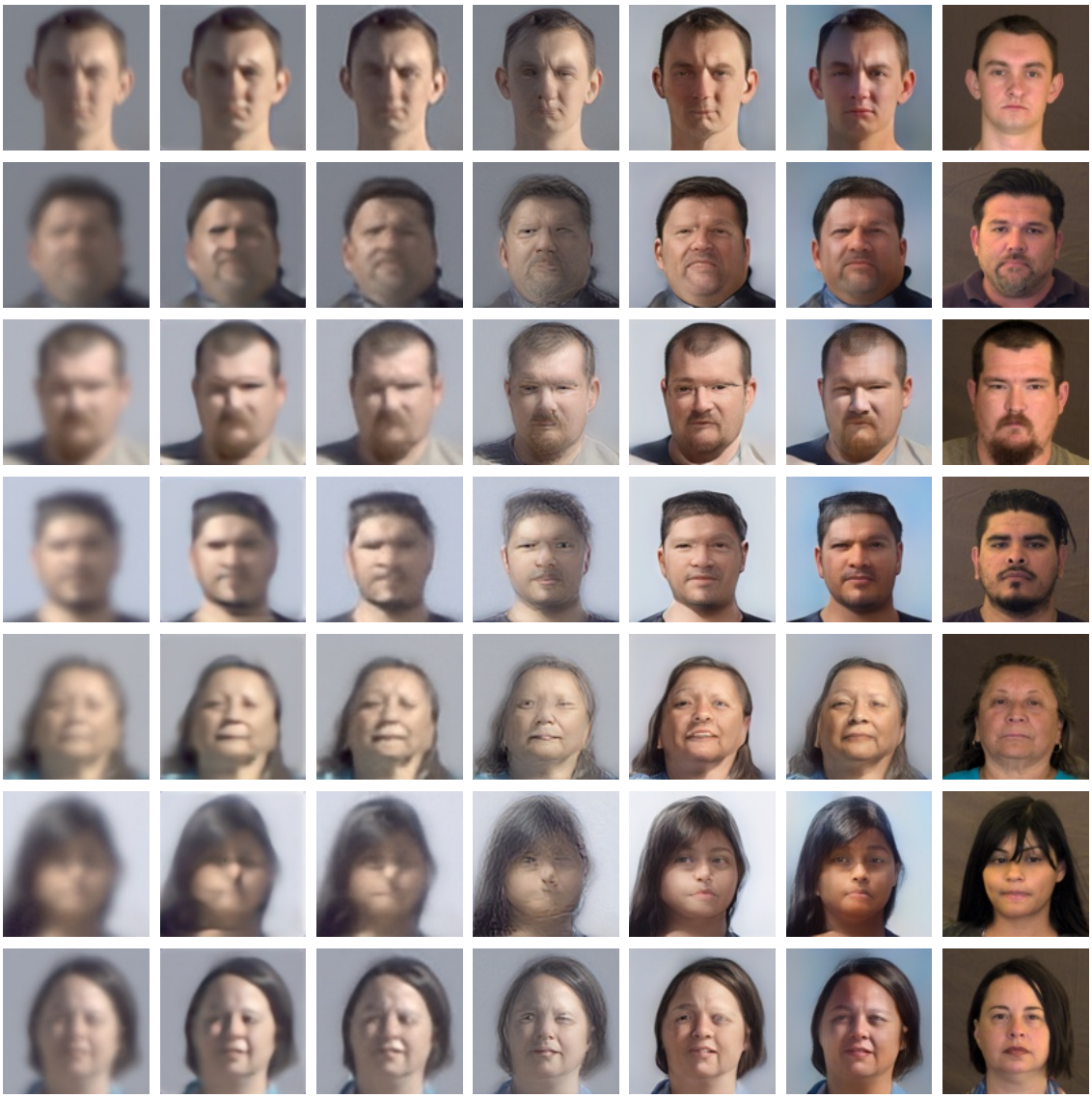


Figure 6.8: Visualization comparisons of cthe ompared methods on the synthesized turbulence face images. Compared with the results with artifacts from other methods, ours achieves the best visual quality and similarity between the restored results and the ground truth.

SOTA methods in the recognition accuracy, in both the center pose and all poses. Compared with the second best methods, the proposed approach achieves over 7.86% and 11.24% improvements. The performance comparisons shown in Table 6.4 further demonstrate the performance superiority of our method.

6.4.3 Ablations

In this section, to comprehensively evaluate the effectiveness of each module of our method, we conduct experiments with different settings of our method on the synthesized turbulence degraded images for visual quality and identity preserving performance. These performance comparisons are shown in Table 6.5 and Table 6.6. For clarification, we remove all the mechanisms proposed by us and name the method as *Baseline*. To simplify, we re-train each method with 80k iterations only, and thus its performance number differs from the final experiments in Table 6.1.

Table 6.5: Performance comparison against several different settings of spatial periodic contextual loss.

	LPIPS ↓	FID ↓	Deg. (%) ↑
Baseline	0.3469	85.36	36.80
Baseline w./*Brute-Force ArcFace Loss*	0.3357	86.62	36.70
Baseline w./*CX*	0.3358	85.08	38.57
Baseline w./*SPCX(r=8)*	0.3360	87.94	38.10
Baseline w./*SPCX(r=16)*	0.3328	82.75	38.67
Baseline w./*SPCX(r=32)*	0.3370	85.90	39.20

The superiority of our proposed spatial periodic contextual loss is demonstrated by comparisons with *Baseline*, *Baseline with Brute-Force ArcFace Loss (larger weights of ArcFace Loss [194])*, and *Baseline with Contextual loss [205]*. As expected, in Table 6.5,

the *Baseline w./SPCX* significantly outperforms the compared methods against the visual quality and identity preserving effectiveness. We further show the influence led by the sub-image scale r, which determines how the image identity is represented in the sub-image contextual space. In Table 6.5, the larger r results in a stronger identity preserving effectiveness (i.e. Deg.(%)); however, $r = 16$ achieves the best visual quality (i.e. LPIPS and FID) compared with $r = 8$ and $r = 32$. Since the larger sub-image consist of more identity feature and more redundant appearance pixels, the results suggest that the first property benefits the identity preserving effectiveness, and the second property latter harms the visual quality in the new contextual space. Therefore, we empirically set the default settings as $r = 16$ for achieving the best trade-off in our applications.

Table 6.6: Performance comparison against several different settings of hierarchical pseudo connections.

	LPIPS ↓	FID ↓	Deg. (%) ↑	Time (s)
Baseline	0.3469	85.36	36.80	0.0296
Baseline w./*SPCX(r=16)*	0.3328	82.75	38.67	0.0296
... w./*HPC(g=8)*	0.3273	92.80	40.03	0.0537
... w./*HPC(g=16)*	0.3324	94.22	40.74	0.0756
... w./*HPC(g=32)*	0.3395	102.64	40.57	0.1239

Since our proposed hierarchical pseudo connections work as the spatial periodic contextual loss augmentation, its superiority is demonstrated by comparisons with *Baseline* and *LTT-GAN with Spatial Periodic Contextual loss* ($r = 16$). By choosing different numbers of g in HPC settings, we show the performance improved led by HPC in Table 6.6. It is easy to notice that the identity preserving performance can be significantly be boosted by hierarchical pseudo connections. Furthermore, the

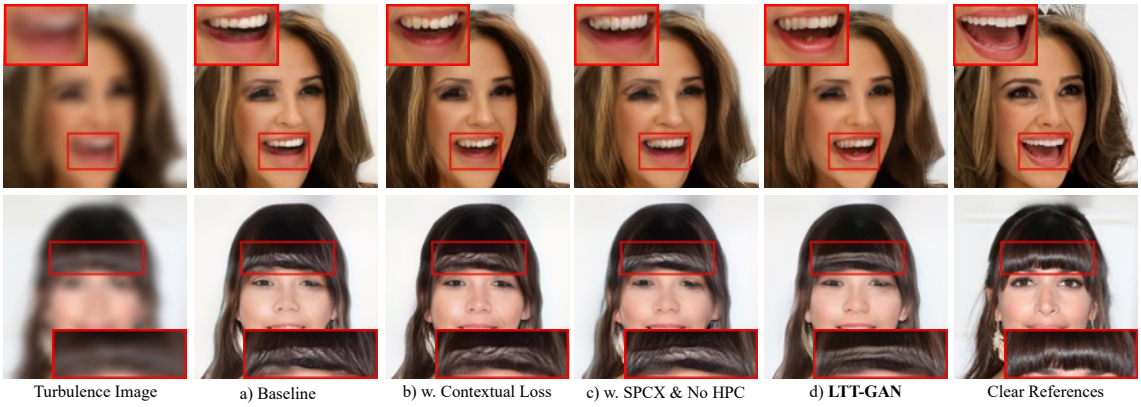
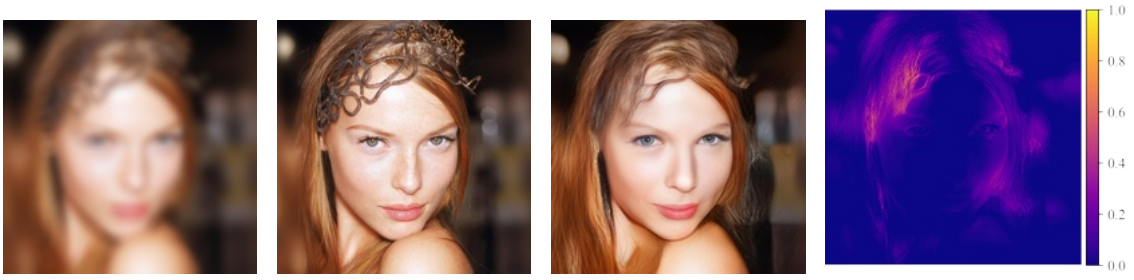


Figure 6.9: Visualization results of our method on the synthesized CelebAHQ100, as we vary the usage of SPCX and HPC. Compared with the baseline, or baselines combined with contextual loss, our method significantly reduce the unnatural artifacts with better details.

performance increase is linear depending on the number of g. However, we can also see that increasing g can hurt the visual quality at some level. The performance drop in terms of FID and LPIPS comes from combining the generated multiple results. We adopt the way of averaging multiple results, which is shown to preserve identity features and improve the Top-K accuracy effectively. However, the proposed way of averaging images blurs the final result and thus reduces the visual quality. We empirically select $g = 8$ as the default setting to achieve the trade-off between visual quality, identity preservation, and running time.

We further show the visual results of different settings for comparison in Figure 6.9. From the results, we can conclude that the proposed two mechanisms are indeed reducing artifacts on the restored results.



Turbulence Image | Reference | Our Result | Uncertainty Map

Figure 6.10: Even though the network generates a sharp result with hair patterns in the original hair band area, the uncertainty map can identify which area is uncertain.

Table 6.7: Identity preserving scored of the compared method with different facial recognition networks. We show the identity similarity and the recognition Top-1 accuracy respectively.

	ArcFace-R18	ArcFace-R50	ArcFace-R101
Turbulence Img	13.59 / 14.61	23.37 / 25.84	29.47 / 32.09
TDRN	10.41 / 13.37	19.63 / 11.23	15.68 / 13.37
ATFaceGAN	17.26 / 23.60	33.79 / 35.96	32.38 / 37.30
PSFRGAN	27.71 / 39.33	43.83 / 43.82	37.33 / 46.51
GPEN	32.46 / 43.82	44.96 / 62.92	40.94 / 67.64
GFPGAN	32.08 / 49.44	45.67 / 57.30	37.61 / 55.01
LTT-GAN (ours)	**35.27 / 57.30**	**45.85 / 65.17**	**47.91 / 88.77**

6.4.4 Uncertainty Visualization

Identifying the uncertainty of restoration results can help both manual inspection and automated methods. Significant efforts have been applied to conventional restoration methods for uncertainty visualization, including Monte Carlo dropouts [253] used in TDRN [215], and multiple forward passes of Style-Mixing in recent GAN inversion methods [200, 254]. However, these efforts neither require network architecture modification nor additional optimization steps that are specialized and hurt the performance. In contrast, benefiting from the diversity in our pseudo results, by

simply measuring the variance of multiple pseudo results, our method can estimate the uncertainty map in a single forward pass. Figure 6.10 shows an example, where the highlighted uncertainty map in the hair area successfully matches with the difference between the results and the ground truth. Hence, as a byproduct of our method, our method has the capability of advising the downstream applications with the uncertainty map.

Chapter 7

Latent Feature-Guided Shadow Removal Diffusion

7.1 Introduction

Images captured in natural illumination often contain shadows caused by objects blocking the light from the illumination source. Shadows can degrade the performance of many computer vision algorithms, such as detection, segmentation, and recognition [255, 256]. Furthermore, removing shadows is essential for photo-editing applications such as distractor removal [257] and relighting [258]; which may rely on instance-level shadow removal. Therefore, it is critical to develop methods that can automatically remove shadows from captured images as works explored in literature.

Recently, diffusion models [85] with hierarchical denoising autoencoders [259] have shown to achieve impressive synthesis performance in terms of sample quality and diversity. The conditional generation ability further allows for iterative refinement and fine-grained control according to certain conditions. Motivated by the success of diffusion-based image restoration models [9, 43], we adapt diffusion models for the task of shadow removal by conditioning on the input shadow image and corresponding shadow mask as a baseline approach to generate shadow-free images. However, preserving and generating high-fidelity textures and colors in the shadow region after removal is non-trivial. The baseline model appears to favor borrowing textures from the surrounding non-shadow areas rather than focusing on restoring the original details underneath the shadow, which results in incorrect color mixtures and loss of detail in

the shadow region. In Fig. 7.2, we show one of the representative issues of image-mask conditioning, *i.e.*, the model synthesizes results containing an incorrect color mixture.

Intuitively, the intensity drop in shadow regions means that diffusion models are typically guided more strongly by the surrounding non-shadow areas. However, this guidance can harm the fidelity of the result if the texture and color under the shadow differs significantly from the surrounding areas. In addition, the multi-head attention module [108] used in diffusion models can exacerbate this issue by extracting global information. This motivates us to consider guiding the conditioned diffusion models with an additional latent feature space that captures external perceptual shadow-free information as the shadow removal priors.

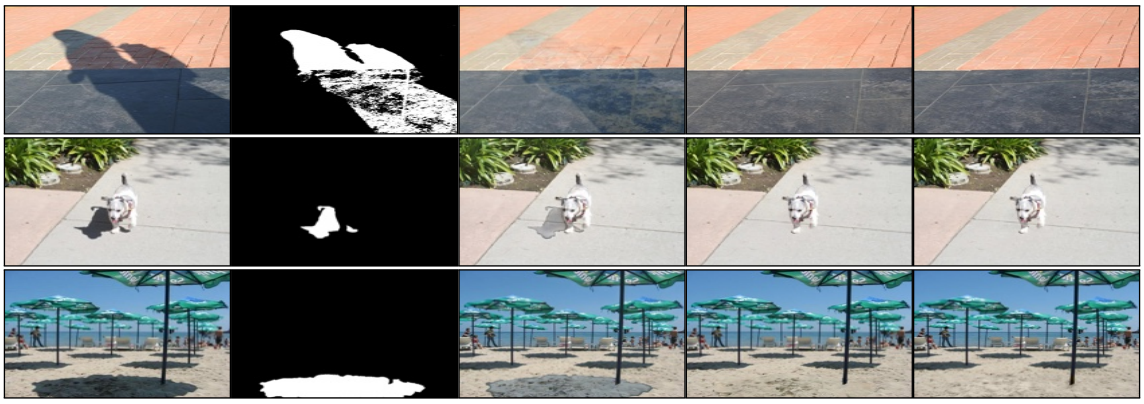
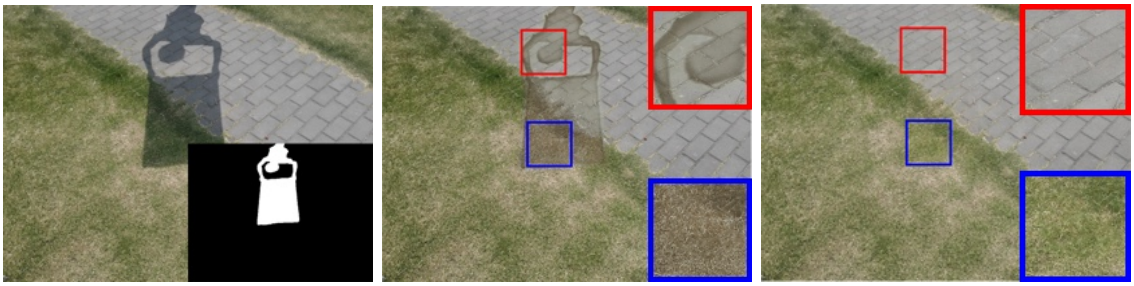


Figure 7.1: Given a shadow mask, our method effectively removes shadows and recovers the underlying details for shadows at the general level (top two rows) or instance level (bottom two rows). From left to right, we show the input image, shadow mask, SG-ShadowNet [260] result, our method result, and shadow-free images for comparisons.

Our proposed method differs from latent diffusion models (LDMs) [9] in that we incorporate a learnable feature encoder to discover a latent feature space. To optimize the latent feature encoder, we minimize the difference between the feature space of



Shadow & Mask | Baseline Result | Our Result

Figure 7.2: Our baseline method, which conditions diffusion models solely on shadow and mask images, produces incorrect results such as color mixing in highlight areas. In contrast, our proposed method generates results with consistent and reasonable colors that match the surrounding area.

shadow images and that of shadow-free images, using it as the loss function. Through experimentation, we have found that optimizing the encoder together with the diffusion models leads to a compact and perceptual latent feature space. Additionally, we demonstrate that pretraining the diffusion model on shadow-free images simplifies the optimization process and is crucial for achieving high-fidelity synthesis. By guiding the diffusion models on the latent feature space, instead of just conditioning on the shadow image and mask, we observe significant improvements in shadow removal capability.

In addition to the proposed latent feature space guidance, we propose an improved diffusion network that addresses the issue of *posterior collapse* [261–263], which refers to the local optima of diffusion models. We identify the local optimum as the degrading effect of the noise variable and introduce a Dense Latent Variable Fusion (DLVF) module that includes dense skip connections between the embedding of the noise and diffusion network. DLVF significantly improves shadow removal results without introducing additional parameters or running complexity. In summary, this paper

makes the following contributions:

- A new shadow removal model that addresses the challenging task of general and instance-level shadow removal. This is the first work, to the best of our knowledge, to demonstrate the applicability of diffusion models for instance shadow removal.
- We show that it is possible to acquire compact and perceptual guidance in a learned feature space that is optimized together with the diffusion models, without relying on handcrafted features or physical quantities.
- We identify the local optimum of diffusion models that degrades the model results and introduce a dense latent variable fusion module to alleviate it, leading to significant performance improvement.

7.2 Related Work

Shadow Removal. The major challenge of modern learning-based shadow removal approaches comes from the large diversity of real-world shadow scenes. The performance of recent shadow removal methods degrades significantly on out-of-distribution scenes [264]. Various approaches have been explored for addressing this issue, such as using physical illumination models, handcrafted priors, and image gradients [265–267]. The recent trend has been in developing learning-based methods that can predict shadow-free scenes [268–272] or intermediate factors [273, 274] for restoration. These methods have improved from previous methods in learning data [272], shadow effects [271], network architecture [269, 275, 276], and learning target decomposition [273, 274, 277]. In particular, generative models have gained some traction for shadow removal.

ARGAN [278] removes the effect of shadow in a progressive manner determined by a discriminator. Nevertheless, these end-to-end GAN-based methods lack generalizability on the out-of-distribution shadow images without significant modifications. Recent diffusion models have shown promising performance in general image restoration tasks but are rarely explored in shadow removal [161, 166, 279]. In this work, we first propose to apply diffusion models for removing shadows, to leverage their impressive capacity of perceptual synthesis, which is shown to be capable of gradually preserving details in denoising sequences.

Latent Feature Space Guidance. Guidance has become an essential component of diffusion models and powers spectacular image generation results in recent works. Typical guidance for diffusion models includes class information [7], text description [57, 160], and even gradients [73]. Nevertheless, these features cannot be easily adopted in shadow removal to provide more guidance than images. In literature, physical quantities and handcrafted features have been heavily explored for guiding the restoration network. Zhu et al. [280] propose to guide the network with an estimated shadow-invariant color map, and Wan et al. [260] propose to guide the network with coarse de-shadowed images. Illumination invariant representations [281–283] are another related approach that aims to decompose intrinsic images by finding quantities invariant to color, density, or shading. In our approach, we define a new latent feature space for guiding diffusion models. By maximizing the similarity between the shadow and shadow-free latents, we empirically demonstrate that it better guides the diffusion model to remove shadows by encapsulating essential perceptual information as a shadow-free prior.

Posterior Collapse. The problem of posterior collapse refers to undesirable local optima first observed in the training of VAE models [3]. Efforts to address it have included aggressive optimization of the inference network proposed by He et al. [262], weakening the generator by Fu et al. [284], and changing the objective function by Tolstikhin et al. [285]. In this work, we show that although this issue has primarily been investigated in VAE models, conditional diffusion models can also suffer from similar issues. Specifically, the conditions used in diffusion models usually provide stronger guidance compared to the latent noise variable. Inspired by previous efforts to address the issue, we propose a new Dense Latent Variable Fusion (DLVF) module for diffusion models and experimentally demonstrate that this design improvement improves shadow removal results without introducing additional costs or modifications. Different from the other latent-based diffusion methods [9, 286], ours uses simpler pixel space and models shadow-free image distribution.

7.3 Proposed Method

7.3.1 Conditional Diffusion Models

Diffusion Forward Process. The denoising diffusion models have been shown to be effective for modeling complex data distributions by reversing a gradual noising process. For the shadow-free image distribution, we define the forward diffusion process that destroys a shadow-free image $\mathbf{y} \sim q(\mathbf{y})$ with T successive standard noises:

$$q(\mathbf{y}_t|\mathbf{y}_{t-1}) = \mathcal{N}\left(\mathbf{y}_t; \sqrt{\beta_t}\mathbf{y}_0, (1-\beta_t)\,\mathbf{I}\right). \tag{7.1}$$

Alternatively, we can use the reparameterization trick [3] to express this as:

$$\begin{aligned} q(\mathbf{y}_t|\mathbf{y}_0) &= \mathcal{N}\left(\mathbf{y}_t; \sqrt{\bar{\alpha}_t}\mathbf{y}_0, (1-\bar{\alpha}_t)\,\mathbf{I}\right) \\ &= \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \epsilon\sqrt{1-\bar{\alpha}_t}, \epsilon \sim \mathcal{N}(0, \mathbf{I}), \end{aligned} \tag{7.2}$$

where the variance schedule $\{\beta_1, \ldots, \beta_T\}$ linear increases and has a closed form $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$.

Diffusion Backward Process. The reversion of $q(\mathbf{y}_t|\mathbf{y}_{t-1})$ is tractable by conditioning on image $\mathbf{y}_0$, and it results in sampling arbitrary shadow-free images from noise $\mathbf{y}_T \sim \mathcal{N}(0, I)$ for removal as:

$$q\left(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{y}_0\right) = \mathcal{N}\left(\mathbf{y}_{t-1}; \tilde{\mu}\left(\mathbf{y}_t, \mathbf{y}_0\right), \tilde{\beta}_t\mathbf{I}\right). \tag{7.3}$$

According to Bayes' rule and Eq. equation 7.2, we represent $\tilde{\mu}_t$ as:

$$\tilde{\mu}_t\left(\mathbf{y}_t, \mathbf{y}_0\right) := \frac{1}{\sqrt{\alpha_t}}(\mathbf{y}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_t). \tag{7.4}$$

Ho et al. [259] suggests modeling the process with p_θ by optimizing the *variational lower bound* (L_{VLB}) as:

$$L_{\text{VLB}} = L_T + L_{T-1} + \cdots + L_0, \tag{7.5}$$

which is defined with Kullback–Leibler (KL) divergence as:

$$\begin{aligned} L_T &= D_{\text{KL}}(q(\mathbf{y}_T|\mathbf{y}_0) \parallel p_\theta(\mathbf{y}_T)), \\ L_t &= D_{\text{KL}}(q(\mathbf{y}_t|\mathbf{y}_{t+1}, \mathbf{y}_0) \parallel p_\theta(\mathbf{y}_t|\mathbf{y}_{t+1})), \\ L_0 &= -\log p_\theta(\mathbf{y}_0|\mathbf{y}_1), \end{aligned} \tag{7.6}$$

and the effective simplification of L_t is

$$L_{\text{simple}} := E\left[\|\epsilon - \epsilon_\theta\left(\mathbf{y}_t, t\right)\|^2\right]. \tag{7.7}$$

Diffusion Conditioning. A straightforward approach to producing shadow-free results is to condition diffusion models on the shadow image $\mathbf{x}$ and shadow mask m

by concatenating them with noise $\mathbf{y}_t$ along the channel dimension:

$$p_\theta\left(\mathbf{y}_{t-1}|\mathbf{y}_t\right) := p_\theta\left(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{x}, m, t\right). \tag{7.8}$$

In the following sections, we will discuss our improvement based on the baseline following Eq. equation 7.8 that takes image $\mathbf{x}$, $\mathbf{y}_t$, and mask m as input and predicts the shadow-free noise $\mathbf{y}_{t-1}$ for effective shadow removal as $p_\theta\left(\mathbf{y}_{t-1}|\mathbf{y}_t, t\right)$.

7.3.2 Latent Feature Guidance

The proposed latent feature encoder $\mathcal{E}_\theta(\cdot)$ uses the same network architecture as the diffusion network $\epsilon_\theta(\cdot)$ with the exception of a timestep embedding and predicts a single-channel feature map that has the same spatial dimension as the shadow image x. It guides the diffusion process Eq. equation 7.8 by concatenating the guidance with conditions:

$$p_\theta\left(\mathbf{y}_{t-1}|\mathbf{y}_t\right) := p_\theta\left(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathcal{E}_\theta(\mathbf{x}, m), t\right). \tag{7.9}$$

We propose to learn to extract shadow-free priors using the latent feature space by minimizing the invariant loss between the encoded shadow-free images and shadow images with shadow masks as:

$$\arg\min_\theta \left\|\mathcal{E}_\theta(\mathbf{y}_0, \mathbf{0}) - \mathcal{E}_\theta(\mathbf{x}, m)\right\|^2. \tag{7.10}$$

In order to extract a compact and perceptual feature space to guide the diffusion model, we optimize the encoder together with the whole network during training based on Eq. equation 7.7:

$$\begin{aligned} L_{\text{simple}} :=& E\left[\left\|\epsilon - \epsilon_\theta\left(\mathbf{y}_t, \mathcal{E}_\theta(\mathbf{x}, m), \mathbf{x}, m, t\right)\right\|^2\right] \\ &+ \left\|\mathcal{E}_\theta(\mathbf{y}_0, \mathbf{0}) - \mathcal{E}_\theta(\mathbf{x}, m)\right\|^2. \end{aligned} \tag{7.11}$$

Moreover, we empirically find that pretraining the diffusion model $\epsilon_\theta(\cdot)$ and then finetuning it accelerates the optimization of Eq. equation 7.11. Intuitively, the pretraining strategy provides a good starting point to finetune the diffusion model, such that the encoder has already learned to model the important characteristics of shadow-free images such as shadow-free textures and colors. This feature space provides strong guidance during finetuning for minimizing shadow features with the invariant loss, allowing the model to achieve higher-quality results.

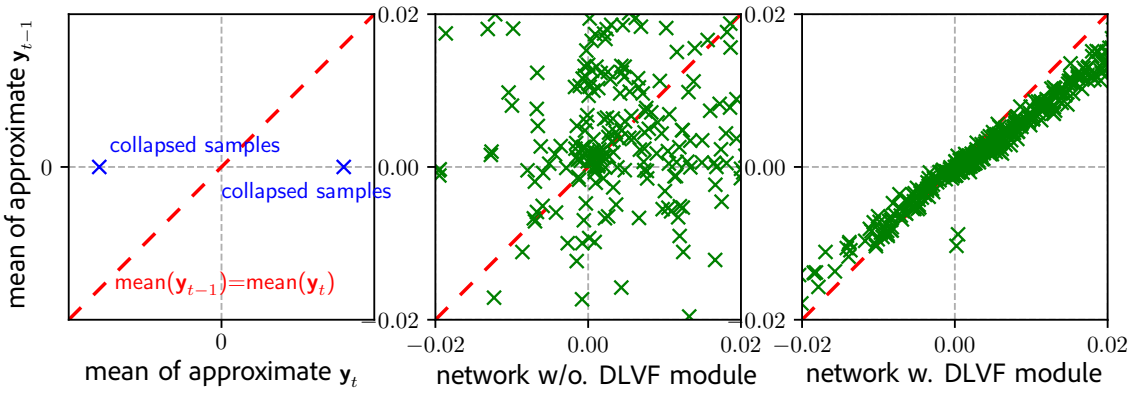


Figure 7.3: We visualize the mean space of variables to show the collapse and our effects. The horizontal and vertical axis represent the mean of predicted $\mathbf{y}_t$ and $\mathbf{y}_{t-1}$, respectively. The dashed diagonal line represents when the approximate noise is relevant. By projecting denoised samples, the results show the network with our DLVF (third) successfully moves points onto the diagonal line and away from collapses compared to without it (second).

Subsequently, we propose a two-stage learning approach for guiding the diffusion models including pretraining and finetuning as shown in Fig. **??** as:

- Optimize the diffusion network ϵ_θ together with the latent encoder $\mathcal{E}_\theta$ for modeling the characteristics of shadow-free images by minimizing the loss:

$$E\left[\left\|\epsilon - \epsilon_\theta\left(\mathbf{y}_t, \mathcal{E}_\theta(\mathbf{y}_0, 0), \mathbf{y}_0, m, t\right)\right\|^2\right]. \tag{7.12}$$

- Finetune the encoder $\mathcal{E}_\theta$ and diffusion network ϵ_θ by optimizing Eq. equation 7.11 to effectively remove shadows and preserve the underlying texture.

(a) (b) (c) (d)

Figure 7.4: Visual comparisons of different guidance strategies in shadow removal literature. **(a)** to **(d)**: shadow image, invariant color map [280], coarse deshadowed image [260], and our learned latent feature. Our approach provides more perceptual information than **(b)** and contains fewer shadow features than **(c)**, which still retains a shadow boundary.

To demonstrate the effectiveness of our proposed latent feature guidance, Fig. 7.4 compares it with existing guidance strategies in shadow removal literature, including [260], which conditions on estimated coarse de-shadowed images, and [280], which conditions on estimated invariant color maps for restoration. Our approach preserves more shadow-free perceptual details compared to the estimated invariant color map, which only consists of large color blocks. Similarly, our approach retains fewer shadow features compared to the estimated coarse de-shadowed image, which still retains shadow boundaries that may lead to incorrect results.

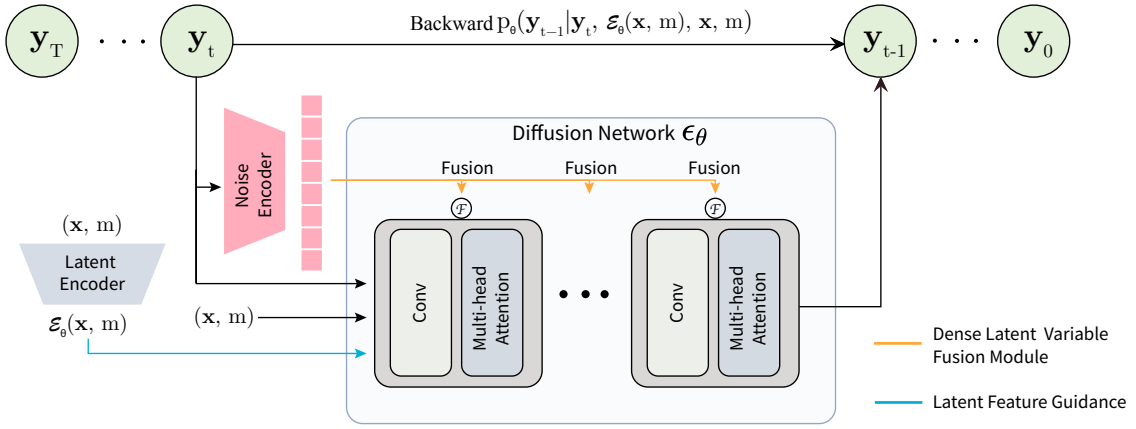


Figure 7.5: Our diffusion model architecture is illustrated in this backward diffusion diagram. The latent feature encoder $\mathcal{E}_\theta(\cdot)$ takes the shadow image $\mathbf{x} \in \mathbb{R}^{3\times H\times W}$ and shadow mask $m \in \mathbb{R}^{1\times H\times W}$ as input, with a resolution of $H \times W$, and acquires the latent feature in a compressed dimension of $1 \times H \times W$. The diffusion network $\epsilon_\theta(\cdot)$ conditioned on $(\mathbf{x}, m)$ takes the latent feature concatenated with the noisy image $\mathbf{y}_t \in \mathbb{R}^{3\times H\times W}$ as input, and estimates the noiseless image $\mathbf{y}_{t-1} \in \mathbb{R}^{3\times H\times W}$ at each diffusion process $p_\theta(\cdot)$. In this process, the noise encoder takes the noise image $\mathbf{y}_t$ as input and acquires a 1-D vector as the noise embedding, which is fused with the diffusion network features by modulation for escaping the local optima.

7.3.3 Dense Latent Variable Fusion Module

The phenomenon known as posterior collapse occurs when the training procedure of generative models falls into a trivial local optimum of L_{VLB}, causing the model to ignore the latent variable and collapse the model posterior to the prior, which has only been discussed in VAE [262]. Given the intrinsic similarity between diffusion models and VAE, we first determine the collapse issue of diffusion models under guidance and then address it with a new module.

In our proposed diffusion models, we parameterize the variational distribution $p_\theta(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{y}_0)$ with the latent variable $\mathbf{y}_t$ under the guidance $\mathcal{E}_\theta(\mathbf{x}, m)$ in Eq. equation 9.7. In this

case, the local optima are characterized by:

$$\begin{aligned} p_\theta(\mathbf{y}_{t-1}) &= p_\theta\left(\mathbf{y}_{t-1} \mid \mathbf{y}_t, \mathcal{E}_\theta(\mathbf{x}, m), t\right) \\ &= p_\theta\left(\mathbf{y}_{t-1} \mid \mathbf{y}_t\right) p_\theta\left(\mathbf{y}_{t-1} \mid \mathcal{E}_\theta(\mathbf{x}, m), t\right) \\ &:= p_\theta\left(\mathbf{y}_{t-1} \mid \mathcal{E}_\theta(\mathbf{x}, m), t\right). \end{aligned} \tag{7.13}$$

This is undesirable since a crucial goal of diffusion models is to produce diverse outputs. This is particularly important for shadow removal, where complex shadow distributions exist that cannot be easily represented by guidance alone.

Much attention has been devoted to remedying the posterior collapse of VAE models. However, some of these methods weaken the encoder or modeling capability of posterior-related components, as observed in [262, 284]. Other approaches, such as those proposed in [285, 287], significantly complicate the optimization.

In this work, we introduce a new Dense Latent Variable Fusion (DLVF) module that works in tandem with the diffusion network to establish strong links between the latent variable and the generated results. To elaborate, for each block $\mathcal{G}(\cdot)^n$ of the diffusion network [7] at level n, the feature h^{n-1} and the embedding of $\text{emb}(\mathbf{y}_t)$ generated by a three-layer MLP are both inputted as:

$$h^n = \mathcal{G}(h^{n-1}, \text{emb}(\mathbf{y}_t) \downarrow_{2n}, t)^n, \tag{7.14}$$

where $\downarrow_{2n}$ denotes the pooling operation with a scaling factor of $2n$ to match the dimension of the features h^{n-1}. To achieve a larger receptive field for the latent variable embedding, we employ fully-connected layers as an additional encoder before inputting them into the network and use adaptive pooling operations to transform

the noise into vectors:

$$\mathbf{y}'_t = \mathcal{P}_{ooling}(\mathcal{E}_{\text{noise}}(\mathbf{y}_t)), \mathbf{y}'_t \in \mathbb{R}^{1\times N}, \tag{7.15}$$

where N is the size of the vector noise. The connection between the embedding $\text{emb}(\mathbf{y}_t)\downarrow_{2n}$ and the network features h^{n-1} is conducted by point-wise summing.

To demonstrate the collapse and effectiveness of our method, we visualize the correspondence between the latent variable $\mathbf{y}_t$ and approximated $\mathbf{y}_{t-1}$, which are randomly selected from T denoising processes, shown in Fig. 7.3. In comparison to the baseline without our fusion strategies, our method shows a stronger correspondence between the two variables, indicating a better optimum in the training dynamics. This ultimately results in more effective removal.

7.4 Experiments

We provide further implementation details, including the settings of the network and optimizer, in the supplemental.

Shadow Removal Benchmarks. We conduct both quantitative and qualitative comparisons on three benchmarks: ISTD [288], AISTD [273], and SRD [289]. The ISTD dataset is a real-world shadow-removal benchmark that consists of 1,330 image triplets for training and 540 image triplets for testing. The image triplet includes the shadow image, shadow mask, and the corresponding shadow-free image. The shadow mask is extracted from the binary difference between the shadow image and the shadow-free image. The AISTD dataset uses the same scene as the ISTD dataset but avoids inconsistent color between the shadow and shadow-free image for accurate

comparisons. SRD contains different scenes and consists of 2,680 image pairs for training and 408 image pairs for testing. Since SRD does not contain binary masks for the shadow regions, we follow the common practice and use the masks generated by Cu et al. [290]. For data processing, we empirically dilate all shadow masks in a kernel size of $k = 21$ to address incomplete shadow masks.

Instance-level Shadow Removal Benchmark. We conduct various experiments with visual comparisons on shadow images collected from the internet. The major difference between the above benchmarks and instance-shadow images is the number of shadows in the image, whereas the latter usually has more than one shadow instances. The major collections of our instance-shadow images come from the shadow object association (SOBA) dataset [291]. We use the manually manipulated shadow-free images of the DESOBA dataset [292] as ground truths for removing shadows at the instance level. For the network training, we synthesize shadow image triplets following the method proposed by Inoue et al. [293]. Please see the supplement for a deep analysis of the synthesized data.

7.4.1 Performance Evaluation

We evaluate our proposed algorithm against state-of-the-art shadow-removal methods, including SP+M-Net [273], DHAN [290], Param+M+D-Net [274], G2R-ShadowNet [272], Auto-Exposure [270], DC-ShadowNet [271], EMDN [294], BMN [280], and SG-ShadowNet [260], as well as two representative image restoration diffusion models, Palette Diffusion [279], and Repaint Diffusion [166]. The evaluation metrics include the Root Mean Square Error (RMSE) between the shadow-free results and the ground

Table 7.1: Quantitative result comparisons of our methods and the state-of-the-art methods on *AISTD*. The best and second-best performance is indicated with **bold** and *italic* respectively. We use ↑ and ↓ to suggest better high/lower score.

		shadow region			*non-shadow region*			*all* image		
Method		RMSE ↓	PSNR ↑	SSIM ↑	RMSE ↓	PSNR ↑	SSIM ↑	RMSE ↓	PSNR ↑	SSIM ↑
SP+M-Net [273]	ICCV-19	5.93	37.96	0.990	3.05	35.77	0.973	3.51	32.90	0.957
DHAN [290]	AAAI-20	11.38	33.18	0.987	7.15	27.10	0.972	7.81	25.65	0.954
Param+M+D-Net [274]	ECCV-20	9.67	33.46	0.985	2.91	34.85	0.974	3.98	30.13	0.945
G2R-ShadowNet [272]	CVPR-21	7.38	36.24	0.988	3.00	35.26	0.975	3.69	31.90	0.953
Auto-Exposure [270]	CVPR-21	6.57	36.30	0.976	3.83	31.10	0.874	4.27	29.44	0.838
DC-ShadowNet [271]	ICCV-21	10.57	32.15	0.976	3.82	34.99	0.969	4.80	28.75	0.925
EMDN [294]	AAAI-22	7.94	36.44	0.986	4.78	31.80	0.962	5.28	29.98	0.940
SG-ShadowNet [260]	ECCV-22	5.93	37.25	0.989	3.00	35.27	0.975	3.46	32.42	0.956
BMN [280]	CVPR-22	*5.69*	*38.00*	**0.991**	*2.52*	*37.35*	**0.981**	*3.02*	*33.93*	*0.966*
Palette Diffusion [166]	SIGGRAPH-22	15.40	-	-	7.82	-	-	6.41	-	-
Repaint Diffusion [279]	CVPR-22	12.90	-	-	10.66	-	-	24.90	-	-
LFG-Diffusion	Ours	**5.15**	**39.36**	**0.991**	**2.47**	**37.69**	**0.981**	**2.90**	**34.69**	**0.968**
shadow image		39.72	20.87	0.944	2.51	36.63	0.980	8.38	20.45	0.908

truth in the LAB color space as well as the Peak Signal-to-Noise Ratio (PSNR) and structural similarity (SSIM) in the RGB space. We also provide the metrics measured on the whole image and non-shadow region for reference. Following previous methods [270, 272, 274], we interpolate the results with a resolution of 256×256 for evaluation. We also present the metrics evaluated on the shadow images for reference.

Tab. 7.1 shows the quantitative results on the AISTD dataset. Compared with the representative end-to-end learning-based methods, including EMDN [294], Auto-Exposure [270] and DHAN [290], ours significantly outperforms them in all regions. The performance gap between them and ours in the *non-shadow* and *full* regions further indicates the superiority of our model in generating high-quality textures of backgrounds. As expected, the comparison between ours and the other generative methods, including BMN [280], DC-ShadowNet [271], and G2R-ShadowNet [272] demonstrate that our method achieves equal performance improvement in different

regions. In contrast, the other methods fail in regions with specific textures. The difference suggests the guidance effectiveness of our modeled latent feature, which is capable of balancing the unbalanced guidance from the surrounding non-shadow areas and shadow regions via the invariant loss function to aid the model in preserving texture and color. The results shown in Tab. 7.2 on the SRD and ISTD datasets further demonstrate the superiority of our method over the others.

Table 7.2: Quantitative comparison results of our methods and the state-of-the-art methods on the *ISTD dataset* and *SRD dataset*. We want to remark on a slight performance drop in the non-shadow region of our method. The reason is that the two benchmarks are un-adjusted, which means the shadow and shadow-free image pairs were captured at different lighting environments. The color inconsistency would result in inaccurate non-shadow region and all image measurement.

(a) ISTD dataset results.

	shadow region			*non-shadow region*			*all image*		
Method	RMSE ↓	PSNR ↑	SSIM ↑	RMSE ↓	PSNR ↑	SSIM ↑	RMSE ↓	PSNR ↑	SSIM ↑
DHAN [290]	7.53	35.82	*0.989*	5.33	30.95	0.971	5.68	29.09	0.953
G2R-ShadowNet [272]	10.72	31.63	0.975	-	-	-	-	-	-
Auto-Exposure [270]	7.82	34.94	0.973	5.59	28.57	0.862	5.94	27.19	0.824
DC-ShadowNet [271]	11.43	31.69	0.976	5.86	28.92	0.956	6.62	26.38	0.917
EMDN [294]	7.94	*36.44*	0.986	4.78	31.80	0.962	5.28	29.98	0.940
SG-ShadowNet [260]	-	-	-	-	-	-	-	-	-
BMN [280]	*7.44*	35.73	*0.989*	**4.61**	**32.73**	*0.976*	*5.06*	*30.26*	*0.957*
LFG-Diffusion (Ours)	**6.41**	**37.19**	**0.990**	*4.65*	*32.60*	**0.977**	**4.93**	**30.64**	**0.963**
shadow image	32.67	22.43	0.953	6.77	27.27	0.974	10.86	20.56	0.908

(b) SRD dataset result.

	shadow region		
Method	RMSE ↓	PSNR ↑	SSIM
DHAN [290]	8.94	33.67	0.978
G2R-ShadowNet [272]	-	-	-
Auto-exposure [270]	8.86	34.93	0.963
DC-ShadowNet [271]	7.86	36.34	0.970
EMDN [294]	9.83	32.48	0.928
SG-ShadowNet [260]	8.00	35.53	0.974
BMN [280]	*7.40*	*36.81*	**0.979**
LFG-Diffusion (Ours)	**6.81**	**37.42**	**0.979**
shadow image	46.24	19.95	0.889

Visual comparison from the AISTD dataset in Fig. 7.6 and SRD dataset in Fig. 7.7 further validates the effectiveness of our method. As shown in Fig. 7.6, our method demonstrates robustness to imperfect shadow mask inputs and preserves the textures as well as removing other subtle shadow effects.

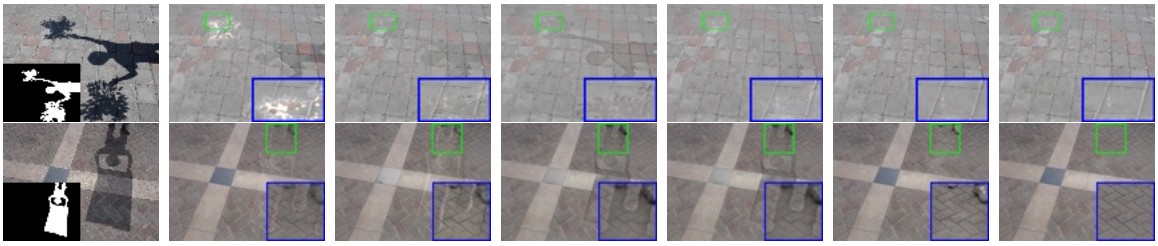
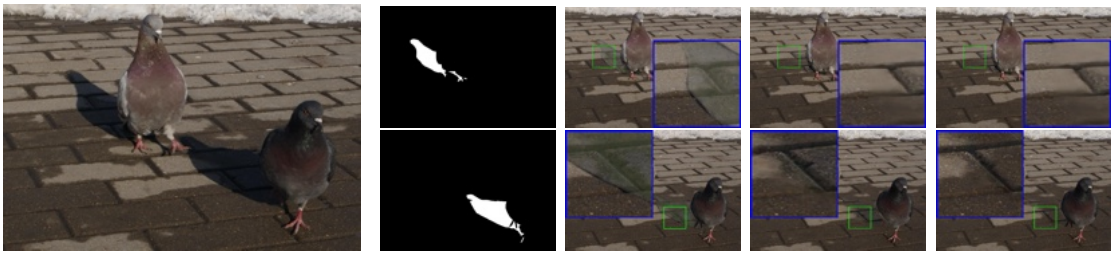


Figure 7.6: Visual comparisons of the representative hard shadow removal results on *AISTD* dataset. Here we highlight the details of shadow regions that are marked with green box in the blue box area, where ours best perseveres details and removes shadow effects. Please see the supplement for additional visual results.

Figure 7.7: Visual comparisons of the representative hard shadow cases from the SRD dataset.

7.4.2 Instance Shadow Removal Evaluation

For real-world applications, shadows cast by objects in the scene are usually instance-level; thus, preserving the other shadows while accurately removing the target instance shadow is crucial. Here, we compare our method with the most recent shadow removal work SG-ShadowNet [260] to demonstrate the generalizability of our method, where we finetuned it with the same dataset synthesized for our experiments. Sample results are shown in Fig. 7.8. Compared with the SG-ShadowNet, ours thoroughly removes the shadow from the images. As far as we know, this is the first work to demonstrate the applicability of instance shadow removal.



shadow image *shadow mask*, *SG-ShadowNet result*, *ours*, and *shadow-free image*

Figure 7.8: Visual comparisons of the real instance shadow removal results on the DeSOBA dataset.

7.4.3 Ablation Study and Analysis

Table 7.3: Effects of different types of strategies for addressing the posterior collapse in diffusion models. We only show shadow region results that are distinguishing.

		AISTD dataset		
Settings	Region	RMSE ↓	PSNR ↑	SSIM ↑
w/o. fusion	*shadow*	6.75	36.34	**0.990**
lagged posterior	*shadow*	*6.65*	*36.27*	**0.990**
dense fusion (Ours)	*shadow*	**5.92**	**37.70**	**0.990**

Effects of the DLVF module. In Tab. 7.3, we investigate the effectiveness of the proposed DLVF module. We use two alternative methods for comparison: a lagged posterior approach [262] for addressing the posterior collapse, which aggressively optimizes the diffusion network before optimizing the latent feature encoder, and the baseline approach that uses a diffusion network without the fusion strategy. The results show that lagged posterior is less effective, with only a slight improvement margin over the baseline, which could be due to the large complexity of diffusion models and difficulty in training. In contrast, our proposed dense fusion scheme outperforms the baseline by a margin of 0.83 RMSE. Moreover, we visually demonstrate its effectiveness

by showing the correspondence of the denoised results $\mathbf{y}_{t-1}$ and $\mathbf{y}_t$ in Fig. 7.3. These results validate the idea proposed in our DLVF, *i.e.*, fusing more noise features into each block of the diffusion network is a promising approach for alleviating the local optima of training diffusion models.

Effects of Latent Feature Space Guidance. Tab. 7.4 compares different types of diffusion model guidance for removing shadows, including **(a)** estimated invariant color map, **(b)** estimated coarse de-shadowed image, **(c)** learned latent feature space without invariant loss, and **(d)** learn latent feature space with our two-stage learning. Our proposed setting achieves a significantly better numerical performance compared to the others. Interestingly, the guidance (*i.e.* coarse deshadowed) that provides the most pixel information performs worse than the guidance (*i.e.* invariant color map) that only provides a simple color map. After deeply looking at their visualization in Fig. 7.4, we observe that even coarse de-shadowed image still contains shadow boundary that may mislead the diffusion models, while the color map omits most shadow features, which demonstrates that only encapsulating shadow-free features is crucial for improving the performance. Correspondingly, our latent feature is acquired by minimizing the difference between the encoded features of shadow and shadow-free images, which implicitly omits shadow features, and it contains more perceptual features because we optimize it together with diffusion models for learning denoising. Therefore, it guides diffusion models with more shadow-free features and outperforms the compared methods.

Model Complexity Analysis. Our work focuses on adapting diffusion models to address shadow removal, and therefore we prioritize exploration over analysis of

Table 7.4: Effects of different types of diffusion model guidance that provides shadow-free priors.

		AISTD dataset		
Settings	Region	RMSE ↓	PSNR ↑	SSIM ↑
invariant color map [280]	*shadow*	*7.72*	36.24	0.986
coarse deshadowed [260]	*shadow*	8.03	35.74	*0.988*
$\bar{\mathcal{E}}_\theta(\mathbf{x}, m)$	*shadow*	7.59	*36.65*	0.984
$\mathcal{E}_\theta(\mathbf{x}, m)$ (Ours)	*shadow*	**5.92**	**37.70**	**0.990**

Table 7.5: Complexity comparisons of our distilled lighter model with the accelerated diffusion solver.

			AISTD dataset (RMSE)		
Method	params	time	shadow	non-shadow	all
BMN	**0.4M**	1.69s	5.69	2.52	3.02
G2R-ShadowNet	22.8M	0.36s	7.38	3.00	3.69
LFG-Diffusion	82.6M	2.76s	**5.15**	2.47	**2.90**
LFG-Diffusion (Distilled)	25.5M	**0.24s**	5.21	**2.34**	2.94

model complexity and inference time. However, we demonstrate the feasibility of our approach in terms of model complexity and inference time using advanced technologies such as those proposed in [295] for reducing model parameters without sacrificing performance, and [34] for accelerating diffusion sampling in Tab. 7.5. We find that even with similar settings, our lighter model outperforms compared methods with better restoration performance and is also faster.

ShadowDiffusion Comparison. Given the similarity between the recent ShadowDiffusion [296] (SD) and our method, which both characterize the shadow-free image distribution by conditioning diffusion models, ours further explores shadow removal at the instance level without any modifications to the model. The other difference

is majorly in the method complexity, *i.e.*, tackling challenges such as color-mixing and collapse, often arising from direct conditioning on shadow images. SD integrates a pre-trained shadow removal network. In contrast, ours models the shadow-free priors through two-stage learning and mitigates collapse using dense fusion modules. Tab. 7.6 demonstrates our efficiency in the shadow region of AISTD.

Table 7.6: Quantitative comparison with ShadowDiffusion.

Model	params(↓)	time(↓)	RMSE(↓)
ShadowDiffusion [296]	602.6M*	7.54s*	4.9†
LFG-Diffusion (Ours)	82.6M	2.76s	5.0‡

Chapter 8
Conditional Diffusion Models through Re-Noising

8.1 Introduction

In recent years, conditional image generation has received significant attention in the computer vision community. Some applications that make use of conditional image generation include text-to-image generation (*e.g.* DALLE-2 [57]) and image restoration (*e.g.* SR3 [210]). The most challenging part of these restoration applications comes from the ill-posedness, *i.e.*, the same degraded images may come from multiple different ground truth images. The ill-posedness affects the performance of traditional methods like sparse coding [297, 298] and makes it difficult for the learning-based algorithms to solve this problem. Although recent learning-based methods have made impressive progress [252], there remains a significant quality gap between the prediction and natural images.

Recent works that utilize pretrained generative networks have shown the superior visual quality of conditional generation compared to the aforementioned end-to-end learning methods. Generative models have shown impressive image generation results in terms of sample quality and diversity, indicating their capacity for encapsulating rich photorealistic priors. Some representative methods include Generative Adversarial Networks (GANs) [76], Variational Autoencoders (VAEs) [3], and Autoregressive models [299]. Their generation process generally starts from the standard normal distribution from which diverse high-fidelity images sampled [26, 91]. Recent work [300]

has shown that the *continuity* in the normal distribution remains preserved in the sampled results. For example, the results produced from two different Gaussian noises with the same model will be close to each other if the two noises are close to each other in Euclidean space. The continuity allows one to perform conditional image generation in an inversion manner that inverts degraded images into standard noises. This inverted noise can then generate clear images by projecting the noise with generative models. Following the protocol, multiple VAE- and GAN-based methods, including optimization- [301] and learning-based [300, 302–305] schemes have been proposed in various image restoration tasks.

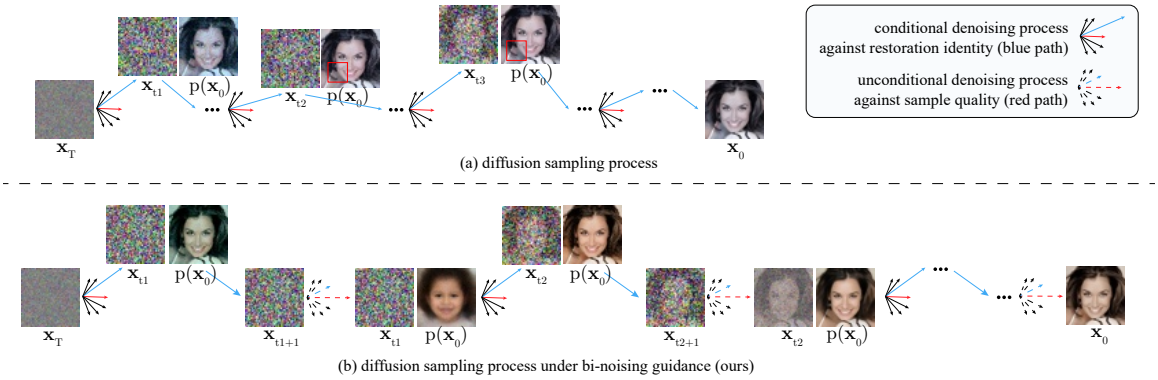


Figure 8.1: The graphical model showing the difference between the previous diffusion sampling process and ours with bi-noising guidance for colorization, where $\mathbf{x}_t$ is the noise of each diffusion process at timestep t, $p(\mathbf{x}_0)$ is the predicted noise-free start point of $\mathbf{x}_0$, and arrows indicate the denoising results of the diffusion models at each denoising process. Top figure shows how the conditional denoising process for colorization gradually accumulates the incorrect noise and results in artifacts. Instead, as shown on the bottom figure, the proposed additional noising and denoising steps diminish the incorrect noise and help in achieving better results.

Denoising diffusion probabilistic models [2, 85] are the most recent deep generative models. They have shown comparable and even better performance at image synthesis than GANs with delicate guidance [7]. These models learn to sequentially denoise

stochastic noise map starting from the normal distribution $\mathcal{N}(0, \mathbf{I})$ to clean images. However, the generation process is stochastic, and the continuity cannot be preserved from the initial sampled noise. For instance, two sampled noises from the same normal distribution with a small divergence may generate significantly different clear images. Such a noncontinuous generation process makes applying diffusion priors to be hard, and much efforts have been devoted into fully exploit such priors include recent works [161, 306–308].

In this work, we introduce an alternative method, named bi-noising diffusion, which is simple and easy to implement, for utilizing rich priors encapsulated in the unconditional pretrained diffusion models. Inspired by implicit sampling that was first developed in the denoising diffusion implicit models [6] for acceleration, we show that the implicit sampling using an unconditional pretrained diffusion model has a capacity for correcting the divergence of distributions modeled by the conditional diffusion models. Specifically, we make a coarse implicit prediction at each intermediate diffusion time step by sampling from the conditional model. We then sample the prediction back to the intermediate step with the forward diffusion process. Finally, we make a refined prediction by utilizing an unconditional model. Fig. 9.2 visualizes the bi-noising procedure and the error by predicting the noiseless start-point image $p(\mathbf{x}_0)$ of the noise image $\mathbf{x}_t$. Using this two-step procedure, one can utilize the embedded rich priors learned by the unconditional model and produce better-quality images. This hypothesis is further validated through extensive experiments demonstrating that the introduced method performs favorably against state-of-the-art conditional diffusion models.

8.2 Related Work

Iterative methods. Finding the corresponding latent code [301, 309–312] or sampled noise [88, 279, 313] of distorted images for restoration is one of the most straightforward ways of utilizing the generative priors. The intuition is that the pretrained generative models tend to produce natural results from their initial distribution. Thus the corresponding latent code or sampled noise can be projected to the restored images without additional optimization or learning. Menon et al. [301] proposed optimizing the latent code based on the difference between the generative results and the distorted images. Gu et al. [312] proposed to optimize multiple latent codes and compose them together for better visual quality. Similar iterative methods based on diffusion models have also been explored. Choi et al. [88] proposed to refine the sampled noise at each reverse diffusion step with the residual of distorted images. However, the applied stochastic iterative process tends to produce significantly different results though slightly changing its input. Therefore, these methods can only be applied to applications that do not require preserving the image identity.

Learning-based methods. Employing additional encoders [58, 300, 314–318] to predict the latent code is another promising way that can bypass the stochastic optimization issues. However, such a method is incompatible with the diffusion models since it is impossible to encode the distribution of each reverse diffusion process for models that employ many sampling timesteps. Existing works learn to model conditional generative restoration [86, 210] instead. Richardson et al. [300] proposed to encode images with a ResNet backbone into an extended $\mathcal{W}+$ latent space, which defines upon features of each input layer of the generative networks. Wang et al. [315]

proposed to encode images with a U-Net backbone and modulate the features of each generative layer of the generative networks. Saharia et al. [210] proposed to learn the noise distribution with the distorted image as the condition. Whang et al. [86] proposed to learn the generative process of residual given restored images. Compared with these GAN-based learning methods, a large number of sampling timesteps significantly increases the complexity of designing the corresponding encoders and thus makes the priors difficult to be learned.

Classifier guidance. Diffusion models have been using class information heavily to perform truncated or low-temperature sampling to increase the sample quality. The initial attempt [1, 7, 85, 319] is to incorporate a pre-trained classifier by using its gradients to guide the diffusion sampling process. However, it complicates the diffusion model because additional training is required for the classifier on noisy data. Classifier-free guidance [73, 320] is another approach for addressing the complexity issue. It alleviates the complexity by combining the existing network with the classifier for guidance, *e.g.*, Ho et al. [73] use conditional diffusion network with an empty condition, and Wang et al. [320] use pretrained segmentation with a null label. Nevertheless, the classifier fails at natural images, and its gradient is meaningless for restoration. Its strength parameters also become less reasonable for the almost definite restoration sampling process. In this paper, we are interested in incorporating the sampling quality superiority of the empty condition and the sampling guidance ability of degraded images. We show that the empty condition can bring the incorrect noisy image back into the high-quality manifold. Compared with the classifier and classifier-free guidance, our bi-noising guided diffusion process keeps the same complexity but better

fits the restoration task.

8.3 Proposed Method

In this section, we discuss the proposed mechanism to add the embedded priors to diffusion models. For consistency, we denote the intermediate output of the unconditional diffusion model as $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\cdot)$, parameterized by θ in the upcoming discussions following Denoising Diffusion Probabilistic Models [2] (DDPM). The additional, conditional diffusion model is denoted by $\boldsymbol{f}_{\boldsymbol{\phi}}(\cdot)$, the condition (*i.e.* degraded images) and natural image pairs are denoted by $\{\mathbf{x}_0, \mathbf{y}_0\}$, where the conditional diffusion model $\boldsymbol{f}_{\boldsymbol{\phi}}(\cdot)$ with parameters ϕ denoises noisy image $\mathbf{x}_t$ at timestep t with the concatenated condition $\mathbf{y}_0$.

8.3.1 Preliminaries

Diffusion probabilistic models belong to a new family of generative models [2, 7, 54, 85, 321] that can effectively model intractable distributions [85]. A diffusion process consists of two parts, *i.e.*, the forward process and the reverse diffusion process. In the forward diffusion process, a clean image is sampled from its data distribution and destroyed in T timesteps by repetitive noising using Gaussians of very small variances. Specifically, the forward process can be formulated as

$$\begin{aligned} q(\mathbf{y}_t|\mathbf{y}_{t-1}) &= \mathcal{N}\left(\mathbf{y}_t; \sqrt{\beta_t}\mathbf{y}_0, (1-\beta_t)\,\mathbf{I}\right) \\ &= \sqrt{\beta_t}\mathbf{y}_0 + \epsilon\sqrt{1-\beta_t}, \epsilon \sim \mathcal{N}(0, \mathbf{I}), \end{aligned} \tag{8.1}$$

or

$$\begin{aligned} q(\mathbf{y}_t|\mathbf{y}_0) &= \mathcal{N}\left(\mathbf{y}_t; \sqrt{\bar{\alpha}_t}\mathbf{y}_0, (1-\bar{\alpha}_t)\,\mathbf{I}\right) \\ &= \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \epsilon\sqrt{1-\bar{\alpha}_t}, \epsilon \sim \mathcal{N}(0, \mathbf{I}), \end{aligned} \tag{8.2}$$

where $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$ come from the variance schedule $\{\beta_1, \ldots, \beta_T\}$. The key idea here is that for large values of T, repetitive noising using Gaussians of small variances lead to a standard Gaussian, *i.e.*,

$$q(\mathbf{y}_T|\mathbf{y}_0) = \mathcal{N}\left(\mathbf{y}_T; 0, \mathbf{I}\right). \tag{8.3}$$

Now at each reverse timestep t, we attempt to reconstruct the noisy y_{t-1} from y_t using a distribution p modeled by a neural network with parameters θ. The parameters of the distribution $p_\theta(\cdot)$, found by optimizing variational lower bound of log-likelihood of $p_\theta(y_0)$, which is simplified by Ho et al. [2] by claiming that the major component in the objective comes from L_{t-1}, and the simplified loss is

$$L_{t-1} = E_{t\sim[1,T],\epsilon\sim\mathcal{N}(0,\mathbf{I})}\left[\|\epsilon - \epsilon_\theta\left(\mathbf{y}_t, t\right)\|^2\right]. \tag{8.4}$$

Here network $\epsilon_\theta(\cdot)$ models the noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ at each timestep t with the denoised one $\mathbf{y}_t$, which can be seen as the process of learning the gradient of distributions with score matching according to Song et al. [321]. Therefore, we can learn the impressive perceptual synthesizing capacity with the simplified loss function between noises.

8.3.2 Learning to Refine Diffusion Process

In our experiments, we denote the recent diffusion models [86, 210] that learn the diffusion process with conditions as the way of Learning to Refine Diffusion Process (LRDP). LRDP models the conditional distribution of a clean image given a degraded image for restoration learning, and thus it requires separate training for different tasks

or datasets. The objective for this learning process is formulated as

$$L_{\text{vlb}} := E_{t\sim[1,T],\epsilon\sim\mathcal{N}(0,\mathbf{I})}\left[\|\epsilon - \epsilon_\theta\left(\mathbf{x}_0, \mathbf{y}_t, t\right)\|^2\right], \tag{8.5}$$

where $\mathbf{y}_t \sim \mathcal{N}(\mathbf{y}_t|\sqrt{\bar{\alpha}_t}\mathbf{y}_0, (1-\bar{\alpha}_t)\mathbf{I})$. The network architecture in LRDP is a slightly changed version from the original U-Net found in DDPM, and the additional input $\mathbf{x}_0$ and $\mathbf{y}_0$ are concatenated and passed to the input layer. Similarly, the reverse diffusion process of LRDP is slightly changed from the original one and formulated as

$$\mathbf{y}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{y}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta\left(\mathbf{x}_0, \mathbf{y}_t, t\right)\right) + \sqrt{1-\alpha_t}\mathbf{z}, \tag{8.6}$$

where $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$ and $\alpha_t, \bar{\alpha}_t$ is the variant of the pre-defined variance schedule $\{\beta_1, \ldots, \beta_T\}$, that is $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$. Since the diffusion process conditions on the specific type of degradation $p(\cdot)$ that produces degraded image $\mathbf{x}_0$ given clear image $\mathbf{y}_0$ as $p(\mathbf{x}_0|\mathbf{y}_0)$, LRDP needs re-training from scratch for different restoration tasks, which further heightens the training cost.

While LRDP offers the advantage of modeling the joint distribution $p(\mathbf{x}_0, \mathbf{y}_0)$, it inherently introduces higher uncertainty compared to modeling the unconditional distribution $p(\mathbf{x}_0)$. This increased uncertainty can lead to significant performance degradation, particularly when the observed data deviates from the estimated distribution [322]. Therefore, we posit that decoupling the diffusion generation process into distinct protocols, rather than relying on a potentially mismatched joint distribution.

8.3.3 Conditioning on Diffusion Process

The recent work [88, 279] falls into another category of utilizing the diffusion process, which uses a pretrained DDPM and changes its reverse diffusion process with distorted images by Conditioning on Diffusion Process (CDP). A similar way was previously

explored in the other generative models, *e.g.*, mGANprior [312] and PULSE [301] invert a trained GAN by optimizing its latent code. However, CDP does not require optimization compared with the previously mentioned GAN-based methods. In contrast, it ensembles the conditions during sampling as

$$\hat{\mathbf{y}}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{y}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta\left(\mathbf{y}_t, t\right)\right) + \sqrt{1-\alpha_t}\mathbf{z} \tag{8.7}$$

$$\mathbf{y}_{t-1} = \hat{\mathbf{y}}_{t-1} + \boldsymbol{\sigma}(x_0, \hat{\mathbf{y}}_{t-1}), \tag{8.8}$$

where $\boldsymbol{\sigma}(\cdot)$ is a handcrafted transformation which aims at combining $\mathbf{x}_0$ with $\hat{\mathbf{y}}_{t-1}$ for accurate restoration. For example, Choi et al. [88] proposed to downsample $\mathbf{x}_0$ and $\hat{\mathbf{y}}_{t-1}$ and take their residual as the conditioning, while Lugmayr et al. [279] proposed to sum the visible region of $\mathbf{x}_0$ with the invisible region $\hat{\mathbf{y}}_{t-1}$ for the inpainting task.

Though CDP avoids the heavy training cost and is suitable for some conditional generation tasks like restoration with minimal modifications, its performance highly depends on the amount of degradation in the conditioned images. For example, when the conditioned images suffer from high amounts of distortion for face image restoration, CDP cannot preserve the face identity and tends to generate pseudo-sharp results with fake details. These fake details introduce further ill-posedness to the restored images and greatly limit the applications of such methods. Therefore, we propose refining the denoised results for correcting such artifacts at each step.

8.3.4 Implicit Error-feedback Diffusion Priors

Since the diffusion models follow a time-sequential process, the error in each step and the visual artifacts propagate and add up, hence severely degrading the quality of some CDP results. However, such issues are rarely observed in the unconditional

diffusion models. We argue that the difference comes from conditioning breaking the inherent probabilistic distribution of noises at each sampling timestep, causing them to deviate from the manifold of natural images. Therefore, we propose to apply generative priors embedded in a pretrained unconditional model to regularize the noise predicted at each timestep from the conditional model. The trained diffusion model with conditioning denoted as $\boldsymbol{f}_{\phi}(\cdot)$ takes as input the predicted image of the previous timestep and makes an implicit prediction $\tilde{\mathbf{y}}_0$ defined by

$$\tilde{\mathbf{y}}_0 = (\mathbf{y}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{f}_{\phi}(\mathbf{x}_0, \mathbf{y}_t, t))/\sqrt{\bar{\alpha}_t}. \tag{8.9}$$

Here y_t denotes the prediction at the previous timestep. We then estimate the noisy version of the implicit prediction, which undergoes further regularization from an unconditional diffusion model. Please note that the unconditional diffusion model that fits the inherent probabilistic distribution. The diffusion process $\mathbf{y}_t \sim q(\mathbf{y}_t|\tilde{\mathbf{y}}_0)$ with $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\cdot)$ is formulated as

$$\begin{aligned} q(\mathbf{y}_t|\tilde{\mathbf{y}}_0) &:= \mathcal{N}(\mathbf{y}_t|\sqrt{\bar{\alpha}_t}\tilde{\mathbf{y}}_0, (1-\bar{\alpha}_t)\mathbf{I}) \\ &= \sqrt{\bar{\alpha}_t}\tilde{\mathbf{y}}_0 + \epsilon\sqrt{1-\bar{\alpha}_t}, \epsilon \sim \mathcal{N}(0, \mathbf{I}), \end{aligned} \tag{8.10}$$

and

$$\mathbf{y}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{y}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}\left(\mathbf{y}_t, t\right)\right) + \sigma_t \mathbf{z}. \tag{8.11}$$

Following this procedure brings in an inherent regularization to the output of the conditional model during the reverse diffusion process. Note that Equation equation 8.11 takes the noised version $\mathbf{y}_t$ sampled from $\tilde{\mathbf{y}}_0$ as input. It is similar to the original reverse diffusion process, which takes the noised version of natural images as input.

In summary, we utilize two diffusion models for conditional image generation. The unconditional diffusion model regularizes the predicted outliers at each prediction

timestep of the conditional diffusion model in an error-feedback way. Moreover, for the complex real-world application like draining where domain gaps may exist, we further discuss the details of applying our bi-noising diffusion with slight modifications to achieve better performance.

One observation from our experiments on image deraining while training by direct conditioning like in SR3 [210] was that the restored images suffered from artifacts and color channel shift which can be seen in Fig. 8.4. On further investigation, we found that this is due to incorrect conditioning of input during the training process. Specifically, for the task of image restoration with source-target pairs denoted as $(\mathbf{x}_0, \mathbf{y}_0)$, existing methods optimize the weights of the network $\boldsymbol{\epsilon_\theta}(\cdot)$ modelling the reverse process of diffusion, by minimizing the L_{simple} function defined in [2] as

$$L_{\text{simple}} := E_{t\sim[1,T],\epsilon\sim\mathcal{N}(0,\mathbf{I})}\left[\|\epsilon - \epsilon_\theta\left(\mathbf{x}_0, \mathbf{y}_t, t\right)\|^2\right]. \tag{8.12}$$

The training objective L_{simple} holds the inherent assumption that during inference time $\mathbf{y}_t$, (i.e. the reconstructed image at time t) is close to the clean target. But for extreme cases where the intermediate diffusion outputs are not accurate during the initial steps of diffusion, the rain streaks continue to propagate through the diffusion process, as can be seen in Fig. 8.4. This is because, inherently, the diffusion model works by predicting the noise present in $\mathbf{y}_t$ than the amount of degradation in it. To account for this, we add a correction prior L_{corr} so that the network can give equally good output for high distortion levels. This term is defined by,

$$L_{\text{corr}} := \alpha_t \left\|\epsilon_\theta\left(\mathbf{x}_t, \mathbf{x}_0, t\right) - \epsilon_\theta\left(\mathbf{y}_t, \mathbf{x}_0, t\right)\right\|^2. \tag{8.13}$$

The final objective for training the network is,

$$L_{\text{final}} = L_{\text{simple}} + \lambda_{corr} L_{\text{corr}}. \tag{8.14}$$

The value of λ_{corr} is empirically set equal to 0.001 for all experiments.

8.4 Experiments

To demonstrate the restoration capacity of our method, we evaluate our method with several experimental settings following the most representative diffusion models, *i.e.*, ILVR [88] and SR3 [210] based on the Guided-diffusion architecture [319]. Following the common practice that pixel-wise metrics, *i.e.*, PSNR and SSIM cannot comprehensively denote the visual quality of restored results, we utilize FID and LPIPS as the additional metrics for evaluation. The tasks in which we evaluate our method on are

- Conditional image restoration which is trained on the FFHQ [188] dataset (70000 images) and evaluated on the CelebA-HQ [323, 324] dataset (first 3000 images) with a resolution of 256×256 pixels.
- Conditional image restoration which is $4\times$ face super-resolution trained on the FFHQ [188] dataset and evaluated on the CelebA-HQ [323, 324] dataset (first 3000 images) with a resolution of 256×256 pixels.
- Image turbulence removal follows the turbulence simulation settings [97] on the FFHQ dataset and conducts evaluation on the real long-range imaging images [241].
- Image deraining which is conducted on the Rain800 [325] dataset and Jorder 200L [326] dataset with their respective train sets. The diffusion models conduct in a resolution of 256×256 pixels.

Note that for the first three tasks, the diffusion models are trained on the FFHQ

dataset for face generation. For the last task, the diffusion models are trained on the ImageNet dataset for natural image generation. The unconditional model utilized has never seen the validation dataset during its training process for all of these cases.

Figure 8.2: Colorization visual result comparisons corresponding to the CelebAHQ dataset. We acquire gray images by averaging their {R,G,B} channels and take them as the restoration input. The *ILVR Diffusion* results come from the inference results of its pretrained face diffusion model. The *Palette* results are acquired from our re-implemented diffusion model, which is trained on the FFHQ dataset and followed the settings mentioned in their paper.

8.4.1 Colorization

Colorization aims at reconstructing grayscale images with colors that are fitted to natural statistics and image semantics. The grayscale image is obtained by averaging the values at red, green, and blue channels of the corresponding colour image. We

empirically observed that conditional denoising diffusion models fail at colorization. Even though they can preserve the fine-grained details, unnatural colors always exist in their reconstructed results. In contrast, the method that adopts our proposed bi-noising diffusion is capable of correcting the reconstruction with more semantics and accurate color descriptions. The quantitative performance comparison is shown in Tab. 8.1, where our method achieves 7.906dB higher PSNR than the one without pretraining. The visual results in Fig. 8.2 further clarify the improvements that come from more globally consistent colors and tones of our results, even though the pretraining had never seen the ground truth before. In contrast, a similar method, i.e., ILVR cannot deal with the colorization task even though it also utilizes a pretrained unconditional model, which demonstrates the superiority of our proposed DDRP in such tasks. Therefore, we argue that utilizing the priors plays a crucial role in ensuring the color naturalism.

Method	PSNR ↑	SSIM ↑	FID ↓	LPIPS ↓
ILVR Diffusion [88]	18.3936	0.5674	86.2642	0.5008
SR3 Diffusion [210]	*19.1647*	*0.8680*	*13.8126*	*0.2959*
Bi-Noising (Ours)	**27.0707**	**0.9531**	**12.6796**	**0.1417**

Table 8.1: Colorization results corresponding to the CelebAHQ dataset. The best and second-best performance is indicated with **blod** and *italic* respectively. We use ↑ and ↓ to suggest higher/lower score should be achieved by better methods.

8.4.2 Face Super-resolution

Face super-resolution is the other representative task in image restoration, and it is widely evaluated in the other denoising diffusion-based restoration works. We follow the experimental settings of SR3 and ILVR, i.e., restore 256×256 face images from

Figure 8.3: $4\times$ **super-resolution visual result comparisons corresponding to the CelebAHQ dataset.** The low-resolution input is dowsampled by using bicubic interpolation.

64×64 face images downsampled by Bicubic interpolation. From Fig. 8.3, one can notice that our method achieves the best visual quality compared with the other methods. Compared with the state-of-the-art face super-resolution method based on GAN priors, our method better preserves the identity of the restored face images. As can be seen from Tab. 8.2, our method significantly outperforms the other methods in terms of the distortion measures, i.e., PSNR and SSIM with 4.8316 dB and 0.04 better than the second one. Though our results in the FID metric are not better than ILVR, FID doesn't denote the reconstruction accuracy that is crucial for super-resolution.

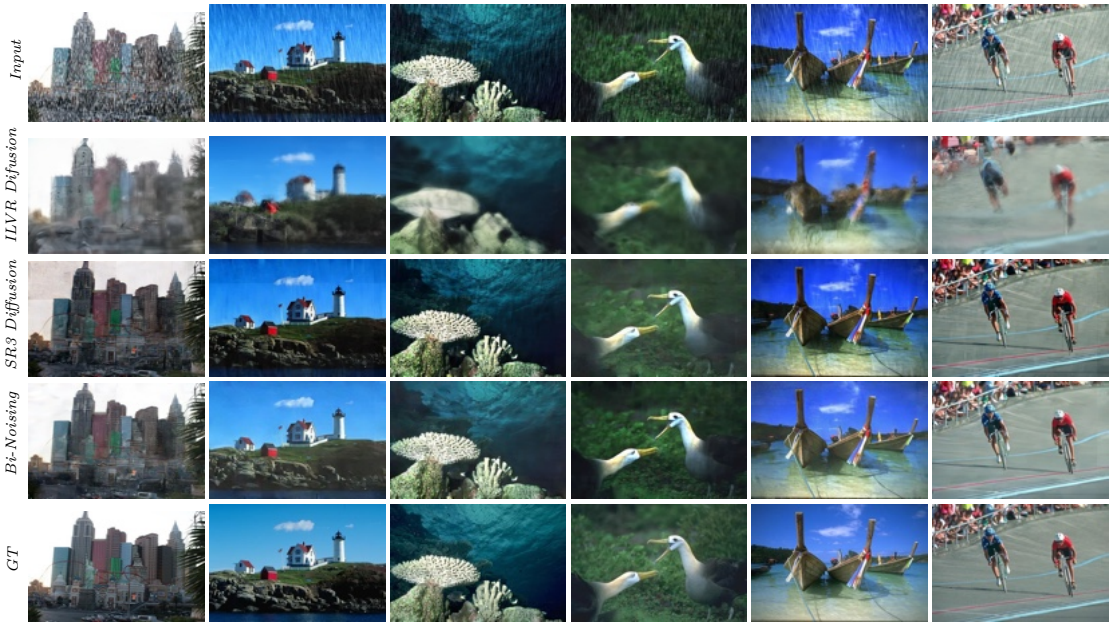


Figure 8.4: Deraining visual result comparisons corresponding to the Rain 800 dataset.

Therefore, the above results demonstrate our method.

Method	PSNR ↑	SSIM ↑	FID ↓	LPIPS ↓
PULSE [301]	*23.5769*	*0.6794*	31.2309	0.3832
ILVR Diffusion [88]	22.5374	0.6150	**20.4621**	0.3393
SR3 Diffusion [210]	22.8290	0.6442	29.8932	*0.3350*
Bi-Noising (Ours)	**29.3996**	**0.8414**	*24.5632*	**0.1809**

Table 8.2: 4× CelebAHQ super-resolution results.

8.4.3 Image Deraining

We perform single image deraining on two popular draining datasets. Namely, the Jorder 200L dataset which contains large rain streaks, and the Rain800 dataset which contains realistic rain. To isolate the contributions of our proposed modules and ensure a fair comparison, we perform comparisons after retraining the network proposed for

super-resolution in the literature. Specifically, we perform comparisons with ILVR diffusion [88] and conditional diffusion models, and we include the improvements brought about by our modules. To evaluate the reconstruction quality, we use the PSNR and SSIM metrics. To assess the quality of images produced by various methods, we use LPIPS and NIQE as metrics. As we can see from Tab. 8.3, the proposed conditioning loss functions bring significant improvement for all metrics in the JORDER 200L dataset [326], obtaining about 2.45 dB PSNR over the exiting method as well as giving realistic natural images. The visual comparisons in Fig. 8.4 further demonstrate our method on the visual quality compared with the other methods.

	Jorder 200L dataset			
Method	PSNR ↑	SSIM ↑	LPIPS ↓	NIQE ↓
Rain Images (Input)	26.70	0.8439	0.2411	4.131
ILVR Diffusion [88]	21.22	0.4942	**0.0972**	6.467
SR3 Diffusion [210]	*31.45*	*0.9091*	*0.1779*	*3.588*
Bi-Noising (Ours)	**33.90**	**0.9555**	**0.0972**	**3.232**

Table 8.3: Restoration results comparison on the Jorder 200L dataset with the other re-trained diffusion models.

	Methods				Ours		
Settings	Ho et al. [2]	Dhariwal et al. [7]	Nichol et al. [319]	Ho et al. [73]	w/o parametric	w/o full guidance	Bi-Noising
classifier guidance [7]		✓					
CLIP guidance [319]			✓				
classifier-free guidance [73]				✓	✓		
alternative guidance						✓	
Bi-Noising					✓	✓	✓
PSNR ↑	19.16	20.10	23.14	25.91	26.46	*26.81*	**27.07**
Parameters (M) ↓	**93.6**	147.7	243.2	**93.6**	**93.6**	187.2	187.2
Running Time (s) ↓	**1.6**	4.9	3.4	3.1	3.1	*2.3*	3.1

Table 8.4: Result comparisons between different prior parameterizations.

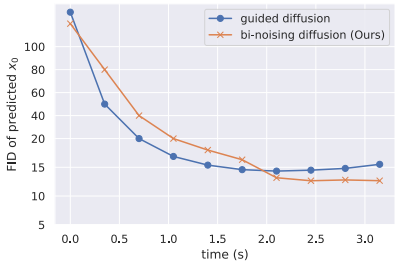


Figure 8.5: FID v.s. Time.

8.4.4 Turbulence Removal

We plug the proposed bi-noising approach into the recent diffusion restoration work [97] to demonstrate the applicability of our method on an extremely ill-posed atmospheric

turbulence mitigation problem [189]. Compared with the diffusion network with single noise conditioning, the results shown in Fig. 8.6 validate that our bi-noising method is able to remove the unnatural textures from the face images resulting from the incorrect denoising results.

Figure 8.6: Atmospheric turbulence mitigation results corresponding to the LRFID dataset [97].

8.4.5 Design Analysis

Nonparametric v.s. Parametric Priors. Inspired by Ho et al. [73], here we analysis the effect of alleviating complexity by parameterizing unconditional models into the conditional restoration model, denoted as *Nonparametric Prior* in Tab. 8.4. The compared methods use the same diffusion model but different guidance settings as the prior for fair comparisons. Specifically, we use a single diffusion model that takes conditions for restoration, and it takes a null token $\varnothing$ for unconditional generation. From Tab. 8.4, we can conclude that the nonparametric prior, *i.e.*, w/o parametric, significantly reduces half of parameters for diffusion sampling, while the model suffers

0.26 dB performance drop compared with the parametric prior, *i.e.*, Bi-Noising, that is our final setting. The reason is that the null token increases the diffusion model training difficulty and thus the model fits worse than the unconditional model used in our final setting. Compared with classifier guidance [7] and clip guidance [319], which are designed for enhancing conditions and orthogonal to our method, our method with bi-noising sampling outperforms them in face super-resolution, even though ours formalizes the model in a similar way as them. This clearly demonstrates the benefits of our parametric prior that can encapsulate the low-level information distribution for restoration. Fig. 8.5 further demonstrates the efficiency of our method.

Figure 8.7: Deraining visual result comparisons that demonstrate the improvement brought by our L_{corr} component.

Priors Correlation. For the complex applications like deraining, we introduce additional correlation priors to further boost the final results. In Tab. 8.5, we present the ablation study of the introduced correlation priors to demonstrate its effectiveness. Since the rain streaks in the JORDER 200L dataset are relatively small, we choose the

Method	PSNR ↑	SSIM ↑	LPIPS ↓	NIQE ↓
Rain Images (Input)	26.70	0.8439	0.2411	4.131
SR3 Diffusion [210]	31.45	0.9091	0.1779	3.588
Ours	*33.23*	*0.9505*	*0.1043*	*3.285*
Ours + L_{corr}	**33.90**	**0.9555**	**0.0972**	**3.232**

Table 8.5: Ablation study on each introduced component.

Rain800 dataset for a fair experiment. The ablation starts with the base model and then adds the two priors one by one to show the improvements. From the improved results due to L_{corr}, we can conclude that the introduced correlation priors allow our diffusion priors to better fit the probabilistic distribution of complex images, which ultimately benefits conditional generation with more realistic results. The comparisons presented in Fig. 8.7 also visually validate the conclusion.

8.4.6 Application on Latent Diffusion Models

The photo-realistic prior in the recent latent text-to-image model [9, 327] has been demonstrated in learning-based image restoration, such as StableSR [328]. We show that our method even though originally built upon the image space, can be easily extended into the latent space by modifying the assumption about image pairs $\{\mathbf{x}_0, \mathbf{y}_0\}$ into the pairs produced by the latent encoder. Compared with the original StableSR on the synthetic Real-ESRGAN benchmarks (*i.e.*, DIV2K Valid), our method can generally improve the StableSR in perceptual metrics like FID and MUSIQ shown in Tab. 8.6. Moreover, our implementation adopts the nonparametric implementation discussed in Sec. 8.4.5, which uses the same number of parameters as the StableSR and leads to a fair comparison with StableSR.

Method	PSNR ↑	SSIM ↑	FID ↓	MUSIQ ↑
StableSR [328]	23.26	**0.5726**	24.44	65.92
Bi-Noising (Ours)	**23.35**	0.5721	**23.60**	**67.12**

Table 8.6: Super-resolution results corresponding to the Real-ESRGAN benchmark on the DIV2K validation set.

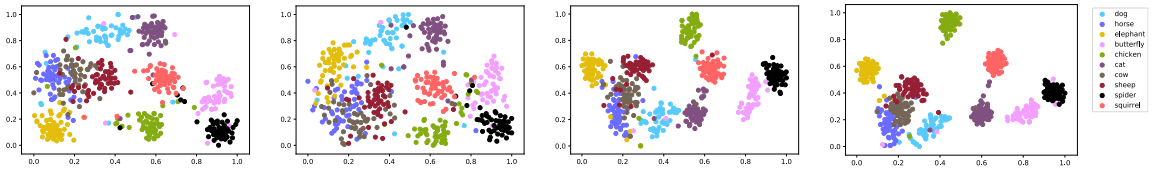
Figure 8.8: Visual comparisons between StableSR and ours on the Real-ESTGAN super-resolution data.

Chapter 9

Deep Semantic Statistics Matching Denoising

9.1 Introduction

Deep learning based methods [329–331] have achieved a dramatic leap in the performance of various image restoration tasks. Typically, they employ a Convolutional Neural Network (CNN) on a set of image pairs, consisting of degraded images and corresponding clear images, for restoration learning. By maximizing the correspondence between each pair of the CNN-restored results and the clear image, the CNN is trained to map images from the degraded domain into the clear domain. However, blur issues always existed in such a manner. Recent work [232] called perceptual loss finds that maximizing the correspondence in the semantic feature space of pre-trained large-scale classification network (*e.g.*, VGG [332] network trained on ImageNet [333]) leads to better visual quality [206, 248]. A more widely used strategy inspired by GANs [334], which employs a discriminator to implicitly enforce the distribution of restored images to be consistent with the distribution of clear images in terms of KL- and JS- divergence, can largely improve the perceptual quality of restored images. But the training procedure is often unstable, mostly because the objective is a zero-sum non-cooperative game that cannot be easily solved. Thus, it is straightforward to wonder whether it is possible to combine the pre-trained large-scale networks in an adversarial or statistical manner to bypass their drawbacks and avail their advantages together?



w. $\mathcal{L}_1$ w. $\mathcal{L}_{Perceptual}$ Ours Clear Images

Figure 9.1: t-SNE of Denoised Images in the Semantic Feature Space By exploiting t-SNE [335] to reduce dimensions of semantic features and project them into 2D coordinates, we visualize the distributions of denoised animal images in the semantic feature space. Ours preserves most semantics as the clear images.

To answer the above question, we first look at the training procedure from a probabilistic view in the semantic feature space (the space of extracted semantic features of each images), which contains many clusters of different semantics. Figure 9.1 visualizes these clusters with t-SNE [335] in animals images selected from ImageNet [333]. An image that belongs to a specific cluster and owns intrinsic semantics is called Single-Semantic Image in this paper. For example, animal images are single-semantic images, because they have common semantics of species, even though individual animals with the same specie look different. Intuitively, restored single-semantic images should preserve the same distribution as the corresponding clear single-semantic images in the semantic feature space. However, the objectives of existed methods for denoising learning cannot preserve this, as w.$\mathcal{L}_1$ and w.$\mathcal{L}_{Perceptual}$ show in Figure 9.1. Therefore, we argue that minimizing the divergence between the probability distributions estimated in the restored domain and the one estimated in the clear domain should be a potential promising solution. Similar idea is also validated by MMDGAN [336] and its variants for generating face and bedroom images, but the idea is rarely researched in the restoration literature.

Different from the single-semantic image, natural images like a cityscape image often consists of multiple objects, and can not be easily identified by a single semantic, here it is called Complex-Semantic image. The semantic feature extracted from such an image may not belong to any simple semantic clusters, but resides on an extremely complicated manifold in the semantic space. For example, cityscape images usually consist of objects/regions of different semantics like *road*, *sidewalk*, and *building*, as classified by the Cityscapes dataset [337]. To approximate and compare the probability distributions of complex-semantic images is nontrivial due to their unique intrinsic uncertainty of semantics.

In the paper, we propose a new distribution-wise objective for denoising learning, and is capable of being extended into general restoration tasks, towards both the single-semantic images and complex-semantic images. It learns to preserve the probability distribution of denoised images in the semantic feature space, and it is called as Deep Semantic Statistics Matching (D2SM) Denoising Network. The objective of D2SM exploits a way similar to Kernel Density Estimation (KDE) [338] to implicitly estimate the probability distributions of semantic features from a set of denoised images and clear images, and then Kullback-Leibler (KL) divergence between two distributions is used as the objective. Here, one of our major novelty comes from the way of density estimation, where we model the probability distributions based on internal patches from a single complex-semantic image or multiple single-semantic images. The way of availing internal patches tends to be more appropriate than modeling the distribution of multiple complex-semantic images. Such a phenomenon is also proved in recent work [339] suggests that the internal visual entropy of a single image is much

smaller than multiple images. Therefore, we propose to use the divergence of patch distributions to guide the learning, called Patch-Wise Internal Probability.

Nevertheless, statistically estimating the density of patches conducted in a single mini-batch usually requires a great large number of samples. To maintain the trade-off between the computational cost and accuracy, another major novelty of the paper, called Memorized Historic Sampling, is proposed, inspired by recent contrastive learning related works [340, 341]. By simply leveraging the statistics among the mini-batch and memorized historic mini-batch in queues, we demonstrate that D2SM significantly outperforms the same network backbone with perceptual loss and other state-of-the-art objectives, without additional information or parameters. Empirical evaluation validates that D2SM largely improves not only the effectiveness of denoising, but also super-resolution and dehazing, and hence it should be able to be generally applied to different tasks and network architectures.

Our contributions are therefore three-fold:

(i) We propose D2SM for image denoising learning, which minimizes the distribution divergence instead of the sample-to-sample distance in the semantic feature space. (ii) D2SM is adapted to complex-semantic images in a patch-wise manner, which can decompose complex semantics in natural images for efficient distribution approximation. (iii) Extensive experiments are conducted to demonstrate that D2SM substantially outperforms the original perceptual loss and other state-of-the-art losses, without modifying the network architecture or accessing the additional data. The superior accuracy in high-level vision tasks further validates that D2SM indeed transfers

semantics for restoration.

9.2 Related Work

Resulted by the emergence of deep neural networks, recent CNN based methods have led to a dramatic leap in image restoration. Among them, most works utilize the pixel-wise similarity metrics as their objective, *e.g.*, $\mathcal{L}_1$ and $\mathcal{L}_{MSE}$. Though higher performance in metrics like PSNR or SSIM [342] is achieved by using these loss functions, recent work [248] finds that these metrics do not reflect human perceptual preferences. In contrast, results generated by CNNs trained with the perceptual objective are more closely correlated with the human judgment [206]. These methods measure the similarity of two images in the pre-trained high-level vision networks, usually VGG classification network [332] trained in ImageNet [333]. Different perceptual objectives have been proposed in this category, *e.g.*, $\mathcal{L}_{MSE}$ of features [232, 248, 343–346], contextual objective [205, 206], and semantic label [347, 348]. However, these methods lack a reasonable explanation of the effectiveness led by the perceptual objective [248]. Furthermore, the frozen network pre-trained on certain datasets, *e.g.*, ImageNet, is not appropriate for the image restoration tasks conducted on the large-scale, diverse natural image datasets [349] or specific semantic image datasets [337, 350, 351]. Here we hypothesize that these issues come from the objectives that estimate the sample-to-sample distance in the feature space. By exploiting the characteristics of single-semantic patches from natural images, which can be associated with an embedded manifold, we implicitly measure the divergence of the probability distributions estimated from restored images and clear images in the semantic feature space, and we use it as the

objective to bypass the above issues.

Similar ideas that minimize the distribution divergence instead of the sample-to-sample distance have been proposed before. In the area of image restoration, Contextual loss [205, 206] that proposed for misaligned image transformation implicitly minimizes the divergence between restored images and clear images. It approximates the divergence by the contextual relationships within patches from a single image (*i.e.* single image statistics [339]), and hence enables image-to-image translation to be conducted on the misaligned image pairs. However, its performance is usually limited by the low accuracy of feature matching [352] and leads to worse restoration performance in aligned image restoration learning. More similar works that avail statistical features are GMMN [353] and GFMN [354]. They achieve the generative ability without adversarial learning in the problematic min/max game. Nevertheless, they are not designed for the image restoration learning that majorly consists of natural images with diverse appearance, and the desired superiority cannot be gained here. In the area of domain adaption, minimizing the statistics feature difference of the high-level vision networks can help networks adapt to unseen domain directly, like CORAL [355] and MMD [356]. However, these methods require semantic labels, which is not practical in the real-world image restoration datasets. By exploiting the internal statistics [339] of natural images, our proposed method successfully facilitates the restoration learning through more accurate divergence approximation.

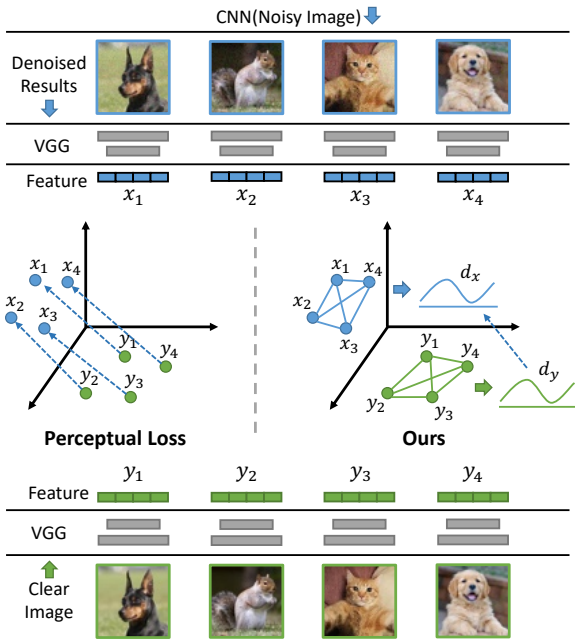


Figure 9.2: Perceptual loss vs. Ours. We minimize the distribution divergence between a set of restored images and the corresponding clear images, instead of the sample-to-sample distance, in the semantic feature space (*e.g.* the penultimate layer of VGG). This procedure better simplifies the restoration learning and ameliorates underfitting compared with the perceptual loss.

9.3 Method

Let $\mathcal{X} \subset \mathbb{R}^{H\times W\times C}$ denotes the domain of degraded images caused by factors like noising, and $\mathcal{Y} \subset \mathbb{R}^{H\times W\times C}$ denotes the domain of corresponding clear images. We wish to restore $x \in \mathcal{X}$ to appear like its corresponding target image $y \in \mathcal{Y}$ by using a denoising network $G(\cdot)$ that outputs $\tilde{y} = G(x)$. To force the outputs $\tilde{y}$ maintains as much the perceptual detail as possible, recent works [205, 206, 232, 343, 344] exploit the pre-trained high-level vision networks (*e.g.*, the intermediate layer of VGG), denoted as $\Phi(\cdot)$, to guide the restoration learning by minimizing the similarity between $\tilde{y}$ and y in the feature space of $\Phi(*)$. This can be formulated as the objective

with the similarity metric $D(\cdot)$:

$$\mathcal{L}(x, y, G) = D(\Phi(y), \Phi(G(x))). \tag{9.1}$$

In practical, the similarity metric $D(\cdot)$ is usually implemented by Mean Square Error (MSE) or Contextual Distance [205].

Contrastively, we take the denoising learning as minimizing the divergence of probability distributions estimated by denoised images and clear images in the semantic feature space. Given N samples of image pairs that consist of $T_x = \{x_1, x_2, \ldots, x_N\}$ and $T_y = \{y_1, y_2, \ldots, y_N\}$, we incorporate the mutual information [357] of them in the feature space of $\Phi(G(\cdot))$ into the restoration learning. Such a manner is empirically proven to be effective to facilitate knowledge transferring [358–364]. By minimizing the divergence of the estimated probability distribution between samples T_x in $\Phi(G(\cdot))$ and T_y in $\Phi(\cdot)$, denoted as $\mathcal{G}'$ and $\mathcal{G}$, we force $G(\cdot)$ to better maintain the geometry of the feature space $\Phi(\cdot)$ estimated in the clear image domain $\mathcal{Y}$. In doing so, we formulate the final objective as

$$\mathcal{L}(T_x, T_y, G) = \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} g'_{j|i} \log(\frac{g'_{j|i}}{g_{j|i}}). \tag{9.2}$$

To elaborate, we will detail the divergence approximation in Section 9.3.1, and the sampling strategy in Section 9.3.2 and Section 9.3.3.

9.3.1 Probability Distribution Divergence

Here we model the correlation of samples from the same domain in the semantic feature space as the probability distribution. Several methods have been proposed for modeling the correlation, including, but not limited to, probabilistic based [358, 364], embedding based [359, 363], graph based [362], and more [361]. In this work,

we exploit the kernel density estimation to estimate the probability distribution of samples in the semantic feature space, which describes the probability of each sample to select its neighbors [335]. It is empirically proven to be effective for describing the geometry of feature space by Passalis et al. [358, 364]. To elaborate, we denote the probability distribution between any two samples i, j from the clear domain as $g_{i|j}$ and the restored domain as $g'_{i|j}$. Based on the extracted feature f^x and f^y from $\Phi(G(\cdot)))$ and $\Phi(\cdot)$, the probability distributions are estimated by:

$$g'_{i|j} = \frac{K_{cosine}(f_i^x, f_j^x)}{\sum_{k=1,k\neq j}^{N} K_{cosine}(f_k^x, f_j^x)} \in [0,1], \tag{9.3}$$

and

$$g_{i|j} = \frac{K_{cosine}(f_i^y, f_j^y)}{\sum_{k=1,k\neq j}^{N} K_{cosine}(f_k^y, f_j^y)} \in [0,1], \tag{9.4}$$

where the cosine kernel function K_{cosine} is employed for estimating the probabilty distribution, formulated with two vectors a and b as:

$$K_{cosine}(a,b) = \frac{1}{2}(\frac{a^\top b}{||a||_2||b||_2} + 1) \in [0,1]. \tag{9.5}$$

As Turlach et al. [365] suggested, this kernel function avoids the bandwidth choosing in Gaussian kernel, and it boosts performance compared with the Euclidean measures as Wang et al. [366] suggested.

To minimize the difference of two estimated probability distributions, we avail the Kullback-Leibler (KL) divergence as the similarity metric, formulated as:

$$D_{KL}(\mathcal{G}'||\mathcal{G}) = \int_{\mathbf{t}} \mathcal{G}'(\mathbf{t}_x) \log \frac{\mathcal{G}'(\mathbf{t}_x)}{\mathcal{G}(\mathbf{t}_y)} d\mathbf{t}. \tag{9.6}$$

where $\mathbf{t}_x \in \mathcal{X}$ and $\mathbf{t}_y \in \mathcal{Y}$. In practical implementation, we avail the mini-batch that consists of N samples for approximation, aiming for acceleration in a parallel fashion.

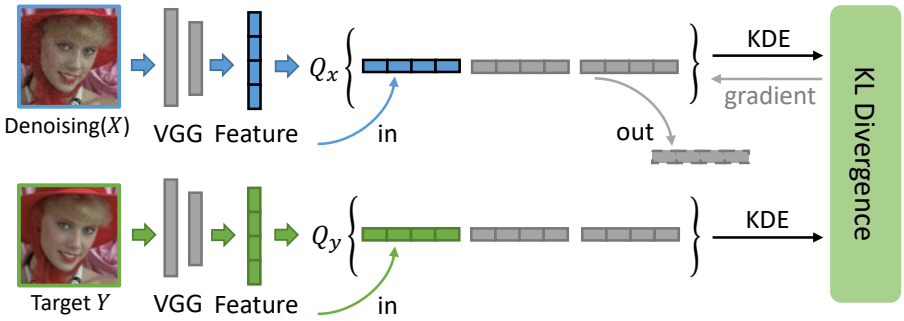


Figure 9.3: Sampling with Historic Gradients. We approximate the divergence with historic sampling by using two queues to bypass the GPU memory limits.

9.3.2 Memorized Historic Sampling

Intuitively, the number of selected samples in a mini-batch should be as large as possible during training. However, in practical implementation, increasing the number of samples is greatly limited by the GPU memory. Such a limitation is more serious in the extracted semantic feature space, and hence greatly limits the effectiveness of our method.

To bypass the limitation, we introduce a Memorized Historic Sampling strategy, visualized in Figure 9.3. It maintains two *queues* of feature samples, *i.e.*, $Q^{\mathcal{X}}$ and $Q^{\mathcal{Y}}$ that can store historic features from previous mini-batches with limited GPU memory cost. In doing so, we can estimate the probability distributions among queues instead of mini-batches. Therefore, it allows a larger number of samples and a relatively smaller mini-batch used at runtime. The queue is updated according to the First-In-First-Out rule, which enforces the historical samples in the queue are always newest, and hence it allows the probability distribution to be more consistent with the immediate state.

Based on such a strategy, we can formulate Equation 9.3 as below:

$$g'_{i|j} = \frac{K_{cosine}(Q_i^{\mathcal{X}}, Q_j^{\mathcal{X}})}{\sum_{k=1,k\neq j}^{q} K_{cosine}(Q_k^{\mathcal{X}}, Q_j^{\mathcal{X}})} \in [0,1], \tag{9.7}$$

and

$$Q_{1...N}^{\mathcal{X}}, Q_{N...q}^{\mathcal{X}} \leftarrow f_{\{1...N\}}^{x}, Q_{\{1...q-N\}}^{\mathcal{X}}. \tag{9.8}$$

where q is the queue size of the appied queue for extracted features f_x from a sigle mini-batch with the number of N, and $q \gg N$.

For example, the maximum size of a mini-batch can only be 32 in a single GPU card with a memory of 12GB, but the number of samples is usually set as 128 to ensure the accuracy of the estimation, which is not practical in a single GPU. By using the queue in the size of 128, we can directly use the current mini-batch with features from 3 historical memorized mini-batch to perform an estimation with 128 samples, while without using additional 12×3 GB memory at running time. It is because the queue that saves historical features costs less GPU memory compared with the procedure of feature extraction. Similar strategies for enlarging the number of samples also exist, *e.g*, memory bank [340] and momentum encoder [341]. Compared with them, our memorized historic queue is simpler but also enlarges the maximum number of samples to be used without additional GPU memory. In the supplement we provide discussion about the effects of different queue sizes.

9.3.3 Patch-Wise Internal Probabilities

The most straightforward way to construct samples for the mini-batch is to randomly choose multiple images from the domains $\mathcal{X}$ and $\mathcal{Y}$, respectively. Even though it is elegant, it fails to exploit another crucial probability in the single image, *i.e.*, internal

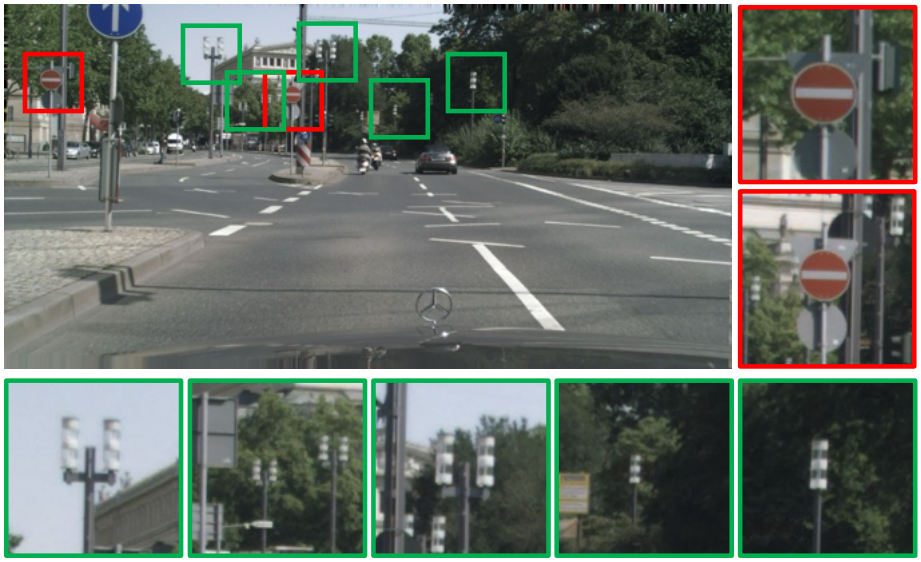


Figure 9.4: Sampling with Internal Patches. Patches cropped from a single image may consist of different semantic objects that showed in different appearances.

probability of patches from a single image, or internal statistics, which has been widely employed and empirically evaluated in many image restoration tasks [339, 367–369]. As illustrated in Figure 9.4, we can notice that the cropped patches from the single image, specifically the complex-semantic image, can be seen as multiple single-semantic images. In addition, the probability estimated on multiple complex-semantic images is not always accurate in some cases, *e.g.*, the denoising learning [330] in mixed noise levels. In such a case, conventional sampling results in an inaccurate estimation, because the restored images in the mini-batch come from different noisy levels. These restored images distribute in different manifolds in the semantic feature space. In contrast, by extracting patches from a single image resulted from extreme similar degradation, their similarity allows the probabilities to be accurately estimated.

In our method, a sliding window in a spatial size of $K \times K$ is availed to extract patches

from the restored image $\tilde{Y}$ and clear image Y. These patches are then inputted into the high-level vision network $\Phi(\cdot)$ as a single mini-batch and transformed into feature sets, which can be formulated as:

$$f^x = \Phi(\text{sliding\_window}(\tilde{Y})). \tag{9.9}$$

Then we exploit Equation 9.7 to estimate the probability distribution in the feature sets respectively for divergence approximation. In the empirical evaluation, shown in Table 9.1, though the vanilla version achieves gains on the restoration performance, the generated images contain less semantic details (lower MIoU performance). This decreasing may boil down to incorrect probability estimation. In contrast, with the help of internal probability, our restoration network achieves a performance leap in both the restoration and semantic evaluation.

9.4 Experiments

Different from conventional denoising works, we focus on not only the restoration performance, but also the semantic accuracy, *i.e.*, how the denoised images can be understood by semantic segmentation networks, as well as whether the method can be extended into the other restoration tasks. Such a similar evaluation protocol is also employed in recent restoration works [370–372]. Therefore, our experiments are divided into three parts, including *Cityscape Denoising and Segmentation* [337], *Face Super-resolution and Alignment* [350], and *Natural Image Restoration* [373]. More details please refer to the supplemental.

9.4.1 Cityscape Denoising and Segmentation

To demonstrate the superiority of our method, we conduct complementary denoising and segmentation experiments on the Cityscapes dataset. The most representative denoising network, *i.e.*, FFDNet [330], as well as the state-of-the-art denoising networks, *i.e.*, CBDNet [374] and SADNet [375] are availed as the generation network $G(\cdot)$. Various objectives are applied, *i.e.*, $\mathcal{L}_1$, $\mathcal{L}_{SSIM}$ [342], $\mathcal{L}_{Perceptual}$ [232], $\mathcal{L}_{LPIPS}$ [248], $\mathcal{L}_{Contextual}$ [205], $\mathcal{L}_{CrossEntropy}$ [348], and ours. Notably, $\mathcal{L}_{CrossEntropy}$ is conducted with the HRNet48 that pre-trained on the Cityscapes dataset, which requires semantic labels during the denoising learning. In contrast, ours does not need any additional data. We then modify the original loss function of CBDNet and SADNet, *i.e.*, $\mathcal{L}_2 + \mathcal{L}_{Asymmetric} + \mathcal{L}_{TV}$ and $\mathcal{L}_2$, by attaching our proposed objective. For convenience, we set the size of the sliding window K as 224 and its stride as 56.

Table 9.1: Quantitative performance comparison on the cityscape denoising and segmentation. The comprasion is conducted with various state-of-the-art denoising objectives and ours on the representative denoising networks.

		Noise-Level σ=25			Noise-Level σ=35			Noise-Level σ=50		
Method (Backbone)	Objective	PSNR ↑	SSIM ↑	MIoU (%) ↑	PSNR ↑	SSIM ↑	MIoU (%) ↑	PSNR ↑	SSIM ↑	MIoU (%) ↑
FFDNet [330]	$\mathcal{L}_1$	$35.033^{(6)}$	$0.925^{(6)}$	$0.605^{(8)}$	$34.074^{(6)}$	$0.912^{(6)}$	$0.537^{(8)}$	$32.845^{(6)}$	$0.895^{(6)}$	$0.451^{(7)}$
	+ $\mathcal{L}_{SSIM}$ [342]	$35.567^{(3)}$	$0.935^{(2)}$	$0.642^{(2)}$	$34.469^{(4)}$	$0.922^{(2)}$	$0.584^{(2)}$	$33.180^{(3)}$	$0.906^{(2)}$	$0.450^{(8)}$
	+ $\mathcal{L}_{Perceptual}$ [232]	$34.319^{(7)}$	$0.912^{(7)}$	$0.629^{(4)}$	$33.486^{(7)}$	$0.899^{(7)}$	$0.582^{(4)}$	$32.383^{(7)}$	$0.881^{(7)}$	$0.509^{(2)}$
	+ $\mathcal{L}_{LPIPS}$ [248]	$35.551^{(4)}$	$0.929^{(4)}$	$0.613^{(6)}$	$34.463^{(5)}$	$0.916^{(4)}$	$0.541^{(7)}$	$33.138^{(5)}$	$0.899^{(4)}$	$0.452^{(6)}$
	+ $\mathcal{L}_{Contextual}$ [205]	$25.115^{(8)}$	$0.762^{(8)}$	$0.628^{(5)}$	$24.938^{(8)}$	$0.758^{(8)}$	$0.583^{(3)}$	$24.775^{(8)}$	$0.753^{(8)}$	$0.509^{(2)}$
	+ $\mathcal{L}_{CrossEntropy}$ [348]	$35.913^{(2)}$	$0.932^{(3)}$	$0.630^{(3)}$	$34.800^{(2)}$	$0.919^{(3)}$	$0.565^{(5)}$	$33.477^{(2)}$	$0.903^{(3)}$	$0.491^{(4)}$
D2SM (Ours)	w/o. Internal	$35.543^{(5)}$	$0.929^{(4)}$	$0.612^{(7)}$	$34.475^{(3)}$	$0.916^{(4)}$	$0.546^{(6)}$	$33.167^{(4)}$	$0.899^{(4)}$	$0.463^{(5)}$
	w/. Internal	$\mathbf{36.454}^{(1)}$	$\mathbf{0.936}^{(1)}$	$\mathbf{0.644}^{(1)}$	$\mathbf{35.206}^{(1)}$	$\mathbf{0.923}^{(1)}$	$\mathbf{0.587}^{(1)}$	$\mathbf{33.807}^{(1)}$	$\mathbf{0.907}^{(1)}$	$\mathbf{0.520}^{(1)}$
CBDNet [374]	-	$36.152^{(3)}$	$0.936^{(2)}$	$0.655^{(3)}$	$34.964^{(3)}$	$0.923^{(3)}$	$0.599^{(3)}$	$33.613^{(3)}$	$0.907^{(3)}$	$0.539^{(3)}$
	w/o. Internal	$36.254^{(2)}$	$0.935^{(3)}$	$0.679^{(2)}$	$35.158^{(2)}$	$0.925^{(2)}$	$0.631^{(2)}$	$33.904^{(2)}$	$0.911^{(2)}$	$0.550^{(2)}$
	w/. Internal	$\mathbf{36.899}^{(1)}$	$\mathbf{0.941}^{(1)}$	$\mathbf{0.691}^{(1)}$	$\mathbf{35.596}^{(1)}$	$\mathbf{0.929}^{(1)}$	$\mathbf{0.652}^{(1)}$	$\mathbf{34.172}^{(1)}$	$\mathbf{0.914}^{(1)}$	$\mathbf{0.600}^{(1)}$
SADNet [375]	-	$36.310^{(3)}$	$0.936^{(3)}$	$0.674^{(3)}$	$35.081^{(3)}$	$0.924^{(2)}$	$0.637^{(3)}$	$33.730^{(3)}$	$0.908^{(3)}$	$0.581^{(3)}$
	w/o. Internal	$36.822^{(2)}$	$0.940^{(2)}$	$0.691^{(2)}$	$35.247^{(2)}$	$0.924^{(2)}$	$0.655^{(2)}$	$34.133^{(2)}$	$0.912^{(2)}$	$0.600^{(2)}$
	w/. Internal	$\mathbf{37.130}^{(1)}$	$\mathbf{0.943}^{(1)}$	$\mathbf{0.701}^{(1)}$	$\mathbf{35.839}^{(1)}$	$\mathbf{0.931}^{(1)}$	$\mathbf{0.670}^{(1)}$	$\mathbf{34.440}^{(1)}$	$\mathbf{0.916}^{(1)}$	$\mathbf{0.634}^{(1)}$

For denoising training, we construct noisy images by adding additive color Gaussian noise of noise level $\sigma \in [0, 75]$ to the clean images from the Cityscapes training set. The images are randomly cropped into 512×512 patches in a mini-batch size of 64. Other settings are kept the same as the settings in FFDNet. For evaluation, we first measure appearance similarities between restored images and corresponding clear images in the Cityscapes validation set, in noisy levels $\{25, 35, 50\}$, which is commonly selected by the denoising community. We then measure the semantic segmentation accuracy on restored images in the term of Mean Intersection-over-Union (MIoU) in 19 pre-defined semantic classes, i.e., *road*, *sidewalk*, *building*, *wall*, *fence*, *pole*, *traffic light*, *traffic sign*, *vegetation*, *terrain*, *sky*, *person*, *rider*, *car*, *truck*, *bus*, *train*, *motorcycle*, and *bicycle*.

Quantitative Comparison. In Table 9.1, it is easy to see that ours outperforms all compared objectives largely in PSNR, SSIM, and MIoU metrics on all noisy levels, when applied in the same backbone. Compared with the state-of-the-art objectives that combine high-level vision tasks, *i.e.*, $\mathcal{L}_{CrossEntropy}$ [348], which requires the semantic label of images during training, ours still outperforms it by 0.542dB in PSNR, 1.4% in MIoU, without using any additional data. Besides, ours shows strong robustness when adopted with different network architectures and objectives, *e.g.*, it helps the original CBDNet improves 0.747dB in PSNR and 0.5% in MIoU. Also, the comparison between using and without using internal probability further demonstrates its superiority for complex-semantic images.

Qualitative Comparison. Though $\mathcal{L}_{Perceptual}$ applied in restoration methods has been proven to lead to better perceptual quality in restored images, we find that ours

significantly outperforms it with more visually pleasant and exact details as shwon in Figure 9.5. As shown in Figure 9.6, our restored results best preserve the edge of the character "**S**" in the red rectangle area. Besides, the blue rectangle area shows the best sharp details in our restored results compared with others. With regard to the segmentation evaluation, restored results from restoration networks trained with ours can best be segmented accurately. For instance, as two green rectangle areas are shown in the Figure 9.6, our result is the only one that is successfully recognized into *traffic light*. This indicates that ours can best preserve semantic details during restoration in the way of divergence minimization.

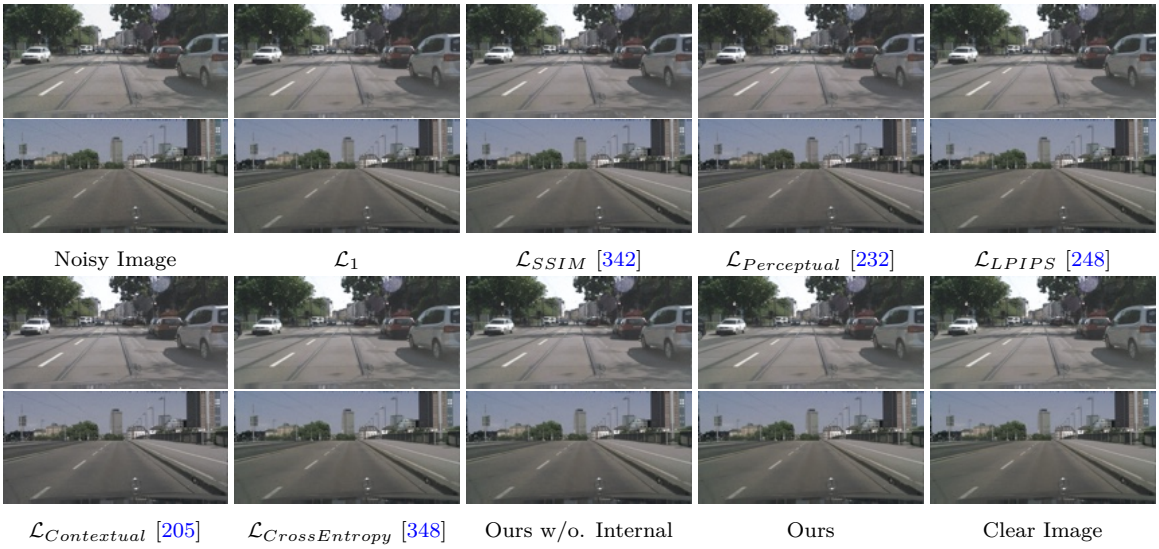


Figure 9.5: Qualitative comparison on the denoising results. Ours results contain the most fine-grained high-frequency information and more visual pleasant details. (400% Zoom is recommended to see their difference in details and color bias.)

Distribution Visualization. In order to get insights into the probability distribution, here we visualize the semantic feature space estimated by restored images and clear images. To elaborate, we randomly select 500 animal images that belong to 10 categories, *i.e.*, *cat*, *dog*, *chicken*, *cow*, *horse*, *sheep*, *squirrel*, *elephant*, *butterfly*, and

spider. We then process their noisy version (*i.e.* adding additive color Gaussian noise of noise level σ=25) with the FFDNet pre-trained on the noisy Cityscapes dataset. After that, the denoised images, as well as the clear images, are inputted into the pre-trained ResNet101 [376] to extract semantic feature maps. As such, we can visualize the distribution of semantic features with the t-SNE [335] in 2D coordinates. Compared with others, the visualized distribution from our restored images best preserves the distribution of clear images in the semantic feature space. This indicates that our proposed method indeed implicitly minimizes the probability distribution divergence between restored images and clear images in the semantic feature space.

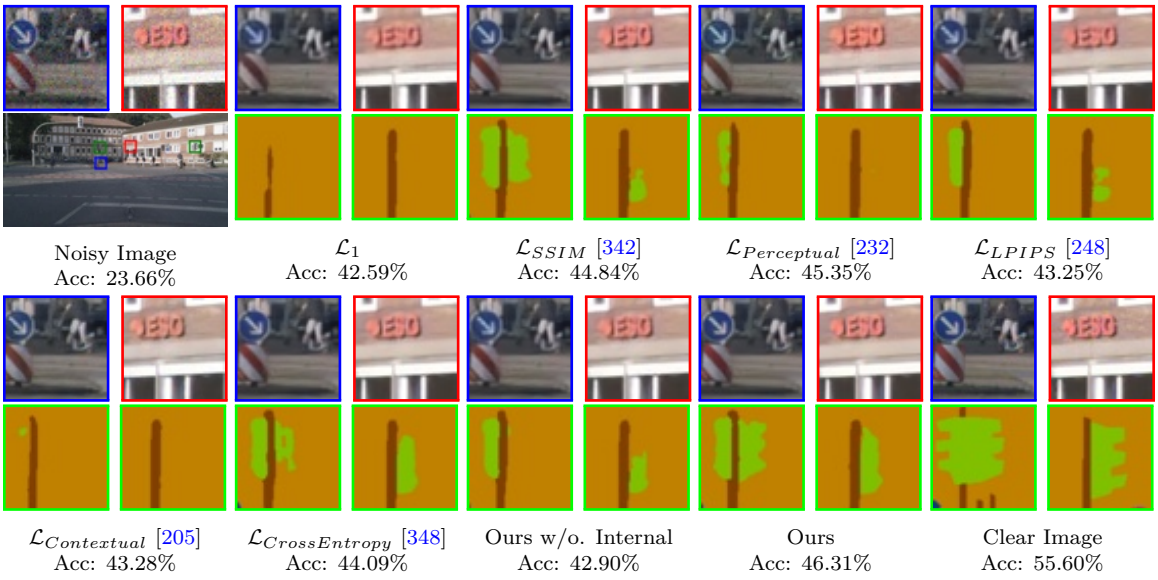


Figure 9.6: Qualitative comparison on the denoising and segmentation results. Ours preserves most of the semantic details, including the human shape and font edge in the highlighted area. Additionally, in the shown segmentation results, our result is the only one that can be successfully recognized into *traffic light*.

9.4.2 Face Super-resolution and Alignment

Here we extend D2SM with historic sampling as the objective and conduct face super-resolution learning under the settings of DICNet [377]. For the evaluation of high-level vision applications, we exploit the face alignment as a measurement, and its accuracy is denoted in the term of NRMSE. The queue size is set as 256 and the mini-batch size is 32, which means the probability is estimated among 32 face images instead interal patches.

In table 9.2, we show the quantitative performance comparison between DICNet and ours, as well as the other state-of-the-art methods. Notably, the face alignment network is pre-trained on the CelebA dataset, and hence its evaluation on the Helen dataset is generally not good enough, which can only be used for reference. By combining our objective with existing $\mathcal{L}_1$ and $\mathcal{L}_{Alignment}$ proposed by DICNet, our modified version successfully outperforms the original DICNet and DICNet in the GAN manner (DICGAN), in both the distortion measurement and alignment measurement. In contrast, the original DICNet can only achieve leading performance in the distortion measurement but is poor in alignment measurement, while the DICGAN can only achieve leading performance in the alignment measurement but is bad at distortion measurement.

9.4.3 Natural Image Restoration

Different from cityscape images collected from limited scenes, natural images contain more diverse and complex semantics. Therefore, the intrinsic semantics of natural images are more complex and diverse, which indicate a more challenging probability

Table 9.2: Quantitative performance comparison on the face super-resolution and alignment. By simply attaching our objective into the DICNet, our method can outperform the state-of-the-art DICNet and DICGAN in both the distortion measurement and high-level vision application measurement.

		CelebA Dataset			Helen Dataset		
Method (Backbone)	Objective	PSNR ↑	SSIM ↑	NRMSE ↓	PSNR ↑	SSIM ↑	NRMSE ↓
Bicubic	-	$23.58^{(10)}$	$0.6285^{(10)}$	$0.3385^{(8)}$	$23.89^{(10)}$	$0.6751^{(10)}$	$0.4577^{(8)}$*
SRResNet [344] **[CVPR-17]**	$\mathcal{L}_2$	$25.82^{(6)}$	$0.7369^{(6)}$	-	$25.30^{(6)}$	$0.7297^{(7)}$	-
URDGN [378] **[ECCV-16]**	$\mathcal{L}_2 + \mathcal{L}_{GAN}$	$24.63^{(8)}$	$0.6851^{(9)}$	-	$24.22^{(9)}$	$0.6909^{(9)}$	-
RDN [379] **[ECCV-18]**	$\mathcal{L}_1$	$26.13^{(5)}$	$0.7412^{(5)}$	$0.1415^{(4)}$	$25.34^{(5)}$	$0.7249^{(8)}$	$0.4437^{(7)}$*
PFSR [380] **[BMVC-18]**	$\mathcal{L}_2 + \mathcal{L}_{Perceptual} + \mathcal{L}_{GAN} + \mathcal{L}_{Heatmap} + \mathcal{L}_{Attention}$	$24.43^{(9)}$	$0.6991^{(8)}$	$0.1917^{(7)}$	$24.73^{(8)}$	$0.7323^{(6)}$	$0.3498^{(4)}$*
FSRNet [381] **[CVPR-18]**	$\mathcal{L}_2 + \mathcal{L}_{Perceptual}$	$26.48^{(3)}$	$0.7718^{(3)}$	$0.1430^{(5)}$	$25.90^{(4)}$	$0.7759^{(3)}$	$0.3723^{(6)}$*
	$\mathcal{L}_2 + \mathcal{L}_{Perceptual} + \mathcal{L}_{GAN}$	$25.06^{(7)}$	$0.7311^{(7)}$	$0.1463^{(6)}$	$24.99^{(7)}$	$0.7424^{(5)}$	$0.3408^{(3)}$*
DICNet [377] **[CVPR-20]**	$\mathcal{L}_1 + \mathcal{L}_{Alignment}$	$27.28^{(2)}$	$0.7929^{(2)}$	$0.1345^{(3)}$	$26.69^{(2)}$	$0.7933^{(2)}$	$0.3674^{(5)}$*
	$\mathcal{L}_1 + \mathcal{L}_{Alignment} + \mathcal{L}_{Perceptual} + \mathcal{L}_{GAN}$	$26.34^{(4)}$	$0.7562^{(4)}$	$0.1319^{(2)}$	$25.96^{(3)}$	$0.7624^{(4)}$	$\mathbf{0.3336}^{(1)}$*
Ours	w/o. Internal	$\mathbf{27.39}^{(1)}$	$\mathbf{0.7973}^{(1)}$	$\mathbf{0.1292}^{(1)}$	$\mathbf{26.94}^{(1)}$	$\mathbf{0.8005}^{(1)}$	$0.3366^{(2)}$*

Table 9.3: Quantitative comparison on the natural image dehazing. Our proposed objective is capable of being extended to the dehazing task based on MSBDN-DFF, which shows superiority in both the indoor and outdoor datasets.

Method	Metric	DCP [382]	MSCNN [383]	DcGAN [384]	GFN [385]	PFFNet [386]	GDN [387]	DuRN [388]	MSBDN-DFF [389]	Ours
I-HAZE	PSNR↑	$14.43^{(9)}$	$15.22^{(8)}$	$16.06^{(5)}$	$15.84^{(7)}$	$16.01^{(6)}$	$16.62^{(4)}$	$21.23^{(3)}$	$23.93^{(2)}$	$\mathbf{24.31}^{(1)}$
	SSIM↑	$0.752^{(6)}$	$0.755^{(5)}$	$0.733^{(9)}$	$0.751^{(7)}$	$0.740^{(8)}$	$0.787^{(4)}$	$0.842^{(3)}$	$0.891^{(2)}$	$\mathbf{0.902}^{(1)}$
O-HAZE	PSNR↑	$16.78^{(9)}$	$17.56^{(8)}$	$19.34^{(4)}$	$18.16^{(7)}$	$18.76^{(6)}$	$18.92^{(5)}$	$20.45^{(3)}$	$24.36^{(2)}$	$\mathbf{24.79}^{(1)}$
	SSIM↑	$0.653^{(8)}$	$0.650^{(9)}$	$0.681^{(4)}$	$0.671^{(6)}$	$0.669^{(7)}$	$0.672^{(5)}$	$0.688^{(3)}$	$0.749^{(2)}$	$\mathbf{0.787}^{(1)}$

distribution estimation for our method. To validate our effectiveness in such cases, here we follow the settings of the state-of-the-art dehazing method, *i.e.*, MSBDN-DFF [389] on the end-to-end dehazing tasks [386], and we extend the dehazing network with our proposed objective. As the quantitative performance comparison shown in Table 9.3, though the density estimation is challenging, our method can successfully outperform the compared method in both the indoor scenes and outdoor scenes without additional cost. In Figure 9.7, we show some randomly highlighted visual results for comparison, and all of our results contain the most clear appearance with less haze remained.

Specifically, we can notice that objects, *e.g.*, *floor*, *chair*, *roof*, *toy* that contain certain semantics, are better restored with accurate color than the method trained with pixel-wise loss functions only. This phenomenon further demonstrates the semantics transferring ability of our method, which regularizes restored objects to be semantic consistent and avoids incorrect color that againsts its semantics.

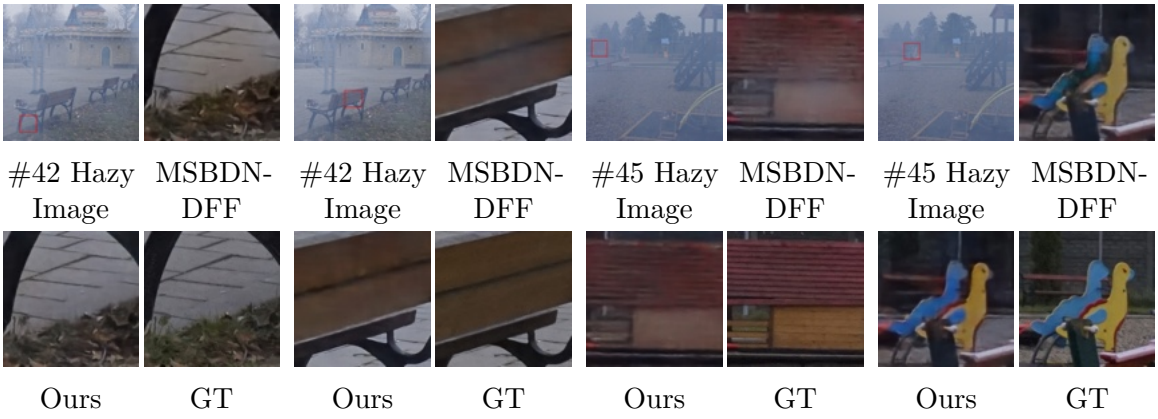


Figure 9.7: Qualitative comparison on the real-world dehazing. Compared with the SOTA method that employs pixel-wise loss functions, our extended version better recover the scenes under severe ill-posed distortion.

Method	σ	PSNR ↑	SSIM ↑	MIoU (%) ↑
FFDNet	25	35.03	0.925	0.605
+ $\mathcal{L}_{iKLD}$	25	35.97	0.931	0.638
+ $\mathcal{L}_{JSD}$	25	36.31	0.935	0.640
+ $\mathcal{L}_{GAN}$	25	35.55	0.931	0.621
Ours	25	**36.45**	**0.936**	**0.644**

Table 9.4: Performance comparison with different distribution divergence.

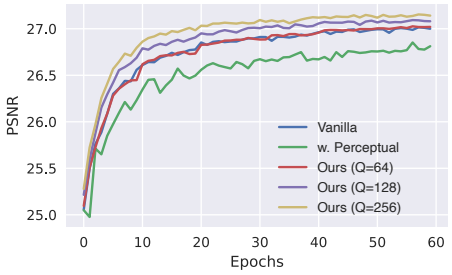


Figure 9.8: Convergence visualization between different queue size.

9.5 Discussion

Designing Choice of KL Divergence. The way of the divergence estimation beween $\mathcal{G}'$ and $\mathcal{G}$ accounts a lot in our method. We empirically employed the Kullback-Leibler (KL) divergence weighted by $\mathcal{G}'$ for D2SM. Indeed, there are some other ways to estimate the divergence. The most similar one, *i.e.*, inverse KL divergence weighted by $\mathcal{G}$, which is also asymmetrical. Based on KL divergence, another way to estimate the divergence is Jensen–Shannon divergence, which is symmetric and can be seen as the smoothed version of KL divergence, formulated as:

$$D_{JS}(\mathcal{G}'||\mathcal{G}) = \frac{1}{2}D_{KL}(\mathcal{G}'||\mathcal{G}) + \frac{1}{2}D_{KL}(\mathcal{G}||\mathcal{G}'). \tag{9.10}$$

According to [390], the optimization procedure of the optimal discriminator $D*$ in GAN yields minimizing the JS divergence, formulated as:

$$\mathcal{L}(D*, G) = 2D_{JS}(\mathcal{G}'||\mathcal{G}) - 2\log 2. \tag{9.11}$$

Here we present the quantitative comparison with the three additional divergence estimation or objectives in Cityscapes. The comprasion empirically proves our superiority of using KL-divergence compared with the others in the denoising tasks.

Designing Choice of Queue Size. The convergence curve visualized in the Figure 9.8 further demonstrates that our proposed method significantly accelerates convergence with the proposed memorized historic sampling. From the figure we can notice that the applied historic sampling with large queue size ($Q > 64$) can greately accelerate the learning, while the vanilla version can only achieve poor performance ($Q \leq 64$). Thus, in our practical implementation, we chose the queue size with the possible max value under the computational limitation.

Chapter 10
Conclusion and Future Work

Generative models have rapidly evovled real-world application, this thesis improved generative models on several foundational aspect and demonstrate these improvement on severl representative tasks.

The thsis first investigated scaling properties of Latent Diffusion Models (LDMs), specifically through scaling model size from 39 million to 5 billion parameters. The most important observation is that, under identical sampling costs, smaller models frequently outperform larger models, suggesting a promising direction for accelerating LDMs in terms of model size. We believe this analysis of scaling sampling efficiency would be instrumental in guiding future developments of LDMs, specifically for balancing model size against performance and efficiency in a broad spectrum of practical applications. However, it is important to acknowledge the potential discrepancy between visual quality and quantitative metrics, which is actively discussed in recent works [391–393]. Moreover, claims regarding the scalability of latent diffusion models are made specifically for the particular model family studied in this work [9]. Extending this analysis to other model families, particularly those incorporating transformer-based backbones such as DiT [17, 29], SiT [394], MM-DiT [395], and DiS [396], and cascaded diffusion models such as Imagen3 [397] and Stable Cascade [398], would be a valuable direction for future research.

The thesis further proposed a new diffusion probabilistic model for video data, which

provides a unique implicit condition paradigm for modeling continuous spatial-temporal changing of videos. The model is capable of sampling frames according to latent that encodes dynamics. We hope the work would benefit and inspire both video generation and conditional diffusion models as a strong baseline in the future. The thesis then introduced a new transformer-based diffusion field model that addresses the limitations of current probabilistic field models in capturing global structures and long-context dependencies. By utilizing a view-wise sampling algorithm for local structure learning and incorporating autoregressive generation to preserve global geometry, our approach overcomes the shortcomings of MLP-based architectures. The proposed model can generate high-fidelity data across multiple modalities, including text-to-video, 3D view generation, and game control while maintaining scalability and unifying diverse modalities. However, the method aligns with the standard practice outlined in DPF, using comparison methods with weights trained separately for each modality. While it performs exceptionally well on individual modalities, achieving strong performance across multiple modalities simultaneously is hindered by the inherent challenges of the multi-task problem. We believe our approach provides a solid foundation for the future versatile generative models.

Moreover, the thesis introduces a new framework for distilling an unconditional diffusion model into a conditional one that allows sampling with very few steps. Our method also enables a new parameter-efficient distillation that allows different distilled models, trained for different tasks, to share most of their parameters. Only a few additional parameters are needed for each different conditional generation task. We believe the method can serve as a strong practical approach for accelerating

large-scale conditional diffusion models. The method shows image conditions benefit our distillation learning. However, the distillation learning depends on the adapter architecture that introduces additional computation in our current framework. As a future work, we would like to explore lightweight network architectures [27] in our distillation technique to further reduce the inference latency.

For real-world tasks, we introduced a novel class of diffusion models that significantly outperform existing shadow removal methods at the general and instance level. Our comprehensive evaluations and analyses have demonstrated the superior effectiveness of our method compared to existing state-of-the-art shadow removal methods. We believe that our proposed diffusion model-based technique has the potential to be applied to other similar ill-posed low-level problems. The thesis also presented LTT-GAN, an enhanced generative embedding network that can achieve a better trade-off between identity and realism of restored results. In contrast to previous methods, LTT-GAN achieves the goal in a similar way of adversarial loss, without a discriminator nor feature extractor needed, but instead, with a simple matrix transformation in the image space, as well as a contextual distance defined on it. By showing its effectiveness on the turbulence mitigation problem, LTT-GAN provided significant performance improvement over the existing state-of-the-art methods, which is the first GAN method that can produce sharp results in turbulence degraded images.

The thesis finally introduces a simple but practical method for facilitating the restoration learning and preserving the semantic attribute. It does not rely on any external information nor introduce additional parameters. By implicitly approximating the divergence on the semantic feature space, we can force existed generation networks to

learn to preserve semantic attributes during restoration learning. We further transfer the method from the single-semantic image to the complex-semantic image *i.e.* natural image by using internal statistics. Empirically evaluation validates that the proposed method can be adapted to various restoration tasks and network architectures with general performance improvement.

Bibliographic references

1. Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S. & Poole, B. *Score-based generative modeling through stochastic differential equations* in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021* (2021).

2. Ho, J., Jain, A. & Abbeel, P. *Denoising diffusion probabilistic models* in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual* (eds Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. & Lin, H.) (2020).

3. Kingma, D. P. & Welling, M. *Auto-encoding variational bayes* in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings* (eds Bengio, Y. & LeCun, Y.) (2014).

4. Song, Y., Garg, S., Shi, J. & Ermon, S. *Sliced score matching: A scalable approach to density and score estimation* in *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2019, Tel Aviv, Israel, July 22-25, 2019* (eds Globerson, A. & Silva, R.) (2019).

5. Kingma, D., Salimans, T., Poole, B. & Ho, J. Variational diffusion models. *Advances in neural information processing systems* (2021).

6. Song, J., Meng, C. & Ermon, S. *Denoising diffusion implicit models* in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021* (2021).

7. Dhariwal, P. & Nichol, A. Q. *Diffusion models beat gans on image synthesis* in *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual* (eds Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P. & Vaughan, J. W.) (2021).

8. Salimans, T. & Ho, J. *Progressive distillation for fast sampling of diffusion models* in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022* (2022).

9. Rombach, R., Blattmann, A., Lorenz, D., Esser, P. & Ommer, B. *High-resolution image synthesis with latent diffusion models* in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2022).

10. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. & Zitnick, C. L. *Microsoft coco: common objects in context* in *ECCV* (2014).

11. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., *et al.* Laion-5b: an open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems* (2022).

12. Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J. & Rombach, R. Sdxl: improving latent diffusion models for high-resolution image synthesis. *ArXiv preprint* (2023).

13. Delbracio, M. & Milanfar, P. Inversion by direct iteration: an alternative to denoising diffusion for image restoration. *Transactions on Machine Learning Research.* Featured Certification (2023).

14. Ren, M., Delbracio, M., Talebi, H., Gerig, G. & Milanfar, P. *Multiscale structure guided diffusion for image deblurring* in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023), 10721–10733.

15. Qi, C., Tu, Z., Ye, K., Delbracio, M., Milanfar, P., Chen, Q. & Talebi, H. Tip: text-driven image processing with semantic and restoration instructions. *ArXiv preprint* (2023).

16. Mei, K. & Patel, V. *Vidm: video implicit diffusion models* in *Proceedings of the AAAI Conference on Artificial Intelligence* **37** (2023), 9117–9125.

17. Mei, K., Zhou, M. & Patel, V. M. T1: scaling diffusion probabilistic fields to high-resolution on unified visual modalities. *ArXiv preprint* (2023).

18. Wu, J. Z., Ge, Y., Wang, X., Lei, S. W., Gu, Y., Shi, Y., Hsu, W., Shan, Y., Qie, X. & Shou, M. Z. *Tune-a-video: one-shot tuning of image diffusion models*

for text-to-video generation in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023).

19. Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., *et al.* Make-a-video: text-to-video generation without text-video data. *ArXiv preprint* (2022).

20. Liu, H., Chen, Z., Yuan, Y., Mei, X., Liu, X., Mandic, D., Wang, W. & Plumbley, M. D. Audioldm: text-to-audio generation with latent diffusion models. *ArXiv preprint* (2023).

21. Lin, C.-H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.-Y. & Lin, T.-Y. *Magic3d: high-resolution text-to-3d content creation* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023).

22. Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S. & Vondrick, C. *Zero-1-to-3: zero-shot one image to 3d object* in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023).

23. Du, H., Zhang, R., Niyato, D., Kang, J., Xiong, Z., Kim, D. I., Shen, X. S. & Poor, H. V. Exploring collaborative distributed diffusion-based ai-generated content (aigc) in wireless networks. *IEEE Network* (2023).

24. Choi, J., Kim, M., Ahn, D., Kim, T., Kim, Y., Jo, D., Jeon, H., Kim, J.-J. & Kim, H. Squeezing large-scale diffusion models for mobile. *ArXiv preprint* (2023).

25. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. Generative adversarial networks. *Communications of the ACM* (2020).

26. Karras, T., Laine, S. & Aila, T. *A style-based generator architecture for generative adversarial networks* in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019* (2019).

27. Li, Y., Wang, H., Jin, Q., Hu, J., Chemerys, P., Fu, Y., Wang, Y., Tulyakov, S. & Ren, J. Snapfusion: text-to-image diffusion model on mobile devices within two seconds. *NeurIPS* (2023).

28. Zhao, Y., Xu, Y., Xiao, Z. & Hou, T. Mobilediffusion: subsecond text-to-image generation on mobile devices. *ArXiv preprint* (2023).

29. Peebles, W. & Xie, S. *Scalable diffusion models with transformers* in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023).

30. Kim, B.-K., Song, H.-K., Castells, T. & Choi, S. *Bk-sdm: architecturally compressed stable diffusion for efficient text-to-image generation* in *Workshop on Efficient Systems for Foundation Models@ ICML2023* (2023).

31. Kim, B.-K., Song, H.-K., Castells, T. & Choi, S. On architectural compression of text-to-image diffusion models. *ArXiv preprint* (2023).

32. Dockhorn, T., Vahdat, A. & Kreis, K. Genie: higher-order denoising diffusion solvers. *Advances in Neural Information Processing Systems* (2022).

33. Karras, T., Aittala, M., Aila, T. & Laine, S. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems* (2022).

34. Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C. & Zhu, J. Dpm-solver: a fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems* (2022).

35. Liu, X., Zhang, X., Ma, J., Peng, J. & Liu, Q. Instaflow: one step is enough for high-quality diffusion-based text-to-image generation. *ArXiv preprint* (2023).

36. Xu, Y., Zhao, Y., Xiao, Z. & Hou, T. Ufogen: you forward once large scale text-to-image generation via diffusion gans. *ArXiv preprint* (2023).

37. Luhman, E. & Luhman, T. Knowledge distillation in iterative generative models for improved sampling speed. *ArXiv preprint* (2021).

38. Song, Y., Dhariwal, P., Chen, M. & Sutskever, I. Consistency models. *ICML* (2023).

39. Sauer, A., Lorenz, D., Blattmann, A. & Rombach, R. Adversarial diffusion distillation. *ArXiv preprint* (2023).

40. Gu, J., Zhai, S., Zhang, Y., Liu, L. & Susskind, J. M. *Boot: data-free distillation of denoising diffusion models with bootstrapping* in *ICML 2023 Workshop on Structured Probabilistic Inference {\&} Generative Modeling* (2023).

41. Mei, K., Delbracio, M., Talebi, H., Tu, Z., Patel, V. M. & Milanfar, P. *Codi: conditional diffusion distillation for higher-fidelity and faster image generation* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024).

42. Jouppi, N., Kurian, G., Li, S., Ma, P., Nagarajan, R., Nai, L., Patil, N., Subramanian, S., Swing, A., Towles, B., *et al. Tpu v4: an optically reconfigurable supercomputer for machine learning with hardware support for embeddings* in *Proceedings of the 50th Annual International Symposium on Computer Architecture* (2023).

43. Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D. J. & Norouzi, M. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022).

44. Sahak, H., Watson, D., Saharia, C. & Fleet, D. Denoising diffusion probabilistic models for robust image super-resolution in the wild. *ArXiv preprint* (2023).

45. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M. & Aberman, K. *Dreambooth: fine tuning text-to-image diffusion models for subject-driven generation* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023).

46. Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C. & Zhu, J. Dpm-solver++: fast solver for guided sampling of diffusion probabilistic models. *ArXiv preprint* (2022).

47. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I. & Amodei, D. *Language models are few-shot learners* in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual* (2020).

48. Anil, R., Dai, A. M., Firat, O., Johnson, M., Lepikhin, D., Passos, A., Shakeri, S., Taropa, E., Bailey, P., Chen, Z., *et al.* Palm 2 technical report. *ArXiv preprint* (2023).

49. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., *et al.* Llama 2: open foundation and fine-tuned chat models. *ArXiv preprint* (2023).

50. Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J. & Amodei, D. Scaling laws for neural language models. *ArXiv preprint* (2020).

51. Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D. d. L., Hendricks, L. A., Welbl, J., Clark, A., *et al.* Training compute-optimal large language models. *ArXiv preprint* (2022).

52. Zhou, Y., Du, N., Huang, Y., Peng, D., Lan, C., Huang, D., Shakeri, S., So, D., Dai, A. M., Lu, Y., *et al. Brainformers: trading simplicity for efficiency* in *International Conference on Machine Learning* (2023).

53. Alabdulmohsin, I. M., Zhai, X., Kolesnikov, A. & Beyer, L. Getting vit in shape: scaling laws for compute-optimal model design. *Advances in Neural Information Processing Systems* (2024).

54. Nichol, A. Q. & Dhariwal, P. *Improved denoising diffusion probabilistic models* in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event* (eds Meila, M. & Zhang, T.) (2021).

55. Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D. & Sutskever, I. *Generative pretraining from pixels* in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event* (2020).

56. Wang, X., Xie, L., Dong, C. & Shan, Y. *Real-esrgan: training real-world blind super-resolution with pure synthetic data* in *IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021* (2021).

57. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C. & Chen, M. Hierarchical text-conditional image generation with clip latents. *ArXiv preprint* (2022).

58. Mei, K., Figueroa, L., Lin, Z., Ding, Z., Cohen, S. & Patel, V. M. *Latent feature-guided diffusion models for shadow removal* in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (2024), 4313–4322.

59. Mei, K., Nair, N. G. & Patel, V. M. *Improving conditional diffusion models through re-noising from unconditional diffusion priors* in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (2025).

60. Rezende, D. J. & Mohamed, S. *Variational inference with normalizing flows* in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015* (2015).

61. Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I. & Frey, B. Adversarial autoencoders. *ArXiv preprint* (2015).

62. Vahdat, A. & Kautz, J. *NVAE: A deep hierarchical variational autoencoder* in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual* (2020).

63. Mao, X., Li, Q., Xie, H., Lau, R. Y. K., Wang, Z. & Smolley, S. P. *Least squares generative adversarial networks* in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017* (2017).

64. Reed, S. E., Akata, Z., Yan, X., Logeswaran, L., Schiele, B. & Lee, H. *Generative adversarial text to image synthesis* in *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016* (2016).

65. Miyato, T., Kataoka, T., Koyama, M. & Yoshida, Y. *Spectral normalization for generative adversarial networks* in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings* (2018).

66. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. *BERT: pre-training of deep bidirectional transformers for language understanding* in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (2019).

67. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W. & Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* (2020).

68. He, K., Chen, X., Xie, S., Li, Y., Dollár, P. & Girshick, R. B. *Masked autoencoders are scalable vision learners* in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022* (2022).

69. Chang, H., Zhang, H., Jiang, L., Liu, C. & Freeman, W. T. *Maskgit: masked generative image transformer* in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022* (2022).

70. Chang, H., Zhang, H., Barber, J., Maschinot, A., Lezama, J., Jiang, L., Yang, M.-H., Murphy, K., Freeman, W. T., Rubinstein, M., *et al.* Muse: text-to-image generation via masked generative transformers. *ArXiv preprint* (2023).

71. Sauer, A., Karras, T., Laine, S., Geiger, A. & Aila, T. Stylegan-t: unlocking the power of gans for fast large-scale text-to-image synthesis. *ArXiv preprint* (2023).

72. Chen, X., Wang, X., Changpinyo, S., Piergiovanni, A., Padlewski, P., Salz, D., Goodman, S., Grycner, A., Mustafa, B., Beyer, L., *et al.* Pali: a jointly-scaled multilingual language-image model. *ArXiv preprint* (2022).

73. Ho, J. & Salimans, T. Classifier-free diffusion guidance. *ArXiv preprint* (2022).

74. Luo, S., Tan, Y., Huang, L., Li, J. & Zhao, H. Latent consistency models: synthesizing high-resolution images with few-step inference. *ArXiv preprint* (2023).

75. Lin, S., Wang, A. & Yang, X. Sdxl-lightning: progressive adversarial diffusion distillation. *ArXiv preprint* (2024).

76. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. *Generative adversarial nets* in *Proceedings of the Advances in Neural Information Processing Systems* (2014).

77. Vondrick, C., Pirsiavash, H. & Torralba, A. *Generating videos with scene dynamics* in *Proceedings of the Advances in Neural Information Processing Systems* (2016).

78. Saito, M., Matsumoto, E. & Saito, S. *Temporal generative adversarial nets with singular value clipping* in *Proceedings of the IEEE international conference on computer vision* (2017).

79. Tulyakov, S., Liu, M.-Y., Yang, X. & Kautz, J. *Mocogan: decomposing motion and content for video generation* in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition* (2018).

80. Yu, S., Tack, J., Mo, S., Kim, H., Kim, J., Ha, J.-W. & Shin, J. *Generating videos with dynamics-aware implicit generative adversarial networks* in *Proceedings of the International Conference on Learning Representations* (2022).

81. Weissenborn, D., Täckström, O. & Uszkoreit, J. *Scaling autoregressive video models* in *Proceedings of the International Conference on Learning Representations* (2020).

82. Tian, Y., Ren, J., Chai, M., Olszewski, K., Peng, X., Metaxas, D. N. & Tulyakov, S. *A good image generator is what you need for high-resolution video synthesis* 2021. arXiv: 2104.15069.

83. Skorokhodov, I., Tulyakov, S. & Elhoseiny, M. *Stylegan-v: a continuous video generator with the price, image quality and perks of stylegan2* in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition* (2022).

84. Srivastava, N., Mansimov, E. & Salakhudinov, R. *Unsupervised learning of video representations using lstms* in *Proceedings of the International Conference on Machine Learning* (2015).

85. Sohl-Dickstein, J., Weiss, E. A., Maheswaranathan, N. & Ganguli, S. *Deep unsupervised learning using nonequilibrium thermodynamics* in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015* (eds Bach, F. R. & Blei, D. M.) (2015).

86. Whang, J., Delbracio, M., Talebi, H., Saharia, C., Dimakis, A. G. & Milanfar, P. *Deblurring via stochastic refinement* in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition* (2022).

87. Ho, J., Saharia, C., Chan, W., Fleet, D. J., Norouzi, M. & Salimans, T. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research* **23,** 1–33 (2022).

88. Choi, J., Kim, S., Jeong, Y., Gwon, Y. & Yoon, S. *Ilvr: conditioning method for denoising diffusion probabilistic models* in *Proceedings of the IEEE International Conference on Computer Vision* (2021).

89. Van Oord, A., Kalchbrenner, N. & Kavukcuoglu, K. *Pixel recurrent neural networks* in *Proceedings of the International Conference on Machine Learning* (2016).

90. Dinh, L., Sohl-Dickstein, J. & Bengio, S. *Density estimation using real nvp* 2016. arXiv: 1605.08803.

91. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J. & Aila, T. *Analyzing and improving the image quality of stylegan* in *IEEE conference on Computer Vision and Pattern Recognition* (2020).

92. Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J. & Aila, T. *Training generative adversarial networks with limited data* in *Proceedings of the Advances in Neural Information Processing Systems* (2020).

93. Brock, A., Donahue, J. & Simonyan, K. *Large scale GAN training for high fidelity natural image synthesis* 2018. arXiv: 1809.11096.

94. Mescheder, L., Geiger, A. & Nowozin, S. *Which training methods for gans do actually converge?* in *Proceedings of the International Conference on Machine Learning* (2018).

95. Mei, K. & Patel, V. M. *Ltt-gan: Looking through turbulence by inverting gans* 2021. arXiv: 2112.02379.

96. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. & Fei-Fei, L. *Imagenet: a large-scale hierarchical image database* in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition* (2009).

97. Nair, N. G., Mei, K. & Patel, V. M. *At-ddpm: restoring faces degraded by atmospheric turbulence using denoising diffusion probabilistic models* in *Proceed-*

ings of the IEEE/CVF Winter Conference on Applications of Computer Vision (2022).

98. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M. & Sutskever, I. *Zero-shot text-to-image generation* in *Proceedings of the International Conference on Machine Learning* (2021).

99. Tran, D., Bourdev, L., Fergus, R., Torresani, L. & Paluri, M. *Learning spatiotemporal features with 3d convolutional networks* in *Proceedings of the International Conference on Computer Vision* (2015).

100. Sitzmann, V., Martel, J., Bergman, A., Lindell, D. & Wetzstein, G. *Implicit neural representations with periodic activation functions* in *Proceedings of the Advances in Neural Information Processing Systems* (2020).

101. Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M. & Fleet, D. J. *Video diffusion models* 2022. arXiv: 2204.03458.

102. Harvey, W., Naderiparizi, S., Masrani, V., Weilbach, C. & Wood, F. *Flexible Diffusion Modeling of Long Videos* 2022. arXiv: 2205.11495.

103. Saito, M., Saito, S., Koyama, M. & Kobayashi, S. Train sparsely, generate densely: memory-efficient unsupervised training of high-resolution temporal gan. *International Journal of Computer Vision* **128,** 2586–2606 (2020).

104. Clark, A., Donahue, J. & Simonyan, K. *Adversarial video generation on complex datasets* 2019. arXiv: 1907.06571.

105. Tian, Y., Ren, J., Chai, M., Olszewski, K., Peng, X., Metaxas, D. N. & Tulyakov, S. *A good image generator is what you need for high-resolution video synthesis* in *Proceedings of the International Conference on Learning Representations* (2021).

106. Fox, G., Tewari, A., Elgharib, M. & Theobalt, C. *Stylevideogan: A temporal generative model using a pretrained stylegan* 2021. arXiv: 2107.07224.

107. Skorokhodov, I., Ignatyev, S. & Elhoseiny, M. *Adversarial generation of continuous images* in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition* (2021).

108. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. *Attention is all you need* in *Proceedings of the Advances in Neural Information Processing Systems* (2017).

109. Wu, Y. & He, K. *Group normalization* in *Proceedings of the European Conference on Computer Vision* (2018).

110. Perez, E., Strub, F., De Vries, H., Dumoulin, V. & Courville, A. *Film: visual reasoning with a general conditioning layer* in *Proceedings of the AAAI Conference on Artificial Intelligence* (2018).

111. Tancik, M., Srinivasan, P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J. & Ng, R. *Fourier features let networks learn high frequency functions in low dimensional domains* in *Proceedings of the Advances in Neural Information Processing Systems* (2020).

112. Horn, B. K. & Schunck, B. G. Determining optical flow. *Artificial intelligence* **17,** 185–203 (1981).

113. Ranjan, A. & Black, M. J. *Optical flow estimation using a spatial pyramid network* in *IEEE conference on Computer Vision and Pattern Recognition* (2017).

114. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., *et al. The kinetics human action video dataset* 2017. arXiv: 1705.06950.

115. Unterthiner, T., van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M. & Gelly, S. *Towards accurate generative models of video: A new metric & challenges* 2018. arXiv: 1812.01717.

116. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A. & Chen, X. *Improved techniques for training gans* in *Proceedings of the Advances in Neural Information Processing Systems* (2016).

117. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. & Hochreiter, S. *Gans trained by a two time-scale update rule converge to a local nash equilibrium* in *Proceedings of the Advances in Neural Information Processing Systems* (2017).

118. Soomro, K., Zamir, A. R. & Shah, M. *UCF101: A dataset of 101 human actions classes from videos in the wild* 2012. arXiv: 1212.0402.

119. Siarohin, A., Lathuilière, S., Tulyakov, S., Ricci, E. & Sebe, N. *First order motion model for image animation* in *Proceedings of the Advances in Neural Information Processing Systems* (2019).

120. Xiong, W., Luo, W., Ma, L., Liu, W. & Luo, J. *Learning to generate time-lapse videos using multi-stage dynamic generative adversarial networks* in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition* (2018).

121. Yi, K., Gan, C., Li, Y., Kohli, P., Wu, J., Torralba, A. & Tenenbaum, J. B. *Clevrer: collision events for video representation and reasoning* in *Proceedings of the International Conference on Machine Learning* (2020).

122. Acharya, D., Huang, Z., Paudel, D. P. & Van Gool, L. *Towards high resolution video generation with progressive growing of sliced wasserstein gans* 2018. arXiv: 1810.02419.

123. Kahembwe, E. & Ramamoorthy, S. Lower dimensional kernels for video discriminators. *Neural Networks* (2020).

124. Yan, W., Zhang, Y., Abbeel, P. & Srinivas, A. *Videogpt: Video generation using vq-vae and transformers* 2021. arXiv: 2104.10157.

125. Ge, S., Hayes, T., Yang, H., Yin, X., Pang, G., Jacobs, D., Huang, J.-B. & Parikh, D. *Long video generation with time-agnostic vqgan and time-sensitive transformer* 2022. arXiv: 2204.03638.

126. Salimans, T., Karpathy, A., Chen, X. & Kingma, D. P. *Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications* 2017. arXiv: 1701.05517.

127. Rombach, R., Blattmann, A., Lorenz, D., Esser, P. & Ommer, B. *High-resolution image synthesis with latent diffusion models* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022).

128. Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M. & Fleet, D. J. *Video diffusion models* in *Advances in Neural Information Processing Systems* (2022).

129. Ho, J., Chan, W., Saharia, C., Whang, J., Gao, R., Gritsenko, A., Kingma, D. P., Poole, B., Norouzi, M. & Fleet, D. J. Imagen video: High definition video generation with diffusion models. *ArXiv preprint* (2022).

130. Dupont, E., Kim, H., Eslami, S., Rezende, D. & Rosenbaum, D. From data to functa: your data point is a function and you should treat it like one. *ArXiv preprint* (2022).

131. Zhuang, P., Abnar, S., Gu, J., Schwing, A., Susskind, J. M. & Bautista, M. Á. *Diffusion Probabilistic Fields* in *International Conference on Learning Representations* (2023).

132. He, Y., Yang, T., Zhang, Y., Shan, Y. & Chen, Q. Latent Video Diffusion Models for High-Fidelity Video Generation with Arbitrary Lengths. *ArXiv preprint* (2022).

133. Bain, M., Nagrani, A., Varol, G. & Zisserman, A. *Frozen in time: A joint video and image encoder for end-to-end retrieval* in *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021* (2021).

134. Yu, J., Zhu, H., Jiang, L., Loy, C. C., Cai, W. & Wu, W. *Celebv-text: A large-scale facial text-video dataset* in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023* (2023).

135. Quinonero-Candela, J. & Rasmussen, C. E. A unifying view of sparse approximate gaussian process regression. *The Journal of Machine Learning Research* (2005).

136. Esser, P., Rombach, R. & Ommer, B. *Taming transformers for high-resolution image synthesis* in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021* (2021).

137. Du, Y., Collins, K., Tenenbaum, J. & Sitzmann, V. *Learning signal-agnostic manifolds of neural fields* in *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual* (2021).

138. Bauer, M., Dupont, E., Brock, A., Rosenbaum, D., Schwarz, J. & Kim, H. Spatial functa: scaling functa to imagenet classification and generation. *ArXiv preprint* (2023).

139. Yu, S., Sohn, K., Kim, S. & Shin, J. *Video probabilistic diffusion models in projected latent space* in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023* (2023).

140. Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G. & Sutskever, I. *Learning transferable visual models from natural language supervision* in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event* (2021).

141. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., *et al.* Language models are unsupervised multitask learners. *OpenAI blog* **1,** 9 (2019).

142. Dupont, E., Kim, H., Eslami, S. M. A., Rezende, D. J. & Rosenbaum, D. *From data to functa: your data point is a function and you can treat it like one* in *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA* (2022).

143. Balaji, Y., Min, M. R., Bai, B., Chellappa, R. & Graf, H. P. *Conditional GAN with discriminative filter generation for text-to-video synthesis* in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019* (2019).

144. Han, L., Ren, J., Lee, H., Barbieri, F., Olszewski, K., Minaee, S., Metaxas, D. N. & Tulyakov, S. *Show me what and tell me how: video synthesis via multimodal conditioning* in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022* (2022).

145. Hong, W., Ding, M., Zheng, W., Liu, X. & Tang, J. *Cogvideo: large-scale pretraining for text-to-video generation via transformers* in *International Conference on Learning Representations* (2023).

146. Ma, X., Wang, Y., Jia, G., Chen, X., Liu, Z., Li, Y.-F., Chen, C. & Qiao, Y. Latte: latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048* (2024).

147. Chan, E. R., Lin, C. Z., Chan, M. A., Nagano, K., Pan, B., Mello, S. D., Gallo, O., Guibas, L. J., Tremblay, J., Khamis, S., Karras, T. & Wetzstein, G. *Efficient geometry-aware 3d generative adversarial networks* in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022* (2022).

148. Kulhánek, J., Derner, E., Sattler, T. & Babuška, R. *Viewformer: nerf-free neural rendering from few images using transformers* in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV* (2022).

149. Yu, A., Ye, V., Tancik, M. & Kanazawa, A. *Pixelnerf: neural radiance fields from one or few images* in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021* (2021).

150. Krizhevsky, A., Hinton, G., *et al.* Learning multiple layers of features from tiny images (2009).

151. Yu, J., Zhu, H., Jiang, L., Loy, C. C., Cai, W. & Wu, W. *Celebv-text: A large-scale facial text-video dataset* in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023* (2023).

152. Han, L., Ren, J., Lee, H., Barbieri, F., Olszewski, K., Minaee, S., Metaxas, D. N. & Tulyakov, S. *Show me what and tell me how: video synthesis via multimodal conditioning* in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022* (2022).

153. Wu, C., Huang, L., Zhang, Q., Li, B., Ji, L., Yang, F., Sapiro, G. & Duan, N. Godiva: generating open-domain videos from natural descriptions. *ArXiv preprint* (2021).

154. Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., *et al.* Shapenet: an information-rich 3d model repository. *ArXiv preprint* (2015).

155. Sitzmann, V., Zollhöfer, M. & Wetzstein, G. *Scene representation networks: continuous 3d-structure-aware neural scene representations* in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada* (2019).

156. Kong, X., Liu, S., Lyu, X., Taher, M., Qi, X. & Davison, A. J. *Eschernet: a generative model for scalable view synthesis* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024), 9503–9513.

157. Long, X., Guo, Y.-C., Lin, C., Liu, Y., Dou, Z., Liu, L., Ma, Y., Zhang, S.-H., Habermann, M., Theobalt, C., *et al. Wonder3d: single image to 3d using cross-domain diffusion* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024), 9970–9980.

158. Voleti, V., Yao, C.-H., Boss, M., Letts, A., Pankratz, D., Tochilkin, D., Laforte, C., Rombach, R. & Jampani, V. *Sv3d: novel multi-view synthesis and 3d generation from a single image using latent video diffusion* in *European Conference on Computer Vision* (2025), 439–457.

159. Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., *et al.* Era5 monthly averaged data on single levels from 1979 to present. *Copernicus Climate Change Service (C3S) Climate Data Store (CDS)* (2019).

160. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., *et al.* Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems* (2022).

161. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J. & Ermon, S. *Sdedit: guided image synthesis and editing with stochastic differential equations* in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022* (2022).

162. Zhang, L., Rao, A. & Agrawala, M. *Adding conditional control to text-to-image diffusion models* in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023), 3836–3847.

163. Whang, J., Delbracio, M., Talebi, H., Saharia, C., Dimakis, A. G. & Milanfar, P. *Deblurring via stochastic refinement* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), 16293–16303.

164. Brooks, T., Holynski, A. & Efros, A. A. *Instructpix2pix: learning to follow image editing instructions* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023).

165. Parmar, G., Kumar Singh, K., Zhang, R., Li, Y., Lu, J. & Zhu, J.-Y. *Zero-shot image-to-image translation* in *ACM SIGGRAPH 2023 Conference Proceedings* (2023).

166. Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D. & Norouzi, M. *Palette: image-to-image diffusion models* in *ACM SIGGRAPH 2022 Conference Proceedings* (2022).

167. Meng, C., Rombach, R., Gao, R., Kingma, D., Ermon, S., Ho, J. & Salimans, T. *On distillation of guided diffusion models* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023).

168. Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J. & Komatsuzaki, A. Laion-400m: open dataset of clip-filtered 400 million image-text pairs. *ArXiv preprint* (2021).

169. Xu, Y., Deng, M., Cheng, X., Tian, Y., Liu, Z. & Jaakkola, T. Restart sampling for improving generative processes. *ArXiv preprint* (2023).

170. Zheng, H., Nie, W., Vahdat, A., Azizzadenesheli, K. & Anandkumar, A. *Fast sampling of diffusion models via operator learning* in *International Conference on Machine Learning* (2023).

171. Xia, B., Zhang, Y., Wang, S., Wang, Y., Wu, X., Tian, Y., Yang, W. & Van Gool, L. Diffir: efficient diffusion model for image restoration. *ICCV* (2023).

172. Yue, Z., Wang, J. & Loy, C. C. *Resshift: efficient diffusion model for image super-resolution by residual shifting* in *Thirty-seventh Conference on Neural Information Processing Systems* (2023).

173. Song, Y. & Dhariwal, P. Improved techniques for training consistency models. *ArXiv preprint* (2023).

174. Kim, D., Lai, C.-H., Liao, W.-H., Murata, N., Takida, Y., Uesaka, T., He, Y., Mitsufuji, Y. & Ermon, S. Consistency trajectory models: learning probability flow ode trajectory of diffusion. *ArXiv preprint* (2023).

175. Lu, H., Lu, Y., Jiang, D., Szabados, S. R., Sun, S. & Yu, Y. *Cm-gan: stabilizing gan training with consistency models* in *ICML 2023 Workshop on Structured Probabilistic Inference {\&} Generative Modeling* (2023).

176. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., de Laroussilhe, Q., Gesmundo, A., Attariyan, M. & Gelly, S. *Parameter-efficient transfer learning for NLP* in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA* (eds Chaudhuri, K. & Salakhutdinov, R.) (2019).

177. Stickland, A. C. & Murray, I. *BERT and pals: projected attention layers for efficient adaptation in multi-task learning* in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA* (eds Chaudhuri, K. & Salakhutdinov, R.) (2019).

178. Rosenfeld, A. & Tsotsos, J. K. Incremental learning through deep adaptation. *IEEE transactions on pattern analysis and machine intelligence* (2018).

179. Rebuffi, S., Bilen, H. & Vedaldi, A. *Efficient parametrization of multi-domain deep neural networks* in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018* (2018).

180. Zhang, K., Mo, L., Chen, W., Sun, H. & Su, Y. Magicbrush: a manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems* **36** (2024).

181. Alaluf, Y., Tov, O., Mokady, R., Gal, R. & Bermano, A. *Hyperstyle: stylegan inversion with hypernetworks for real image editing* in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022* (2022).

182. Dinh, T. M., Tran, A. T., Nguyen, R. & Hua, B. *Hyperinverter: improving stylegan inversion via hypernetwork* in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022* (2022).

183. Mou, C., Wang, X., Xie, L., Zhang, J., Qi, Z., Shan, Y. & Qie, X. T2i-adapter: learning adapters to dig out more controllable ability for text-to-image diffusion models. *ArXiv preprint* (2023).

184. Balaji, Y., Nah, S., Huang, X., Vahdat, A., Song, J., Kreis, K., Aittala, M., Aila, T., Laine, S., Catanzaro, B., *et al.* Ediffi: text-to-image diffusion models with an ensemble of expert denoisers. *ArXiv preprint* (2022).

185. Wang, X., Xie, L., Dong, C. & Shan, Y. Realesrgan: training real-world blind super-resolution with pure synthetic data supplementary material. *Computer Vision Foundation open access* (2022).

186. Wang, J., Yue, Z., Zhou, S., Chan, K. C. & Loy, C. C. Exploiting diffusion prior for real-world image super-resolution. *ArXiv preprint* (2023).

187. Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R. & Van Gool, L. R. *Inpainting using denoising diffusion probabilistic models* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022* ().

188. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., *et al.* Imagenet large scale visual recognition challenge. *International journal of computer vision* (2015).

189. Nair, N. G., Mei, K. & Patel, V. M. *A comparison of different atmospheric turbulence simulation methods for image restoration* in *IEEE international conference on image processing* (2022).

190. Menon, S., Damian, A., Hu, S., Ravi, N. & Rudin, C. *Pulse: Self-supervised photo upsampling via latent space exploration of generative models* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), 2437–2445.

191. Chan, K. C., Wang, X., Xu, X., Gu, J. & Loy, C. C. *Glean: Generative latent bank for large-factor image super-resolution* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 14245–14254.

192. Wu, Y., Wang, X., Li, Y., Zhang, H., Zhao, X. & Shan, Y. *Towards Vivid and Diverse Image Colorization with Generative Color Prior* in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), 14377–14386.

193. Yang, T., Ren, P., Xie, X. & Zhang, L. *GAN Prior Embedded Network for Blind Face Restoration in the Wild* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 672–681.

194. Wang, X., Li, Y., Zhang, H. & Shan, Y. *Towards Real-World Blind Face Restoration with Generative Facial Prior* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 9168–9178.

195. Gu, J., Shen, Y. & Zhou, B. *Image processing using multi-code gan prior* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), 3012–3021.

196. Pan, X., Zhan, X., Dai, B., Lin, D., Loy, C. C. & Luo, P. Exploiting deep generative prior for versatile image restoration and manipulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021).

197. Zhu, J.-Y., Krähenbühl, P., Shechtman, E. & Efros, A. A. *Generative visual manipulation on the natural image manifold* in *Proceedings of the European Conference on Computer Vision* (2016), 597–613.

198. Lin, J., Zhang, R., Ganz, F., Han, S. & Zhu, J.-Y. *Anycost gans for interactive image synthesis and editing* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 14986–14996.

199. Pidhorskyi, S., Adjeroh, D. A. & Doretto, G. *Adversarial latent autoencoders* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), 14104–14113.

200. Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S. & Cohen-Or, D. *Encoding in style: a stylegan encoder for image-to-image translation* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 2287–2296.

201. Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O. & Cohen-Or, D. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics* **40,** 1–14 (2021).

202. Deng, J., Guo, J., Xue, N. & Zafeiriou, S. *Arcface: Additive angular margin loss for deep face recognition* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), 4690–4699.

203. Karras, T., Laine, S. & Aila, T. *A style-based generator architecture for generative adversarial networks* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), 4401–4410.

204. Mei, K., Patel, V. M. & Huang, R. *Deep semantic statistics matching (d2sm) denoising network* in *Proceedings of the European conference on computer vision (ECCV)* (2022).

205. Mechrez, R., Talmi, I. & Zelnik-Manor, L. *The contextual loss for image transformation with non-aligned data* in *Proceedings of the European Conference on Computer Vision* (2018), 768–783.

206. Mechrez, R., Talmi, I., Shama, F. & Zelnik-Manor, L. *Maintaining natural image statistics with the contextual loss* in *Asian Conference on Computer Vision* (2018), 427–443.

207. Arjovsky, M. & Bottou, L. *Towards principled methods for training generative adversarial networks* in *International Conference on Learning Representations* (2017).

208. Deng, J., Guo, J., Zhou, Y., Yu, J., Kotsia, I. & Zafeiriou, S. Retinaface: Single-stage dense face localisation in the wild. *arXiv preprint arXiv:1905.00641* (2019).

209. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, D. & Wang, Z. *Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), 1874–1883.

210. Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D. J. & Norouzi, M. Image super-resolution via iterative refinement. *arXiv preprint arXiv:2104.07636* (2021).

211. Roggemann, M. C. & Welsh, B. M. *Imaging through turbulence* (CRC press, 2018).

212. Goodman, J. W. *Statistical optics* (John Wiley & Sons, 2015).

213. Chak, W. H., Lau, C. P. & Lui, L. M. Subsampled turbulence removal network. *arXiv preprint arXiv:1807.04418* (2018).

214. Lau, C. P. & Lui, L. M. Subsampled turbulence removal network. *Mathematics, Computation and Geometry of Data* **1,** 1–33 (2021).

215. Yasarla, R. & Patel, V. M. Learning to Restore a Single Face Image Degraded by Atmospheric Turbulence using CNNs. *arXiv preprint arXiv:2007.08404* (2020).

216. Lau, C. P., Castillo, C. D. & Chellappa, R. ATFaceGAN: Single Face Semantic Aware Image Restoration and Recognition From Atmospheric Turbulence. *IEEE Transactions on Biometrics, Behavior, and Identity Science* **3,** 240–251 (2021).

217. Yasarla, R. & Patel, V. M. Cnn-based restoration of a single face image degraded by atmospheric turbulence. *IEEE Transactions on Biometrics, Behavior, and Identity Science* (2022).

218. Nair, N. G. & Patel, V. M. *Confidence Guided Network For Atmospheric Turbulence Mitigation* in *2021 IEEE International Conference on Image Processing* (2021), 1359–1363.

219. Gou, Y., Hu, P., Lv, J. & Peng, X. Multi-scale adaptive network for single image denoising. *arXiv preprint arXiv:2203.04313* (2022).

220. Mehri, A., Ardakani, P. B. & Sappa, A. D. *Mprnet: multi-path residual network for lightweight image super resolution* in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (2021), 2704–2713.

221. Chimitt, N. & Chan, S. H. *Simulating anisoplanatic turbulence by sampling correlated Zernike coefficients* in *2020 IEEE International Conference on Computational Photography (ICCP)* (2020), 1–12.

222. Mao, Z., Chimitt, N. & Chan, S. H. *Accelerating atmospheric turbulence simulation via learned phase-to-space transform* in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), 14759–14768.

223. Bos, J. P. & Roggemann, M. C. Technique for simulating anisoplanatic image formation over long horizontal paths. *Optical Engineering* **51,** 101704 (2012).

224. Hardie, R. C., Power, J. D., LeMaster, D. A., Droege, D. R., Gladysz, S. & Bose-Pillai, S. Simulation of anisoplanatic imaging through optical turbulence using numerical wave propagation with new validation analysis. *Optical Engineering* **56,** 071502 (2017).

225. Li, X., Liu, M., Ye, Y., Zuo, W., Lin, L. & Yang, R. *Learning warped guidance for blind face restoration* in *Proceedings of the European Conference on Computer Vision* (2018), 272–289.

226. Li, X., Chen, C., Zhou, S., Lin, X., Zuo, W. & Zhang, L. *Blind face restoration via deep multi-scale component dictionaries* in *Proceedings of the European Conference on Computer Vision* (2020), 399–415.

227. Yang, L., Wang, S., Ma, S., Gao, W., Liu, C., Wang, P. & Ren, P. *Hifacegan: Face renovation via collaborative suppression and replenishment* in *Proceedings of the 28th ACM International Conference on Multimedia* (2020), 1551–1560.

228. Xia, W., Zhang, Y., Yang, Y., Xue, J.-H., Zhou, B. & Yang, M.-H. Gan inversion: A survey. *arXiv preprint arXiv:2101.05278* (2021).

229. Creswell, A. & Bharath, A. A. Inverting the generator of a generative adversarial network. *IEEE Transactions on Neural Networks and Learning Systems* **30,** 1967–1974 (2018).

230. Ma, F., Ayaz, U. & Karaman, S. *Invertibility of convolutional generative networks from partial measurements* in *Proceedings of the International Conference on Neural Information Processing Systems* (2018), 9651–9660.

231. Bau, D., Zhu, J.-Y., Wulff, J., Peebles, W., Strobelt, H., Zhou, B. & Torralba, A. *Seeing what a gan cannot generate* in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), 4502–4511.

232. Johnson, J., Alahi, A. & Fei-Fei, L. *Perceptual losses for real-time style transfer and super-resolution* in *Proceedings of the European Conference on Computer Vision* (2016), 694–711.

233. Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B. & Smola, A. A kernel two-sample test. *The Journal of Machine Learning Research* **13,** 723–773 (2012).

234. Zhang, Z., Zhang, R., Li, Z., Bengio, Y. & Paull, L. *Perceptual generative autoencoders* in *International Conference on Machine Learning* (2020), 11298–11306.

235. Zoran, D. & Weiss, Y. *From learning models of natural image patches to whole image restoration* in *International Conference on Computer Vision* (2011), 479–486.

236. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, D. & Wang, Z. *Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2016), 1874–1883.

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

238. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y. & Change Loy, C. *Esrgan: enhanced super-resolution generative adversarial networks* in *Proceedings of the European Conference on Computer Vision Workshops* (2018), 0–0.

239. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J. & Aila, T. *Analyzing and Improving the Image Quality of StyleGAN* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), 8110–8119.

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

241. Miller, K. J., Preece, B., Du Bosq, T. W. & Leonard, K. R. *A data-constrained algorithm for the emulation of long-range turbulence-degraded video* in *Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXX* **11001** (2019), 110010J.

242. Chen, C., Li, X., Yang, L., Lin, X., Zhang, L. & Wong, K.-Y. K. *Progressive semantic-aware style transformation for blind face restoration* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 11896–11905.

243. Simard, P. Y., Steinkraus, D. & Platt, J. C. *Best practices for convolutional neural networks applied to visual document analysis.* in *ICDAR* **3** (2003).

244. Zamir, S. W., Arora, A., Khan, S., Hayat, M., Khan, F. S., Yang, M.-H. & Shao, L. *Multi-stage progressive image restoration* in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2021), 14821–14831.

245. Yasarla, R. & Patel, V. M. *Learning to restore images degraded by atmospheric turbulence using uncertainty* in *2021 IEEE International Conference on Image Processing (ICIP)* (2021), 1694–1698.

246. Lau, C. P., Souri, H. & Chellappa, R. *ATFaceGAN: Single face image restoration and recognition from atmospheric turbulence* in *IEEE International Conference on Automatic Face and Gesture Recognition* (2020), 32–39.

247. Choi, J., Lee, J., Jeong, Y. & Yoon, S. Toward Spatially Unbiased Generative Models. *arXiv preprint arXiv:2108.01285* (2021).

248. Zhang, R., Isola, P., Efros, A. A., Shechtman, E. & Wang, O. *The unreasonable effectiveness of deep features as a perceptual metric* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), 586–595.

249. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. & Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems* **30** (2017).

250. Mittal, A., Soundararajan, R. & Bovik, A. C. Making a "completely blind" image quality analyzer. *IEEE Signal processing letters* **20,** 209–212 (2012).

251. Blau, Y., Mechrez, R., Timofte, R., Michaeli, T. & Zelnik-Manor, L. *The 2018 pirm challenge on perceptual image super-resolution* in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops* (2018), 0–0.

252. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., *et al. Photo-realistic single image super-resolution using a generative adversarial network* in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition* (2017).

253. Kendall, A. & Gal, Y. What uncertainties do we need in bayesian deep learning for computer vision? *arXiv preprint arXiv:1703.04977* (2017).

254. Shen, Y. & Zhou, B. *Closed-form factorization of latent semantics in gans* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 1532–1540.

255. Ma, H., Qin, Q. & Shen, X. *Shadow segmentation and compensation in high resolution satellite images* in *IGARSS 2008-2008 IEEE International Geoscience and Remote Sensing Symposium* **2** (2008), II–1036.

256. Zhang, W., Zhao, X., Morvan, J.-M. & Chen, L. Improving shadow suppression for illumination robust face recognition. *IEEE TPAMI* (2018).

257. Zhang, E., Martin-Brualla, R., Kontkanen, J. & Curless, B. L. *No shadow left behind: removing objects and their shadows using approximate lighting and geometry* in *CVPR* (2021).

258. Hou, A., Sarkis, M., Bi, N., Tong, Y. & Liu, X. *Face relighting with geometrically consistent shadows* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), 4217–4226.

259. Ho, J., Jain, A. & Abbeel, P. Denoising diffusion probabilistic models. *NeurIPS* (2020).

260. Wan, J., Yin, H., Wu, Z., Wu, X., Liu, Y. & Wang, S. *Style-guided shadow removal* in *ECCV* (2022).

261. Zhao, S., Song, J. & Ermon, S. Infovae: information maximizing variational autoencoders. *arXiv preprint arXiv:1706.02262* (2017).

262. He, J., Spokoyny, D., Neubig, G. & Berg-Kirkpatrick, T. Lagging inference networks and posterior collapse in variational autoencoders. *arXiv preprint arXiv:1901.05534* (2019).

263. Dieng, A. B., Kim, Y., Rush, A. M. & Blei, D. M. *Avoiding latent variable collapse with generative skip models* in *International Conference on Artificial Intelligence and Statistics* (2019).

264. Arbel, E. & Hel-Or, H. Shadow removal using intensity surfaces and texture anchor points. *IEEE TPAMI* (2010).

265. Finlayson, G. D., Hordley, S. D., Lu, C. & Drew, M. S. On the removal of shadows from images. *IEEE TPAMI* (2005).

266. Finlayson, G. D., Drew, M. S. & Lu, C. Entropy minimization for shadow removal. *IJCV* (2009).

267. Guo, R., Dai, Q. & Hoiem, D. Paired regions for shadow detection and removal. *IEEE TPMAI* (2012).

268. Hu, X., Jiang, Y., Fu, C.-W. & Heng, P.-A. *Mask-shadowgan: learning to remove shadows from unpaired data* in *ICCV* (2019).

269. Chen, Z., Long, C., Zhang, L. & Xiao, C. *Canet: a context-aware network for shadow removal* in *ICCV* (2021).

270. Fu, L., Zhou, C., Guo, Q., Juefei-Xu, F., Yu, H., Feng, W., Liu, Y. & Wang, S. *Auto-exposure fusion for single-image shadow removal* in *CVPR* (2021).

271. Jin, Y., Sharma, A. & Tan, R. T. *Dc-shadownet: single-image hard and soft shadow removal using unsupervised domain-classifier guided network* in *ICCV* (2021).

272. Liu, Z., Yin, H., Wu, X., Wu, Z., Mi, Y. & Wang, S. *From shadow generation to shadow removal* in *CVPR* (2021).

273. Le, H. & Samaras, D. *Shadow removal via shadow image decomposition* in *ICCV* (2019).

274. Le, H. & Samaras, D. *From shadow segmentation to shadow removal* in *ECCV* (2020).

275. Wan, J., Yin, H., Wu, Z., Wu, X., Liu, Z. & Wang, S. Crformer: a cross-region transformer for shadow removal. *arXiv preprint arXiv:2207.01600* (2022).

276. Guo, L., Huang, S., Liu, D., Cheng, H. & Wen, B. Shadowformer: global context helps image shadow removal. *arXiv preprint arXiv:2302.01650* (2023).

277. Zhang, L., Long, C., Zhang, X. & Xiao, C. *Ris-gan: explore residual and illumination with generative adversarial networks for shadow removal* in *AAAI* (2020).

278. Ding, B., Long, C., Zhang, L. & Xiao, C. *Argan: attentive recurrent generative adversarial network for shadow detection and removal* in *ICCV* (2019).

279. Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R. & Van Gool, L. *Repaint: inpainting using denoising diffusion probabilistic models* in *Proceedings*

of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022).

280. Zhu, Y., Huang, J., Fu, X., Zhao, F., Sun, Q. & Zha, Z.-J. *Bijective Mapping Network for Shadow Removal* in *CVPR* (2022).

281. Funt, B. V. & Finlayson, G. D. Color constant color indexing. *IEEE TPAMI* (1995).

282. Stricker, M. A. & Orengo, M. *Similarity of color images* in *Storage and retrieval for image and video databases III* **2420** (1995), 381–392.

283. Finlayson, G. D., Chatterjee, S. S. & Funt, B. V. *Color angular indexing* in *ECCV* (1996).

284. Fu, H., Li, C., Liu, X., Gao, J., Celikyilmaz, A. & Carin, L. Cyclical annealing schedule: a simple approach to mitigating kl vanishing. *arXiv preprint arXiv:1903.10145* (2019).

285. Tolstikhin, I., Bousquet, O., Gelly, S. & Schoelkopf, B. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558* (2017).

286. Preechakul, K., Chatthee, N., Wizadwongsa, S. & Suwajanakorn, S. *Diffusion autoencoders: toward a meaningful and decodable representation* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), 10619–10629.

287. Yang, Z., Hu, Z., Salakhutdinov, R. & Berg-Kirkpatrick, T. *Improved variational autoencoders for text modeling using dilated convolutions* in *International conference on machine learning* (2017), 3881–3890.

288. Wang, J., Li, X. & Yang, J. *Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal* in *CVPR* (2018).

289. Qu, L., Tian, J., He, S., Tang, Y. & Lau, R. W. *Deshadownet: a multi-context embedding deep network for shadow removal* in *CVPR* (2017).

290. Cun, X., Pun, C.-M. & Shi, C. *Towards ghost-free shadow removal via dual hierarchical aggregation network and shadow matting GAN* in *AAAI* (2020).

291. Wang, T., Hu, X., Wang, Q., Heng, P.-A. & Fu, C.-W. *Instance shadow detection* in *CVPR* (2020).

292. Hong, Y., Niu, L. & Zhang, J. *Shadow generation for composite image in real-world scenes* in *AAAI* (2022).

293. Inoue, N. & Yamasaki, T. Learning from synthetic shadows for shadow detection and removal. *IEEE TCSVT* (2020).

294. Zhu, Y., Xiao, Z., Fang, Y., Fu, X., Xiong, Z. & Zha, Z.-J. *Efficient model-driven network for shadow removal* in *AAAI* (2022).

295. Lu, C., Zhang, J., Chu, Y., Chen, Z., Zhou, J., Wu, F., Chen, H. & Yang, H. Knowledge distillation of transformer-based language models revisited. *arXiv preprint arXiv:2206.14366* (2022).

296. Guo, L., Wang, C., Yang, W., Huang, S., Wang, Y., Pfister, H. & Wen, B. *Shadowdiffusion: when degradation prior meets diffusion model for shadow removal* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), 14049–14058.

297. Mairal, J., Elad, M. & Sapiro, G. Sparse representation for color image restoration. *IEEE TIP* (2007).

298. Mairal, J., Bach, F., Ponce, J., Sapiro, G. & Zisserman, A. *Non-local sparse models for image restoration* in *ICCV* (2009).

299. Larochelle, H. & Murray, I. *The neural autoregressive distribution estimator* in *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (2011).

300. Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S. & Cohen-Or, D. *Encoding in style: a stylegan encoder for image-to-image translation* in *IEEE Conference on Computer Vision and Pattern Recognition* (2021).

301. Menon, S., Damian, A., Hu, S., Ravi, N. & Rudin, C. *Pulse: self-supervised photo upsampling via latent space exploration of generative models* in *CVPR* (2020).

302. Mei, K., Jiang, A., Li, J., Ye, J. & Wang, M. *An effective single-image super-resolution model using squeeze-and-excitation networks* in *Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13–16, 2018, Proceedings, Part VI 25* (2018), 542–553.

303. Liu, Z.-S., Siu, W.-C. & Wang, L.-W. *Variational autoencoder for reference based image super-resolution* in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2021), 516–525.

304. Xiao, J., Jiang, X., Zheng, N., Yang, H., Yang, Y., Yang, Y., Li, D. & Lam, K.-M. Online video super-resolution with convolutional kernel bypass grafts. *IEEE Transactions on Multimedia* **25,** 8972–8987 (2023).

305. Mei, K., Patel, V. M. & Huang, R. *Deep semantic statistics matching (d2sm) denoising network* in *Proceedings of the European Conference on Computer Vision* (2022).

306. Wu, C. H. & De la Torre, F. *A latent space of stochastic diffusion models for zero-shot image editing and guidance* in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023), 7378–7387.

307. Nie, S., Guo, H. A., Lu, C., Zhou, Y., Zheng, C. & Li, C. The blessing of randomness: sde beats ode in general diffusion-based image editing. *arXiv preprint arXiv:2311.01410* (2023).

308. Kawar, B., Elad, M., Ermon, S. & Song, J. Denoising diffusion restoration models. *Advances in Neural Information Processing Systems* **35,** 23593–23606 (2022).

309. Creswell, A. & Bharath, A. A. Inverting the generator of a generative adversarial network. *IEEE TNNLS* (2018).

310. Abdal, R., Qin, Y. & Wonka, P. *Image2stylegan: how to embed images into the stylegan latent space?* in *ICCV* (2019).

311. Abdal, R., Qin, Y. & Wonka, P. *Image2stylegan++: how to edit the embedded images?* in *CVPR* (2020).

312. Gu, J., Shen, Y. & Zhou, B. *Image processing using multi-code gan prior* in *CVPR* (2020).

313. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.-Y. & Ermon, S. *SDEdit: guided image synthesis and editing with stochastic differential equations* in *International Conference on Learning Representations* (2022).

314. Zhu, J.-Y., Krähenbühl, P., Shechtman, E. & Efros, A. A. *Generative visual manipulation on the natural image manifold* in *ECCV* (2016).

315. Wang, X., Li, Y., Zhang, H. & Shan, Y. *Towards real-world blind face restoration with generative facial prior* in *CVPR* (2021).

316. Chan, K. C., Wang, X., Xu, X., Gu, J. & Loy, C. C. *Glean: generative latent bank for large-factor image super-resolution* in *CVPR* (2021).

317. Mei, K., Tu, Z., Delbracio, M., Talebi, H., Patel, V. M. & Milanfar, P. Bigger is not always better: scaling properties of latent diffusion models. *arXiv preprint arXiv:2404.01367* (2024).

318. Mei, K., Delbracio, M., Talebi, H., Tu, Z., Patel, V. M. & Milanfar, P. *Codi: conditional diffusion distillation for higher-fidelity and faster image generation* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024), 9048–9058.

319. Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I. & Chen, M. Glide: towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741* (2021).

320. Wang, W., Bao, J., Zhou, W., Chen, D., Chen, D., Yuan, L. & Li, H. Semantic image synthesis via diffusion models. *arXiv preprint arXiv:2207.00050* (2022).

321. Song, Y. & Ermon, S. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems* **32** (2019).

322. Grandvalet, Y. & Bengio, Y. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems* **17** (2004).

323. Liu, Z., Luo, P., Wang, X. & Tang, X. *Deep learning face attributes in the wild* in *ICCV* (2015).

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

325. Zhang, H., Sindagi, V. & Patel, V. M. Image de-raining using a conditional generative adversarial network. *IEEE TCSVT* (2019).

326. Yang, W., Tan, R. T., Feng, J., Liu, J., Guo, Z. & Yan, S. *Deep joint rain detection and removal from a single image* in *CVPR* (2017).

327. Chen, J., Yu, J., Ge, C., Yao, L., Xie, E., Wu, Y., Wang, Z., Kwok, J., Luo, P., Lu, H., *et al.* Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426* (2023).

328. Wang, J., Yue, Z., Zhou, S., Chan, K. C. & Loy, C. C. Exploiting diffusion prior for real-world image super-resolution. *International Journal of Computer Vision,* 1–21 (2024).

329. Dong, C., Loy, C. C., He, K. & Tang, X. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence* (2015).

330. Zhang, K., Zuo, W. & Zhang, L. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *TIP* (2018).

331. Li, J., Fang, F., Mei, K. & Zhang, G. *Multi-scale residual network for image super-resolution* in *ECCV* (2018).

332. Simonyan, K. & Zisserman, A. *Very deep convolutional networks for large-scale image recognition* in *ICLR* (2015).

333. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. & Fei-Fei, L. *Imagenet: A large-scale hierarchical image database* in *CVPR* (2009).

334. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. Generative adversarial nets. *Advances in neural information processing systems* **27** (2014).

335. Van der Maaten, L. & Hinton, G. Visualizing data using t-SNE. *JMLR* (2008).

336. Li, C.-L., Chang, W.-C., Cheng, Y., Yang, Y. & Póczos, B. Mmd gan: towards deeper understanding of moment matching network. *arXiv preprint arXiv:1705.08584* (2017).

337. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S. & Schiele, B. *The cityscapes dataset for semantic urban scene understanding* in *CVPR* (2016).

338. Scott, D. W. *Multivariate density estimation: theory, practice, and visualization* (2015).

339. Zontak, M. & Irani, M. *Internal statistics of a single natural image* in *CVPR* (2011).

340. Wu, Z., Xiong, Y., Yu, S. X. & Lin, D. *Unsupervised Feature Learning via Non-parametric Instance Discrimination* in *CVPR* (2018).

341. He, K., Fan, H., Wu, Y., Xie, S. & Girshick, R. *Momentum Contrast for Unsupervised Visual Representation Learning* in *CVPR* (2020).

342. Wang, Z., Bovik, A. C., Sheikh, H. R. & Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* **13,** 600–612 (2004).

343. Dosovitskiy, A. & Brox, T. *Generating images with perceptual similarity metrics based on deep networks* in *NeurIPS* (2016).

344. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J. & Wang, Z. *Photo-realistic single image super-resolution using a generative adversarial network* in *CVPR* (2017).

345. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y. & Change Loy, C. *Esrgan: Enhanced super-resolution generative adversarial networks* in *ECCV Workshops* (2018).

346. Mei, K., Ye, S. & Huang, R. *Sdan: squared deformable alignment network for learning misaligned optical zoom* in *ICME* (2021).

347. Liu, D. *Connecting Low-Level Image Processing and High-Level Vision via Deep Learning* in *IJCAI* (2018).

348. Liu, D., Wen, B., Jiao, J., Liu, X., Wang, Z. & Huang, T. S. Connecting image denoising and high-level vision tasks via deep learning. *TIP* (2020).

349. Agustsson, E. & Timofte, R. *Ntire 2017 challenge on single image super-resolution: Dataset and study* in *CVPR Workshops* (2017).

350. Liu, Z., Luo, P., Wang, X. & Tang, X. *Deep learning face attributes in the wild* in *ICCV* (2015).

351. Le, V., Brandt, J., Lin, Z., Bourdev, L. & Huang, T. S. *Interactive facial feature localization* in *ECCV* (2012).

352. Zhang, X., Chen, Q., Ng, R. & Koltun, V. *Zoom to Learn, Learn to Zoom* in *CVPR* (2019).

353. Li, Y., Swersky, K. & Zemel, R. *Generative moment matching networks* in *ICML* (2015).

354. Santos, C. N. d., Mroueh, Y., Padhi, I. & Dognin, P. *Learning implicit generative models by matching perceptual features* in *CVPR* (2019).

355. Sun, B. & Saenko, K. *Deep coral: Correlation alignment for deep domain adaptation* in *ECCV* (2016).

356. Tolstikhin, I. O., Sriperumbudur, B. K. & Schölkopf, B. *Minimax estimation of maximum mean discrepancy with radial kernels* in *NeurIPS* (2016).

357. Torkkola, K. Feature extraction by non-parametric mutual information maximization. *JMLR* (2003).

358. Passalis, N. & Tefas, A. *Learning deep representations with probabilistic knowledge transfer* in *ECCV* (2018).

359. Peng, B., Jin, X., Liu, J., Li, D., Wu, Y., Liu, Y., Zhou, S. & Zhang, Z. *Correlation congruence for knowledge distillation* in *ICCV* (2019).

360. Tung, F. & Mori, G. *Similarity-preserving knowledge distillation* in *ICCV* (2019).

361. Park, W., Kim, D., Lu, Y. & Cho, M. *Relational knowledge distillation* in *CVPR* (2019).

362. Liu, Y., Cao, J., Li, B., Yuan, C., Hu, W., Li, Y. & Duan, Y. *Knowledge distillation via instance relationship graph* in *CVPR* (2019).

363. Chen, H., Wang, Y., Xu, C., Xu, C. & Tao, D. Learning student networks via feature embedding. *TNNLS* (2020).

364. Passalis, N., Tzelepi, M. & Tefas, A. Probabilistic Knowledge Transfer for Lightweight Deep Representation Learning. *TNNLS* (2020).

365. Turlach, B. Bandwidth Selection in Kernel Density Estimation: A Review. *CORE and Institut de Statistique* (1999).

366. Wang, D., Lu, H. & Bo, C. Visual tracking via weighted local cosine similarity. *TCYB* (2014).

367. Shocher, A., Cohen, N. & Irani, M. *Zero-Shot Super-Resolution Using Deep Internal Learning* in *CVPR* (2018).

368. Shaham, T. R., Dekel, T. & Michaeli, T. *Singan: Learning a generative model from a single natural image* in *ICCV* (2019).

369. Park, T., Efros, A. A., Zhang, R. & Zhu, J.-Y. *Contrastive learning for unpaired image-to-image translation* in *ECCV* (2020).

370. Li, B., Peng, X., Wang, Z., Xu, J. & Feng, D. *Aod-net: All-in-one dehazing network* in *ICCV* (2017).

371. Li, J., Fang, F., Li, J., Mei, K. & Zhang, G. MDCN: Multi-scale Dense Cross Network for Image Super-Resolution. *TCSVT* (2020).

372. Tian, Y., Krishnan, D. & Isola, P. *Contrastive Representation Distillation* in *ICLR* (2019).

373. Ancuti, C., Ancuti, C. O., Timofte, R. & De Vleeschouwer, C. *I-HAZE: a dehazing benchmark with real hazy and haze-free indoor images* in *ICACIVS* (2018).

374. Guo, S., Yan, Z., Zhang, K., Zuo, W. & Zhang, L. *Toward Convolutional Blind Denoising of Real Photographs* in *CVPR* (2019).

375. Chang, M., Li, Q., Feng, H. & Xu, Z. *Spatial-Adaptive Network for Single Image Denoising* in *ECCV* (2020).

376. He, K., Zhang, X., Ren, S. & Sun, J. *Deep Residual Learning for Image Recognition* in *CVPR* (2016).

377. Ma, C., Jiang, Z., Rao, Y., Lu, J. & Zhou, J. *Deep face super-resolution with iterative collaboration between attentive recovery and landmark estimation* in *CVPR* (2020).

378. Yu, X. & Porikli, F. *Ultra-resolving face images by discriminative generative networks* in *ECCV* (2016).

379. Zhang, Y., Tian, Y., Kong, Y., Zhong, B. & Fu, Y. *Residual dense network for image super-resolution* in *CVPR* (2018).

380. Kim, D., Kim, M., Kwon, G. & Kim, D.-S. *Progressive face super-resolution via attention to facial landmark* in *BMVC* (2019).

381. Chen, Y., Tai, Y., Liu, X., Shen, C. & Yang, J. *Fsrnet: End-to-end learning face super-resolution with facial priors* in *CVPR* (2018).

382. He, K., Sun, J. & Tang, X. Single image haze removal using dark channel prior. *TPAMI* (2010).

383. Ren, W., Liu, S., Zhang, H., Pan, J., Cao, X. & Yang, M.-H. *Single image dehazing via multi-scale convolutional neural networks* in *ECCV* (2016).

384. Li, R., Pan, J., Li, Z. & Tang, J. *Single image dehazing via conditional generative adversarial network* in *CVPR* (2018).

385. Ren, W., Ma, L., Zhang, J., Pan, J., Cao, X., Liu, W. & Yang, M.-H. *Gated fusion network for single image dehazing* in *CVPR* (2018).

386. Mei, K., Jiang, A., Li, J. & Wang, M. *Progressive feature fusion network for realistic image dehazing* in *ACCV* (2018).

387. Liu, X., Ma, Y., Shi, Z. & Chen, J. *Griddehazenet: Attention-based multi-scale network for image dehazing* in *ICCV* (2019).

388. Liu, X., Suganuma, M., Sun, Z. & Okatani, T. *Dual residual networks leveraging the potential of paired operations for image restoration* in *CVPR* (2019).

389. Dong, H., Pan, J., Xiang, L., Hu, Z., Zhang, X., Wang, F. & Yang, M.-H. *Multi-scale boosted dehazing network with dense feature fusion* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), 2157–2167.

390. Arjovsky, M. & Bottou, L. Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862* (2017).

391. Zhang, H., Koh, J. Y., Baldridge, J., Lee, H. & Yang, Y. *Cross-modal contrastive learning for text-to-image generation* in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021* (2021).

392. Jayasumana, S., Ramalingam, S., Veit, A., Glasner, D., Chakrabarti, A. & Kumar, S. Rethinking fid: towards a better evaluation metric for image generation. *ArXiv preprint* (2024).

393. Cho, J., Hu, Y., Garg, R., Anderson, P., Krishna, R., Baldridge, J., Bansal, M., Pont-Tuset, J. & Wang, S. Davidsonian scene graph: improving reliability in fine-grained evaluation for text-image generation. *ArXiv preprint* (2023).

394. Ma, N., Goldstein, M., Albergo, M. S., Boffi, N. M., Vanden-Eijnden, E. & Xie, S. Sit: exploring flow and diffusion-based generative models with scalable interpolant transformers. *ArXiv preprint* (2024).

395. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., *et al. Scaling rectified flow transformers for high-resolution image synthesis* in *Forty-first International Conference on Machine Learning* (2024).

396. Fei, Z., Fan, M., Yu, C. & Huang, J. Scalable diffusion models with state space backbone. *arXiv preprint arXiv:2402.05608* (2024).

397. Baldridge, J., Bauer, J., Bhutani, M., Brichtova, N., Bunner, A., Chan, K., Chen, Y., Dieleman, S., Du, Y., Eaton-Rosen, Z., *et al.* Imagen 3. *arXiv preprint arXiv:2408.07009* (2024).

398. Pernias, P., Rampas, D., Richter, M. L., Pal, C. J. & Aubreville, M. Würstchen: an efficient architecture for large-scale text-to-image diffusion models. *arXiv preprint arXiv:2306.00637* (2023).

Appendix A
Proofs

A.1 Self-consistency in Noise Prediction

Remark A.1. If a diffusion model, parameterized by $\hat{\mathbf{v}}_\theta(\mathbf{z}_t, t)$, satisfies the self-consistency property on the noise prediction $\hat{\epsilon}_\theta(\mathbf{z}_t, t) = \alpha_t \hat{\mathbf{v}}_\theta(\mathbf{z}_t, t) + \sigma_t \mathbf{z}_t$, then it also satisfies the self-consistency property on the signal prediction $\hat{\mathbf{x}}_\theta(\mathbf{z}_t, t) = \alpha_t \mathbf{z}_t - \sigma_t \hat{\mathbf{v}}_\theta(\mathbf{z}_t, t)$.

Proof. The diffusion model that satisfies the self-consistency in the noise prediction implies:

$$\begin{aligned} \hat{\epsilon}_\theta(\mathbf{z}_{t'}, t') &= \hat{\epsilon}_\theta(\mathbf{z}_t, t), \\ \alpha_{t'} \hat{\mathbf{v}}_\theta(\mathbf{z}_{t'}, t') + \sigma_{t'} \mathbf{z}_{t'} &= \alpha_t \hat{\mathbf{v}}_\theta(\mathbf{z}_t, t) + \sigma_t \mathbf{z}_t, \\ \hat{\mathbf{v}}_\theta(\mathbf{z}_{t'}, t') &= \frac{\alpha_t \hat{\mathbf{v}}_\theta(\mathbf{z}_t, t) + \sigma_t \mathbf{z}_t - \sigma_{t'} \mathbf{z}_{t'}}{\alpha_{t'}}, \end{aligned} \tag{A.1}$$

Based on the above equivalence, the transformation between the signal prediction $\mathbf{x}_\theta(\mathbf{z}_{t'}, t')$ and $\mathbf{x}_\theta(\mathbf{z}_t, t)$ by using the update ruler in equation 1.8 and the reparameterization trick is:

$$\begin{aligned} \mathbf{x}_\theta(\mathbf{z}_{t'}, t') &= \alpha_{t'} \mathbf{z}_{t'} - \sigma_{t'} \hat{\mathbf{v}}_\theta(\mathbf{z}_{t'}, t') \\ &= \alpha_{t'} \mathbf{z}_{t'} - \sigma_{t'} \frac{\alpha_t \hat{\mathbf{v}}_\theta(\mathbf{z}_t, t) + \sigma_t \mathbf{z}_t - \sigma_{t'} \mathbf{z}_{t'}}{\alpha_{t'}} && \text{// integrating equation A.1} \\ &= \frac{\alpha_{t'}^2 \mathbf{z}_{t'} - \sigma_{t'} \alpha_t \hat{\mathbf{v}}_\theta(\mathbf{z}_t, t) - \sigma_{t'} \sigma_t \mathbf{z}_t + \sigma_{t'}^2 \mathbf{z}_{t'}}{\alpha_{t'}} \end{aligned}$$

$$
\begin{aligned}
&= \frac{(1-\sigma_{t'}^2)\mathbf{z}_{t'} - \sigma_{t'}\alpha_t\hat{\mathbf{v}}_\theta(\mathbf{z}_t,t) - \sigma_{t'}\sigma_t\mathbf{z}_t + \sigma_{t'}^2\mathbf{z}_{t'}}{\alpha_{t'}} \\
&= \frac{\mathbf{z}_{t'} - \sigma_{t'}(\alpha_t\hat{\mathbf{v}}_\theta(\mathbf{z}_t,t) + \sigma_t\mathbf{z}_t)}{\alpha_{t'}} \\
&= \frac{\mathbf{z}_{t'} - \sigma_{t'}(\hat{\epsilon}_\theta(\mathbf{z}_t,t))}{\alpha_{t'}} && \text{// transformed with equation 1.9} \\
&= \frac{\alpha_{t'}\mathbf{x}_\theta(\mathbf{z}_t,t) + \sigma_{t'}\hat{\epsilon}_\theta(\mathbf{z}_t,t) - \sigma_{t'}(\hat{\epsilon}_\theta(\mathbf{z}_t,t))}{\alpha_{t'}} && \text{// update ruler equation 1.10 of DDIM} \\
&= \mathbf{x}_\theta(\mathbf{z}_t,t).
\end{aligned}
$$

The derived equivalence shows that enforcing the self-consistency in the noise prediction, which is implemented by learning to minimize our distillation loss, enforcing the self-consistency in the signal prediction and distilling the pre-trained diffusion model. □

Appendix B
List of Publications

This thesis includes a list of publications utilized in its content.

- Mei, Kangfu, Zhengzhong Tu, Mauricio Delbracio, Hossein Talebi, Vishal M. Patel, and Peyman Milanfar. "Bigger is not Always Better: Scaling Properties of Latent Diffusion Models." Transactions on Machine Learning Research. (2024).

- Mei, Kangfu, and Vishal Patel. "Vidm: Video implicit diffusion models." In Proceedings of the AAAI conference on artificial intelligence, vol. 37, no. 8, pp. 9117-9125. 2023.

- Mei, Kangfu, Mauricio Delbracio, Hossein Talebi, Zhengzhong Tu, Vishal M. Patel, and Peyman Milanfar. "CoDi: Conditional Diffusion Distillation for Higher-Fidelity and Faster Image Generation." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9048-9058. 2024.

- Mei, Kangfu, and Vishal M. Patel. "Ltt-gan: Looking through turbulence by inverting gans." IEEE Journal of Selected Topics in Signal Processing 17, no. 3 (2023): 587-598.

- Mei, Kangfu, Luis Figueroa, Zhe Lin, Zhihong Ding, Scott Cohen, and Vishal M. Patel. "Latent feature-guided diffusion models for shadow removal." In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer

Vision, pp. 4313-4322. 2024.

- Mei, Kangfu, Nithin Gopalakrishnan Nair, and Vishal M. Patel. "Bi-noising diffusion: Towards conditional diffusion models with generative restoration priors." In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2025.

- Mei, Kangfu, Vishal M. Patel, and Rui Huang. "Deep semantic statistics matching (D2SM) denoising network." In European Conference on Computer Vision, pp. 384-400. Cham: Springer Nature Switzerland, 2022.