arXiv:2112.10752v2 [cs.CV] 13 Apr 2022

High-Resolution Image Synthesis with Latent Diffusion Models

Robin Rombach[1] * Andreas Blattmann[1] * Dominik Lorenz[1] Patrick Esser[R] Björn Ommer[1]
[1]Ludwig Maximilian University of Munich & IWR, Heidelberg University, Germany [R]Runway ML
https://github.com/CompVis/latent-diffusion

Abstract

By decomposing the image formation process into a sequential application of denoising autoencoders, diffusion models (DMs) achieve state-of-the-art synthesis results on image data and beyond. Additionally, their formulation allows for a guiding mechanism to control the image generation process without retraining. However, since these models typically operate directly in pixel space, optimization of powerful DMs often consumes hundreds of GPU days and inference is expensive due to sequential evaluations. To enable DM training on limited computational resources while retaining their quality and flexibility, we apply them in the latent space of powerful pretrained autoencoders. In contrast to previous work, training diffusion models on such a representation allows for the first time to reach a near-optimal point between complexity reduction and detail preservation, greatly boosting visual fidelity. By introducing cross-attention layers into the model architecture, we turn diffusion models into powerful and flexible generators for general conditioning inputs such as text or bounding boxes and high-resolution synthesis becomes possible in a convolutional manner. Our latent diffusion models (LDMs) achieve new state-of-the-art scores for image inpainting and class-conditional image synthesis and highly competitive performance on various tasks, including text-to-image synthesis, unconditional image generation and super-resolution, while significantly reducing computational requirements compared to pixel-based DMs.

1. Introduction

Image synthesis is one of the computer vision fields with the most spectacular recent development, but also among those with the greatest computational demands. Especially high-resolution synthesis of complex, natural scenes is presently dominated by scaling up likelihood-based models, potentially containing billions of parameters in autoregressive (AR) transformers [66,67]. In contrast, the promising results of GANs [3, 27, 40] have been revealed to be mostly confined to data with comparably limited variability as their adversarial learning procedure does not easily scale to modeling complex, multi-modal distributions. Recently, diffusion models [82], which are built from a hierarchy of denoising autoencoders, have shown to achieve impressive

Figure 1. Boosting the upper bound on achievable quality with less agressive downsampling. Since diffusion models offer excellent inductive biases for spatial data, we do not need the heavy spatial downsampling of related generative models in latent space, but can still greatly reduce the dimensionality of the data via suitable autoencoding models, see Sec. 3. Images are from the DIV2K [1] validation set, evaluated at 512^2 px. We denote the spatial downsampling factor by f. Reconstruction FIDs [29] and PSNR are calculated on ImageNet-val. [12]; see also Tab. 8.

results in image synthesis [30,85] and beyond [7,45,48,57], and define the state-of-the-art in class-conditional image synthesis [15,31] and super-resolution [72]. Moreover, even unconditional DMs can readily be applied to tasks such as inpainting and colorization [85] or stroke-based synthesis [53], in contrast to other types of generative models [19,46,69]. Being likelihood-based models, they do not exhibit mode-collapse and training instabilities as GANs and, by heavily exploiting parameter sharing, they can model highly complex distributions of natural images without involving billions of parameters as in AR models [67].

Democratizing High-Resolution Image Synthesis DMs belong to the class of likelihood-based models, whose mode-covering behavior makes them prone to spend excessive amounts of capacity (and thus compute resources) on modeling imperceptible details of the data [16, 73]. Although the reweighted variational objective [30] aims to address this by undersampling the initial denoising steps, DMs are still computationally demanding, since training and evaluating such a model requires repeated function evaluations (and gradient computations) in the high-dimensional space of RGB images. As an example, training the most powerful DMs often takes hundreds of GPU days (*e.g.* 150 - 1000 V100 days in [15]) and repeated evaluations on a noisy version of the input space render also inference expensive,

*The first two authors contributed equally to this work.

so that producing 50k samples takes approximately 5 days [15] on a single A100 GPU. This has two consequences for the research community and users in general: Firstly, training such a model requires massive computational resources only available to a small fraction of the field, and leaves a huge carbon footprint [65, 86]. Secondly, evaluating an already trained model is also expensive in time and memory, since the same model architecture must run sequentially for a large number of steps (*e.g.* 25 - 1000 steps in [15]).

To increase the accessibility of this powerful model class and at the same time reduce its significant resource consumption, a method is needed that reduces the computational complexity for both training and sampling. Reducing the computational demands of DMs without impairing their performance is, therefore, key to enhance their accessibility.

Departure to Latent Space Our approach starts with the analysis of already trained diffusion models in pixel space: Fig. 2 shows the rate-distortion trade-off of a trained model. As with any likelihood-based model, learning can be roughly divided into two stages: First is a *perceptual compression* stage which removes high-frequency details but still learns little semantic variation. In the second stage, the actual generative model learns the semantic and conceptual composition of the data (*semantic compression*). We thus aim to first find a *perceptually equivalent, but computationally more suitable space*, in which we will train diffusion models for high-resolution image synthesis.

Following common practice [11, 23, 66, 67, 96], we separate training into two distinct phases: First, we train an autoencoder which provides a lower-dimensional (and thereby efficient) representational space which is perceptually equivalent to the data space. Importantly, and in contrast to previous work [23, 66], we do not need to rely on excessive spatial compression, as we train DMs in the learned latent space, which exhibits better scaling properties with respect to the spatial dimensionality. The reduced complexity also provides efficient image generation from the latent space with a single network pass. We dub the resulting model class *Latent Diffusion Models* (LDMs).

A notable advantage of this approach is that we need to train the universal autoencoding stage only once and can therefore reuse it for multiple DM trainings or to explore possibly completely different tasks [81]. This enables efficient exploration of a large number of diffusion models for various image-to-image and text-to-image tasks. For the latter, we design an architecture that connects transformers to the DM's UNet backbone [71] and enables arbitrary types of token-based conditioning mechanisms, see Sec. 3.3.

In sum, our work makes the following **contributions**: (i) In contrast to purely transformer-based approaches [23, 66], our method scales more graceful to higher dimensional data and can thus (a) work on a compression level which provides more faithful and detailed reconstructions than previous work (see Fig. 1) and (b) can be efficiently applied to high-resolution synthesis of megapixel images.

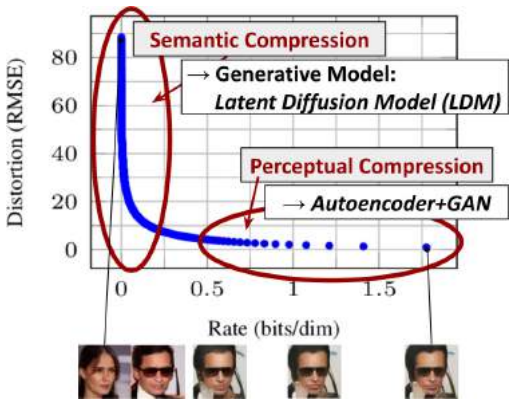

Figure 2. Illustrating perceptual and semantic compression: Most bits of a digital image correspond to imperceptible details. While DMs allow to suppress this semantically meaningless information by minimizing the responsible loss term, gradients (during training) and the neural network backbone (training and inference) still need to be evaluated on all pixels, leading to superfluous computations and unnecessarily expensive optimization and inference. We propose *latent diffusion models (LDMs)* as an effective generative model and a separate mild compression stage that only eliminates imperceptible details. Data and images from [30].

(ii) We achieve competitive performance on multiple tasks (unconditional image synthesis, inpainting, stochastic super-resolution) and datasets while significantly lowering computational costs. Compared to pixel-based diffusion approaches, we also significantly decrease inference costs.

(iii) We show that, in contrast to previous work [93] which learns both an encoder/decoder architecture and a score-based prior simultaneously, our approach does not require a delicate weighting of reconstruction and generative abilities. This ensures extremely faithful reconstructions and requires very little regularization of the latent space.

(iv) We find that for densely conditioned tasks such as super-resolution, inpainting and semantic synthesis, our model can be applied in a convolutional fashion and render large, consistent images of $\sim 1024^2$ px.

(v) Moreover, we design a general-purpose conditioning mechanism based on cross-attention, enabling multi-modal training. We use it to train class-conditional, text-to-image and layout-to-image models.

(vi) Finally, we release pretrained latent diffusion and autoencoding models at `https://github.com/CompVis/latent-diffusion` which might be reusable for a various tasks besides training of DMs [81].

2. Related Work

Generative Models for Image Synthesis The high dimensional nature of images presents distinct challenges to generative modeling. Generative Adversarial Networks (GAN) [27] allow for efficient sampling of high resolution images with good perceptual quality [3, 42], but are diffi-

cult to optimize [2, 28, 54] and struggle to capture the full data distribution [55]. In contrast, likelihood-based methods emphasize good density estimation which renders optimization more well-behaved. Variational autoencoders (VAE) [46] and flow-based models [18, 19] enable efficient synthesis of high resolution images [9, 44, 92], but sample quality is not on par with GANs. While autoregressive models (ARM) [6, 10, 94, 95] achieve strong performance in density estimation, computationally demanding architectures [97] and a sequential sampling process limit them to low resolution images. Because pixel based representations of images contain barely perceptible, high-frequency details [16, 73], maximum-likelihood training spends a disproportionate amount of capacity on modeling them, resulting in long training times. To scale to higher resolutions, several two-stage approaches [23, 67, 101, 103] use ARMs to model a compressed latent image space instead of raw pixels.

Recently, **Diffusion Probabilistic Models** (DM) [82], have achieved state-of-the-art results in density estimation [45] as well as in sample quality [15]. The generative power of these models stems from a natural fit to the inductive biases of image-like data when their underlying neural backbone is implemented as a UNet [15, 30, 71, 85]. The best synthesis quality is usually achieved when a reweighted objective [30] is used for training. In this case, the DM corresponds to a lossy compressor and allow to trade image quality for compression capabilities. Evaluating and optimizing these models in pixel space, however, has the downside of low inference speed and very high training costs. While the former can be partially adressed by advanced sampling strategies [47, 75, 84] and hierarchical approaches [31, 93], training on high-resolution image data always requires to calculate expensive gradients. We adress both drawbacks with our proposed *LDMs*, which work on a compressed latent space of lower dimensionality. This renders training computationally cheaper and speeds up inference with almost no reduction in synthesis quality (see Fig. 1).

Two-Stage Image Synthesis To mitigate the shortcomings of individual generative approaches, a lot of research [11, 23, 67, 70, 101, 103] has gone into combining the strengths of different methods into more efficient and performant models via a two stage approach. VQ-VAEs [67, 101] use autoregressive models to learn an expressive prior over a discretized latent space. [66] extend this approach to text-to-image generation by learning a joint distributation over discretized image and text representations. More generally, [70] uses conditionally invertible networks to provide a generic transfer between latent spaces of diverse domains. Different from VQ-VAEs, VQGANs [23, 103] employ a first stage with an adversarial and perceptual objective to scale autoregressive transformers to larger images. However, the high compression rates required for feasible ARM training, which introduces billions of trainable parameters [23, 66], limit the overall performance of such approaches and less compression comes at the price of high computational cost [23, 66]. Our work prevents such trade-offs, as our proposed *LDMs* scale more gently to higher dimensional latent spaces due to their convolutional backbone. Thus, we are free to choose the level of compression which optimally mediates between learning a powerful first stage, without leaving too much perceptual compression up to the generative diffusion model while guaranteeing high-fidelity reconstructions (see Fig. 1).

While approaches to jointly [93] or separately [80] learn an encoding/decoding model together with a score-based prior exist, the former still require a difficult weighting between reconstruction and generative capabilities [11] and are outperformed by our approach (Sec. 4), and the latter focus on highly structured images such as human faces.

3. Method

To lower the computational demands of training diffusion models towards high-resolution image synthesis, we observe that although diffusion models allow to ignore perceptually irrelevant details by undersampling the corresponding loss terms [30], they still require costly function evaluations in pixel space, which causes huge demands in computation time and energy resources.

We propose to circumvent this drawback by introducing an explicit separation of the compressive from the generative learning phase (see Fig. 2). To achieve this, we utilize an autoencoding model which learns a space that is perceptually equivalent to the image space, but offers significantly reduced computational complexity.

Such an approach offers several advantages: (i) By leaving the high-dimensional image space, we obtain DMs which are computationally much more efficient because sampling is performed on a low-dimensional space. (ii) We exploit the inductive bias of DMs inherited from their UNet architecture [71], which makes them particularly effective for data with spatial structure and therefore alleviates the need for aggressive, quality-reducing compression levels as required by previous approaches [23, 66]. (iii) Finally, we obtain general-purpose compression models whose latent space can be used to train multiple generative models and which can also be utilized for other downstream applications such as single-image CLIP-guided synthesis [25].

3.1. Perceptual Image Compression

Our perceptual compression model is based on previous work [23] and consists of an autoencoder trained by combination of a perceptual loss [106] and a patch-based [33] adversarial objective [20, 23, 103]. This ensures that the reconstructions are confined to the image manifold by enforcing local realism and avoids bluriness introduced by relying solely on pixel-space losses such as L_2 or L_1 objectives.

More precisely, given an image $x \in \mathbb{R}^{H\times W\times 3}$ in RGB space, the encoder $\mathcal{E}$ encodes x into a latent representa-

相关工作一上来就讲出GAN的优点和缺点，欲抑先扬，对抗神经网络虽然在高像素的图片上采样很高的效率，但是他很难被优化，很难捕捉到完整的数据分布。相比之下，基于似然的方法更注重对概率密度估计的良好表现，这使得优化过程更加稳定。传统的变分自动编码器 (VAE) [46] 和基于流的模型 [18, 19] 能够有效合成高分辨率图像，但样本质量无法与 GAN 相提并论。

虽然自回归模型 (ARM) [6,10,94,95] 在概率密度估计方面实现了强大的性能，但计算要求较高的架构 [97] 和顺序采样过程将它们限制在低分辨率图像上。由于基于像素的图像表示包含几乎不可察觉的高频细节[16,73]，因此最大似然训练在对它们进行建模时花费了不成比例的容量，从而导致训练时间较长。为了扩展到更高分辨率，几种两阶段方法 [23,67,101,103] 接下来说一下传统的扩散概率模型的弊端，当这些模型的底层神经主干被实现为 UNet 时，这些模型的生成能力源于对类图像数据的归纳偏差的自然拟合，当使用重新加权的目标 0]进行训练时，通常可以实现最佳的综合质量。在这种情况下，DM 相当于有损压缩器，并允许以图像质量换取压缩能力。然而，在像素空间中评估和优化这些模型具有推理速度低和训练成本非常高的缺点。虽然前者可以通过先进的采样策略 [47, 75, 84] 和分层方法 [31, 93] 部分解决，但高分辨率图像数据的训练总是需要计算昂贵的梯度。我们通过提出的 LDM 解决了这两个缺点。该 LDM

开始讲目前两阶段图像合成的一些做法和问题。为了减轻普通人因为较高的算力要求导致普通个体无法使用的缺点，大量研究 [11,23,67,70,101,103] 已经通过两阶段方法将不同方法的优点结合到更高效和性能更高的模型中，VQ-VAE [67, 101] 使用自回归模型来学习离散潜在空间的表达先验。[66]通过学习离散图像和文本表示的联合分布，将这种方法扩展到文本到图像的生成。更一般地，[70]使用条件可逆网络来提供

前面讲了说传统的自回归模型模型采用具有对抗性和感知目标的第一阶段，将自回归 transf
ormer缩放到更大的图像。然而可行的 ARM（自回归模型）
训练所需的高压缩率引入了数十亿个可训练参数 [23,
66]，限制了此类方法的整体性能，并且较少的压缩是以高计算成本为代价的，这里说出和之前工作的不同以及如何结局这个问题的。他们的工作作防止了这种权衡，因为我们提出的 LDM
由于其卷积主干可以更温和地扩展到更高维的潜在空间。因此，我们可以自由选择在学习

虽然存在联合[93]或单独[80]学习编码/解码模型以及基于分数的先验的方法，但前者仍然需要在重建和生成能力之间进行困难的权重[11]，并且我们的方法优于（sec

为了降低训练扩散模型对高分辨率图像合成的计算需求，我们观察到，尽管扩散模型允许通过对相应的损失项进行欠采样来忽略感知上不相关的细节[30]，但它们仍然需要在像素空间中进行昂贵的函数评估

我们决定通过引入压缩学习阶段和生成学习阶段的明确分离来规避这个缺点（见图 2）。为了实现这一目标，我们利用自动编码模型，该模型学习感知上与图像空间等效的空间，但显

这种方法有几个优点：(i) 通过离开高维图像空间，我们获得了计算效率更高的 DM，因为采样是在低维空间上执行的。(ii) 我们利用从 UNet 架构继承的 DM 的归纳偏差 [71]，这使得它们对于具有空间结构的数据特别有效，因此减轻了先前方法所需的激进的、降低质量的压缩级别的需求 [23，66]。(iii) 最后，我们获得通用压缩模型，其潜在空间可

这里提到的23、106、20、103是指先前的研究工作中使用的方法和技术。这些方法和技术结合起来，确保了通过强制局部真实性将重建限制在图像流形内，并避免了仅依赖像素空间损失（例如L2或L1目标）而引入的模糊。具体来说，这些方法和技

与 VQ-VAE 不同，VQGAN [23, 103] 采用具有对抗性和感知目标的第一阶段，将自回归 transformer缩放到更大的图像。然而可行的 ARM（自回归模型）训练所需的高压缩率引入了数十亿个可训练参数 [23, 66]，限制了此类方法的整体性能，并且较少的压缩是以高计算成本为代价的

-23：这个方法是指先前的工作中使用的一种自动编码器模型，通过组合感知损失和基于补丁的对抗目标来训练。这种方法可以确保重建图像在图像流形内，避免了模糊现象的出现。
106：这个方法是指先前的工作中使用的一种感知损失函数。感知损失函数基于神经网络的特征表示，可以更好地捕捉图像的感知质量，避免了仅依赖像素空间损失引入的模糊。20：这个方法是指先前的工作中使用的一种基于补丁的对抗目标。通过对图像的局部区域进行对抗训练，可以确保重建图像在局部区域内具有真实性，避免了模糊现象的出现。103：这个方法是指先前的工作中使用的一种基于对抗和感知目标的训练方法。通过同时优化对抗目标和感知目标，可以更好地保持重建图像的真实性和感知质量，避免了模糊现象的出现。
综合使用这些方法和技术，可以确保重建图像在图像流形内具有局部真实性，并避免了仅依赖像素空间损失引入的模糊现象。这样可以得到更清晰、更真实的重建图像。

tion $z = \mathcal{E}(x)$, and the decoder $\mathcal{D}$ reconstructs the image from the latent, giving $\tilde{x} = \mathcal{D}(z) = \mathcal{D}(\mathcal{E}(x))$, where $z \in \mathbb{R}^{h \times w \times c}$. Importantly, the encoder *downsamples* the image by a factor $f = H/h = W/w$, and we investigate different downsampling factors $f = 2^m$, with $m \in \mathbb{N}$.

In order to avoid arbitrarily high-variance latent spaces, we experiment with two different kinds of regularizations. The first variant, *KL-reg.*, imposes a slight KL-penalty towards a standard normal on the learned latent, similar to a VAE [46, 69], whereas *VQ-reg.* uses a vector quantization layer [96] within the decoder. This model can be interpreted as a VQGAN [23] but with the quantization layer absorbed by the decoder. Because our subsequent DM is designed to work with the two-dimensional structure of our learned latent space $z = \mathcal{E}(x)$, we can use relatively mild compression rates and achieve very good reconstructions. This is in contrast to previous works [23, 66], which relied on an arbitrary 1D ordering of the learned space z to model its distribution autoregressively and thereby ignored much of the inherent structure of z. Hence, our compression model preserves details of x better (see Tab. 8). The full objective and training details can be found in the supplement.

3.2. Latent Diffusion Models

Diffusion Models [82] are probabilistic models designed to learn a data distribution $p(x)$ by gradually denoising a normally distributed variable, which corresponds to learning the reverse process of a fixed Markov Chain of length T. For image synthesis, the most successful models [15, 30, 72] rely on a reweighted variant of the variational lower bound on $p(x)$, which mirrors denoising score-matching [85]. These models can be interpreted as an equally weighted sequence of denoising autoencoders $\epsilon_\theta(x_t, t)$; $t = 1 \dots T$, which are trained to predict a denoised variant of their input x_t, where x_t is a noisy version of the input x. The corresponding objective can be simplified to (Sec. B)

这个就是传统扩散模型训练过程中基于像素空间的目标函数

$$L_{DM} = \mathbb{E}_{x,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|_2^2\right], \qquad (1)$$

with t uniformly sampled from $\{1, \dots, T\}$.

Generative Modeling of Latent Representations With our trained perceptual compression models consisting of $\mathcal{E}$ and $\mathcal{D}$, we now have access to an efficient, low-dimensional latent space in which high-frequency, imperceptible details are abstracted away. Compared to the high-dimensional pixel space, this space is more suitable for likelihood-based generative models, as they can now (i) focus on the important, semantic bits of the data and (ii) train in a lower dimensional, computationally much more efficient space.

Unlike previous work that relied on autoregressive, attention-based transformer models in a highly compressed, discrete latent space [23, 66, 103], we can take advantage of image-specific inductive biases that our model offers. This

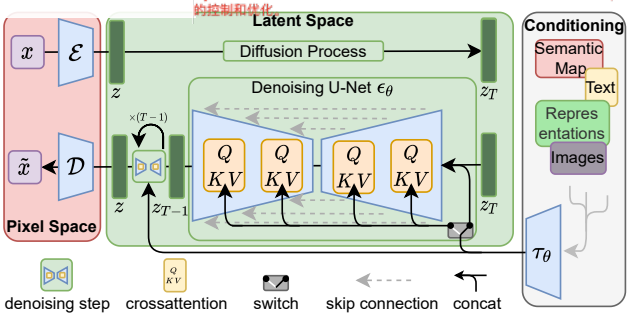

Figure 3. We condition LDMs either via concatenation or by a more general cross-attention mechanism. See Sec. 3.3

includes the ability to build the underlying UNet primarily from 2D convolutional layers, and further focusing the objective on the perceptually most relevant bits using the reweighted bound, which now reads

这个是优化后的基于潜空间训练的的目标函数

$$L_{LDM} := \mathbb{E}_{\mathcal{E}(x),\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t)\|_2^2\right]. \qquad (2)$$

The neural backbone $\epsilon_\theta(\circ, t)$ of our model is realized as a time-conditional UNet [71]. Since the forward process is fixed, z_t can be efficiently obtained from $\mathcal{E}$ during training, and samples from $p(z)$ can be decoded to image space with a single pass through $\mathcal{D}$.

3.3. Conditioning Mechanisms

Similar to other types of generative models [56, 83], diffusion models are in principle capable of modeling conditional distributions of the form $p(z|y)$. This can be implemented with a conditional denoising autoencoder $\epsilon_\theta(z_t, t, y)$ and paves the way to controlling the synthesis process through inputs y such as text [68], semantic maps [33, 61] or other image-to-image translation tasks [34].

In the context of image synthesis, however, combining the generative power of DMs with other types of conditionings beyond class-labels [15] or blurred variants of the input image [72] is so far an under-explored area of research.

We turn DMs into more flexible conditional image generators by augmenting their underlying UNet backbone with the cross-attention mechanism [97], which is effective for learning attention-based models of various input modalities [35, 36]. To pre-process y from various modalities (such as language prompts) we introduce a domain specific encoder τ_θ that projects y to an intermediate representation $\tau_\theta(y) \in \mathbb{R}^{M \times d_\tau}$, which is then mapped to the intermediate layers of the UNet via a cross-attention layer implementing $\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \cdot V$, with

$$Q = W_Q^{(i)} \cdot \varphi_i(z_t), \; K = W_K^{(i)} \cdot \tau_\theta(y), \; V = W_V^{(i)} \cdot \tau_\theta(y).$$

Here, $\varphi_i(z_t) \in \mathbb{R}^{N \times d_\epsilon^i}$ denotes a (flattened) intermediate representation of the UNet implementing ϵ_θ and $W_V^{(i)} \in$

对学习到的潜在特征的正态分布施加轻微的KL惩罚，类似于VAE [46, 69]，而VQ-reg。尝试两种不同类型的正则化。第一个变体KL-reg.，对学习潜伏的标准正态施加轻微的KL惩罚，类似于VAE [46, 69]，反之VQ-reg。在解码器中使用矢量量化层[96]。该模型可以被解释为VQGAN [23]，但量化层被解码器吸收。因为我们后续的DM被设计为与我们学习的潜在空间z = E（x）的二维结构一起工作，所以我们可以使用相对温和的压缩率并实现非常好的重建。这与以前的工作[23，66]相反，后者依赖于学习空间z的任意1D排序来对其分布自回归建模，从而忽略了z的大部分固有结构。因此，我们的压缩模型很好的保留了细节，完整的目标和训练细节可以再补充文档中找到

扩散模型[82]是设计用于通过逐渐对正态分布变量进行去噪来学习数据分布p（x）的概率模型，这对应于学习长度为T的固定马尔可夫链的逆过程。对于图像合成，最成功的模型[15，30，72]依赖于p（x）的变分下限的重新加权变体，这反映了去噪得分匹配[85]。

公式中的下标2代表矩阵的2范数，就是A的 转置共轭矩阵 与矩阵A的 积 的 最大 特征 根 的 平方 根值，是指空间上两个向量矩阵的直线距离。类似于求棋盘上两点间的直线距离。上标2代表2范数的平方，潜空间的数学期望描述中，先求后面分布的2范数，然后对2范数做平方的目的是为了衡量生成图像与真实图像之间的差异。通过计算生成图像与真实图像在潜空间中的差异的平方和，可以得到一个衡量生成图像质量的指标。这个指标越小，表示生成图像与真实图像越接近，质量越好。因此，对2范数做平方可以放大差异，使得差异更加明显，便于评估生成图像的质量。

基于潜在空间的生成式模型由我们训练的感知压缩模型（由encoder和decoder组成），我们现在可以访问一个高效的低维潜在空间，在这个空间中，高频的、不可感知的细节被抽象掉了。与高维像素空间相比，该空间更适合基于似然的生成模型，因为它们现在可以（i）专注于数据的重要语义位，以及（ii）在较低维度，计算效率更高的空间中进行训练。

与之前依赖于高度压缩、离散潜在空间中的自回归、基于注意力的变压器模型的工作不同[23,66,103]，我们可以利用我们的模型提供的特定于图像的归纳偏差，归纳偏置（inductive bias）在我们的模型中的作用。归纳偏置是指模型对数据的先验假设或偏好，它可以帮助模型更好地学习和表示数据。在我们的模型中，归纳偏置是指我们利用了图像特定的归纳偏好，通过使用2D卷积层构建UNet，并通过重新加权的边界将目标集中在感知上最相关的位上。这种归纳偏置使得我们的模型能够更好地处理具有空间结构的数据，并且不需要像之前的方法那样进行过度的压缩，从而保留了图像的细节。因此，我们的模型在高维像素空间中的表现更好。

Perceptual layer是通过使用感知损失函数来实现的。感知损失函数是基于预训练的卷积神经网络（通常是VGG网络）的中间层特征来计算的。具体而言，它通过比较生成图像和真实图像在这些中间层特征上的差异来度量图像的感知质量。

在训练过程中，生成器生成一张图像。然后将生成的图像和真实图像都输入到预训练的卷积神经网络中，提取它们在中间层的特征表示。然后，通过比较这些特征表示的差异来计算感知损失。生成器的目标是最小化感知损失，以使生成的图像在感知上与真实图像更接近。

感知损失函数的使用可以帮助生成器生成更加逼真和高质量的图像。因为它考虑了人类感知的因素。相比于传统的像素级别的损失函数（如均方误差），感知损失函数更加关注图像的结构和语义信息，可以更好地捕捉到人眼对图像质量的敏感度。总结起来，Perceptual layer通过使用感知损失函数来衡量生成图像和真实图像之间的感知差异，从而实现对图像的感知质量的控制和优化。

这里提到了两个能力。首先，它指的是使用二维的卷积层构建底层UNet的能力。UNet是一种常用的神经网络架构，用于图像分割和图像生成等任务。通过使用2D卷积层，可以有效地提取图像中的特征，并将其用于生成高质量的图像。

其次，它提到了重新加权边界的能力。这是指在生成图像时，对于感知上最相关的比特（即图像中最重要的细节）进行重点关注。通过重新加权边界，可以确保生成的图像在保留重要细节的同时，具有更好的感知质量。

综上所述，这句话强调了使用二维卷积层构建UNet和重新加权边界的能力，以提高生成图像的质量和感知效果。

与其他类型的生成模型相似[56,83]，扩散模型原则上能够建模形式为p(z|y)的条件分布。这可以通过一个条件去噪自动编码器来实现，并为通过输入y来控制合成过程铺平了道路，例如文本[68]、语义映射[33,61]或其他图像到图像的翻译任务[34]。

接下来开始转折，在图像合成方面，除了类标签 [15] 或输入图像的模糊变体 [72] 之外，将扩散模型的生成能力与其他类型的条件相结合，这个为止还是一个尚未充分开发的研究领域。

在给定文本条件的图像合成中，除了本文提到的LDM模型，还有其他一些条件结合方法。以下是一些常见的方法：1. 条件生成对抗网络（Conditional Generative Adversarial Networks，CGAN）：CGAN是一种通过在生成器和判别器中引入条件信息来实现图像合成的方法。生成器和判别器都接收条件信息作为输入，生成器根据条件信息生成图像，判别器则评估生成图像的真实性。通过对抗训练，生成器学习根据条件信息生成逼真的图像。2. 变分自编码器（Variational Autoencoder，VAE）：VAE是一种生成模型，它通过学习数据的潜在分布来生成新的样本。在条件图像合成中，可以将条件信息作为VAE的输入，通过学习条件分布来生成与条件相匹配的图像。3. 注意力机制（Attention Mechanism）：注意力机制可以用于将条件信息与生成器的中间表示进行结合，以指导生成器生成与条件相一致的图像。通过引入注意力机制，生成器可以在生成图像时更加关注与条件相关的特征。

这些方法在条件图像合成任务中都有广泛的应用，并且都有各自的优势和适用场景。具体选择哪种方法取决于任务需求和数据特点。

在这篇论文中，交叉注意力机制中的Q、K、V代表的是不同的输入。Q（Query）表示查询，K（Key）表示键，V（Value）表示值。这种注意力机制是用于条件化扩散模型的生成过程中。在这个过程中，Q和K是来自于多模态训练的UNet的输出，而V是噪音。通过将Q、K、V输入到交叉注意力机制中，可以实现对生成过程的控制。噪音是通过对输入图像进行逐渐去噪的过程引入的。具体来说，作者使用了一个固定长度为T的马尔可夫链，通过逐步去噪一个服从正态分布的变量来学习数据分布。每个去噪步骤都对输入图像进行噪音处理，然后通过训练一个去噪自编码器来预测去噪后的图像。这个过程中引入的噪音可以被看作是对输入图像的扰动，用于训练模型学习数据分布。

Figure 4. Samples from *LDMs* trained on CelebAHQ [39], FFHQ [41], LSUN-Churches [102], LSUN-Bedrooms [102] and class-conditional ImageNet [12], each with a resolution of 256×256. Best viewed when zoomed in. For more samples *cf*. the supplement.

$\mathbb{R}^{d \times d'_\epsilon}$, $W_Q^{(i)} \in \mathbb{R}^{d \times d_\tau}$ & $W_K^{(i)} \in \mathbb{R}^{d \times d_\tau}$ are learnable projection matrices [36, 97]. See Fig. 3 for a visual depiction.

Based on image-conditioning pairs, we then learn the conditional LDM via

$$L_{LDM} := \mathbb{E}_{\mathcal{E}(x),y,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t, \tau_\theta(y))\|_2^2\right], \quad (3)$$

where both τ_θ and ϵ_θ are jointly optimized via Eq. 3. This conditioning mechanism is flexible as τ_θ can be parameterized with domain-specific experts, *e.g.* (unmasked) transformers [97] when y are text prompts (see Sec. 4.3.1)

4. Experiments

LDMs provide means to flexible and computationally tractable diffusion based image synthesis of various image modalities, which we empirically show in the following. Firstly, however, we analyze the gains of our models compared to pixel-based diffusion models in both training and inference. Interestingly, we find that *LDMs* trained in *VQ*-regularized latent spaces sometimes achieve better sample quality, even though the reconstruction capabilities of *VQ*-regularized first stage models slightly fall behind those of their continuous counterparts, *cf*. Tab. 8. A visual comparison between the effects of first stage regularization schemes on *LDM* training and their generalization abilities to resolutions $> 256^2$ can be found in Appendix D.1. In E.2 we list details on architecture, implementation, training and evaluation for all results presented in this section.

4.1. On Perceptual Compression Tradeoffs

This section analyzes the behavior of our LDMs with different downsampling factors $f \in \{1, 2, 4, 8, 16, 32\}$ (abbreviated as *LDM-f*, where *LDM-1* corresponds to pixel-based DMs). To obtain a comparable test-field, we fix the computational resources to a single NVIDIA A100 for all experiments in this section and train all models for the same number of steps and with the same number of parameters.

Tab. 8 shows hyperparameters and reconstruction performance of the first stage models used for the *LDMs* compared in this section. Fig. 6 shows sample quality as a function of training progress for 2M steps of class-conditional models on the ImageNet [12] dataset. We see that, i) small downsampling factors for *LDM-{1,2}* result in slow training progress, whereas ii) overly large values of f cause stagnating fidelity after comparably few training steps. Revisiting the analysis above (Fig. 1 and 2) we attribute this to i) leaving most of perceptual compression to the diffusion model and ii) too strong first stage compression resulting in information loss and thus limiting the achievable quality. *LDM-{4-16}* strike a good balance between efficiency and perceptually faithful results, which manifests in a significant FID [29] gap of 38 between pixel-based diffusion (*LDM-1*) and *LDM-8* after 2M training steps.

In Fig. 7, we compare models trained on CelebA-HQ [39] and ImageNet in terms sampling speed for different numbers of denoising steps with the DDIM sampler [84] and plot it against FID-scores [29]. *LDM-{4-8}* outperform models with unsuitable ratios of perceptual and conceptual compression. Especially compared to pixel-based *LDM-1*, they achieve much lower FID scores while simultaneously significantly increasing sample throughput. Complex datasets such as ImageNet require reduced compression rates to avoid reducing quality. In summary, *LDM-4* and *-8* offer the best conditions for achieving high-quality synthesis results.

4.2. Image Generation with Latent Diffusion

We train unconditional models of 256^2 images on CelebA-HQ [39], FFHQ [41], LSUN-Churches and -Bedrooms [102] and evaluate the i) sample quality and ii) their coverage of the data manifold using ii) FID [29] and ii) Precision-and-Recall [50]. Tab. 1 summarizes our results. On CelebA-HQ, we report a new state-of-the-art FID of 5.11, outperforming previous likelihood-based models as well as GANs. We also outperform LSGM [93] where a latent diffusion model is trained jointly together with the first stage. In contrast, we train diffusion models in a fixed space

Inception Score（Inception分数）是一种用于评估生成模型生成图像质量的指标。它是由Ian Goodfellow等人在2016年提出的。Inception Score结合了两个方面的评估：图像质量和图像多样性。
具体来说，Inception Score使用了预训练的Inception模型来计算生成图像的类别概率分布。然后，通过计算生成图像的KL散度（Kullback-Leibler散度）来衡量生成图像的多样性。较高的Inception Score表示生成图像具有较高的质量和多样性。 需要注意的是，Inception Score并不是完美的评估指标，它有一些局限性。例如，它只考虑了图像的类别概率分布，而没有考虑图像的细节和真实性。因此，在评估生成模型时，还需要综合考虑其他指标和人类主观评价。
KL散度（Kullback-Leibler divergence），也称为相对熵，是一种用于衡量两个概率分布之间差异的指标。它由两个概率分布P和Q的对数差的期望值得到，可以表示为D(P||Q)。KL散度不是对称的，即D(P||Q) ≠ D(Q||P)。KL散度的计算公式如下： D(P||Q) = Σ P(x) * log(P(x)/Q(x))
KL散度的值越大，表示两个概率分布之间的差异越大。当KL散度为0时，表示两个概率分布完全相同。KL散度常用于信息论、统计学和机器学习中，例如在生成模型中用于衡量生成的样本分布与真实数据分布之间的差异。

FID（Fréchet Inception Distance）和LPIPS（Learned Perceptual Image Patch Similarity）是用于评估图像生成模型质量的指标。
FID是一种用于衡量生成图像与真实图像之间差异的指标。它基于真实图像和生成图像在预训练的Inception网络中提取的特征之间的统计距离。FID值越低，表示生成图像与真实图像越相似，质量越高。
LPIPS是一种用于衡量图像之间感知相似性的指标。它基于学习的模型，通过比较图像中的感知特征来计算图像之间的相似性。LPIPS值越低，表示图像之间的感知差异越小，质量越高。
这两个指标都被广泛应用于评估图像生成模型的性能，可以帮助研究人员和开发者了解生成图像的质量和真实图像之间的差距。

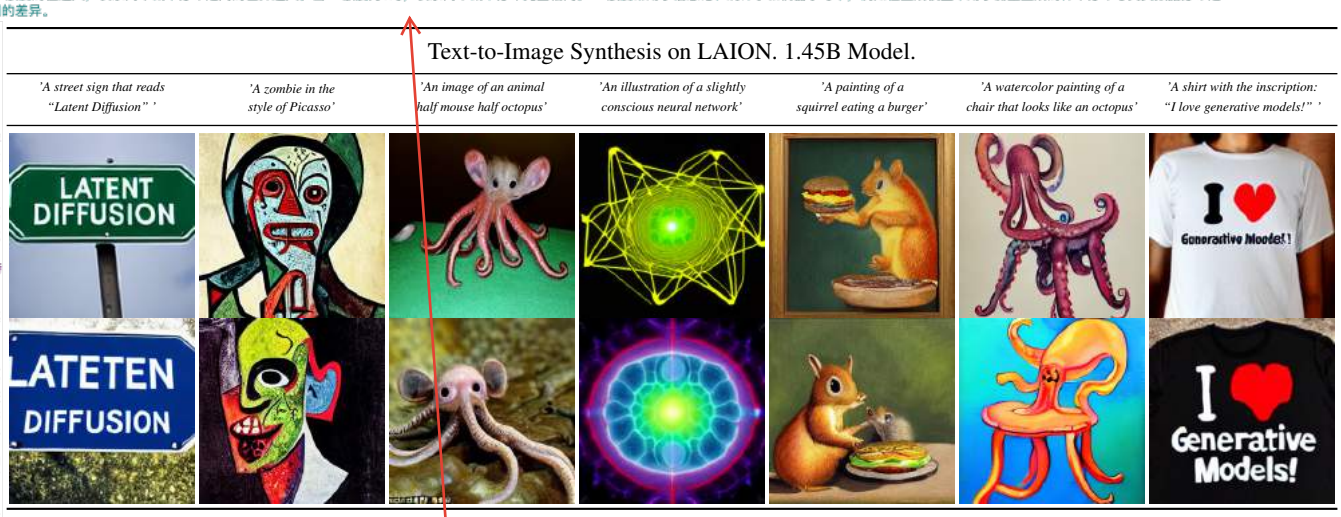

Figure 5. Samples for user-defined text prompts from our model for text-to-image synthesis, *LDM-8 (KL)*, which was trained on the LAION [78] database. Samples generated with 200 DDIM steps and $\eta = 1.0$. We use unconditional guidance [32] with $s = 10.0$.

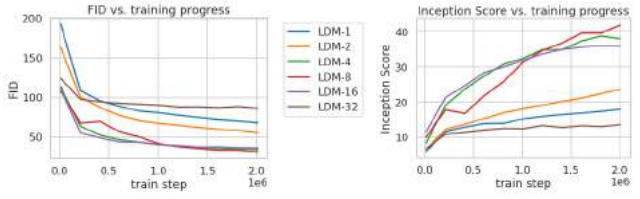

Figure 6. Analyzing the training of class-conditional *LDMs* with different downsampling factors f over 2M train steps on the ImageNet dataset. Pixel-based *LDM-1* requires substantially larger train times compared to models with larger downsampling factors (*LDM-*$\{4\text{-}16\}$). Too much perceptual compression as in *LDM-32* limits the overall sample quality. All models are trained on a single NVIDIA A100 with the same computational budget. Results obtained with 100 DDIM steps [84] and $\kappa = 0$.

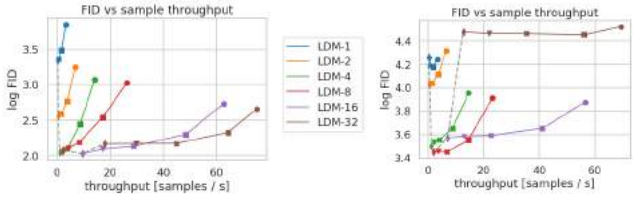

Figure 7. Comparing *LDMs* with varying compression on the CelebA-HQ (left) and ImageNet (right) datasets. Different markers indicate $\{10, 20, 50, 100, 200\}$ sampling steps using DDIM, from right to left along each line. The dashed line shows the FID scores for 200 steps, indicating the strong performance of *LDM-*$\{4\text{-}8\}$. FID scores assessed on 5000 samples. All models were trained for 500k (CelebA) / 2M (ImageNet) steps on an A100.

and avoid the difficulty of weighing reconstruction quality against learning the prior over the latent space, see Fig. 1-2.

We outperform prior diffusion based approaches on all but the LSUN-Bedrooms dataset, where our score is close to ADM [15], despite utilizing half its parameters and requiring 4-times less train resources (see Appendix E.3.5).

CelebA-HQ 256 × 256				FFHQ 256 × 256			
Method	FID ↓	Prec. ↑	Recall ↑	**Method**	FID ↓	Prec. ↑	Recall ↑
DC-VAE [63]	15.8	-	-	ImageBART [21]	9.57	-	-
VQGAN+T. [23] (k=400)	10.2	-	-	U-Net GAN (+aug) [77]	10.9 (7.6)	-	-
PGGAN [39]	8.0	-	-	UDM [43]	5.54	-	-
LSGM [93]	7.22	-	-	StyleGAN [41]	4.16	0.71	0.46
UDM [43]	7.16	-	-	ProjectedGAN [76]	**3.08**	0.65	0.46
LDM-4 (ours, 500-s†)	**5.11**	0.72	0.49	*LDM-4* (ours, 200-s)	4.98	**0.73**	**0.50**

LSUN-Churches 256 × 256				LSUN-Bedrooms 256 × 256			
Method	FID ↓	Prec. ↑	Recall ↑	**Method**	FID ↓	Prec. ↑	Recall ↑
DDPM [30]	7.89	-	-	ImageBART [21]	5.51	-	-
ImageBART [21]	7.32	-	-	DDPM [30]	4.9	-	-
PGGAN [39]	6.42	-	-	UDM [43]	4.57	-	-
StyleGAN [41]	4.21	-	-	StyleGAN [41]	2.35	0.59	0.48
StyleGAN2 [42]	3.86	-	-	ADM [15]	1.90	**0.66**	**0.51**
ProjectedGAN [76]	**1.59**	0.61	0.44	ProjectedGAN [76]	**1.52**	0.61	0.34
*LDM-8** (ours, 200-s)	4.02	**0.64**	**0.52**	*LDM-4* (ours, 200-s)	2.95	**0.66**	0.48

Table 1. Evaluation metrics for unconditional image synthesis. CelebA-HQ results reproduced from [43, 63, 100], FFHQ from [42, 43]. †: N-s refers to N sampling steps with the DDIM [84] sampler. *: trained in *KL*-regularized latent space. Additional results can be found in the supplementary.

Text-Conditional Image Synthesis				
Method	FID ↓	IS↑	*N*params	
CogView† [17]	27.10	18.20	4B	self-ranking, rejection rate 0.017
LAFITE† [109]	26.94	26.02	75M	
GLIDE* [59]	12.24	-	6B	277 DDIM steps, c.f.g. [32] $s = 3$
Make-A-Scene* [26]	**11.84**	-	4B	c.f.g for AR models [98] $s = 5$
LDM-KL-8	23.31	$20.03_{\pm 0.33}$	1.45B	250 DDIM steps
*LDM-KL-8-G**	12.63	$\mathbf{30.29}_{\pm \mathbf{0.42}}$	1.45B	250 DDIM steps, c.f.g. [32] $s = 1.5$

Table 2. Evaluation of text-conditional image synthesis on the 256 × 256-sized MS-COCO [51] dataset: with 250 DDIM [84] steps our model is on par with the most recent diffusion [59] and autoregressive [26] methods despite using significantly less parameters. †/*:Numbers from [109]/ [26]

Moreover, *LDMs* consistently improve upon GAN-based methods in Precision and Recall, thus confirming the advantages of their mode-covering likelihood-based training objective over adversarial approaches. In Fig. 4 we also show qualitative results on each dataset.

这里使用FID作为指标，在相同的样本集上做训练，LDM的效果相对来说遥遥领先

作者介绍了他们的方法与之前其他工作的区别，并解释了为什么他们的方法在效果上更好。
首先，与之前的方法相比，作者的方法引入了一个明确的压缩和生成学习阶段的分离。他们利用一个自编码模型来学习一个与图像空间在感知上等价但计算复杂度显著降低的低维空间。这样做有几个优势：首先，通过离开高维图像空间，他们得到了在低维空间上更高效的扩散模型，因为采样是在低维空间上进行的。其次，他们利用扩散模型从UNet架构中继承的归纳偏差，使其在具有空间结构的数据上特别有效，因此不需要像之前的方法那样进行过度的、降低质量的压缩。最后，他们得到了通用的压缩模型，其潜在空间可以用于训练多个生成模型，并且还可以用于其他下游应用，如基于单图像的CLIP引导合成。
其次，作者的方法在实验中展示了与现有方法相比的有利结果。他们在各种条件图像合成任务上与最先进的方法进行了比较，并取得了良好的结果。他们的方法在图像合成质量和多样性方面都表现出色，并且在各种评估指标上都取得了较低的值，如FID和LPIPS。这表明他们的方法在生成高分辨率图像方面具有更好的性能。
综上所述，作者的方法通过引入压缩和生成学习阶段的分离以及利用归纳偏差，提高了扩散模型的训练和采样效率，并在各种条件图像合成任务中取得了优越的结果。

KL-reg是指KL正则化（KL-regularization），它是一种在训练模型时用于控制模型生成样本的多样性和质量的技术。KL正则化通过在模型的训练过程中引入KL散度（Kullback-Leibler divergence）来约束模型生成的样本分布与目标分布之间的差异。
在图像生成任务中，KL正则化通常用于生成对抗网络（GAN）和变分自编码器（VAE）等模型中。它通过在模型的损失函数中添加一个KL散度项，使得模型在生成样本时更加接近目标分布，从而提高生成样本的质量和多样性。
在文献中，LDM-KL-8和LDM-KL-8-G是基于KL正则化的条件潜在扩散模型（Latent Diffusion Models）的变体。它们通过在模型的训练过程中使用KL正则化来控制生成图像的质量和多样性

Figure 8. Layout-to-image synthesis with an *LDM* on COCO [4], see Sec. 4.3.1. Quantitative evaluation in the supplement D.3.

4.3. Conditional Latent Diffusion

4.3.1 Transformer Encoders for LDMs

这里讲使用BERT tokenizer和transformer来实现τθ，以推断出一个潜在的编码，然后通过(multi-head) cross-attention将其映射到UNet中（参见第3.3节）。这种结合领域特定的专家来学习语言表示和视觉合成的方法，产生了一个强大的模型，可以很好地推广到复杂的、用户定义的文本提示。

By introducing cross-attention based conditioning into LDMs we open them up for various conditioning modalities previously unexplored for diffusion models. For **text-to-image** image modeling, we train a 1.45B parameter *KL*-regularized *LDM* conditioned on language prompts on LAION-400M [78]. We employ the BERT-tokenizer [14] and implement τ_θ as a transformer [97] to infer a latent code which is mapped into the UNet via (multi-head) cross-attention (Sec. 3.3). This combination of domain specific experts for learning a language representation and visual synthesis results in a powerful model, which generalizes well to complex, user-defined text prompts, *cf*. Fig. 8 and 5. For quantitative analysis, we follow prior work and evaluate text-to-image generation on the MS-COCO [51] validation set, where our model improves upon powerful AR [17, 66] and GAN-based [109] methods, *cf*. Tab. 2. We note that applying classifier-free diffusion guidance [32] greatly boosts sample quality, such that the guided *LDM-KL-8-G* is on par with the recent state-of-the-art AR [26] and diffusion models [59] for text-to-image synthesis, while substantially reducing parameter count. To further analyze the flexibility of the cross-attention based conditioning mechanism we also train models to synthesize images based on **semantic layouts** on OpenImages [49], and finetune on COCO [4], see Fig. 8. See Sec. D.3 for the quantitative evaluation and implementation details.

这里总结一下子，经过上述实现，终于发现采样因子在(4,8)时性能最好，优于最先进的扩散模型ADM [15]，同时显着降低计算要求和参数计数

Lastly, following prior work [3, 15, 21, 23], we evaluate our best-performing **class-conditional** ImageNet models with $f \in \{4, 8\}$ from Sec. 4.1 in Tab. 3, Fig. 4 and Sec. D.4. Here we outperform the state of the art diffusion model ADM [15] while significantly reducing computational requirements and parameter count, *cf*. Tab 18.

4.3.2 Convolutional Sampling Beyond 256^2

By concatenating spatially aligned conditioning information to the input of ϵ_θ, *LDMs* can serve as efficient general-purpose image-to-image translation models. We use this to train models for semantic synthesis, super-resolution (Sec. 4.4) and inpainting (Sec. 4.5). For semantic synthesis, we use images of landscapes paired with semantic maps [23, 61] and concatenate downsampled versions of the semantic maps with the latent image representation of a $f = 4$ model (VQ-reg., see Tab. 8). We train on an input resolution of 256^2 (crops from 384^2) but find that our model generalizes to larger resolutions and can generate images up to the megapixel regime when evaluated in a convolutional manner (see Fig. 9). We exploit this behavior to also apply the super-resolution models in Sec. 4.4 and the inpainting models in Sec. 4.5 to generate large images between 512^2 and 1024^2. For this application, the signal-to-noise ratio (induced by the scale of the latent space) significantly affects the results. In Sec. D.1 we illustrate this when learning an LDM on (i) the latent space as provided by a $f = 4$ model (KL-reg., see Tab. 8), and (ii) a rescaled version, scaled by the component-wise standard deviation.

The latter, in combination with classifier-free guidance [32], also enables the direct synthesis of $> 256^2$ images for the text-conditional *LDM-KL-8-G* as in Fig. 13.

作者通过将空间上对齐的条件信息与输入的潜空间噪音连接起来，构建了高效的通用图像到图像翻译模型（LDM）。作者利用这个模型来训练语义综合、超分辨率和修复模型。对于语义综合，作者使用景观图像和语义地图进行配对，并将下采样的语义地图与f=4模型的潜在图像表示连接起来。作者在256^2的输入分辨率上进行训练，但发现模型可以推广到更大的分辨率，并在卷积方式下生成高达百万像素级别的图像。作者还利用这种行为将超分辨率模型应用于生成512^2到1024^2之间的大型图像，并观察到信噪比对结果有显著影响。在第D.1节中，作者通过在潜在空间上学习LDM，验证了信噪比对结果的影响，并比较了使用f=4模型的潜在空间和使用经过标准差调整的潜在空间的结果

Method	FID↓	IS↑	Precision↑	Recall↑	N_{params}	
BigGan-deep [3]	6.95	203.6$_{\pm 2.6}$	**0.87**	0.28	340M	-
ADM [15]	10.94	100.98	0.69	**0.63**	554M	250 DDIM steps
ADM-G [15]	4.59	186.7	0.82	0.52	608M	250 DDIM steps
LDM-4 (ours)	10.56	103.49$_{\pm 1.24}$	0.71	0.62	400M	250 DDIM steps
LDM-4-G (ours)	**3.60**	**247.67**$_{\pm 5.59}$	**0.87**	0.48	400M	250 steps, c.f.g [32], $s = 1.5$

Table 3. Comparison of a class-conditional ImageNet *LDM* with recent state-of-the-art methods for class-conditional image generation on ImageNet [12]. A more detailed comparison with additional baselines can be found in D.4, Tab. 10 and F. *c.f.g.* denotes classifier-free guidance with a scale s as proposed in [32].

Figure 9. A *LDM* trained on 256^2 resolution can generalize to larger resolution (here: 512×1024) for spatially conditioned tasks such as semantic synthesis of landscape images. See Sec. 4.3.2.

4.4. Super-Resolution with Latent Diffusion

LDMs can be efficiently trained for super-resolution by diretly conditioning on low-resolution images via concatenation (*cf*. Sec. 3.3). In a first experiment, we follow SR3

Figure 10. ImageNet 64→256 super-resolution on ImageNet-Val. *LDM-SR* has advantages at rendering realistic textures but SR3 can synthesize more coherent fine structures. See appendix for additional samples and cropouts. SR3 results from [72].

[72] and fix the image degradation to a bicubic interpolation with 4×-downsampling and train on ImageNet following SR3's data processing pipeline. We use the $f = 4$ autoencoding model pretrained on OpenImages (VQ-reg., *cf*. Tab. 8) and concatenate the low-resolution conditioning y and the inputs to the UNet, *i.e*. τ_θ is the identity. Our qualitative and quantitative results (see Fig. 10 and Tab. 5) show competitive performance and LDM-SR outperforms SR3 in FID while SR3 has a better IS. A simple image regression model achieves the highest PSNR and SSIM scores; however these metrics do not align well with human perception [106] and favor blurriness over imperfectly aligned high frequency details [72]. Further, we conduct a user study comparing the pixel-baseline with LDM-SR. We follow SR3 [72] where human subjects were shown a low-res image in between two high-res images and asked for preference. The results in Tab. 4 affirm the good performance of LDM-SR. PSNR and SSIM can be pushed by using a post-hoc guiding mechanism [15] and we implement this *image-based guider* via a perceptual loss, see Sec. D.6.

User Study	SR on ImageNet		Inpainting on Places	
	Pixel-DM ($f1$)	*LDM-4*	LAMA [88]	*LDM-4*
Task 1: Preference vs GT ↑	16.0%	**30.4%**	13.6%	**21.0%**
Task 2: Preference Score ↑	29.4%	**70.6%**	31.9%	**68.1%**

Table 4. Task 1: Subjects were shown ground truth and generated image and asked for preference. Task 2: Subjects had to decide between two generated images. More details in E.3.6

Since the bicubic degradation process does not generalize well to images which do not follow this pre-processing, we also train a generic model, *LDM-BSR*, by using more diverse degradation. The results are shown in Sec. D.6.1.

Method	FID ↓	IS ↑	PSNR ↑	SSIM ↑	N_{params}	[samples/s](*)
Image Regression [72]	15.2	121.1	**27.9**	**0.801**	625M	N/A
SR3 [72]	5.2	**180.1**	26.4	0.762	625M	N/A
LDM-4 (ours, 100 steps)	2.8†/4.8‡	166.3	24.4±3.8	0.69±0.14	**169M**	4.62
emphLDM-4 (ours, big, 100 steps)	**2.4**†/**4.3**‡	174.9	24.7±4.1	0.71±0.15	552M	4.5
LDM-4 (ours, 50 steps, guiding)	4.4†/6.4‡	153.7	25.8±3.7	0.74±0.12	184M	0.38

Table 5. ×4 upscaling results on ImageNet-Val. (256^2); †: FID features computed on validation split, ‡: FID features computed on train split; *: Assessed on a NVIDIA A100

Model (*reg.*-type)	train throughput samples/sec.	sampling throughput† @256	@512	train+val hours/epoch	FID@2k epoch 6
LDM-1 (no first stage)	0.11	0.26	0.07	20.66	24.74
LDM-4 (*KL*, w/ attn)	0.32	0.97	0.34	7.66	15.21
LDM-4 (*VQ*, w/ attn)	0.33	0.97	0.34	7.04	14.99
LDM-4 (*VQ*, w/o attn)	0.35	0.99	0.36	6.66	15.95

Table 6. Assessing inpainting efficiency. †: Deviations from Fig. 7 due to varying GPU settings/batch sizes *cf*. the supplement.

4.5. Inpainting with Latent Diffusion

Inpainting is the task of filling masked regions of an image with new content either because parts of the image are are corrupted or to replace existing but undesired content within the image. We evaluate how our general approach for conditional image generation compares to more specialized, state-of-the-art approaches for this task. Our evaluation follows the protocol of LaMa [88], a recent inpainting model that introduces a specialized architecture relying on Fast Fourier Convolutions [8]. The exact training & evaluation protocol on Places [108] is described in Sec. E.2.2.

We first analyze the effect of different design choices for the first stage. In particular, we compare the inpainting efficiency of *LDM-1* (*i.e*. a pixel-based conditional DM) with *LDM-4*, for both *KL* and *VQ* regularizations, as well as *VQ-LDM-4* without any attention in the first stage (see Tab. 8), where the latter reduces GPU memory for decoding at high resolutions. For comparability, we fix the number of parameters for all models. Tab. 6 reports the training and sampling throughput at resolution 256^2 and 512^2, the total training time in hours per epoch and the FID score on the validation split after six epochs. Overall, we observe a speed-up of at least 2.7× between pixel- and latent-based diffusion models while improving FID scores by a factor of at least 1.6×.

The comparison with other inpainting approaches in Tab. 7 shows that our model with attention improves the overall image quality as measured by FID over that of [88]. LPIPS between the unmasked images and our samples is slightly higher than that of [88]. We attribute this to [88] only producing a single result which tends to recover more of an average image compared to the diverse results produced by our LDM *cf*. Fig. 21. Additionally in a user study (Tab. 4) human subjects favor our results over those of [88].

Based on these initial results, we also trained a larger diffusion model (*big* in Tab. 7) in the latent space of the *VQ*-regularized first stage without attention. Following [15], the UNet of this diffusion model uses attention layers on three levels of its feature hierarchy, the BigGAN [3] residual block for up- and downsampling and has 387M parameters

Figure 11. Qualitative results on object removal with our *big, w/ ft* inpainting model. For more results, see Fig. 22.

instead of 215M. After training, we noticed a discrepancy in the quality of samples produced at resolutions 256^2 and 512^2, which we hypothesize to be caused by the additional attention modules. However, fine-tuning the model for half an epoch at resolution 512^2 allows the model to adjust to the new feature statistics and sets a new state of the art FID on image inpainting (*big, w/o attn, w/ ft* in Tab. 7, Fig. 11.).

5. Limitations & Societal Impact

Limitations While LDMs significantly reduce computational requirements compared to pixel-based approaches, their sequential sampling process is still slower than that of GANs. Moreover, the use of LDMs can be questionable when high precision is required: although the loss of image quality is very small in our $f = 4$ autoencoding models (see Fig. 1), their reconstruction capability can become a bottleneck for tasks that require fine-grained accuracy in pixel space. We assume that our superresolution models (Sec. 4.4) are already somewhat limited in this respect.

Societal Impact Generative models for media like imagery are a double-edged sword: On the one hand, they enable various creative applications, and in particular approaches like ours that reduce the cost of training and inference have the potential to facilitate access to this technology and democratize its exploration. On the other hand, it also means that it becomes easier to create and disseminate manipulated data or spread misinformation and spam. In particular, the deliberate manipulation of images ("deep fakes") is a common problem in this context, and women in particular are disproportionately affected by it [13, 24].

Generative models can also reveal their training data [5, 90], which is of great concern when the data contain sensitive or personal information and were collected without explicit consent. However, the extent to which this also applies to DMs of images is not yet fully understood.

Finally, deep learning modules tend to reproduce or exacerbate biases that are already present in the data [22, 38, 91]. While diffusion models achieve better coverage of the data distribution than *e.g.* GAN-based approaches, the extent to which our two-stage approach that combines adversarial training and a likelihood-based objective misrepresents the data remains an important research question.

For a more general, detailed discussion of the ethical considerations of deep generative models, see *e.g.* [13].

	40-50% masked		All samples	
Method	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓
LDM-4 (ours, big, w/ ft)	**9.39**	0.246± 0.042	**1.50**	0.137± 0.080
LDM-4 (ours, big, w/o ft)	12.89	0.257± 0.047	2.40	0.142± 0.085
LDM-4 (ours, w/ attn)	11.87	0.257± 0.042	2.15	0.144± 0.084
LDM-4 (ours, w/o attn)	12.60	0.259± 0.041	2.37	0.145± 0.084
LaMa [88]†	12.31	**0.243**± 0.038	2.23	**0.134**± 0.080
LaMa [88]	12.0	**0.24**	2.21	0.14
CoModGAN [107]	10.4	0.26	1.82	0.15
RegionWise [52]	21.3	0.27	4.75	0.15
DeepFill v2 [104]	22.1	0.28	5.20	0.16
EdgeConnect [58]	30.5	0.28	8.37	0.16

Table 7. Comparison of inpainting performance on 30k crops of size 512×512 from test images of Places [108]. The column *40-50%* reports metrics computed over hard examples where 40-50% of the image region have to be inpainted. †recomputed on our test set, since the original test set used in [88] was not available.

6. Conclusion

We have presented latent diffusion models, a simple and efficient way to significantly improve both the training and sampling efficiency of denoising diffusion models without degrading their quality. Based on this and our cross-attention conditioning mechanism, our experiments could demonstrate favorable results compared to state-of-the-art methods across a wide range of conditional image synthesis tasks without task-specific architectures.

This work has been supported by the German Federal Ministry for Economic Affairs and Energy within the project 'KI-Absicherung - Safe AI for automated driving' and by the German Research Foundation (DFG) project 421703927.

References

[1] Eirikur Agustsson and Radu Timofte. NTIRE 2017 challenge on single image super-resolution: Dataset and study. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 1122–1131. IEEE Computer Society, 2017. 1

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017. 3

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *Int. Conf. Learn. Represent.*, 2019. 1, 2, 7, 8, 22, 28

[4] Holger Caesar, Jasper R. R. Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1209–1218. Computer Vision Foundation / IEEE Computer Society, 2018. 7, 20, 22

[5] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650, 2021. 9

[6] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, pages 1691–1703. PMLR, 2020. 3

[7] Nanxin Chen, Yu Zhang, Heiga Zen, Ron J. Weiss, Mohammad Norouzi, and William Chan. Wavegrad: Estimating gradients for waveform generation. In *ICLR*. OpenReview.net, 2021. 1

[8] Lu Chi, Borui Jiang, and Yadong Mu. Fast fourier convolution. In *NeurIPS*, 2020. 8

[9] Rewon Child. Very deep vaes generalize autoregressive models and can outperform them on images. *CoRR*, abs/2011.10650, 2020. 3

[10] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *CoRR*, abs/1904.10509, 2019. 3

[11] Bin Dai and David P. Wipf. Diagnosing and enhancing VAE models. In *ICLR (Poster)*. OpenReview.net, 2019. 2, 3

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. IEEE Computer Society, 2009. 1, 5, 7, 22

[13] Emily Denton. Ethical considerations of generative ai. AI for Content Creation Workshop, CVPR, 2021. 9

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. 7

[15] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *CoRR*, abs/2105.05233, 2021. 1, 2, 3, 4, 6, 7, 8, 18, 22, 25, 26, 28

[16] Sander Dieleman. Musings on typicality, 2020. 1, 3

[17] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, and Jie Tang. Cogview: Mastering text-to-image generation via transformers. *CoRR*, abs/2105.13290, 2021. 6, 7

[18] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation, 2015. 3

[19] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 1, 3

[20] Alexey Dosovitskiy and Thomas Brox. Generating images with perceptual similarity metrics based on deep networks. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Adv. Neural Inform. Process. Syst.*, pages 658–666, 2016. 3

[21] Patrick Esser, Robin Rombach, Andreas Blattmann, and Björn Ommer. Imagebart: Bidirectional context with multinomial diffusion for autoregressive image synthesis. *CoRR*, abs/2108.08827, 2021. 6, 7, 22

[22] Patrick Esser, Robin Rombach, and Björn Ommer. A note on data biases in generative models. *arXiv preprint arXiv:2012.02516*, 2020. 9

[23] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. *CoRR*, abs/2012.09841, 2020. 2, 3, 4, 6, 7, 21, 22, 29, 34, 36

[24] Mary Anne Franks and Ari Ezra Waldman. Sex, lies, and videotape: Deep fakes and free speech delusions. *Md. L. Rev.*, 78:892, 2018. 9

[25] Kevin Frans, Lisa B. Soros, and Olaf Witkowski. Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. *ArXiv*, abs/2106.14843, 2021. 3

[26] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. *CoRR*, abs/2203.13131, 2022. 6, 7, 16

[27] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial networks. *CoRR*, 2014. 1, 2

[28] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans, 2017. 3

[29] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Adv. Neural Inform. Process. Syst.*, pages 6626–6637, 2017. 1, 5, 26

[30] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 1, 2, 3, 4, 6, 17

[31] Jonathan Ho, Chitwan Saharia, William Chan, David J. Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *CoRR*, abs/2106.15282, 2021. 1, 3, 22

[32] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 6, 7, 16, 22, 28, 37, 38

[33] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, pages 5967–5976. IEEE Computer Society, 2017. 3, 4

[34] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, 2017. 4

[35] Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, Olivier J. Hénaff, Matthew M. Botvinick, Andrew Zisserman, Oriol Vinyals, and João Carreira. Perceiver IO: A general architecture for structured inputs &outputs. *CoRR*, abs/2107.14795, 2021. 4

[36] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and João Carreira. Perceiver: General perception with iterative attention. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 4651–4664. PMLR, 2021. 4, 5

[37] Manuel Jahn, Robin Rombach, and Björn Ommer. High-resolution complex scene synthesis with transformers. *CoRR*, abs/2105.06458, 2021. 20, 22, 27

[38] Niharika Jain, Alberto Olmo, Sailik Sengupta, Lydia Manikonda, and Subbarao Kambhampati. Imperfect imaganation: Implications of gans exacerbating biases on facial data augmentation and snapchat selfie lenses. *arXiv preprint arXiv:2001.09528*, 2020. 9

[39] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *CoRR*, abs/1710.10196, 2017. 5, 6

[40] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4401–4410, 2019. 1

[41] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 5, 6

[42] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. *CoRR*, abs/1912.04958, 2019. 2, 6, 28

[43] Dongjun Kim, Seungjae Shin, Kyungwoo Song, Wanmo Kang, and Il-Chul Moon. Score matching model for unbounded data score. *CoRR*, abs/2106.05527, 2021. 6

[44] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, 2018. 3

[45] Diederik P. Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *CoRR*, abs/2107.00630, 2021. 1, 3, 16

[46] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR*, 2014. 1, 3, 4, 29

[47] Zhifeng Kong and Wei Ping. On fast sampling of diffusion probabilistic models. *CoRR*, abs/2106.00132, 2021. 3

[48] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In *ICLR*. OpenReview.net, 2021. 1

[49] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, and Vittorio Ferrari. The open images dataset V4: unified image classification, object detection, and visual relationship detection at scale. *CoRR*, abs/1811.00982, 2018. 7, 20, 22

[50] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *CoRR*, abs/1904.06991, 2019. 5, 26

[51] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014. 6, 7, 27

[52] Yuqing Ma, Xianglong Liu, Shihao Bai, Le-Yi Wang, Aishan Liu, Dacheng Tao, and Edwin Hancock. Region-wise generative adversarial imageinpainting for large missing areas. *ArXiv*, abs/1909.12507, 2019. 9

[53] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *CoRR*, abs/2108.01073, 2021. 1

[54] Lars M. Mescheder. On the convergence properties of GAN training. *CoRR*, abs/1801.04406, 2018. 3

[55] Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled generative adversarial networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 3

[56] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014. 4

[57] Gautam Mittal, Jesse H. Engel, Curtis Hawthorne, and Ian Simon. Symbolic music generation with diffusion models. *CoRR*, abs/2103.16091, 2021. 1

[58] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z. Qureshi, and Mehran Ebrahimi. Edgeconnect: Generative image inpainting with adversarial edge learning. *ArXiv*, abs/1901.00212, 2019. 9

[59] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: towards photorealistic image generation and editing with text-guided diffusion models. *CoRR*, abs/2112.10741, 2021. 6, 7, 16

[60] Anton Obukhov, Maximilian Seitzer, Po-Wei Wu, Semen Zhydenko, Jonathan Kyl, and Elvis Yu-Jing Lin.

High-fidelity performance metrics for generative models in pytorch, 2020. Version: 0.3.0, DOI: 10.5281/zenodo.4957738. 26, 27

[61] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 4, 7

[62] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 22

[63] Gaurav Parmar, Dacheng Li, Kwonjoon Lee, and Zhuowen Tu. Dual contradistinctive generative autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 823–832. Computer Vision Foundation / IEEE, 2021. 6

[64] Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On buggy resizing libraries and surprising subtleties in fid calculation. *arXiv preprint arXiv:2104.11222*, 2021. 26

[65] David A. Patterson, Joseph Gonzalez, Quoc V. Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David R. So, Maud Texier, and Jeff Dean. Carbon emissions and large neural network training. *CoRR*, abs/2104.10350, 2021. 2

[66] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *CoRR*, abs/2102.12092, 2021. 1, 2, 3, 4, 7, 21, 27

[67] Ali Razavi, Aäron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In *NeurIPS*, pages 14837–14847, 2019. 1, 2, 3, 22

[68] Scott E. Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *ICML*, 2016. 4

[69] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on International Conference on Machine Learning, ICML*, 2014. 1, 4, 29

[70] Robin Rombach, Patrick Esser, and Björn Ommer. Network-to-network translation with conditional invertible neural networks. In *NeurIPS*, 2020. 3

[71] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI (3)*, volume 9351 of *Lecture Notes in Computer Science*, pages 234–241. Springer, 2015. 2, 3, 4

[72] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *CoRR*, abs/2104.07636, 2021. 1, 4, 8, 16, 22, 23, 27

[73] Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P. Kingma. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *CoRR*, abs/1701.05517, 2017. 1, 3

[74] Dave Salvator. NVIDIA Developer Blog. https://developer.nvidia.com/blog/getting-immediate-speedups-with-a100-tf32, 2020. 28

[75] Robin San-Roman, Eliya Nachmani, and Lior Wolf. Noise estimation for generative diffusion models. *CoRR*, abs/2104.02600, 2021. 3

[76] Axel Sauer, Kashyap Chitta, Jens Müller, and Andreas Geiger. Projected gans converge faster. *CoRR*, abs/2111.01007, 2021. 6

[77] Edgar Schönfeld, Bernt Schiele, and Anna Khoreva. A u-net based discriminator for generative adversarial networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 8204–8213. Computer Vision Foundation / IEEE, 2020. 6

[78] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs, 2021. 6, 7

[79] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *Int. Conf. Learn. Represent.*, 2015. 29, 43, 44, 45

[80] Abhishek Sinha, Jiaming Song, Chenlin Meng, and Stefano Ermon. D2C: diffusion-denoising models for few-shot conditional generation. *CoRR*, abs/2106.06819, 2021. 3

[81] Charlie Snell. Alien Dreams: An Emerging Art Scene. https://ml.berkeley.edu/blog/posts/clip-art/, 2021. [Online; accessed November-2021]. 2

[82] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *CoRR*, abs/1503.03585, 2015. 1, 3, 4, 18

[83] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. 4

[84] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*. OpenReview.net, 2021. 3, 5, 6, 22

[85] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *CoRR*, abs/2011.13456, 2020. 1, 3, 4, 18

[86] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for modern deep learning research. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 13693–13696. AAAI Press, 2020. 2

[87] Wei Sun and Tianfu Wu. Learning layout and style reconfigurable gans for controllable image synthesis. *CoRR*, abs/2003.11571, 2020. 22, 27

[88] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor S. Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. *ArXiv*, abs/2109.07161, 2021. 8, 9, 26, 32

[89] Tristan Sylvain, Pengchuan Zhang, Yoshua Bengio, R. Devon Hjelm, and Shikhar Sharma. Object-centric image generation from layouts. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 2647–2655. AAAI Press, 2021. 20, 22, 27

[90] Patrick Tinsley, Adam Czajka, and Patrick Flynn. This face does not exist... but it might be yours! identity leakage in generative models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1320–1328, 2021. 9

[91] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011. 9

[92] Arash Vahdat and Jan Kautz. NVAE: A deep hierarchical variational autoencoder. In *NeurIPS*, 2020. 3

[93] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *CoRR*, abs/2106.05931, 2021. 2, 3, 5, 6

[94] Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, koray kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems*, 2016. 3

[95] Aäron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. *CoRR*, abs/1601.06759, 2016. 3

[96] Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *NIPS*, pages 6306–6315, 2017. 2, 4, 29

[97] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 3, 4, 5, 7

[98] Rivers Have Wings. Tweet on Classifier-free guidance for autoregressive models. https://twitter.com/RiversHaveWings/status/1478093658716966912, 2022. 6

[99] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771, 2019. 26

[100] Zhisheng Xiao, Karsten Kreis, Jan Kautz, and Arash Vahdat. VAEBM: A symbiosis between variational autoencoders and energy-based models. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 6

[101] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using VQ-VAE and transformers. *CoRR*, abs/2104.10157, 2021. 3

[102] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. LSUN: construction of a large-scale image dataset using deep learning with humans in the loop. *CoRR*, abs/1506.03365, 2015. 5

[103] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan, 2021. 3, 4

[104] Jiahui Yu, Zhe L. Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Free-form image inpainting with gated convolution. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4470–4479, 2019. 9

[105] K. Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. *ArXiv*, abs/2103.14006, 2021. 23

[106] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3, 8, 19

[107] Shengyu Zhao, Jianwei Cui, Yilun Sheng, Yue Dong, Xiao Liang, Eric I-Chao Chang, and Yan Xu. Large scale image completion via co-modulated generative adversarial networks. *ArXiv*, abs/2103.10428, 2021. 9

[108] Bolei Zhou, Àgata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:1452–1464, 2018. 8, 9, 26

[109] Yufan Zhou, Ruiyi Zhang, Changyou Chen, Chunyuan Li, Chris Tensmeyer, Tong Yu, Jiuxiang Gu, Jinhui Xu, and Tong Sun. LAFITE: towards language-free training for text-to-image generation. *CoRR*, abs/2111.13792, 2021. 6, 7, 16

Appendix

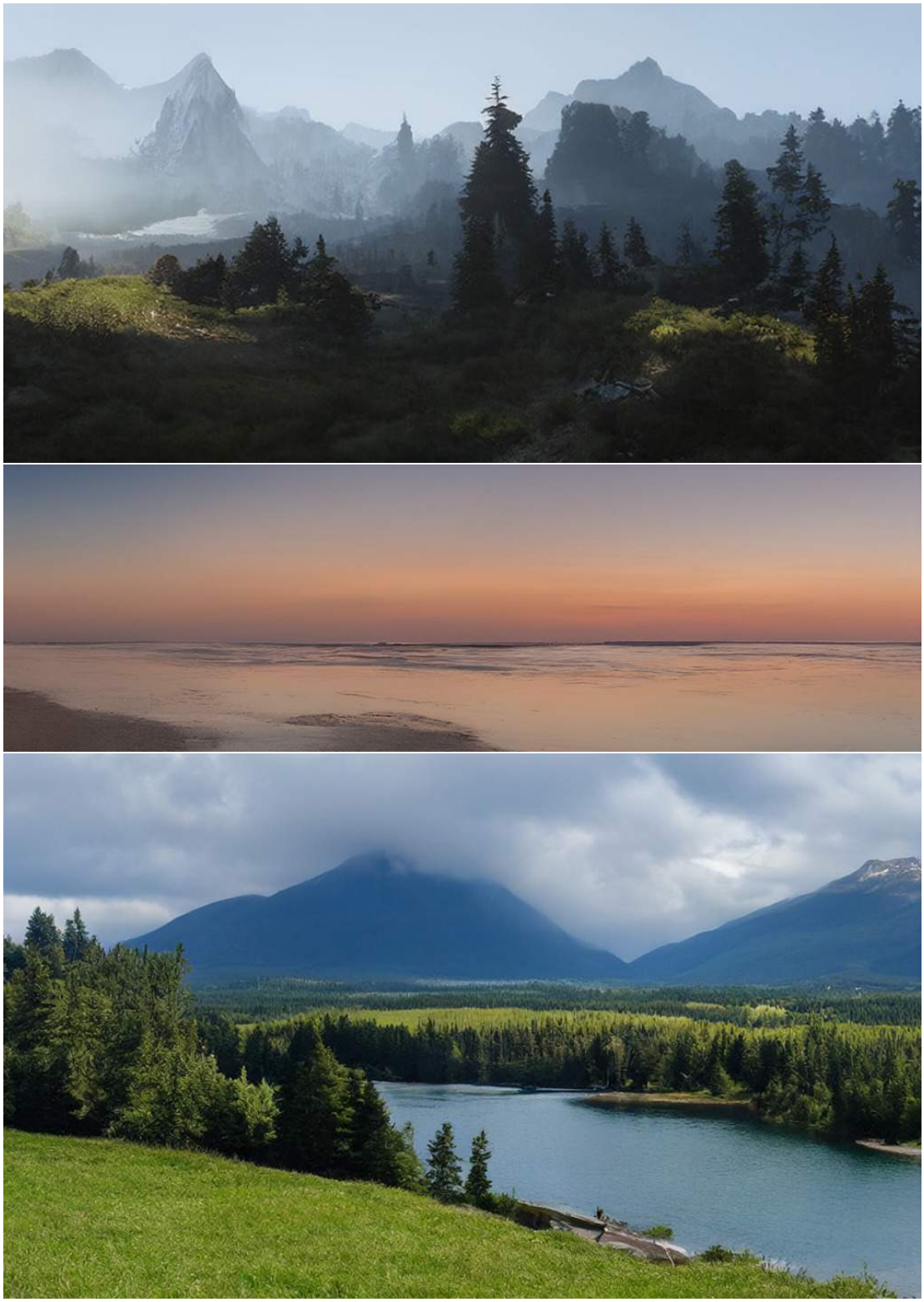

Figure 12. Convolutional samples from the semantic landscapes model as in Sec. 4.3.2, finetuned on 512^2 images.

'A painting of the last supper by Picasso.'

'An oil painting of a latent space.'

'An epic painting of Gandalf the Black summoning thunder and lightning in the mountains.'

'A sunset over a mountain range, vector image.'

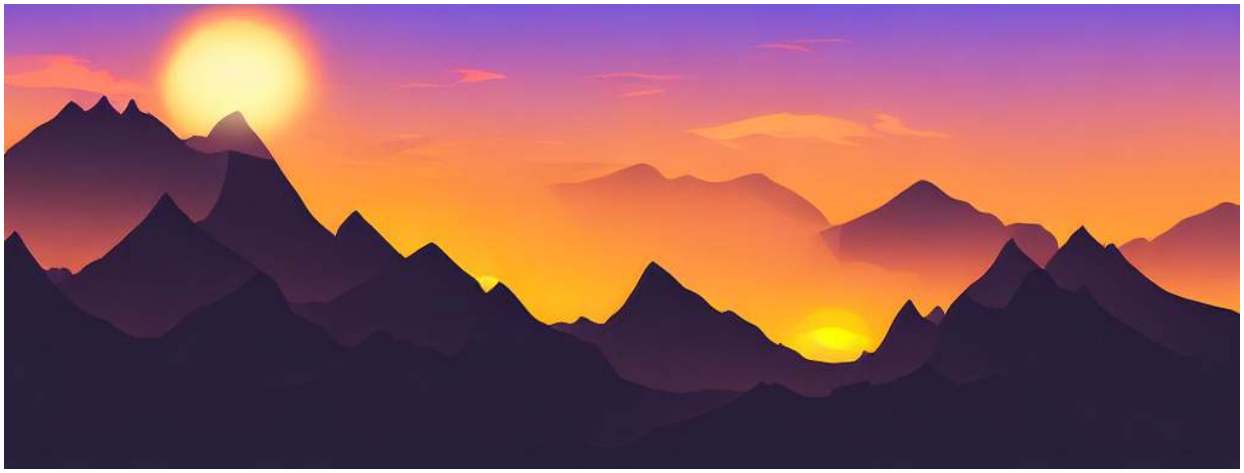

Figure 13. Combining classifier free diffusion guidance with the convolutional sampling strategy from Sec. 4.3.2, our 1.45B parameter text-to-image model can be used for rendering images larger than the native 256^2 resolution the model was trained on.

A. Changelog

Here we list changes between this version (https://arxiv.org/abs/2112.10752v2) of the paper and the previous version, *i.e.* https://arxiv.org/abs/2112.10752v1.

- We updated the results on text-to-image synthesis in Sec. 4.3 which were obtained by training a new, larger model (1.45B parameters). This also includes a new comparison to very recent competing methods on this task that were published on arXiv at the same time as ([59, 109]) or after ([26]) the publication of our work.
- We updated results on class-conditional synthesis on ImageNet in Sec. 4.1, Tab. 3 (see also Sec. D.4) obtained by retraining the model with a larger batch size. The corresponding qualitative results in Fig. 26 and Fig. 27 were also updated. Both the updated text-to-image and the class-conditional model now use classifier-free guidance [32] as a measure to increase visual fidelity.
- We conducted a user study (following the scheme suggested by Saharia et al [72]) which provides additional evaluation for our inpainting (Sec. 4.5) and superresolution models (Sec. 4.4).
- Added Fig. 5 to the main paper, moved Fig. 18 to the appendix, added Fig. 13 to the appendix.

B. Detailed Information on Denoising Diffusion Models

Diffusion models can be specified in terms of a signal-to-noise ratio $\text{SNR}(t) = \frac{\alpha_t^2}{\sigma_t^2}$ consisting of sequences $(\alpha_t)_{t=1}^T$ and $(\sigma_t)_{t=1}^T$ which, starting from a data sample x_0, define a forward diffusion process q as

$$q(x_t|x_0) = \mathcal{N}(x_t|\alpha_t x_0, \sigma_t^2 \mathbb{I}) \tag{4}$$

with the Markov structure for $s < t$:

$$q(x_t|x_s) = \mathcal{N}(x_t|\alpha_{t|s} x_s, \sigma_{t|s}^2 \mathbb{I}) \tag{5}$$

$$\alpha_{t|s} = \frac{\alpha_t}{\alpha_s} \tag{6}$$

$$\sigma_{t|s}^2 = \sigma_t^2 - \alpha_{t|s}^2 \sigma_s^2 \tag{7}$$

Denoising diffusion models are generative models $p(x_0)$ which revert this process with a similar Markov structure running backward in time, *i.e.* they are specified as

$$p(x_0) = \int_z p(x_T) \prod_{t=1}^{T} p(x_{t-1}|x_t) \tag{8}$$

The evidence lower bound (ELBO) associated with this model then decomposes over the discrete time steps as

$$-\log p(x_0) \leq \mathbb{KL}(q(x_T|x_0)|p(x_T)) + \sum_{t=1}^{T} \mathbb{E}_{q(x_t|x_0)} \mathbb{KL}(q(x_{t-1}|x_t, x_0)|p(x_{t-1}|x_t)) \tag{9}$$

The prior $p(x_T)$ is typically choosen as a standard normal distribution and the first term of the ELBO then depends only on the final signal-to-noise ratio $\text{SNR}(T)$. To minimize the remaining terms, a common choice to parameterize $p(x_{t-1}|x_t)$ is to specify it in terms of the true posterior $q(x_{t-1}|x_t, x_0)$ but with the unknown x_0 replaced by an estimate $x_\theta(x_t, t)$ based on the current step x_t. This gives [45]

$$p(x_{t-1}|x_t) \coloneqq q(x_{t-1}|x_t, x_\theta(x_t, t)) \tag{10}$$

$$= \mathcal{N}(x_{t-1}|\mu_\theta(x_t, t), \sigma_{t|t-1}^2 \frac{\sigma_{t-1}^2}{\sigma_t^2} \mathbb{I}), \tag{11}$$

where the mean can be expressed as

$$\mu_\theta(x_t, t) = \frac{\alpha_{t|t-1}\sigma_{t-1}^2}{\sigma_t^2} x_t + \frac{\alpha_{t-1}\sigma_{t|t-1}^2}{\sigma_t^2} x_\theta(x_t, t). \tag{12}$$

In this case, the sum of the ELBO simplify to

$$\sum_{t=1}^{T} \mathbb{E}_{q(x_t|x_0)} \mathbb{KL}(q(x_{t-1}|x_t, x_0)|p(x_{t-1}) = \sum_{t=1}^{T} \mathbb{E}_{\mathcal{N}(\epsilon|0,\mathbb{I})} \frac{1}{2} (\text{SNR}(t-1) - \text{SNR}(t)) \|x_0 - x_\theta(\alpha_t x_0 + \sigma_t \epsilon, t)\|^2 \tag{13}$$

Following [30], we use the reparameterization

$$\epsilon_\theta(x_t, t) = (x_t - \alpha_t x_\theta(x_t, t))/\sigma_t \tag{14}$$

to express the reconstruction term as a denoising objective,

$$\|x_0 - x_\theta(\alpha_t x_0 + \sigma_t \epsilon, t)\|^2 = \frac{\sigma_t^2}{\alpha_t^2} \|\epsilon - \epsilon_\theta(\alpha_t x_0 + \sigma_t \epsilon, t)\|^2 \tag{15}$$

and the reweighting, which assigns each of the terms the same weight and results in Eq. (1).

C. Image Guiding Mechanisms

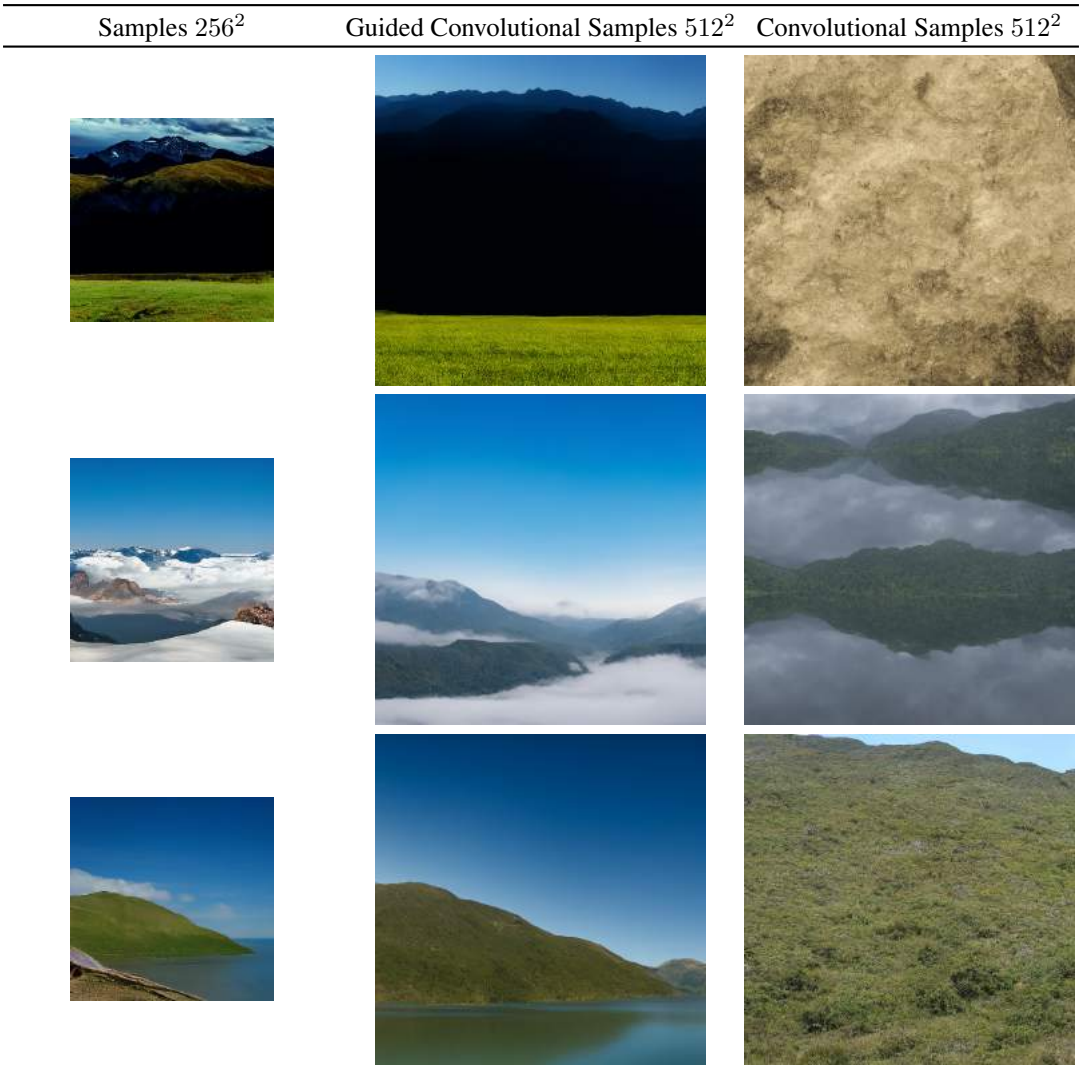

Figure 14. On landscapes, convolutional sampling with unconditional models can lead to homogeneous and incoherent global structures (see column 2). L_2-guiding with a low resolution image can help to reestablish coherent global structures.

An intriguing feature of diffusion models is that unconditional models can be conditioned at test-time [15, 82, 85]. In particular, [15] presented an algorithm to guide both unconditional and conditional models trained on the ImageNet dataset with a classifier $\log p_\Phi(y|x_t)$, trained on each x_t of the diffusion process. We directly build on this formulation and introduce post-hoc *image-guiding*:

For an epsilon-parameterized model with fixed variance, the guiding algorithm as introduced in [15] reads:

$$\hat{\epsilon} \leftarrow \epsilon_\theta(z_t, t) + \sqrt{1-\alpha_t^2}\, \nabla_{z_t} \log p_\Phi(y|z_t)\ . \tag{16}$$

This can be interpreted as an update correcting the "score" ϵ_θ with a conditional distribution $\log p_\Phi(y|z_t)$.

So far, this scenario has only been applied to single-class classification models. We re-interpret the guiding distribution $p_\Phi(y|T(\mathcal{D}(z_0(z_t))))$ as a general purpose image-to-image translation task given a target image y, where T can be any differentiable transformation adopted to the image-to-image translation task at hand, such as the identity, a downsampling operation or similar.

As an example, we can assume a Gaussian guider with fixed variance $\sigma^2 = 1$, such that

$$\log p_\Phi(y|z_t) = -\frac{1}{2}\|y - T(\mathcal{D}(z_0(z_t)))\|_2^2 \tag{17}$$

becomes a L_2 regression objective.

Fig. 14 demonstrates how this formulation can serve as an upsampling mechanism of an unconditional model trained on 256^2 images, where unconditional samples of size 256^2 guide the convolutional synthesis of 512^2 images and T is a $2\times$ bicubic downsampling. Following this motivation, we also experiment with a perceptual similarity guiding and replace the L_2 objective with the LPIPS [106] metric, see Sec. 4.4.

D. Additional Results

D.1. Choosing the Signal-to-Noise Ratio for High-Resolution Synthesis

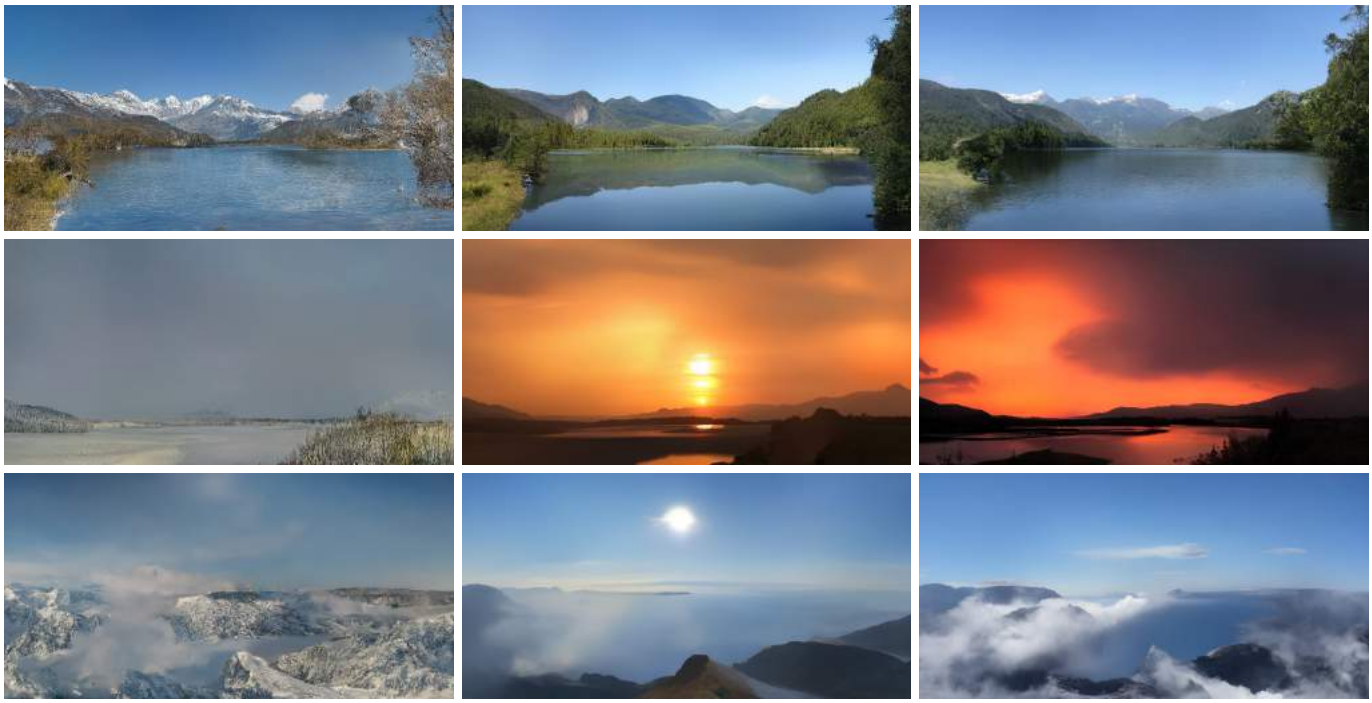

Figure 15. Illustrating the effect of latent space rescaling on convolutional sampling, here for semantic image synthesis on landscapes. See Sec. 4.3.2 and Sec. D.1.

As discussed in Sec. 4.3.2, the signal-to-noise ratio induced by the variance of the latent space (*i.e.* $\mathrm{Var}(z)/\sigma_t^2$) significantly affects the results for convolutional sampling. For example, when training a LDM directly in the latent space of a KL-regularized model (see Tab. 8), this ratio is very high, such that the model allocates a lot of semantic detail early on in the reverse denoising process. In contrast, when rescaling the latent space by the component-wise standard deviation of the latents as described in Sec. G, the SNR is descreased. We illustrate the effect on convolutional sampling for semantic image synthesis in Fig. 15. Note that the VQ-regularized space has a variance close to 1, such that it does not have to be rescaled.

D.2. Full List of all First Stage Models

We provide a complete list of various autoenconding models trained on the OpenImages dataset in Tab. 8.

D.3. Layout-to-Image Synthesis

Here we provide the quantitative evaluation and additional samples for our layout-to-image models from Sec. 4.3.1. We train a model on the COCO [4] and one on the OpenImages [49] dataset, which we subsequently additionally finetune on COCO. Tab 9 shows the result. Our COCO model reaches the performance of recent state-of-the art models in layout-to-image synthesis, when following their training and evaluation protocol [89]. When finetuning from the OpenImages model, we surpass these works. Our OpenImages model surpasses the results of Jahn et al [37] by a margin of nearly 11 in terms of FID. In Fig. 16 we show additional samples of the model finetuned on COCO.

D.4. Class-Conditional Image Synthesis on ImageNet

Tab. 10 contains the results for our class-conditional LDM measured in FID and Inception score (IS). LDM-8 requires significantly fewer parameters and compute requirements (see Tab. 18) to achieve very competitive performance. Similar to previous work, we can further boost the performance by training a classifier on each noise scale and guiding with it,

f	$\lvert\mathcal{Z}\rvert$	c	**R-FID** ↓	**R-IS** ↑	**PSNR** ↑	**PSIM** ↓	**SSIM** ↑
16 *VQGAN* [23]	16384	256	4.98	–	19.9 ±3.4	1.83 ±0.42	0.51 ±0.18
16 *VQGAN* [23]	1024	256	7.94	–	19.4 ±3.3	1.98 ±0.43	0.50 ±0.18
8 *DALL-E* [66]	8192	-	32.01	–	22.8 ±2.1	1.95 ±0.51	0.73 ±0.13
32	16384	16	31.83	40.40 ±1.07	17.45 ±2.90	2.58 ±0.48	0.41 ±0.18
16	16384	8	5.15	144.55 ±3.74	20.83 ±3.61	1.73 ±0.43	0.54 ±0.18
8	16384	4	1.14	201.92 ±3.97	23.07 ±3.99	1.17 ±0.36	0.65 ±0.16
8	256	4	1.49	194.20 ±3.87	22.35 ±3.81	1.26 ±0.37	0.62 ±0.16
4	8192	3	0.58	224.78 ±5.35	27.43 ±4.26	0.53 ±0.21	0.82 ±0.10
4†	8192	3	1.06	221.94 ±4.58	25.21 ±4.17	0.72 ±0.26	0.76 ±0.12
4	256	3	0.47	223.81 ±4.58	26.43 ±4.22	0.62 ±0.24	0.80 ±0.11
2	2048	2	0.16	232.75 ±5.09	30.85 ±4.12	0.27 ±0.12	0.91 ±0.05
2	64	2	0.40	226.62 ±4.83	29.13 ±3.46	0.38 ±0.13	0.90 ±0.05
32	KL	64	2.04	189.53 ±3.68	22.27 ±3.93	1.41 ±0.40	0.61 ±0.17
32	KL	16	7.3	132.75 ±2.71	20.38 ±3.56	1.88 ±0.45	0.53 ±0.18
16	KL	16	0.87	210.31 ±3.97	24.08 ±4.22	1.07 ±0.36	0.68 ±0.15
16	KL	8	2.63	178.68 ±4.08	21.94 ±3.92	1.49 ±0.42	0.59 ±0.17
8	KL	4	0.90	209.90 ±4.92	24.19 ±4.19	1.02 ±0.35	0.69 ±0.15
4	KL	3	0.27	227.57 ±4.89	27.53 ±4.54	0.55 ±0.24	0.82 ±0.11
2	KL	2	0.086	232.66 ±5.16	32.47 ±4.19	0.20 ±0.09	0.93 ±0.04

Table 8. Complete autoencoder zoo trained on OpenImages, evaluated on ImageNet-Val. † denotes an attention-free autoencoder.

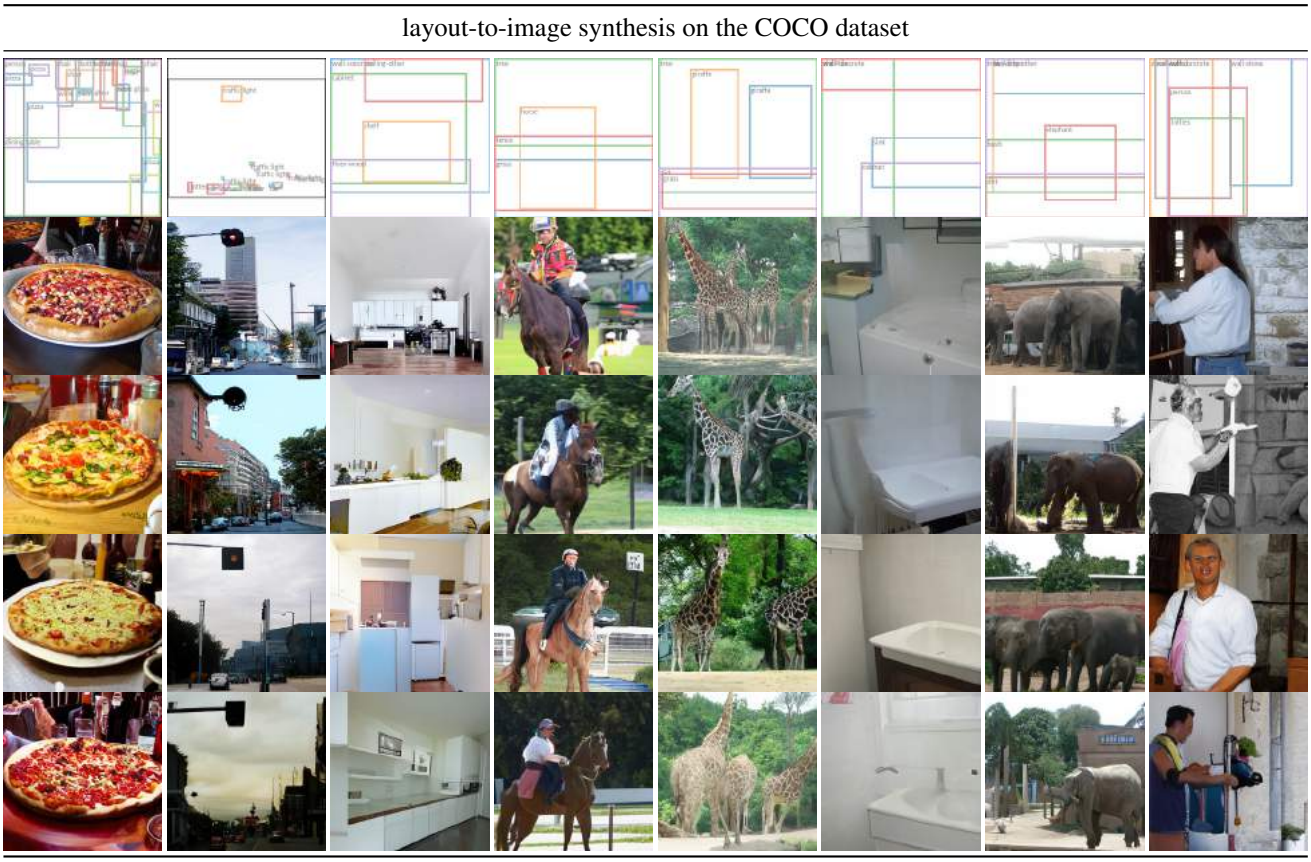

Figure 16. More samples from our best model for layout-to-image synthesis, *LDM-4*, which was trained on the OpenImages dataset and finetuned on the COCO dataset. Samples generated with 100 DDIM steps and $\eta = 0$. Layouts are from the COCO validation set.

see Sec. C. Unlike the pixel-based methods, this classifier is trained very cheaply in latent space. For additional qualitative results, see Fig. 26 and Fig. 27.

	COCO256 × 256	OpenImages 256 × 256	OpenImages 512 × 512
Method	FID↓	FID↓	FID↓
LostGAN-V2 [87]	42.55	-	-
OC-GAN [89]	41.65	-	-
SPADE [62]	41.11	-	-
VQGAN+T [37]	56.58	45.33	48.11
LDM-8 (100 steps, ours)	42.06†	-	-
LDM-4 (200 steps, ours)	**40.91***	**32.02**	**35.80**

Table 9. Quantitative comparison of our layout-to-image models on the COCO [4] and OpenImages [49] datasets. †: Training from scratch on COCO; *: Finetuning from OpenImages.

Method	FID↓	IS↑	Precision↑	Recall↑	N_{params}	
SR3 [72]	11.30	-	-	-	625M	-
ImageBART [21]	21.19	-	-	-	3.5B	-
ImageBART [21]	7.44	-	-	-	3.5B	0.05 acc. rate*
VQGAN+T [23]	17.04	70.6±1.8	-	-	1.3B	-
VQGAN+T [23]	5.88	**304.8**±3.6	-	-	1.3B	0.05 acc. rate*
BigGan-deep [3]	6.95	203.6±2.6	**0.87**	0.28	340M	-
ADM [15]	10.94	100.98	0.69	**0.63**	554M	250 DDIM steps
ADM-G [15]	4.59	186.7	0.82	0.52	608M	250 DDIM steps
ADM-G,ADM-U [15]	3.85	221.72	0.84	0.53	n/a	2 × 250 DDIM steps
CDM [31]	4.88	158.71±2.26	-	-	n/a	2 × 100 DDIM steps
LDM-8 (ours)	17.41	72.92±2.6	0.65	0.62	395M	200 DDIM steps, 2.9M train steps, batch size 64
LDM-8-G (ours)	8.11	190.43±2.60	0.83	0.36	506M	200 DDIM steps, classifier scale 10, 2.9M train steps, batch size 64
LDM-8 (ours)	15.51	79.03±1.03	0.65	**0.63**	395M	200 DDIM steps, 4.8M train steps, batch size 64
LDM-8-G (ours)	7.76	209.52±4.24	0.84	0.35	506M	200 DDIM steps, classifier scale 10, 4.8M train steps, batch size 64
LDM-4 (ours)	10.56	103.49±1.24	0.71	0.62	400M	250 DDIM steps, 178K train steps, batch size 1200
LDM-4-G (ours)	3.95	178.22±2.43	0.81	0.55	400M	250 DDIM steps, unconditional guidance [32] scale 1.25, 178K train steps, batch size 1200
LDM-4-G (ours)	**3.60**	247.67±5.59	**0.87**	0.48	400M	250 DDIM steps, unconditional guidance [32] scale 1.5, 178K train steps, batch size 1200

Table 10. Comparison of a class-conditional ImageNet *LDM* with recent state-of-the-art methods for class-conditional image generation on the ImageNet [12] dataset.*: Classifier rejection sampling with the given rejection rate as proposed in [67].

D.5. Sample Quality vs. V100 Days (Continued from Sec. 4.1)

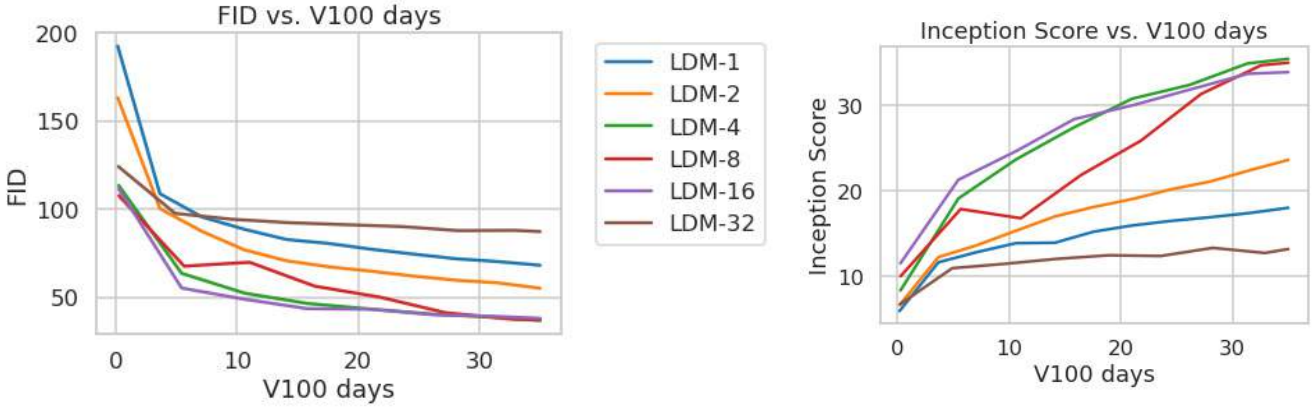

Figure 17. For completeness we also report the training progress of class-conditional *LDMs* on the ImageNet dataset for a fixed number of 35 V100 days. Results obtained with 100 DDIM steps [84] and $\kappa = 0$. FIDs computed on 5000 samples for efficiency reasons.

For the assessment of sample quality over the training progress in Sec. 4.1, we reported FID and IS scores as a function of train steps. Another possibility is to report these metrics over the used resources in V100 days. Such an analysis is additionally provided in Fig. 17, showing qualitatively similar results.

Method	FID ↓	IS ↑	PSNR ↑	SSIM ↑
Image Regression [72]	15.2	121.1	**27.9**	**0.801**
SR3 [72]	5.2	**180.1**	26.4	0.762
LDM-4 (ours, 100 steps)	**2.8**†/**4.8**‡	166.3	24.4$\pm$3.8	0.69$\pm$0.14
LDM-4 (ours, 50 steps, guiding)	4.4†/6.4‡	153.7	25.8$\pm$3.7	0.74$\pm$0.12
LDM-4 (ours, 100 steps, guiding)	4.4†/6.4‡	154.1	25.7$\pm$3.7	0.73$\pm$0.12
LDM-4 (ours, 100 steps, +15 ep.)	**2.6**† / **4.6**‡	169.76$\pm$5.03	24.4$\pm$3.8	0.69$\pm$0.14
Pixel-DM (100 steps, +15 ep.)	5.1† / 7.1‡	163.06$\pm$4.67	24.1$\pm$3.3	0.59$\pm$0.12

Table 11. $\times 4$ upscaling results on ImageNet-Val. (256^2); †: FID features computed on validation split, ‡: FID features computed on train split. We also include a pixel-space baseline that receives the same amount of compute as *LDM-4*. The last two rows received 15 epochs of additional training compared to the former results.

D.6. Super-Resolution

For better comparability between LDMs and diffusion models in pixel space, we extend our analysis from Tab. 5 by comparing a diffusion model trained for the same number of steps and with a comparable number [1] of parameters to our LDM. The results of this comparison are shown in the last two rows of Tab. 11 and demonstrate that LDM achieves better performance while allowing for significantly faster sampling. A qualitative comparison is given in Fig. 20 which shows random samples from both LDM and the diffusion model in pixel space.

D.6.1 LDM-BSR: General Purpose SR Model via Diverse Image Degradation

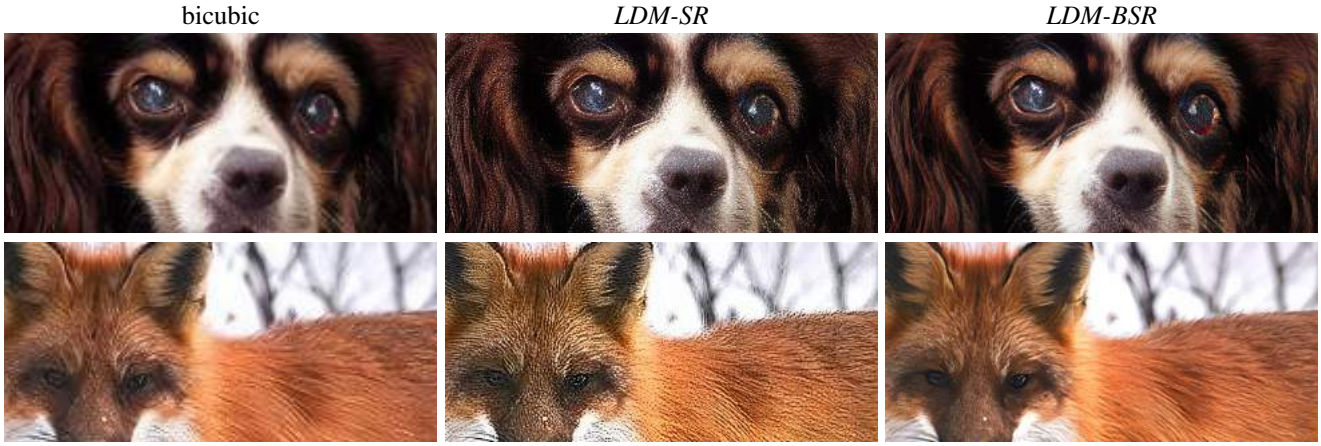

Figure 18. *LDM-BSR* generalizes to arbitrary inputs and can be used as a general-purpose upsampler, upscaling samples from a class-conditional *LDM* (image *cf*. Fig. 4) to 1024^2 resolution. In contrast, using a fixed degradation process (see Sec. 4.4) hinders generalization.

To evaluate generalization of our LDM-SR, we apply it both on synthetic LDM samples from a class-conditional ImageNet model (Sec. 4.1) and images crawled from the internet. Interestingly, we observe that LDM-SR, trained only with a bicubicly downsampled conditioning as in [72], does not generalize well to images which do not follow this pre-processing. Hence, to obtain a superresolution model for a wide range of real world images, which can contain complex superpositions of camera noise, compression artifacts, blurr and interpolations, we replace the bicubic downsampling operation in LDM-SR with the degration pipeline from [105]. The BSR-degradation process is a degradation pipline which applies JPEG compressions noise, camera sensor noise, different image interpolations for downsampling, Gaussian blur kernels and Gaussian noise in a random order to an image. We found that using the bsr-degredation process with the original parameters as in [105] leads to a very strong degradation process. Since a more moderate degradation process seemed apppropiate for our application, we adapted the parameters of the bsr-degradation (our adapted degradation process can be found in our code base at `https://github.com/CompVis/latent-diffusion`). Fig. 18 illustrates the effectiveness of this approach by directly comparing *LDM-SR* with *LDM-BSR*. The latter produces images much sharper than the models confined to a fixed pre-processing, making it suitable for real-world applications. Further results of *LDM-BSR* are shown on LSUN-cows in Fig. 19.

[1] It is not possible to exactly match both architectures since the diffusion model operates in the pixel space

E. Implementation Details and Hyperparameters

E.1. Hyperparameters

We provide an overview of the hyperparameters of all trained *LDM* models in Tab. 12, Tab. 13, Tab. 14 and Tab. 15.

	CelebA-HQ 256×256	FFHQ 256×256	LSUN-Churches 256×256	LSUN-Bedrooms 256×256
f	4	4	8	4
z-shape	$64 \times 64 \times 3$	$64 \times 64 \times 3$	-	$64 \times 64 \times 3$
$\lvert\mathcal{Z}\rvert$	8192	8192	-	8192
Diffusion steps	1000	1000	1000	1000
Noise Schedule	linear	linear	linear	linear
N_{params}	274M	274M	294M	274M
Channels	224	224	192	224
Depth	2	2	2	2
Channel Multiplier	1,2,3,4	1,2,3,4	1,2,2,4,4	1,2,3,4
Attention resolutions	32, 16, 8	32, 16, 8	32, 16, 8, 4	32, 16, 8
Head Channels	32	32	24	32
Batch Size	48	42	96	48
Iterations*	410k	635k	500k	1.9M
Learning Rate	9.6e-5	8.4e-5	5.e-5	9.6e-5

Table 12. Hyperparameters for the unconditional *LDMs* producing the numbers shown in Tab. 1. All models trained on a single NVIDIA A100.

	LDM-1	*LDM-2*	*LDM-4*	*LDM-8*	*LDM-16*	*LDM-32*
z-shape	$256 \times 256 \times 3$	$128 \times 128 \times 2$	$64 \times 64 \times 3$	$32 \times 32 \times 4$	$16 \times 16 \times 8$	$88 \times 8 \times 32$
$\lvert\mathcal{Z}\rvert$	-	2048	8192	16384	16384	16384
Diffusion steps	1000	1000	1000	1000	1000	1000
Noise Schedule	linear	linear	linear	linear	linear	linear
Model Size	396M	391M	391M	395M	395M	395M
Channels	192	192	192	256	256	256
Depth	2	2	2	2	2	2
Channel Multiplier	1,1,2,2,4,4	1,2,2,4,4	1,2,3,5	1,2,4	1,2,4	1,2,4
Number of Heads	1	1	1	1	1	1
Batch Size	7	9	40	64	112	112
Iterations	2M	2M	2M	2M	2M	2M
Learning Rate	4.9e-5	6.3e-5	8e-5	6.4e-5	4.5e-5	4.5e-5
Conditioning	CA	CA	CA	CA	CA	CA
CA-resolutions	32, 16, 8	32, 16, 8	32, 16, 8	32, 16, 8	16, 8, 4	8, 4, 2
Embedding Dimension	512	512	512	512	512	512
Transformers Depth	1	1	1	1	1	1

Table 13. Hyperparameters for the conditional *LDMs* trained on the ImageNet dataset for the analysis in Sec. 4.1. All models trained on a single NVIDIA A100.

E.2. Implementation Details

E.2.1 Implementations of τ_θ for conditional *LDMs*

For the experiments on text-to-image and layout-to-image (Sec. 4.3.1) synthesis, we implement the conditioner τ_θ as an unmasked transformer which processes a tokenized version of the input y and produces an output $\zeta := \tau_\theta(y)$, where $\zeta \in \mathbb{R}^{M \times d_\tau}$. More specifically, the transformer is implemented from N transformer blocks consisting of global self-attention layers, layer-normalization and position-wise MLPs as follows[2]:

[2]adapted from https://github.com/lucidrains/x-transformers

	LDM-1	*LDM-2*	*LDM-4*	*LDM-8*	*LDM-16*	*LDM-32*
z-shape	$256 \times 256 \times 3$	$128 \times 128 \times 2$	$64 \times 64 \times 3$	$32 \times 32 \times 4$	$16 \times 16 \times 8$	$88 \times 8 \times 32$
$\lvert\mathcal{Z}\rvert$	-	2048	8192	16384	16384	16384
Diffusion steps	1000	1000	1000	1000	1000	1000
Noise Schedule	linear	linear	linear	linear	linear	linear
Model Size	270M	265M	274M	258M	260M	258M
Channels	192	192	224	256	256	256
Depth	2	2	2	2	2	2
Channel Multiplier	1,1,2,2,4,4	1,2,2,4,4	1,2,3,4	1,2,4	1,2,4	1,2,4
Attention resolutions	32, 16, 8	32, 16, 8	32, 16, 8	32, 16, 8	16, 8, 4	8, 4, 2
Head Channels	32	32	32	32	32	32
Batch Size	9	11	48	96	128	128
Iterations*	500k	500k	500k	500k	500k	500k
Learning Rate	9e-5	1.1e-4	9.6e-5	9.6e-5	1.3e-4	1.3e-4

Table 14. Hyperparameters for the unconditional *LDMs* trained on the CelebA dataset for the analysis in Fig. 7. All models trained on a single NVIDIA A100. *: All models are trained for 500k iterations. If converging earlier, we used the best checkpoint for assessing the provided FID scores.

Task	Text-to-Image	Layout-to-Image		Class-Label-to-Image	Super Resolution	Inpainting	Semantic-Map-to-Image
Dataset	LAION	OpenImages	COCO	ImageNet	ImageNet	Places	Landscapes
f	8	4	8	4	4	4	8
z-shape	$32 \times 32 \times 4$	$64 \times 64 \times 3$	$32 \times 32 \times 4$	$64 \times 64 \times 3$	$64 \times 64 \times 3$	$64 \times 64 \times 3$	$32 \times 32 \times 4$
$\lvert\mathcal{Z}\rvert$	-	8192	16384	8192	8192	8192	16384
Diffusion steps	1000	1000	1000	1000	1000	1000	1000
Noise Schedule	linear	linear	linear	linear	linear	linear	linear
Model Size	1.45B	306M	345M	395M	169M	215M	215M
Channels	320	128	192	192	160	128	128
Depth	2	2	2	2	2	2	2
Channel Multiplier	1,2,4,4	1,2,3,4	1,2,4	1,2,3,5	1,2,2,4	1,4,8	1,4,8
Number of Heads	8	1	1	1	1	1	1
Dropout	-	-	0.1	-	-	-	-
Batch Size	680	24	48	1200	64	128	48
Iterations	390K	4.4M	170K	178K	860K	360K	360K
Learning Rate	1.0e-4	4.8e-5	4.8e-5	1.0e-4	6.4e-5	1.0e-6	4.8e-5
Conditioning	CA	CA	CA	CA	concat	concat	concat
(C)A-resolutions	32, 16, 8	32, 16, 8	32, 16, 8	32, 16, 8	-	-	-
Embedding Dimension	1280	512	512	512	-	-	-
Transformer Depth	1	3	2	1	-	-	-

Table 15. Hyperparameters for the conditional *LDMs* from Sec. 4. All models trained on a single NVIDIA A100 except for the inpainting model which was trained on eight V100.

$$\zeta \leftarrow \text{TokEmb}(y) + \text{PosEmb(y)} \tag{18}$$

$$\text{for } i = 1, \dots, N:$$

$$\zeta_1 \leftarrow \text{LayerNorm}(\zeta) \tag{19}$$

$$\zeta_2 \leftarrow \text{MultiHeadSelfAttention}(\zeta_1) + \zeta \tag{20}$$

$$\zeta_3 \leftarrow \text{LayerNorm}(\zeta_2) \tag{21}$$

$$\zeta \leftarrow \text{MLP}(\zeta_3) + \zeta_2 \tag{22}$$

$$\zeta \leftarrow \text{LayerNorm}(\zeta) \tag{23}$$

$$\tag{24}$$

With ζ available, the conditioning is mapped into the UNet via the cross-attention mechanism as depicted in Fig. 3. We modify the "ablated UNet" [15] architecture and replace the self-attention layer with a shallow (unmasked) transformer consisting of T blocks with alternating layers of (i) self-attention, (ii) a position-wise MLP and (iii) a cross-attention layer;

see Tab. 16. Note that without (ii) and (iii), this architecture is equivalent to the "ablated UNet".

While it would be possible to increase the representational power of τ_θ by additionally conditioning on the time step t, we do not pursue this choice as it reduces the speed of inference. We leave a more detailed analysis of this modification to future work.

For the text-to-image model, we rely on a publicly available[3] tokenizer [99]. The layout-to-image model discretizes the spatial locations of the bounding boxes and encodes each box as a (l, b, c)-tuple, where l denotes the (discrete) top-left and b the bottom-right position. Class information is contained in c.
See Tab. 17 for the hyperparameters of τ_θ and Tab. 13 for those of the UNet for both of the above tasks.

Note that the class-conditional model as described in Sec. 4.1 is also implemented via cross-attention, where τ_θ is a single learnable embedding layer with a dimensionality of 512, mapping classes y to $\zeta \in \mathbb{R}^{1\times 512}$.

input	$\mathbb{R}^{h\times w\times c}$
LayerNorm	$\mathbb{R}^{h\times w\times c}$
Conv1x1	$\mathbb{R}^{h\times w\times d\cdot n_h}$
Reshape	$\mathbb{R}^{h\cdot w\times d\cdot n_h}$
$\times T$ { SelfAttention	$\mathbb{R}^{h\cdot w\times d\cdot n_h}$
$\times T$ { MLP	$\mathbb{R}^{h\cdot w\times d\cdot n_h}$
$\times T$ { CrossAttention	$\mathbb{R}^{h\cdot w\times d\cdot n_h}$
Reshape	$\mathbb{R}^{h\times w\times d\cdot n_h}$
Conv1x1	$\mathbb{R}^{h\times w\times c}$

Table 16. Architecture of a transformer block as described in Sec. E.2.1, replacing the self-attention layer of the standard "ablated UNet" architecture [15]. Here, n_h denotes the number of attention heads and d the dimensionality per head.

	Text-to-Image	**Layout-to-Image**
seq-length	77	92
depth N	32	16
dim	1280	512

Table 17. Hyperparameters for the experiments with transformer encoders in Sec. 4.3.

E.2.2 Inpainting

For our experiments on image-inpainting in Sec. 4.5, we used the code of [88] to generate synthetic masks. We use a fixed set of 2k validation and 30k testing samples from Places [108]. During training, we use random crops of size 256×256 and evaluate on crops of size 512×512. This follows the training and testing protocol in [88] and reproduces their reported metrics (see † in Tab. 7). We include additional qualitative results of *LDM-4, w/ attn* in Fig. 21 and of *LDM-4, w/o attn, big, w/ ft* in Fig. 22.

E.3. Evaluation Details

This section provides additional details on evaluation for the experiments shown in Sec. 4.

E.3.1 Quantitative Results in Unconditional and Class-Conditional Image Synthesis

We follow common practice and estimate the statistics for calculating the FID-, Precision- and Recall-scores [29,50] shown in Tab. 1 and 10 based on 50k samples from our models and the entire training set of each of the shown datasets. For calculating FID scores we use the `torch-fidelity` package [60]. However, since different data processing pipelines might lead to different results [64], we also evaluate our models with the script provided by Dhariwal and Nichol [15]. We find that results

[3] https://huggingface.co/transformers/model_doc/bert.html#berttokenizerfast

mainly coincide, except for the ImageNet and LSUN-Bedrooms datasets, where we notice slightly varying scores of 7.76 (`torch-fidelity`) vs. 7.77 (Nichol and Dhariwal) and 2.95 vs 3.0. For the future we emphasize the importance of a unified procedure for sample quality assessment. Precision and Recall are also computed by using the script provided by Nichol and Dhariwal.

E.3.2 Text-to-Image Synthesis

Following the evaluation protocol of [66] we compute FID and Inception Score for the Text-to-Image models from Tab. 2 by comparing generated samples with 30000 samples from the validation set of the MS-COCO dataset [51]. FID and Inception Scores are computed with `torch-fidelity`.

E.3.3 Layout-to-Image Synthesis

For assessing the sample quality of our Layout-to-Image models from Tab. 9 on the COCO dataset, we follow common practice [37, 87, 89] and compute FID scores the 2048 unaugmented examples of the COCO Segmentation Challenge split. To obtain better comparability, we use the exact same samples as in [37]. For the OpenImages dataset we similarly follow their protocol and use 2048 center-cropped test images from the validation set.

E.3.4 Super Resolution

We evaluate the super-resolution models on ImageNet following the pipeline suggested in [72], *i.e*. images with a shorter size less than 256 px are removed (both for training and evaluation). On ImageNet, the low-resolution images are produced using bicubic interpolation with anti-aliasing. FIDs are evaluated using `torch-fidelity` [60], and we produce samples on the validation split. For FID scores, we additionally compare to reference features computed on the train split, see Tab. 5 and Tab. 11.

E.3.5 Efficiency Analysis

For efficiency reasons we compute the sample quality metrics plotted in Fig. 6, 17 and 7 based on 5k samples. Therefore, the results might vary from those shown in Tab. 1 and 10. All models have a comparable number of parameters as provided in Tab. 13 and 14. We maximize the learning rates of the individual models such that they still train stably. Therefore, the learning rates slightly vary between different runs *cf*. Tab. 13 and 14.

E.3.6 User Study

For the results of the user study presented in Tab. 4 we followed the protocoll of [72] and and use the 2-alternative force-choice paradigm to assess human preference scores for two distinct tasks. In Task-1 subjects were shown a low resolution/masked image between the corresponding ground truth high resolution/unmasked version and a synthesized image, which was generated by using the middle image as conditioning. For SuperResolution subjects were asked: *'Which of the two images is a better high quality version of the low resolution image in the middle?'*. For Inpainting we asked *'Which of the two images contains more realistic inpainted regions of the image in the middle?'*. In Task-2, humans were similarly shown the low-res/masked version and asked for preference between two corresponding images generated by the two competing methods. As in [72] humans viewed the images for 3 seconds before responding.

F. Computational Requirements

Method	Generator Compute	Classifier Compute	Overall Compute	Inference Throughput*	N_{params}	FID↓	IS↑	Precision↑	Recall↑
LSUN Churches 256^2									
StyleGAN2 [42]†	64	-	64	-	59M	3.86	-	-	-
LDM-8 (ours, 100 steps, 410K)	18	-	18	6.80	256M	4.02	-	0.64	0.52
LSUN Bedrooms 256^2									
ADM [15]† (1000 steps)	232	-	232	0.03	552M	1.9	-	0.66	0.51
LDM-4 (ours, 200 steps, 1.9M)	60	-	55	1.07	274M	2.95	-	0.66	0.48
CelebA-HQ 256^2									
LDM-4 (ours, 500 steps, 410K)	14.4	-	14.4	0.43	274M	5.11	-	0.72	0.49
FFHQ 256^2									
StyleGAN2 [42]	32.13‡	-	32.13†	-	59M	3.8	-	-	-
LDM-4 (ours, 200 steps, 635K)	26	-	26	1.07	274M	4.98	-	0.73	0.50
ImageNet 256^2									
VQGAN-f-4 (ours, first stage)	29	-	29	-	55M	0.58††	-	-	-
VQGAN-f-8 (ours, first stage)	66	-	66	-	68M	1.14††	-	-	-
BigGAN-deep [3]†	128-256		128-256	-	340M	6.95	$203.6_{\pm 2.6}$	0.87	0.28
ADM [15] (250 steps) †	916	-	916	0.12	554M	10.94	100.98	0.69	0.63
ADM-G [15] (25 steps) †	916	46	962	0.7	608M	5.58	-	0.81	0.49
ADM-G [15] (250 steps)†	916	46	962	0.07	608M	4.59	186.7	0.82	0.52
ADM-G,ADM-U [15] (250 steps)†	329	30	349	n/a	n/a	3.85	221.72	0.84	0.53
LDM-8-G (ours, 100, 2.9M)	79	12	91	1.93	506M	8.11	$190.4_{\pm 2.6}$	0.83	0.36
LDM-8 (ours, 200 ddim steps 2.9M, batch size 64)	79	-	79	1.9	395M	17.41	72.92	0.65	0.62
LDM-4 (ours, 250 ddim steps 178K, batch size 1200)	271	-	271	0.7	400M	10.56	$103.49_{\pm 1.24}$	0.71	0.62
LDM-4-G (ours, 250 ddim steps 178K, batch size 1200, classifier-free guidance [32] scale 1.25)	271	-	271	0.4	400M	3.95	$178.22_{\pm 2.43}$	0.81	0.55
LDM-4-G (ours, 250 ddim steps 178K, batch size 1200, classifier-free guidance [32] scale 1.5)	271	-	271	0.4	400M	3.60	$247.67_{\pm 5.59}$	0.87	0.48

Table 18. Comparing compute requirements during training and inference throughput with state-of-the-art generative models. Compute during training in V100-days, numbers of competing methods taken from [15] unless stated differently;*: Throughput measured in samples/sec on a single NVIDIA A100;†: Numbers taken from [15] ;‡: Assumed to be trained on 25M train examples; ††: R-FID vs. ImageNet validation set

In Tab 18 we provide a more detailed analysis on our used compute ressources and compare our best performing models on the CelebA-HQ, FFHQ, LSUN and ImageNet datasets with the recent state of the art models by using their provided numbers, *cf*. [15]. As they report their used compute in V100 days and we train all our models on a single NVIDIA A100 GPU, we convert the A100 days to V100 days by assuming a $\times 2.2$ speedup of A100 vs V100 [74][4]. To assess sample quality, we additionally report FID scores on the reported datasets. We closely reach the performance of state of the art methods as StyleGAN2 [42] and ADM [15] while significantly reducing the required compute resources.

[4]This factor corresponds to the speedup of the A100 over the V100 for a U-Net, as defined in Fig. 1 in [74]

G. Details on Autoencoder Models

We train all our autoencoder models in an adversarial manner following [23], such that a patch-based discriminator D_ψ is optimized to differentiate original images from reconstructions $\mathcal{D}(\mathcal{E}(x))$. To avoid arbitrarily scaled latent spaces, we regularize the latent z to be zero centered and obtain small variance by introducing an regularizing loss term L_{reg}.
We investigate two different regularization methods: (i) a low-weighted Kullback-Leibler-term between $q_\mathcal{E}(z|x) = \mathcal{N}(z; \mathcal{E}_\mu, \mathcal{E}_{\sigma^2})$ and a standard normal distribution $\mathcal{N}(z; 0, 1)$ as in a standard variational autoencoder [46, 69], and, (ii) regularizing the latent space with a vector quantization layer by learning a codebook of $|\mathcal{Z}|$ different exemplars [96].
To obtain high-fidelity reconstructions we only use a very small regularization for both scenarios, *i.e.* we either weight the $\mathbb{KL}$ term by a factor $\sim 10^{-6}$ or choose a high codebook dimensionality $|\mathcal{Z}|$.

The full objective to train the autoencoding model $(\mathcal{E}, \mathcal{D})$ reads:

$$L_{\text{Autoencoder}} = \min_{\mathcal{E},\mathcal{D}} \max_{\psi} \Big(L_{rec}(x, \mathcal{D}(\mathcal{E}(x))) - L_{adv}(\mathcal{D}(\mathcal{E}(x))) + \log D_\psi(x) + L_{reg}(x; \mathcal{E}, \mathcal{D}) \Big) \tag{25}$$

DM Training in Latent Space Note that for training diffusion models on the learned latent space, we again distinguish two cases when learning $p(z)$ or $p(z|y)$ (Sec. 4.3): (i) For a KL-regularized latent space, we sample $z = \mathcal{E}_\mu(x) + \mathcal{E}_\sigma(x) \cdot \varepsilon =: \mathcal{E}(x)$, where $\varepsilon \sim \mathcal{N}(0, 1)$. When rescaling the latent, we estimate the component-wise variance

$$\hat{\sigma}^2 = \frac{1}{bchw} \sum_{b,c,h,w} (z^{b,c,h,w} - \hat{\mu})^2$$

from the first batch in the data, where $\hat{\mu} = \frac{1}{bchw} \sum_{b,c,h,w} z^{b,c,h,w}$. The output of $\mathcal{E}$ is scaled such that the rescaled latent has unit standard deviation, *i.e.* $z \leftarrow \frac{z}{\hat{\sigma}} = \frac{\mathcal{E}(x)}{\hat{\sigma}}$. (ii) For a VQ-regularized latent space, we extract z *before* the quantization layer and absorb the quantization operation into the decoder, *i.e.* it can be interpreted as the first layer of $\mathcal{D}$.

H. Additional Qualitative Results

Finally, we provide additional qualitative results for our landscapes model (Fig. 12, 23, 24 and 25), our class-conditional ImageNet model (Fig. 26 - 27) and our unconditional models for the CelebA-HQ, FFHQ and LSUN datasets (Fig. 28 - 31). Similar as for the inpainting model in Sec. 4.5 we also fine-tuned the semantic landscapes model from Sec. 4.3.2 directly on 512^2 images and depict qualitative results in Fig. 12 and Fig. 23. For our those models trained on comparably small datasets, we additionally show nearest neighbors in VGG [79] feature space for samples from our models in Fig. 32 - 34.

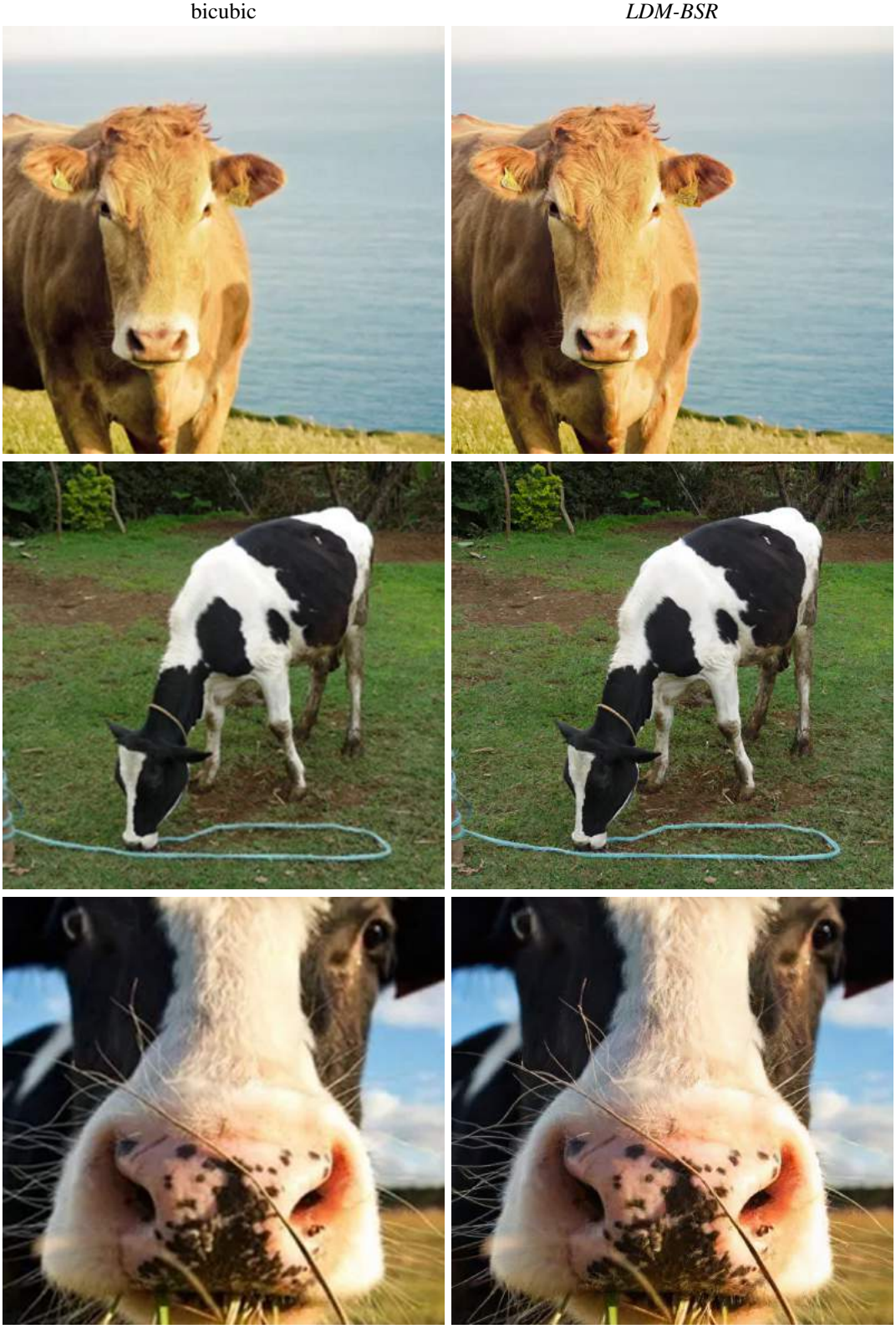

Figure 19. *LDM-BSR* generalizes to arbitrary inputs and can be used as a general-purpose upsampler, upscaling samples from the LSUN-Cows dataset to 1024^2 resolution.

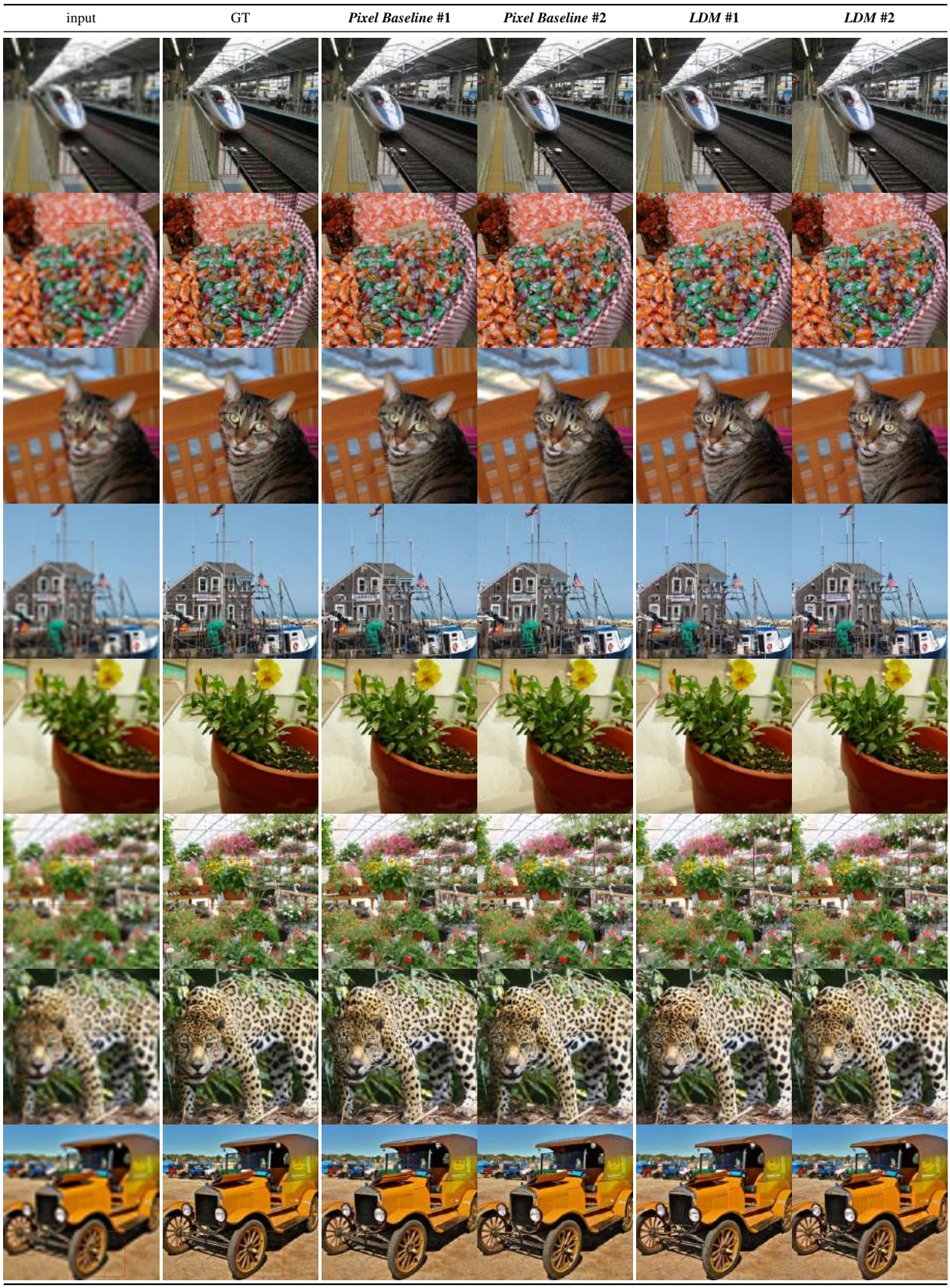

Figure 20. Qualitative superresolution comparison of two random samples between LDM-SR and baseline-diffusionmodel in Pixelspace. Evaluated on imagenet validation-set after same amount of training steps.

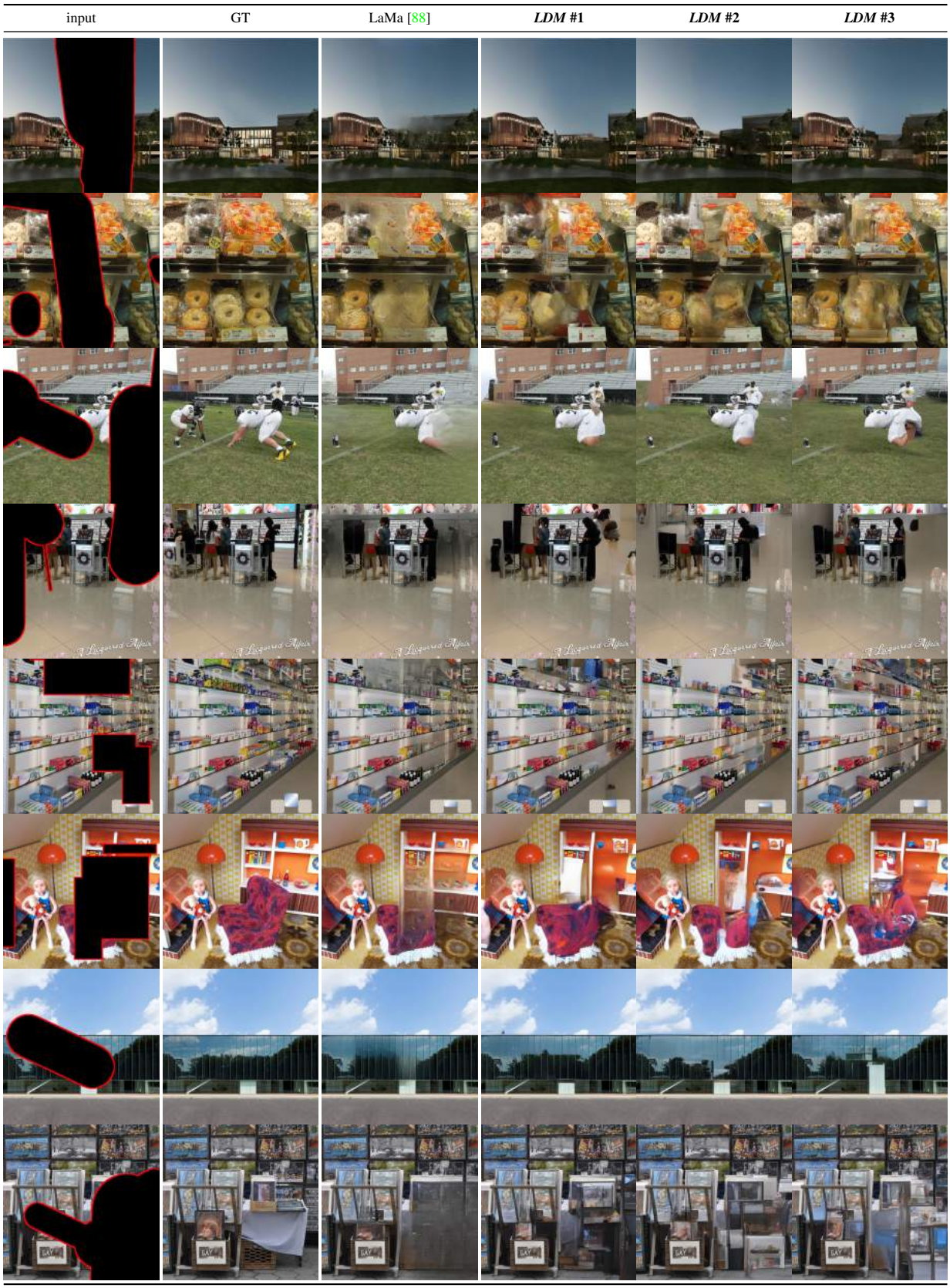

Figure 21. Qualitative results on image inpainting. In contrast to [88], our generative approach enables generation of multiple diverse samples for a given input.

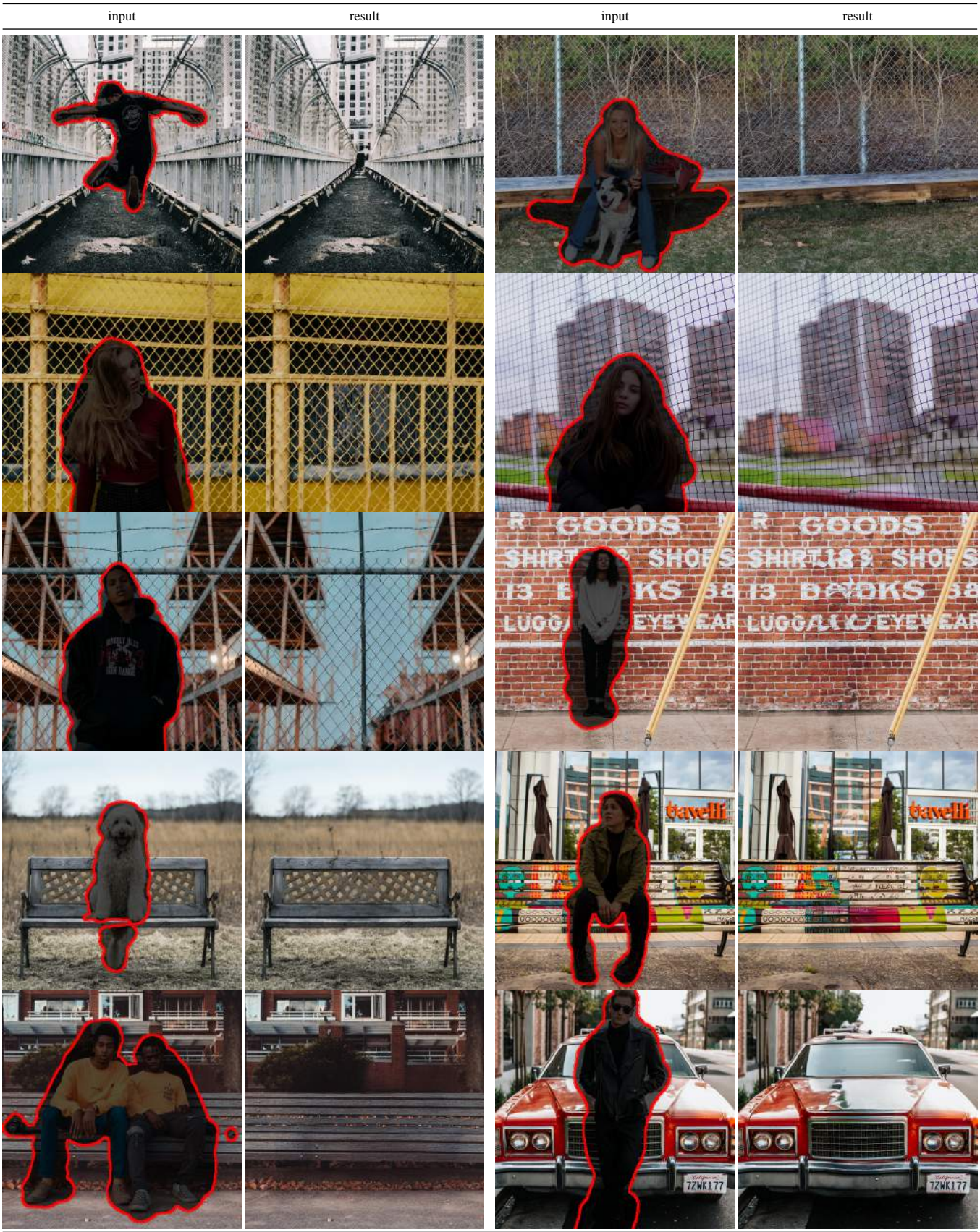

Figure 22. More qualitative results on object removal as in Fig. 11.

Semantic Synthesis on Flickr-Landscapes [23] (512^2 finetuning)

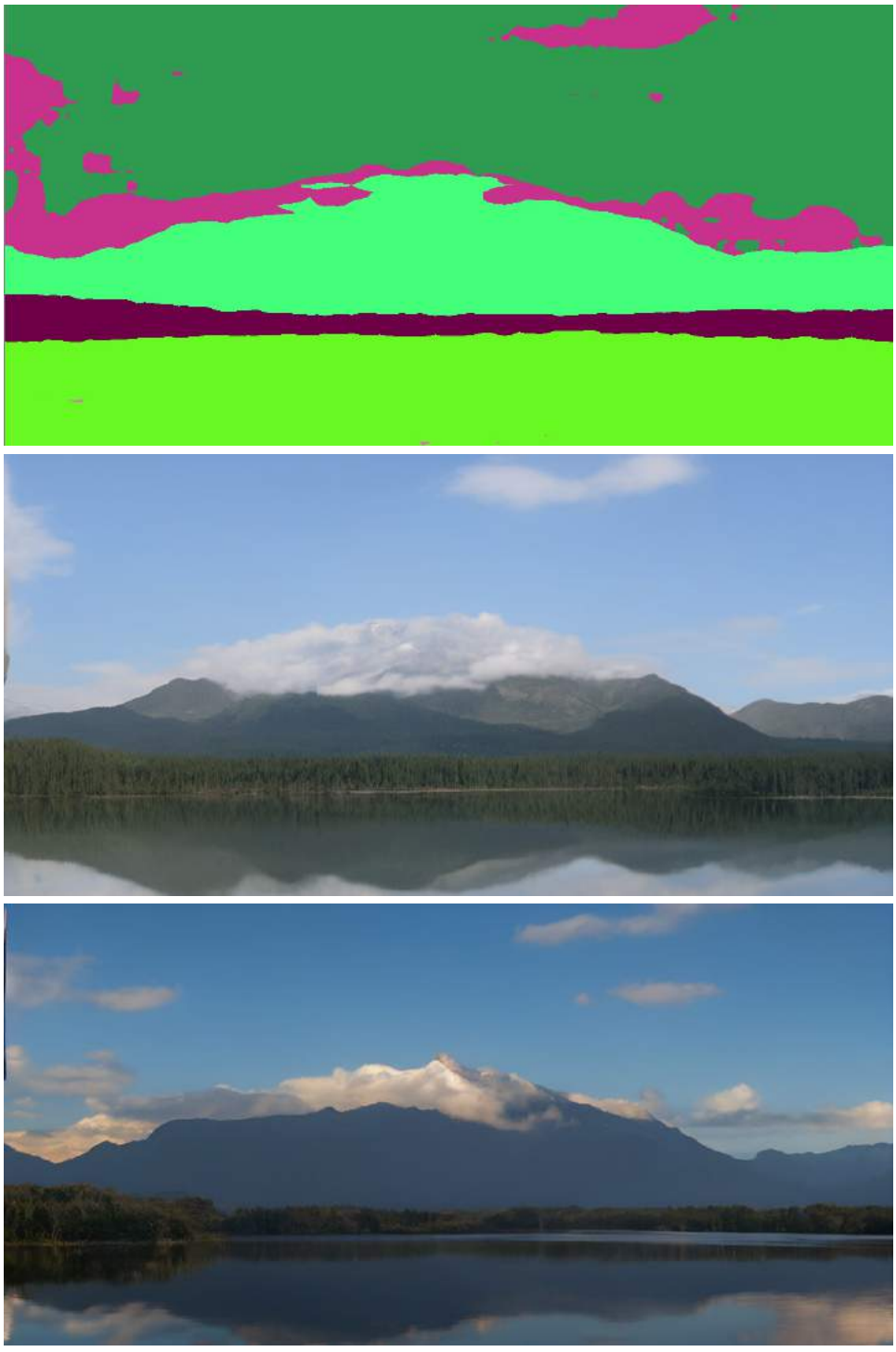

Figure 23. Convolutional samples from the semantic landscapes model as in Sec. 4.3.2, finetuned on 512^2 images.

Figure 24. A *LDM* trained on 256^2 resolution can generalize to larger resolution for spatially conditioned tasks such as semantic synthesis of landscape images. See Sec. 4.3.2.

Semantic Synthesis on Flickr-Landscapes [23]

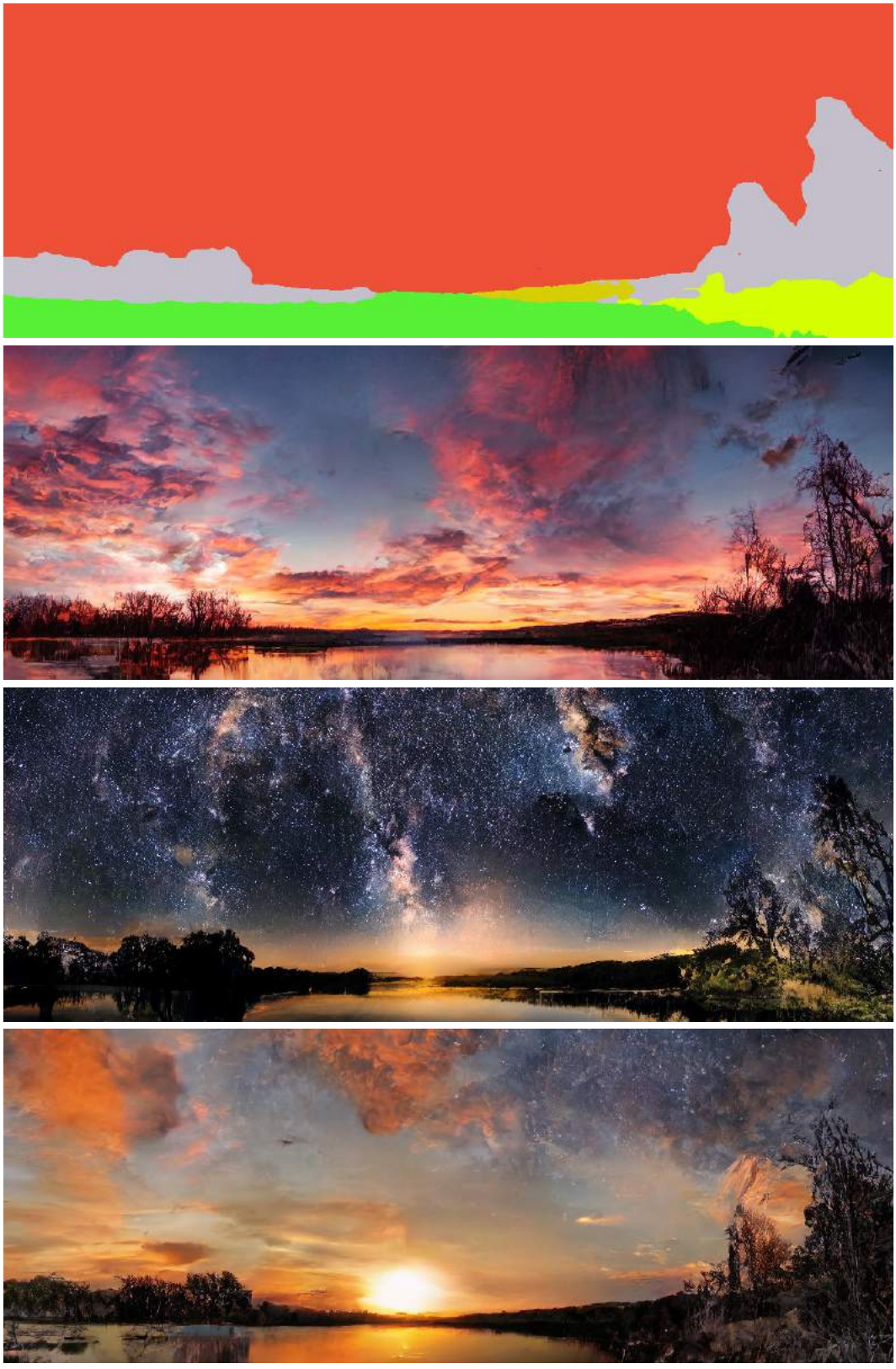

Figure 25. When provided a semantic map as conditioning, our *LDMs* generalize to substantially larger resolutions than those seen during training. Although this model was trained on inputs of size 256^2 it can be used to create high-resolution samples as the ones shown here, which are of resolution 1024×384.

Random class conditional samples on the ImageNet dataset

Figure 26. Random samples from *LDM-4* trained on the ImageNet dataset. Sampled with classifier-free guidance [32] scale $s = 5.0$ and 200 DDIM steps with $\eta = 1.0$.

Random class conditional samples on the ImageNet dataset

Figure 27. Random samples from *LDM-4* trained on the ImageNet dataset. Sampled with classifier-free guidance [32] scale $s = 3.0$ and 200 DDIM steps with $\eta = 1.0$.

Random samples on the CelebA-HQ dataset

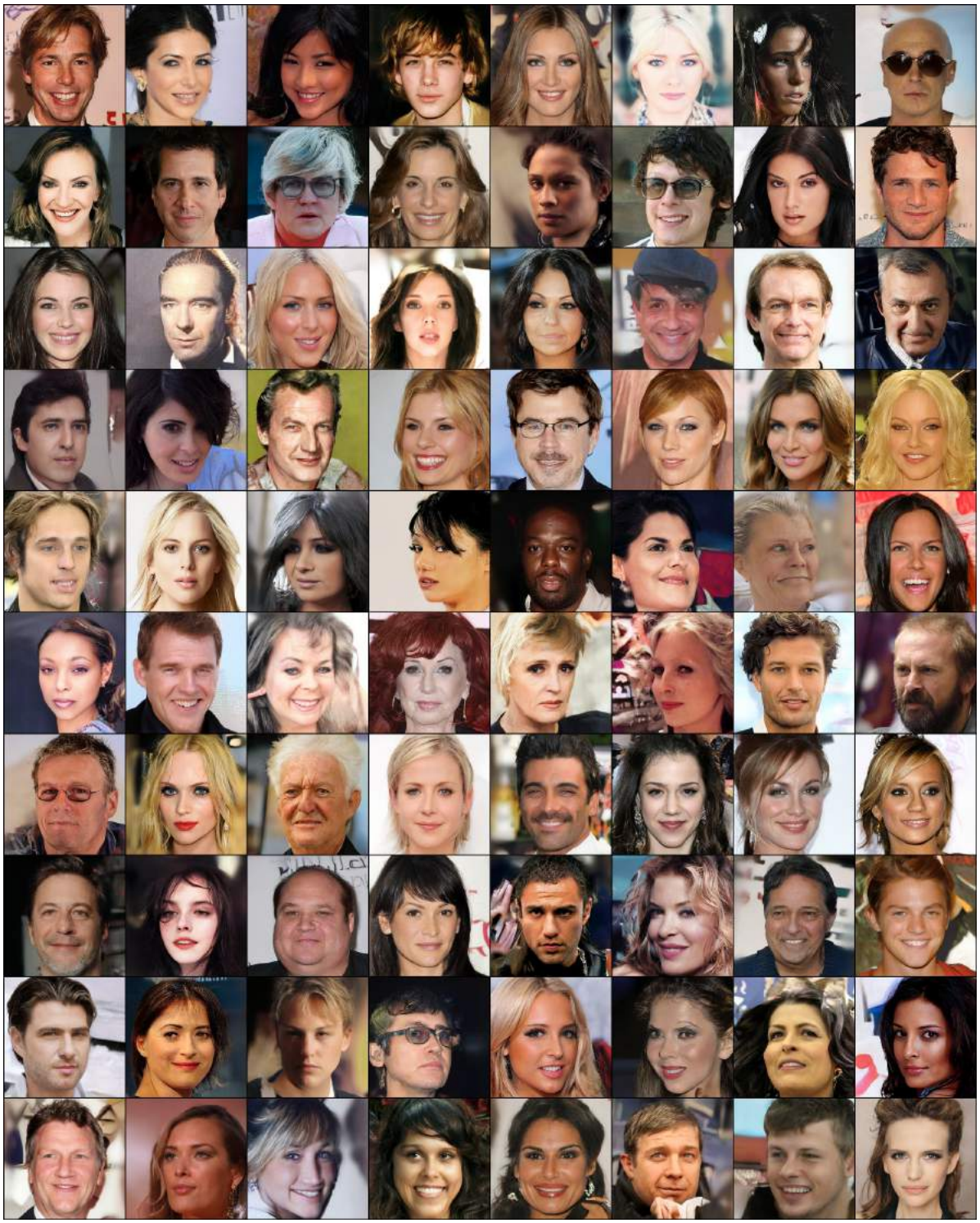

Figure 28. Random samples of our best performing model *LDM-4* on the CelebA-HQ dataset. Sampled with 500 DDIM steps and $\eta = 0$ (FID = 5.15).

Random samples on the FFHQ dataset

Figure 29. Random samples of our best performing model *LDM-4* on the FFHQ dataset. Sampled with 200 DDIM steps and $\eta = 1$ (FID = 4.98).

Random samples on the LSUN-Churches dataset

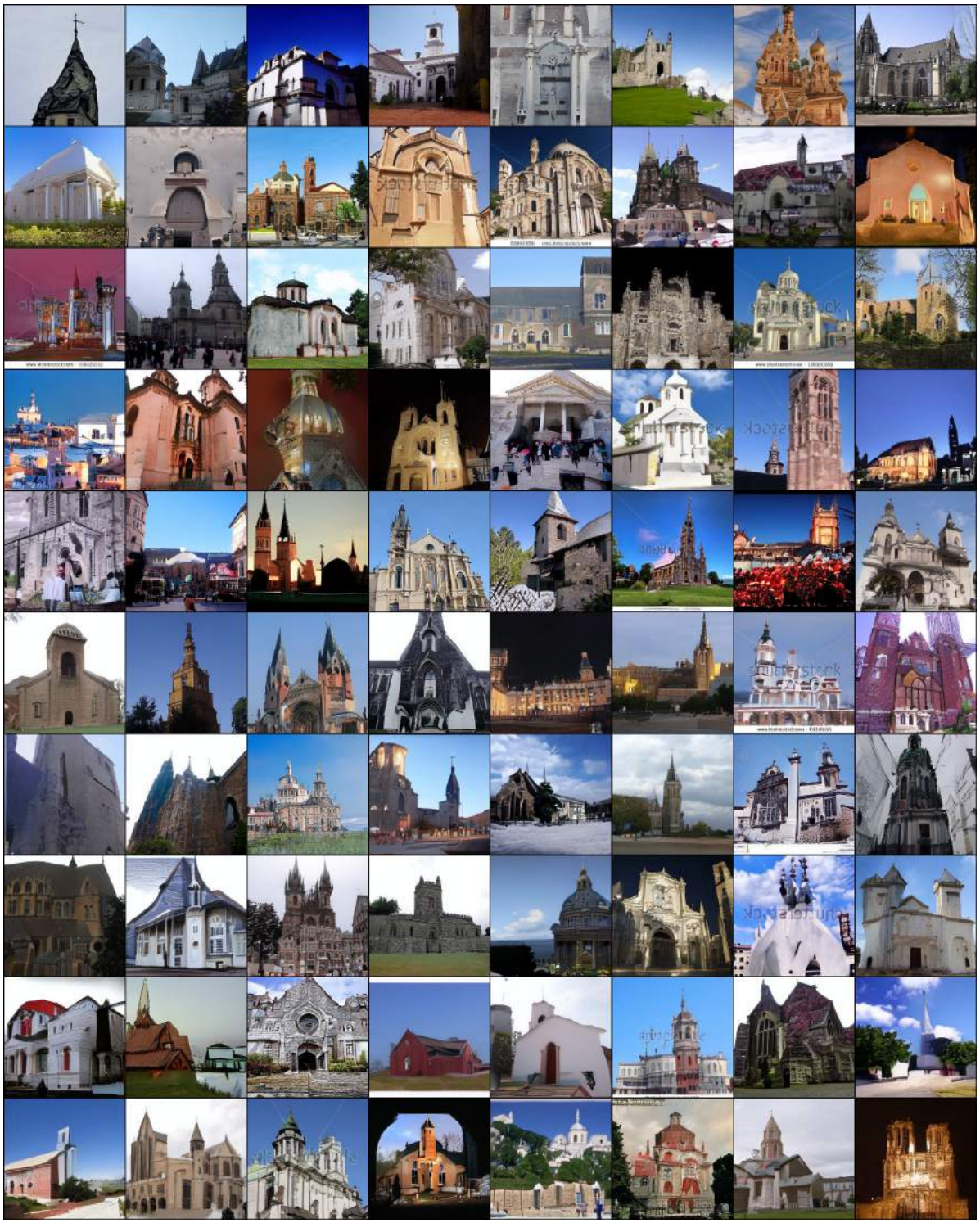

Figure 30. Random samples of our best performing model *LDM-8* on the LSUN-Churches dataset. Sampled with 200 DDIM steps and $\eta = 0$ (FID = 4.48).

Random samples on the LSUN-Bedrooms dataset

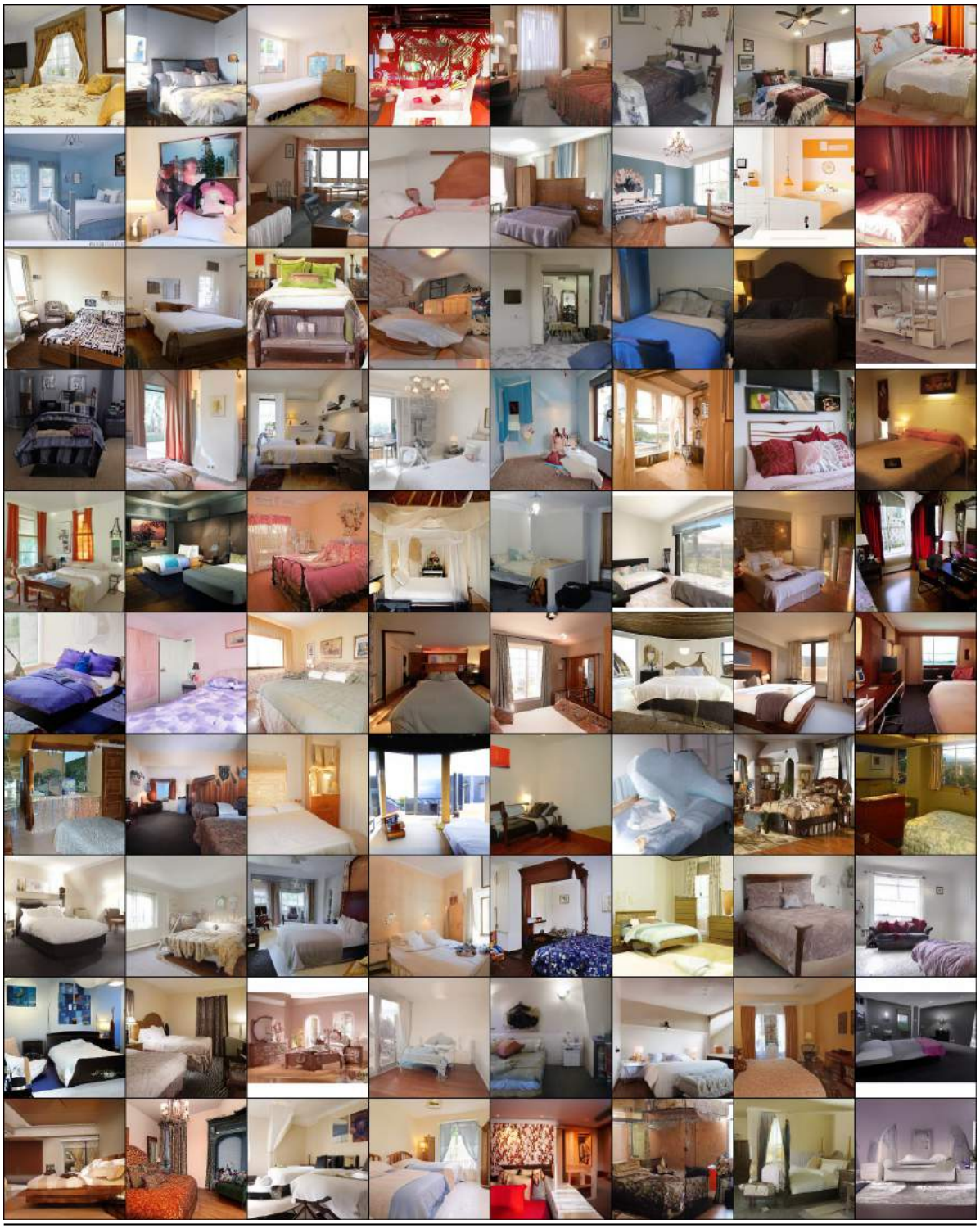

Figure 31. Random samples of our best performing model *LDM-4* on the LSUN-Bedrooms dataset. Sampled with 200 DDIM steps and $\eta = 1$ (FID = 2.95).

Nearest Neighbors on the CelebA-HQ dataset

Figure 32. Nearest neighbors of our best CelebA-HQ model, computed in the feature space of a VGG-16 [79]. The leftmost sample is from our model. The remaining samples in each row are its 10 nearest neighbors.

Nearest Neighbors on the FFHQ dataset

Figure 33. Nearest neighbors of our best FFHQ model, computed in the feature space of a VGG-16 [79]. The leftmost sample is from our model. The remaining samples in each row are its 10 nearest neighbors.

Nearest Neighbors on the LSUN-Churches dataset

Figure 34. Nearest neighbors of our best LSUN-Churches model, computed in the feature space of a VGG-16 [79]. The leftmost sample is from our model. The remaining samples in each row are its 10 nearest neighbors.