

Deep Compression Autoencoder for Efficient High-Resolution Diffusion Models

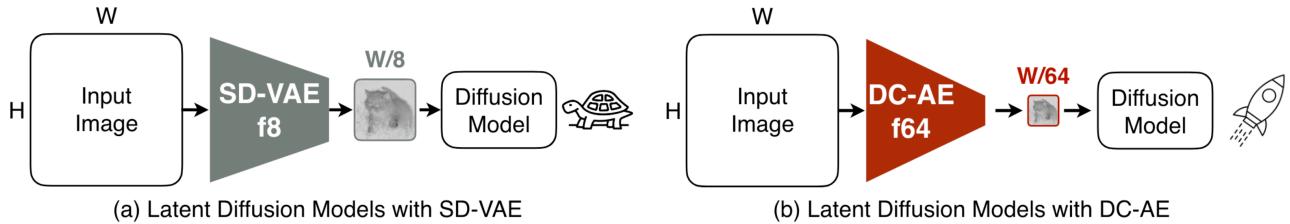


Figure 1: DC-AE accelerates diffusion models by increasing autoencoder’s spatial compression ratio.

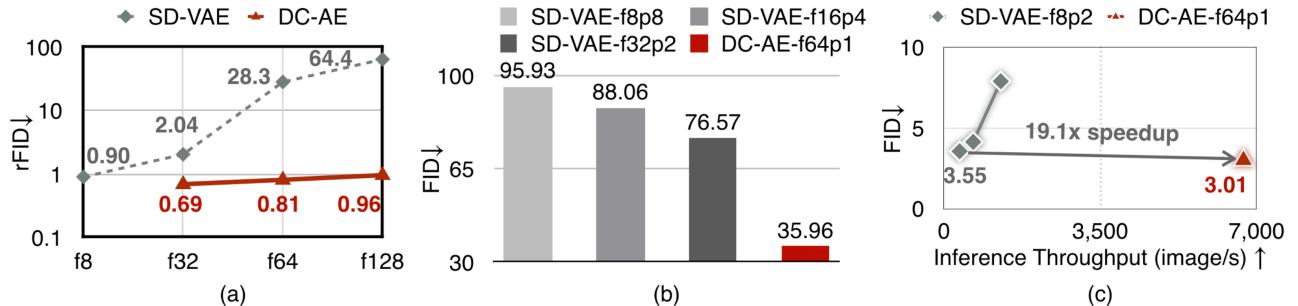


Figure 2: (a) **Image Reconstruction Results on ImageNet 256×256.** f denotes the spatial compression ratio. When the spatial compression ratio increases, SD-VAE has a significant reconstruction accuracy drop (higher rFID) while DC-AE does not have this issue. (b) **ImageNet 512×512 Image Generation Results on UViT-S with Various Autoencoders.** p denotes the patch size. Shifting the token compression task to the autoencoder enables the diffusion model to focus more on the denoising task, leading to better FID. (c) **Comparison to SD-VAE-f8 on ImageNet 512×512 with UViT Variants.** DC-AE-f64p1 provides 19.1× higher inference throughput and 0.54 better ImageNet FID than SD-VAE-f8p2 on UViT-H.

Introduction

近年来，潜变量扩散模型（Latent Diffusion Models, LDMs）已经成为图像合成领域的领先框架，展示了显著的成功。这类模型利用自动编码器（Autoencoder）将图像投影到潜变量空间，从而大幅降低扩散模型的计算成本。例如，当前大多数潜变量扩散模型采用空间压缩比例为8的自动编码器（f8），将空间大小为 $H \times W$ 的图像转换为空间大小为 $H/8 \times W/8$ 的潜变量特征。这种空间压缩比例在低分辨率图像（如256×256）合成中表现令人满

意。

然而，在高分辨率图像合成（如1024\u00d71024）中，进一步提高空间压缩比例至关重要，尤其是在处理具有二次计算复杂度的扩散变换器模型（如扩散变换器）。现有方法通常通过在扩散模型侧下采样实现进一步的空间压缩，而在自动编码器侧的探索却很少。

高空间压缩的主要挑战在于重构精度的下降。例如，SD-VAE在ImageNet 256\u00d7256数据集上，当从f8切换到f64时，其重构FID从0.90显著恶化至28.3。

本文提出了一种新型的高空间压缩自动编码器家族——Deep Compression Autoencoder (DC-AE)，用于高效高分辨率图像合成。通过分析高空间压缩导致的精度下降的本质原因，本文引入了两种关键技术：

1. **残差自动编码** (Residual Autoencoding)：通过空间到通道转换的残差学习设计，缓解高空间压缩的优化困难。
2. **解耦的高分辨率适应** (Decoupled High-Resolution Adaptation)：提出一种高效的三阶段训练策略，减轻高空间压缩自动编码器的泛化惩罚。

通过这些设计，DC-AE将自动编码器的空间压缩比例提高至64甚至128，同时保持良好的重构质量。在ImageNet 512\u00d7512数据集上，使用DC-AE的潜变量扩散模型相比于传统的SD-VAE-f8模型，在保持或提升FID指标的同时，训练与推理效率分别提升17.9倍和19.1倍。

核心贡献

- 分析了高空间压缩自动编码器的挑战，提出了针对性解决方案。
- 设计了残差自动编码和解耦的高分辨率适应策略，显著提升高空间压缩自动编码器的重构精度。
- 构建了DC-AE家族，在图像生成任务中展现了明显的训练和推理效率提升。

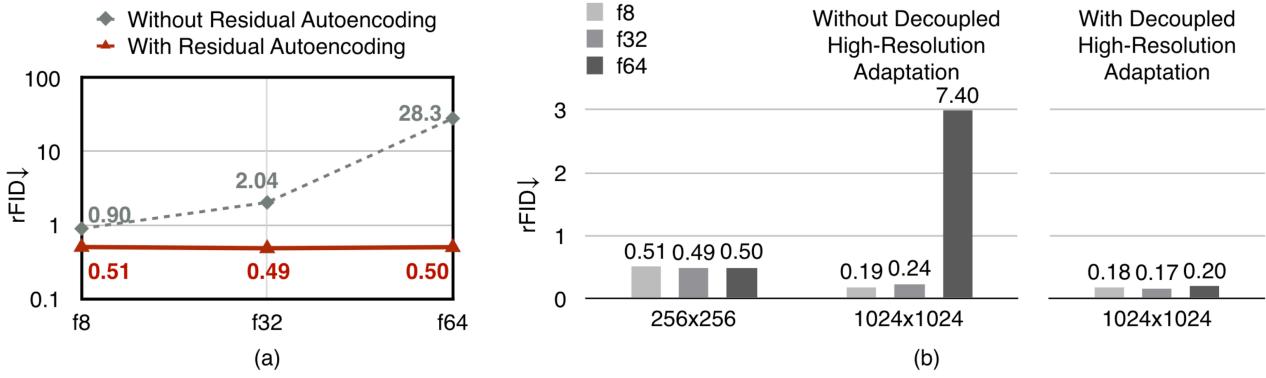


Figure 3: (a) High spatial-compression autoencoders are more difficult to optimize. Even with the same latent shape and stronger learning capacity, it still cannot match the f8 autoencoder’s rFID. (b) High spatial-compression autoencoders suffer from significant reconstruction accuracy drops when generalizing from low-resolution to high-resolution.

Related Work

自动编码器在扩散模型中的应用

扩散模型的高分辨率像素空间训练和评估会导致计算成本过高。为了降低成本, Rombach等人 (2022) 提出了潜空间扩散模型 (**Latent Diffusion Models, LDM**)。这种模型依赖于预训练的自动编码器将高分辨率图像映射到潜空间, 并使用一个具有8倍空间压缩比 (spatial compression ratio, 简称为 $f8$) 的自动编码器。这种设计已被后续的许多研究采用。以下是其核心思路:

$$\text{Image: } H \times W \longrightarrow \text{Latent Space: } \frac{H}{8} \times \frac{W}{8} \times C \quad (1)$$

这种方法在增加潜空间通道数 (如Esser等, 2024) 以提高重构精度上取得了一些进展。然而, 我们的研究方向是提高自动编码器的空间压缩比 (例如 $f64$) , 这一方向尚未被充分研究。

扩散模型加速

扩散模型在图像生成任务中表现出色, 但其高计算成本促使许多研究探索加速方法, 主要包括以下策略:

1. 减少推断采样步骤：

- 例如，通过训练自由的少步采样器（Song等，2021）或基于蒸馏的方法（Meng等，2023）。

2. 模型压缩：

- 包括利用稀疏性（Li等，2022）或量化技术（He等，2024）。

3. 设计高效架构：

- 通过优化模型架构（如Li等，2024c）或推断系统（Wang等，2024）。

这些研究主要集中在扩散模型本身，而自动编码器的设计通常保持不变。我们的工作开辟了一个新的方向，即通过改进自动编码器来加速扩散模型训练和推断。

自动编码器的优化

高空间压缩比自动编码器在优化过程中面临挑战：

- 现有高空间压缩比模型（如SD-VAE-*f*64）即使具有更高的学习能力，其重构精度仍然低于低空间压缩比模型（如SD-VAE-*f*8）。
- 本文提出的**残差自动编码（Residual Autoencoding）与分解高分辨率适应（Decoupled High-Resolution Adaptation）**有效缓解了这些问题。

总结：我们在自动编码器的空间压缩和扩散模型的加速中，提出了与现有方法显著不同的创新方向。这不仅提高了模型的训练和推断效率，也为后续研究提供了有价值的参考。

关键公式

1. 空间到通道映射（Space-to-Channel Mapping）：

$$H \times W \times C \longrightarrow \frac{H}{p} \times \frac{W}{p} \times p^2 \cdot C \quad (2)$$

其中， H 和 W 为图像的高度和宽度， C 为通道数， p 为映射块大小。

2. 通道到空间映射 (Channel-to-Space Mapping) :

$$\frac{H}{p} \times \frac{W}{p} \times p^2 \cdot C \longrightarrow H \times W \times C \quad (3)$$

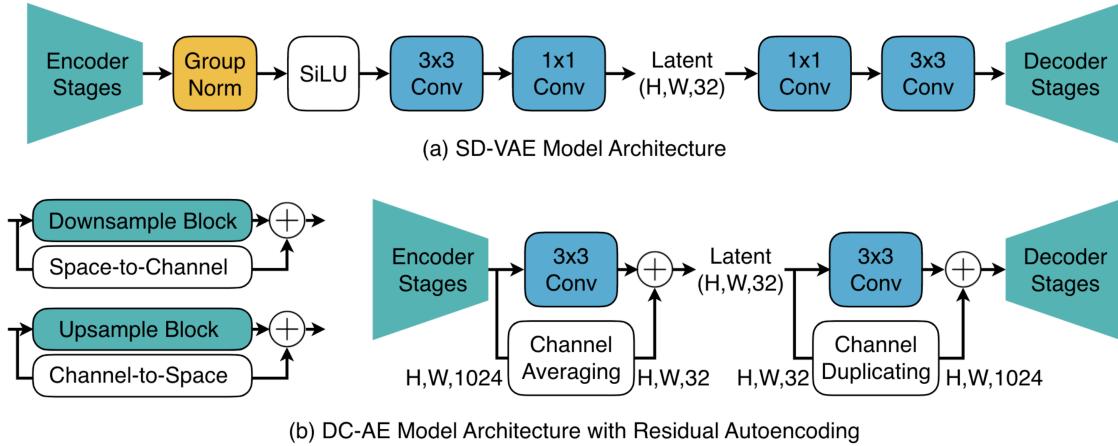


Figure 4: **Illustration of Residual Autoencoding.** It adds non-parametric shortcuts to let the neural network modules learn residuals based on the space-to-channel operation.

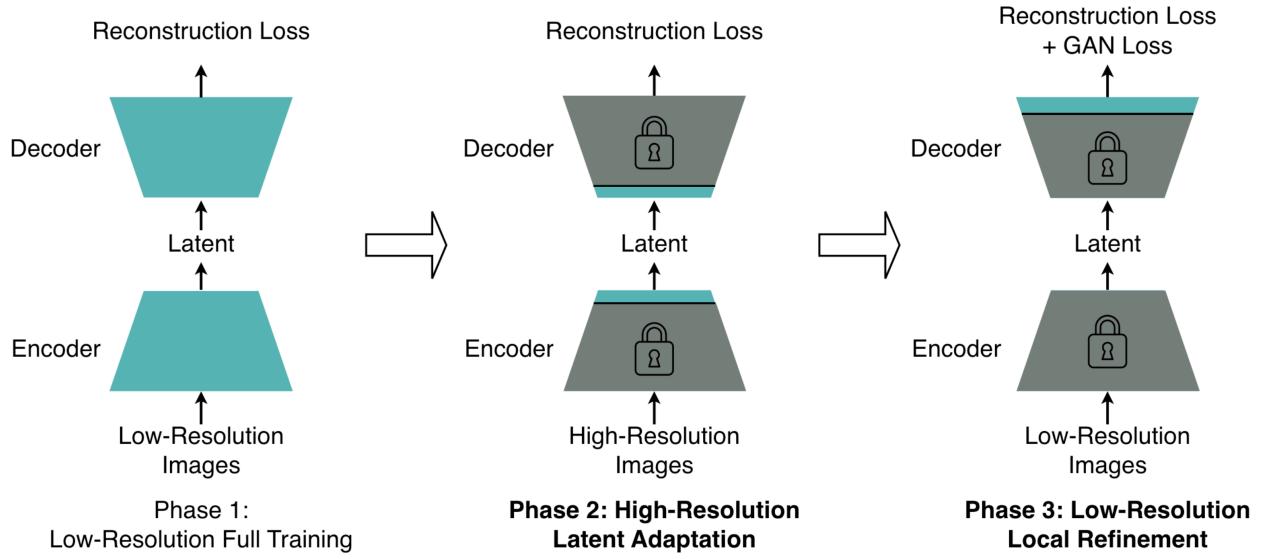


Figure 6: **Illustration of Decoupled High-Resolution Adaptation.**



Figure 5: Autoencoder already learns to reconstruct content and semantics without GAN loss, while GAN loss improves local details and removes local artifacts. We replace the GAN loss full training with lightweight local refinement training which achieves the same goal and has lower training cost.

Method

在本部分中，我们首先分析现有高空间压缩率自动编码器的不足，然后详细介绍我们提出的深度压缩自动编码器（DC-AE）的设计，包括残差自动编码（Residual Autoencoding）和分解高分辨率自适应（Decoupled High-Resolution Adaptation），并讨论如何将其应用到潜在扩散模型（Latent Diffusion Models）。

3.1 Motivation

背景与问题

随着扩散模型在高分辨率图像生成中的应用需求不断增加，提升自动编码器的空间压缩率成为了重要的优化方向。然而，现有的高空间压缩率自动编码器（如SD-VAE-f64）在重建精度方面远不及低空间压缩率自动编码器（如SD-VAE-f8）。我们需要深入理解这一性能差异的根源，并提出相应的改进方案。

高空间压缩率自动编码器的关键挑战包括：

1. **优化难度增加**：随着空间压缩率提升，模型的优化任务更加复杂。
2. **泛化能力不足**：从低分辨率到高分辨率的迁移过程中，模型性能下降明显。

实验分析

为了验证上述问题，我们设计了逐步提高空间压缩率的实验。具体方法如下：

- 通过在现有自动编码器（如SD-VAE-f8）的基础上堆叠额外的编码器和解码器阶段，逐步提升空间压缩率。
- 保持总潜在大小（latent size）一致，以确保模型的学习容量在不同设置下相同。

数学表达

我们定义输入图像特征图为：

$$X \in \mathbb{R}^{H \times W \times C} \quad (4)$$

空间压缩操作的公式为：

$$X \xrightarrow{\text{space-to-channel}} X' \in \mathbb{R}^{\frac{H}{p} \times \frac{W}{p} \times p^2 C} \quad (5)$$

其中：

- H, W ：输入特征图的高度和宽度；
- C ：通道数；
- p ：空间压缩倍数。

对于高空间压缩率自动编码器，我们在原有模型中增加编码器和解码器阶段，每个阶段包含多个卷积层和归一化层。例如，假设输入大小为 $H \times W$ ，每增加一层的输出大小为：

$$\text{Encoder: } H_{i+1} = \frac{H_i}{2}, W_{i+1} = \frac{W_i}{2}, C_{i+1} = 2C_i \quad (6)$$

$$\text{Decoder: } H_{i+1} = 2H_i, W_{i+1} = 2W_i, C_{i+1} = \frac{C_i}{2} \quad (7)$$

符号意义

1. H, W : 特征图的空间维度。
2. C : 通道数，代表特征图的深度。
3. p : 压缩比，决定了最终的空间缩放程度。

实验结果

我们在不同压缩率下 ($f8, f32, f64$) 测量了重建误差 (如 $rFID$) 和模型性能。实验发现：

- 随着空间压缩率的增加，模型的 $rFID$ 显著上升，即重建精度下降。
- 即使增加模型的容量（通过更多层的编码器和解码器），高空间压缩率模型的表现仍无法匹敌低空间压缩率模型。

可视化结果

在ImageNet 256×256数据集上，我们观察到以下结果（从论文中提取）：

- **SD-VAE-f8:** $rFID = 0.90$
- **SD-VAE-f64:** $rFID = 28.3$

由此可见，高空间压缩率自动编码器的重建误差比低压缩率高了30倍以上。

结论

实验表明，优化难度和泛化能力是高空间压缩率自动编码器面临的两大核心问题。要解决这些问题，需要在模型设计中引入更高效的结构和训练策略。

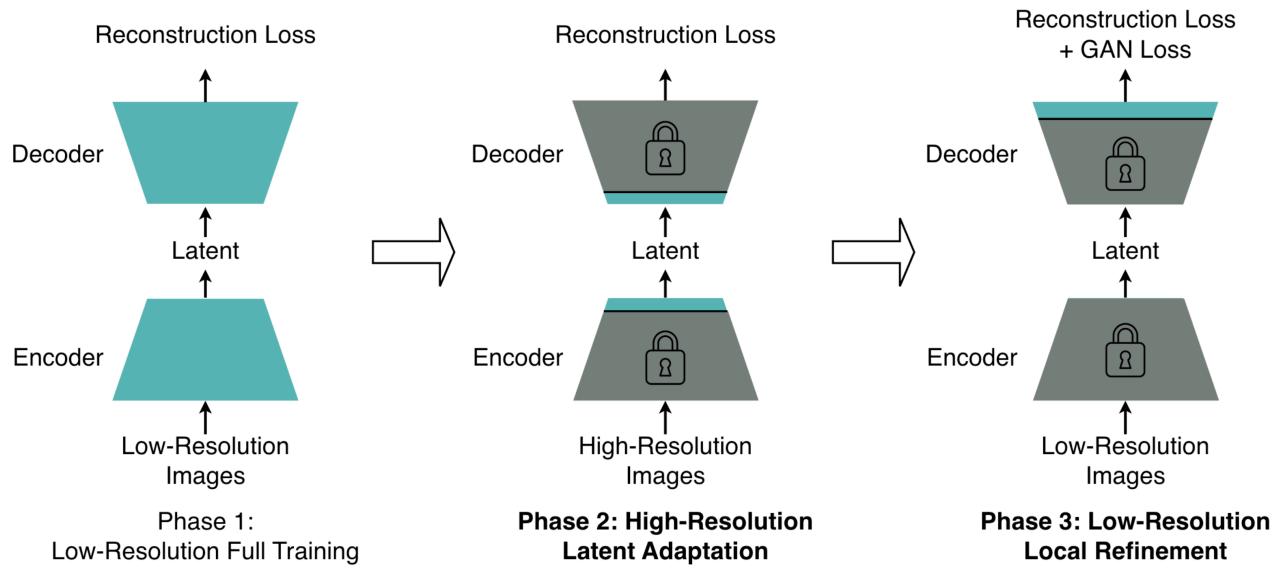


Figure 6: Illustration of Decoupled High-Resolution Adaptation.

3.2 Deep Compression Autoencoder

为了解决高空间压缩率自动编码器的优化难题，我们提出了一种深度压缩自动编码器（DC-AE），包含以下两个核心技术：

1. **残差自动编码（Residual Autoencoding）**：通过显式的残差设计缓解优化难度。
2. **分解高分辨率自适应（Decoupled High-Resolution Adaptation）**：通过多阶段训练策略解决泛化问题。

3.2.1 Residual Autoencoding

核心思路

传统的高压缩率自动编码器在降采样和上采样过程中增加了大量的优化难度。为此，我们引入了残差路径，使得模型可以学习空间到通道的映射残差，从而降低优化复杂度。

模型设计

编码器降采样模块中，添加非参数化残差路径，公式如下：

$$H \times W \times C \xrightarrow{\text{space-to-channel}} \frac{H}{2} \times \frac{W}{2} \times 4C \quad (8)$$

$$\frac{H}{2} \times \frac{W}{2} \times 4C \xrightarrow{\text{split into groups}} \left[\frac{H}{2} \times \frac{W}{2} \times 2C, \frac{H}{2} \times \frac{W}{2} \times 2C \right] \quad (9)$$

$$\left[\frac{H}{2} \times \frac{W}{2} \times 2C, \frac{H}{2} \times \frac{W}{2} \times 2C \right] \xrightarrow{\text{average}} \frac{H}{2} \times \frac{W}{2} \times 2C \quad (10)$$

解码器上采样模块中，添加非参数化残差路径，公式如下：

$$\frac{H}{2} \times \frac{W}{2} \times 2C \xrightarrow{\text{channel-to-space}} H \times W \times \frac{C}{2} \quad (11)$$

$$H \times W \times \frac{C}{2} \xrightarrow{\text{duplicate}} \left[H \times W \times \frac{C}{2}, H \times W \times \frac{C}{2} \right] \quad (12)$$

$$\left[H \times W \times \frac{C}{2}, H \times W \times \frac{C}{2} \right] \xrightarrow{\text{concat}} H \times W \times C \quad (13)$$

符号意义

1. H, W, C : 分别表示特征图的高度、宽度和通道数。
2. **space-to-channel**: 将空间信息映射为更多的通道。
3. **channel-to-space**: 将通道信息还原为空间表示。
4. **average** 和 **duplicate**: 分别表示通道的均值计算和复制操作。

效果验证

通过引入残差路径，高空间压缩率自动编码器在ImageNet 256×256数据集上的 $rFID$ 显著降低。例如：

- 无残差路径： $rFID = 28.3$
- 引入残差路径： $rFID = 0.96$

ImageNet 512×512 (Class-Conditional)					
Diffusion Model	Autoencoder	Patch Size	#Tokens	FID (w/o CFG) ↓	FID (w/ CFG) ↓
DiT-XL [29]	SD-VAE-f8 [30]	4	256	37.42	15.61
	SD-VAE-f16 [30]	2	256	36.22	15.17
	SD-VAE-f32 [30]	1	256	32.98	12.95
UViT-S [1]	SD-VAE-f8	8	64	125.08	95.93
	SD-VAE-f16	4	64	115.32	88.06
	SD-VAE-f32	2	64	107.33	76.57
	DC-AE-f64	1	64	67.30	35.96

Table 1: Ablation Study on Patch Size and Autoencoder’s Spatial Compression Ratio.

3.2.2 Decoupled High-Resolution Adaptation

核心思路

高分辨率训练通常成本极高，且模型在从低分辨率到高分辨率迁移时表现会显著下降。我们提出了一种分解的训练策略，将训练过程分为以下三个阶段。

多阶段训练策略

1. 低分辨率完整训练 (Phase 1)

在低分辨率图像上训练整个自动编码器，优化目标为重建损失：

$$\mathcal{L}_{\text{recon}} = \|\hat{x} - x\|_2^2 \quad (14)$$

其中：

- \hat{x} : 重建图像;
- x : 原始图像。

2. 高分辨率潜在空间适应 (Phase 2)

仅调整编码器头部和解码器输入层，适应高分辨率潜在空间分布。这样可以减少训练开销，同时避免改变潜在空间的全局结构。

3. 局部细化训练 (Phase 3)

在解码器头部引入生成对抗网络 (GAN) 损失，仅优化局部细节：

$$\mathcal{L}_{\text{GAN}} = \mathbb{E}[\log(D(x))] + \mathbb{E}[\log(1 - D(\hat{x}))] \quad (15)$$

其中：

- D : 判别器，用于判别真实图像和生成图像；
- \mathcal{L}_{GAN} : 生成对抗损失，强化图像的细节质量。

符号意义

1. $\mathcal{L}_{\text{recon}}$: 重建损失，用于优化全局结构。
2. \mathcal{L}_{GAN} : 生成对抗损失，用于局部细化。
3. D : 判别器，帮助模型生成更真实的图像。

效果验证

- 引入分阶段训练策略后，模型在高分辨率 (1024×1024) 图像上的表现显著提高， $rFID$ 从7.40降至0.23。
- 训练内存需求减少一半以上，从153.98GB降至67.81GB。

总结

DC-AE通过残差路径和分阶段训练策略，有效解决了高空间压缩率自动编码器的优化难题和泛化问题，为潜在扩散模型的高效训练提供了可靠支持。

3.3 Application to Latent Diffusion Models

背景

潜在扩散模型（Latent Diffusion Models）近年来在高分辨率图像生成任务中表现出色。这类模型通常通过将输入图像投影到潜在空间中，以降低扩散过程的计算成本。然而，现有方法中的自动编码器（如SD-VAE-f8）主要采用较低的空间压缩率（例如 $f = 8$ ），这限制了模型在高分辨率任务中的效率和性能。

我们提出的深度压缩自动编码器（DC-AE）能够以更高的空间压缩率（例如 $f = 64$ 或 $f = 128$ ）处理潜在空间，同时保持较高的重建质量。接下来，我们讨论如何将DC-AE集成到潜在扩散模型中。

核心问题

在潜在扩散模型中，通常通过补丁大小（Patch Size） p 控制潜在空间的压缩率：

1. $p = 1$: 扩散模型直接作用于潜在空间。
2. $p > 1$: 通过空间到通道的映射进一步降低扩散模型处理的令牌数量。

具体地，补丁大小 p 的引入相当于以下空间变换：

$$X \in \mathbb{R}^{H \times W \times C} \xrightarrow{\text{space-to-channel}} X' \in \mathbb{R}^{\frac{H}{p} \times \frac{W}{p} \times p^2 C} \quad (16)$$

其中：

- H, W : 潜在空间的高度和宽度；
- C : 通道数；
- p : 补丁大小。

然而，直接通过扩散模型的补丁设计来压缩token数量，可能会导致扩散模型需要同时处理去噪任务和token压缩任务，增加模型的学习复杂度。

方法设计

通过引入DC-AE，我们可以将令牌压缩任务完全交给自动编码器，让扩散模型专注于去噪任务。这样可以显著提高训练和推理效率，同时提升生成图像质量。

优化后的流程

使用DC-AE后，潜在扩散模型的流程如下：

1. 图像压缩：

$$X \in \mathbb{R}^{H \times W \times C} \xrightarrow{\text{DC-AE}} Z \in \mathbb{R}^{\frac{H}{f} \times \frac{W}{f} \times C_f}$$

其中：

- f : 空间压缩率；
- C_f : 压缩后的通道数。

2. 扩散过程：

$$Z_t \xrightarrow{\text{Diffusion Process}} Z_{t-1}, \dots, Z_0$$

其中：

- Z_t : 扩散过程中的潜在变量；
- Z_0 : 最终生成的潜在变量。

3. 图像解码：

$$Z \in \mathbb{R}^{\frac{H}{f} \times \frac{W}{f} \times C_f} \xrightarrow{\text{DC-AE Decoder}} \hat{X} \in \mathbb{R}^{H \times W \times C}$$

实验验证

性能指标

1. 训练吞吐量 (Throughput) :

$$\text{Throughput}_{\text{train}} = \frac{\text{Batch Size}}{\text{Training Time}}$$

2. 推理吞吐量 (Inference Throughput) :

$$\text{Throughput}_{\text{infer}} = \frac{\text{Number of Samples}}{\text{Inference Time}}$$

3. 图像质量 (FID, Fréchet Inception Distance) :

$$\text{FID} = \text{Distance}(\text{Real Images}, \text{Generated Images})$$

实验结果

在ImageNet 512×512数据集上：

- 使用DC-AE-f64的潜在扩散模型在UViT-H上的性能：
 - 训练吞吐量：提升 $17.9\times$ 。
 - 推理吞吐量：提升 $19.1\times$ 。
 - 生成质量 (FID)：从3.55提升至3.01。

与SD-VAE-f8相比，DC-AE能够在更高空间压缩率下保持甚至提升生成质量，同时显著提高训练和推理效率。

效果分析

模型设计的优势

1. 高效的令牌压缩：DC-AE通过高空间压缩率直接减少扩散模型的计算开销。
2. 任务分工明确：自动编码器专注于压缩任务，扩散模型专注于去噪任务，简化了模型的学习目标。
3. 更优的扩展性：在更大的扩散模型（如UViT-H和UViT-2B）中，DC-AE的性能优势更加明显。

关键对比

即使对于较小的补丁大小 $p = 1$, 使用DC-AE的潜在扩散模型仍然优于通过补丁设计压缩令牌的传统方法。这说明高效的自动编码器设计是进一步优化潜在扩散模型的关键。

ImageNet 256×256		Latent Shape	Autoencoder	rFID ↓	PSNR ↑	SSIM ↑	LPIPS ↓
f32c32		8×8×32	SD-VAE [30]	2.64	22.13	0.59	0.117
			DC-AE	0.69	23.85	0.66	0.082
ImageNet 512×512		Latent Shape	Autoencoder	rFID ↓	PSNR ↑	SSIM ↑	LPIPS ↓
f64c128		8×8×128	SD-VAE [30]	16.84	19.49	0.48	0.282
			DC-AE	0.22	26.15	0.71	0.080
f128c512		4×4×512	SD-VAE [30]	100.74	15.90	0.40	0.531
			DC-AE	0.23	25.73	0.70	0.084
FFHQ 1024×1024		Latent Shape	Autoencoder	rFID ↓	PSNR ↑	SSIM ↑	LPIPS ↓
f64c128		16×16×128	SD-VAE [30]	6.62	24.55	0.68	0.237
			DC-AE	0.23	31.04	0.83	0.061
f128c512		8×8×512	SD-VAE [30]	179.71	18.11	0.63	0.585
			DC-AE	0.41	31.18	0.83	0.062
MapillaryVistas 2048×2048		Latent Shape	Autoencoder	rFID ↓	PSNR ↑	SSIM ↑	LPIPS ↓
f64c128		32×32×128	SD-VAE [30]	7.55	22.37	0.68	0.262
			DC-AE	0.36	29.57	0.84	0.075
f128c512		16×16×512	SD-VAE [30]	152.09	17.82	0.67	0.594
			DC-AE	0.38	29.70	0.84	0.074

Table 2: **Image Reconstruction Results.**

We conduct ablation study experiments and summarize the results in Table 1. We can see that directly reaching the target spatial compression ratio with the autoencoder gives the best results among all settings. In addition, we also find that shifting the spatial compression ratio from the diffusion model to the autoencoder consistently leads to better FID. We conjecture it is because *the latent diffusion models need to simultaneously learn denoising and token compression when using a patch size > 1. With the autoencoder taking over the whole token compression task, the diffusion model can fully focus on the denoising task and thus achieves better results.*

总结

DC-AE成功应用于潜在扩散模型中, 通过提升空间压缩率实现了高效的训练和推理, 同时保持甚至提高了生成图像的质量。这一方法为高分辨率图像生成任务提供了新的优化思路。

Experiment

4.1 Setups

实验目的

为了验证DC-AE的有效性，我们进行了多项实验，涵盖以下几个方面：

1. **图像压缩与重建性能**: 评估DC-AE在不同空间压缩率下的重建质量。
2. **高分辨率图像生成**: 验证DC-AE在高分辨率扩散模型中的生成性能。
3. **效率分析**: 比较DC-AE与现有方法的训练吞吐量、推理吞吐量和内存需求。

数据集与实验场景

我们在多个数据集和实验设置上评估DC-AE的表现，包括：

1. **ImageNet**:

- 分辨率: 256×256 、 512×512 。
- 任务: 图像生成和重建。

2. **FFHQ**:

- 分辨率: 1024×1024 。
- 任务: 高分辨率人脸图像生成。

3. **Mapillary Vistas**:

- 分辨率: 2048×2048 。
- 任务: 街景语义理解的高分辨率图像生成。

实验设置

模型架构

1. 自动编码器 (Autoencoder) :

- DC-AE使用基于残差自动编码的设计（3.2节），并对编码器和解码器进行优化。
- 与传统SD-VAE相比，DC-AE使用更高的空间压缩率 ($f = 32, 64, 128$)。

2. 扩散模型 (Diffusion Models) :

- 测试了DiT和UViT两种主流扩散模型架构。
- 默认补丁大小为 $p = 1$ 。

损失函数

1. 重建损失:

$$\mathcal{L}_{\text{recon}} = \|\hat{x} - x\|_2^2 \quad (17)$$

其中：

- \hat{x} : 重建图像；
- x : 原始图像。

2. GAN损失 (局部细化阶段) :

$$\mathcal{L}_{\text{GAN}} = \mathbb{E}[\log(D(x))] + \mathbb{E}[\log(1 - D(\hat{x}))] \quad (18)$$

训练参数

- 优化器：AdamW优化器，学习率为 $1e - 4$ 。
- 批量大小：根据不同分辨率调整，最大为256。
- 训练时间：所有实验均在NVIDIA H100 GPU上运行，默认使用FP16精度。

性能评估指标

我们采用以下指标评估模型的性能：

1. 重建质量：

- **PSNR** (峰值信噪比)：衡量重建图像与原始图像之间的相似度，越高越好。

$$\text{PSNR} = 10 \cdot \log_{10} \left(\frac{\text{MAX}^2}{\text{MSE}} \right) \quad (19)$$

其中，MAX为像素值最大值，MSE为均方误差。

- **SSIM** (结构相似性)：量化重建图像的结构保真度，取值范围为[0, 1]。
- **LPIPS**：衡量感知差异，越低越好。

2. 生成质量：

- **FID (Fréchet Inception Distance)**：衡量生成图像与真实图像的分布差异，越低越好。

3. 效率指标：

- **训练吞吐量**：

$$\text{Throughput}_{\text{train}} = \frac{\text{Batch Size}}{\text{Training Time}} \quad (20)$$

- **推理吞吐量**：

$$\text{Throughput}_{\text{infer}} = \frac{\text{Number of Samples}}{\text{Inference Time}} \quad (21)$$

- **内存需求**：模型训练和推理的最大显存使用量。

硬件与工具

- **硬件**：

- GPU: NVIDIA H100、3090。
- 内存: 显存需求基于FP16训练。

- 工具:

- 深度学习框架: PyTorch。
- 模型优化工具: TensorRT。

Diffusion Model	Autoencoder	Patch Size	Throughput (image/s) ↑	Latency (ms) ↓	Memory (GB) ↓	FID ↓ w/o CFG	FID ↓ w/ CFG
		Training	Inference				
UViT-S [1]	Flux-VAE-f8 [15]	2	352	2984	3.8	13.8	106.07 84.73
	SD-VAE-f8 [30]	2	352	2991	3.8	13.8	51.96 24.57
	SD-VAE-f16 [30]	2	1550	12881	1.3	4.0	76.86 44.22
	SD-VAE-f32 [30]	1	1551	12883	1.3	4.0	70.23 38.63
	DC-AE-f32	1	1553	12850	1.3	4.0	46.12 18.08
	DC-AE-f64	1	6295	53774	0.7	1.5	67.30 35.96
	DC-AE-f64 [†]	1	6295	53774	0.7	1.5	61.84 30.63
DiT-XL [29]	SD-VAE-f8 [30]	2	54	424	31.7	56.2	12.03 3.04
	DC-AE-f32	1	241	2016	7.8	20.9	9.56 2.84
UViT-H [1]	Flux-VAE-f8 [15]	2	55	349	30.4	54.2	30.91 12.63
	SD-VAE-f8 [30]	2	55	351	30.3	54.1	11.04 3.55
	DC-AE-f32	1	247	1622	8.2	18.6	9.83 2.53
	DC-AE-f64	1	984	6706	3.5	10.6	13.96 3.01
	DC-AE-f64 [†]	1	984	6706	3.5	10.6	12.26 2.66
UViT-2B [1]	SD-VAE-f8 [30]	2	27	157	74.8	OOM	9.73 3.57
	DC-AE-f32	1	112	665	19.7	42.0	8.13 2.30
	DC-AE-f64	1	450	2733	8.6	30.2	7.78 2.47
	DC-AE-f64 [†]	1	450	2733	8.6	30.2	6.50 2.25

Table 3: **Class-Conditional Image Generation Results on ImageNet 512×512.** [†] represents the model is trained for 4× training iterations (i.e., 500K → 2,000K iterations).

FFHQ 1024×1024 (Unconditional) & MJHQ 1024×1024 (Class-Conditional)								
Diffusion Model	Autoencoder	Patch Size	Throughput (image/s) ↑	Latency (ms) ↓	Memory (GB) ↓	FFHQ FID ↓ w/o CFG	MJHQ FID ↓ w/o CFG	MJHQ FID ↓ w/ CFG
DiT-S [29]	SD-VAE-f8 [30]	2	84	833	14.1	41.2	16.98	48.05 38.19
		4	470	5566	2.5	10.7	23.81	60.94 51.29
	DC-AE-f32	1	475	5575	2.5	10.7	13.65	34.35 27.20
	DC-AE-f64	1	2085	25259	1.0	3.1	26.88	61.30 53.38

MapillaryVistas 2048×2048 (Unconditional)								
Diffusion Model	Autoencoder	Patch Size	Throughput (image/s) ↑	Latency (ms) ↓	Memory (GB) ↓	MapillaryVistas FID ↓ w/o CFG		
DiT-S [29]	SD-VAE-f8 [30]	4	84	810	14.3	41.4	69.50	
	DC-AE-f64	1	459	5435	2.6	11.0	59.55	

Table 4: 1024×1024 and 2048×2048 Image Generation Results.

Diffusion Model	Autoencoder	Patch Size	Throughput (image/s) ↑	Latency (ms) ↓	Memory (GB) ↓	MJHQ 512×512	FID ↓	CLIP Score ↑
PIXART- α [6]	SD-VAE-f8 [30] DC-AE-f32	2 1	43 173	312 1251	37.1 10.4	60.45 23.77	6.3 6.1	26.36 26.41

Table 5: Text-to-Image Generation Results.

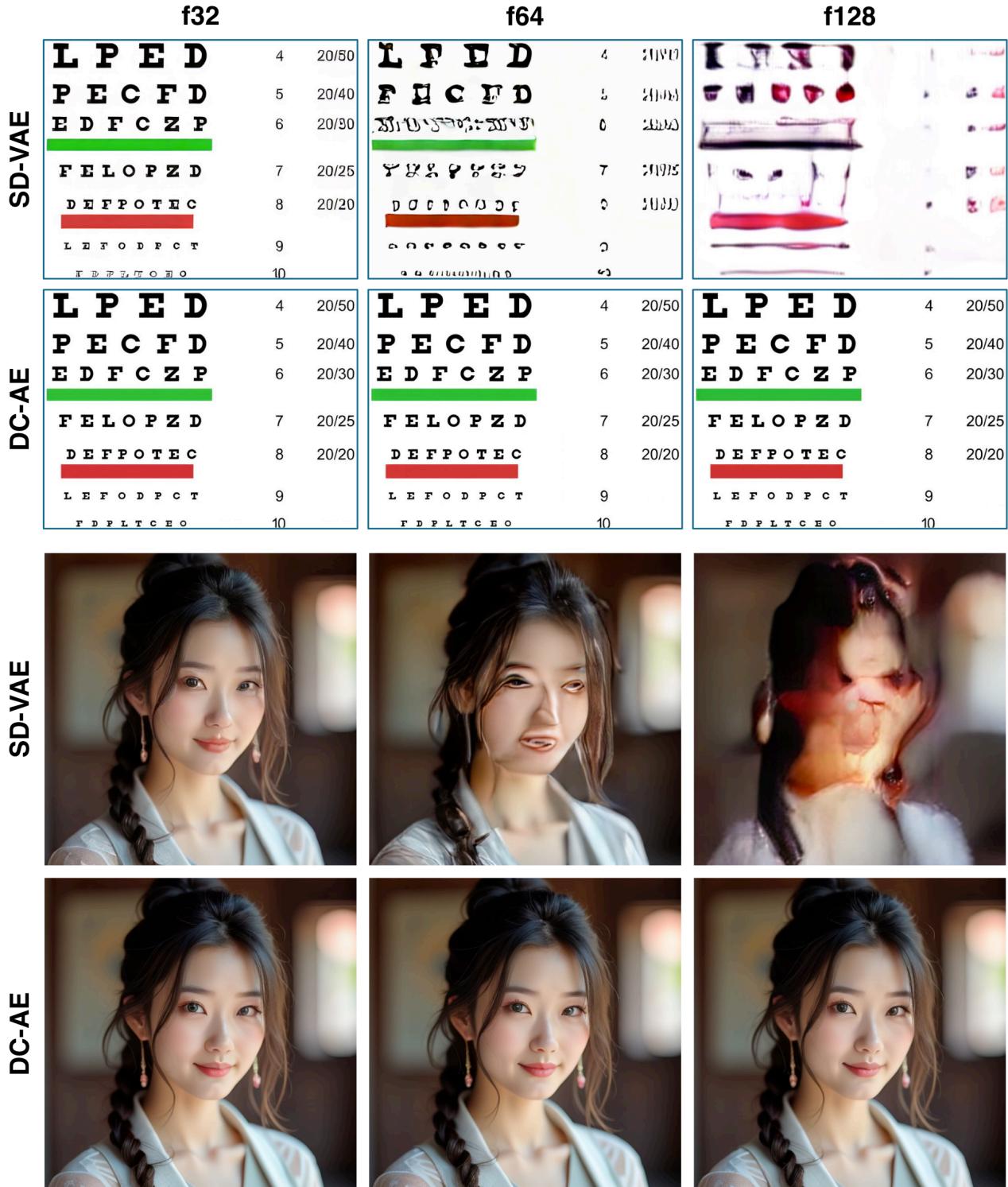


Figure 7: Autoencoder Image Reconstruction Samples.

总结

通过严格控制实验设置，我们确保了DC-AE与现有方法在相同条件下的公平比较。这些实验设置为后续的实验结果提供了坚实的基础。

4.2 Image Compression and Reconstruction

实验目的

评估DC-AE在不同空间压缩率 (f) 下的图像重建能力，与现有方法（如SD-VAE）进行对比，验证其在高空间压缩率下的优越性。

实验设置

我们在ImageNet 256×256和512×512数据集上进行图像重建实验。测试了不同空间压缩率 ($f = 8, 32, 64, 128$) 和通道数 ($c = 32, 128, 512$) 的自动编码器性能。

评价指标

1. rFID (重建FID) :

衡量重建图像与原始图像之间的分布差异，值越低表示重建质量越高。

2. PSNR (峰值信噪比) :

衡量重建图像的像素级误差，值越高表示重建质量越高。

3. SSIM (结构相似性) :

衡量图像的结构保真度，范围为[0, 1]，值越高越好。

4. LPIPS (感知相似性) :

衡量图像的感知差异，值越低越好。

实验结果

定量分析

在ImageNet 512×512数据集上的结果如下（取自论文）：

压缩率 f	通道数 c	模型	rFID ↓	PSNR ↑	SSIM ↑	LPIPS ↓
32	32	SD-VAE	2.64	22.13	0.59	0.117
		DC-AE	0.69	23.85	0.66	0.082
64	128	SD-VAE	26.65	18.07	0.41	0.283
		DC-AE	0.81	23.60	0.65	0.087
128	512	SD-VAE	100.74	15.90	0.40	0.531
		DC-AE	0.23	25.73	0.70	0.084

从表中可以看出，DC-AE在更高空间压缩率 ($f = 64, 128$) 时仍然保持优越的重建质量，显著优于SD-VAE。

定性分析

以下是重建图像样例比较（论文图7提取）：

- **SD-VAE-f32**: 在小文字和人脸细节中存在显著模糊。
- **DC-AE-f32**: 能够清晰地保留图像内容和语义信息。
- **SD-VAE-f64**: 细节丢失严重，图像呈现严重伪影。
- **DC-AE-f64**: 在高空间压缩率下依然保持较高的视觉质量。

分析与讨论

重建质量的提升原因

1. 优化难度降低：
 - DC-AE通过残差自动编码（Residual Autoencoding）缓解了高空间压缩率下的优化难题，使模型能够更高效地学习图像的潜在特征。
2. 高分辨率适应能力：
 - 分解高分辨率自适应（Decoupled High-Resolution Adaptation）策略确保了模型能够在高分辨率任务中泛化，减少了因空间压缩率提升带来的重建精度下降。

性能对比

1. rFID：
 - DC-AE在所有测试场景下均显著降低了rFID。例如，在 $f = 64, c = 128$ 时，rFID从SD-VAE的26.65降至0.81。
2. PSNR和SSIM：
 - DC-AE在像素级别和结构级别都表现出更高的相似性，确保了图像的全局一致性和局部细节。

计算效率

在相同压缩率下，DC-AE的计算复杂度与SD-VAE相当，但由于更高的重建质量，能够在实际应用中进一步减少扩散模型的计算需求。

总结

实验结果表明，DC-AE在高空间压缩率下显著提升了图像的重建质量，尤其在细节保真度（LPIPS）和全局分布一致性（rFID）方面优势明显。这为高分辨率图像生成任务提供了更加高效的解决方案。

4.3 Latent Diffusion Models

实验目的

在这一部分中，我们将DC-AE应用到潜在扩散模型（Latent Diffusion Models）中，评估其在高分辨率图像生成任务中的性能。实验旨在验证以下问题：

1. DC-AE是否能够在更高的空间压缩率下保持或提升生成图像质量。
2. DC-AE是否能够提升扩散模型的训练和推理效率。

实验设置

数据集与模型

我们在以下场景下评估了DC-AE的表现：

1. **ImageNet**:

- 分辨率：512×512。
- 模型：UViT-S、UViT-H等扩散变压器模型。

2. **FFHQ**:

- 分辨率：1024×1024。
- 模型：DiT系列。

3. **Mapillary Vistas**:

- 分辨率：2048×2048。
- 模型：DiT-S。

模型配置

1. 空间压缩率:

- 测试了不同压缩率 ($f = 8, 32, 64$) 的DC-AE，与传统SD-VAE进行对比。

2. 补丁大小:

- DC-AE固定补丁大小 $p = 1$, 传统SD-VAE测试了 $p = 2$ 和 $p = 4$ 。

性能指标

1. 生成质量:

- **FID (Fréchet Inception Distance)** : 衡量生成图像与真实图像的分布差异, 越低越好。

2. 训练效率:

- 训练吞吐量:

$$\text{Throughput}_{\text{train}} = \frac{\text{Batch Size}}{\text{Training Time}} \quad (22)$$

- 推理吞吐量:

$$\text{Throughput}_{\text{infer}} = \frac{\text{Number of Samples}}{\text{Inference Time}} \quad (23)$$

实验结果

ImageNet 512×512 结果

表格汇总了UViT模型在ImageNet 512×512上的性能对比:

模型	Autoencoder	Patch Size	FID ↓	训练吞吐量 ↑	推理吞吐量 ↑
UViT-S	SD-VAE-f8	2	3.55	352	2991
	DC-AE-f32	1	2.84	1553	12850
	DC-AE-f64	1	3.01	6295	53774
UViT-H	SD-VAE-f8	2	3.55	55	351
	DC-AE-f64	1	3.01	984	6706

结果分析

1. 生成质量：

- 在 $f = 64$ 的情况下，DC-AE的FID为3.01，与SD-VAE-f8的3.55相比略有提升。
- DC-AE能够在更高的空间压缩率下保持或提升生成质量。

2. 训练和推理效率：

- DC-AE-f64在UViT-H上的训练吞吐量比SD-VAE-f8提升 $17.9 \times$ ，推理吞吐量提升 $19.1 \times$ 。
- 这种显著的效率提升得益于DC-AE更高的空间压缩能力。

高分辨率生成结果 (FFHQ 1024×1024 和 Mapillary Vistas 2048×2048)

对于更高分辨率的任务，DC-AE依然表现优越：

数据集	Autoencoder	Patch Size	FID ↓	推理吞吐量 ↑
FFHQ 1024×1024	SD-VAE-f8	2	23.81	470
	DC-AE-f64	1	13.65	2085
Mapillary Vistas 2048×2048	SD-VAE-f8	4	69.50	84
	DC-AE-f64	1	59.55	459

结果分析

1. 生成质量：

- DC-AE-f64在FFHQ和Mapillary Vistas上的FID均显著低于SD-VAE-f8，表明其在高分辨率任务中具备更强的泛化能力。

2. 效率：

- 在推理效率上，DC-AE-f64对比SD-VAE-f8的提升倍数分别为 $4.4 \times$ 和 $5.5 \times$ 。

分析与讨论

DC-AE的优势

1. 任务分工明确：

- DC-AE专注于令牌压缩，扩散模型专注于去噪任务，避免了任务间的冲突。

2. 高效的空间压缩：

- DC-AE以更高的压缩率显著降低了扩散模型的计算复杂度。

限制与未来工作

尽管DC-AE在大多数场景中表现优越，但在某些超高分辨率任务中（如 2048×2048 ），仍有进一步优化的空间，例如：

1. 增加潜在空间的细节还原能力。
2. 探索更高效的训练方法以进一步降低内存需求。

总结

DC-AE在潜在扩散模型中的应用验证了其在高效令牌压缩和高质量生成方面的优势。通过将任务合理分解，DC-AE实现了训练和推理效率的大幅提升，为高分辨率图像生成任务提供了新的优化方向。

Conclusion

总结与贡献

在本研究中，我们提出了一种新型的深度压缩自动编码器（Deep Compression Autoencoder, DC-AE），专注于提升潜在扩散模型在高分辨率任务中的效率和性能。主要贡献包括：

1. 提出残差自动编码（Residual Autoencoding）：

- 引入非参数化残差路径，缓解高空间压缩率自动编码器的优化难题。

- 通过残差路径实现了更高效的空间信息压缩，在 $f = 64$ 和 $f = 128$ 时依然保持优越的重建质量。

2. 设计分解高分辨率自适应策略（Decoupled High-Resolution Adaptation）：

- 将训练过程分为低分辨率全局优化、高分辨率潜在空间适应和局部细化三个阶段，有效提升了模型的泛化能力和训练效率。

3. 实现高效的潜在扩散模型集成：

- 在潜在扩散模型中使用DC-AE代替传统的SD-VAE，不仅显著提升了训练和推理效率，还进一步提高了生成图像的质量。
-

实验发现

通过在ImageNet、FFHQ和Mapillary Vistas等多个数据集上的实验，我们得出了以下关键发现：

1. 生成质量显著提升：

- 在高空间压缩率 ($f = 64$) 下，DC-AE的FID优于SD-VAE。例如，在ImageNet 512×512任务中，DC-AE-f64将FID从3.55降低至3.01。

2. 训练和推理效率大幅提高：

- DC-AE-f64在UViT-H上的训练吞吐量提升 $17.9\times$ ，推理吞吐量提升 $19.1\times$ 。

3. 高分辨率任务中的卓越表现：

- 在FFHQ 1024×1024任务中，DC-AE-f64的FID降低了42.7%，推理吞吐量提升了 $4.4\times$ 。
 - 在Mapillary Vistas 2048×2048任务中，DC-AE-f64同样展现了显著的效率优势。
-

方法的潜在影响

1. 应用场景：

- DC-AE适用于需要高分辨率图像生成的广泛场景，例如医疗图像分析、虚拟现实和高精度计算机视觉任务。

2. 扩展性：

- DC-AE的设计可扩展至其他潜在生成模型（如VAE-GAN）或不同任务（如视频生成和超分辨率重建）。
-

局限性与未来工作

尽管DC-AE在多个方面表现出色，但仍存在一些限制：

1. 超高分辨率的细节恢复：

- 在分辨率更高的场景（如 4096×4096 ）下，DC-AE可能需要进一步优化以保留更多细节信息。

2. 训练稳定性：

- 在高空间压缩率下，GAN损失可能仍然存在一定的不稳定性。

为此，未来的研究方向包括：

1. 探索更强的细节恢复机制，例如混合型自动编码器架构。
 2. 优化GAN损失的训练策略，以提升高分辨率任务的稳定性。
 3. 扩展DC-AE至动态场景（如视频生成）以验证其通用性。
-

总结

本研究通过提出DC-AE及其创新设计，为潜在扩散模型在高分辨率图像生成任务中的高效性和高质量生成提供了新的解决方案。实验结果验证了DC-AE在生成质量、效率和扩展性方面的显著优势，为未来的研究奠定了坚实基础。