Latent Diffusion Models with Masked AutoEncoders

Supplementary Material

A. Autoencoders

AutoEncoders (AEs) [33] are optimized by solely relying on a reconstruction loss, compressing the input into a more compact latent space and reconstruct it back. For its objective, Mean Squared Error (MSE) is commonly used:

$$\mathcal{L}_{\text{AE}} = \mathbb{E}_{p_{\text{data}}(\boldsymbol{x})}\Big[\|\boldsymbol{x} - g_\theta(f_\phi(\boldsymbol{x}))\|^2\Big], \tag{3}$$

where $p_{\text{data}}(\boldsymbol{x})$ is the data distribution. The encoder $f_\phi(\boldsymbol{x})$ maps the input into the latent $\boldsymbol{z}$, and the decoder $g_\theta(\boldsymbol{z})$ recovers the latent $\boldsymbol{z}$ back to the input. ϕ and θ are the parameters of the encoder and decoder networks, respectively.

Denoising AutoEncoders (DAEs) [44] is trained to recover the original input from the noised one for the robust latent features with improved generalization. The objective is

$$\mathcal{L}_{\text{DAE}} = \mathbb{E}_{p_{\text{data}}(\boldsymbol{x}),p_c(\tilde{\boldsymbol{x}}|\boldsymbol{x})}\Big[\|\boldsymbol{x} - g_\theta(f_\phi(\tilde{\boldsymbol{x}}))\|^2\Big], \tag{4}$$

where $p_c(\tilde{\boldsymbol{x}}|\boldsymbol{x})$ represents a corrupted data distribution.

Variational AutoEncoders (VAEs) [20] adopt a probabilistic framework by encoding the input data into a latent variable distribution instead of a fixed vector, facilitating sampling. VAEs are trained using the Evidence Lower Bound Loss (ELBO), which combines a reconstruction loss with a prior matching term, *i.e.*, KL-divergence to regularize the latent space towards a Gaussian distribution:

$$\begin{aligned}\mathcal{L}_{\text{VAE}} = \mathbb{E}_{p_{\text{data}}(\boldsymbol{x})}\Big[&\underbrace{\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[-\log p_\theta(\boldsymbol{x}|\boldsymbol{z})]}_{\text{reconstruction}} \\ &+ \lambda_{\text{KL}} \cdot \underbrace{\text{D}_{\text{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x})\|p(\boldsymbol{z}))}_{\text{prior matching}}\Big],\end{aligned} \tag{5}$$

where $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ is the distribution of latent $\boldsymbol{z}$ encoded from the input $\boldsymbol{x} \sim p_{\text{data}}$ and $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ is the distribution of reconstructed $\boldsymbol{x}$ from $\boldsymbol{z}$. KL-divergence is scaled by λ_{KL}.

StableDiffusion VAEs (SD-VAEs) [11, 32] are built upon the VQGAN [10], the autoencoder adopted in traditional LDMs [32]. Following the VQGAN, SD-VAEs integrate an additional adversarial network and train with perceptual loss (LPIPS) [46] for an improved perceptual quality in the compressed space. Unlike VQGAN, however, SD-VAEs omit the quantization layer entirely; instead, it simply adopts continuous features. In this paper, we follow the SD-VAEs settings from StableDiffusion3 [11], unless noted otherwise.

The overall objective of SD-VAEs is

$$\begin{aligned}\mathcal{L}_{\text{SD-VAE}} = \mathbb{E}_{p_{\text{data}}(\boldsymbol{x})}\Big[&\underbrace{\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[-\log p_\theta(\boldsymbol{x}|\boldsymbol{z})]}_{\text{reconstruction}} \\ &+ \lambda_{\text{KL}} \cdot \underbrace{\text{D}_{\text{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x})|p(\boldsymbol{z}))}_{\text{prior matching}}\Big] \\ &+ \lambda_{\text{D}} \cdot \underbrace{\big(\mathbb{E}_{p_{\text{data}}}[\log D(\boldsymbol{x})] + \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log(1 - D(p_\theta(\boldsymbol{x}|\boldsymbol{z})))]\big)}_{\text{adversarial loss}} \\ &+ \lambda_{\text{LPIPS}} \cdot \underbrace{\mathbb{E}_{p_{\text{data}}(\boldsymbol{x}),q_\phi(\boldsymbol{z}|\boldsymbol{x}),p_\theta(\hat{\boldsymbol{x}}|\boldsymbol{z})}\Big[\sum_l w_l \|\psi_l(\boldsymbol{x}) - \psi_l(\hat{\boldsymbol{x}})\|_2^2\Big]}_{\text{LPIPS loss}}.\end{aligned} \tag{6}$$

The third term corresponds to the adversarial loss scaled by λ_{D}, where $D(x)$ denotes the discriminator function. The last LPIPS loss term incorporates the feature extraction function ψ_l up to l layers of a pre-trained network, with w_l as the layer-specific weight, scaled by λ_{LPIPS}.

Masked AutoEncoders (MAEs) [13] were originally proposed as a self-supervised learning method for representation learning based on Vision Transformers (ViTs) [9]. The MAEs encoder $f_\phi(\boldsymbol{x}_v)$ maps a masked-out image $\boldsymbol{x}_v \sim p(\boldsymbol{x}_v|\boldsymbol{x})$ into a latent $\boldsymbol{z}$, and its decoder $g_\theta(\boldsymbol{z})$ reconstructs the original input $\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})$ from $\boldsymbol{z}$ along with learnable mask tokens. Since MSE loss applies only to mask tokens, the actual loss acts more as a prediction loss than a reconstruction loss. Omitting the mask tokens for simplicity, the objective can be expressed as

$$\mathcal{L}_{\text{MAE}} = \mathbb{E}_{p_{\text{data}}(\boldsymbol{x}),p(\boldsymbol{x}_v|\boldsymbol{x})}\Big[\|\text{M} \odot (\boldsymbol{x} - g_\theta(f_\phi(\boldsymbol{x}_v)))\|^2\Big], \tag{7}$$

where M is a fixed-ratio random binary mask.

B. Implementation Details

B.1. Experimental Setup for Autoencoders

Datasets. We use ImageNet-1K [7] training set for training autoencoders, and evaluate them on the ImageNet-1K test set, ADE20k [47] test set, and CelebAMask-HQ [23]. ADE20K and CelebAMask-HQ are segmentation datasets containing ground truth masks in pixel-level.

Implementation Details. Our VMAE adopts a symmetric ViT-based encoder–decoder architecture. The encoder partitions an input image $\mathbf{x} \in \mathbb{R}^{H\times W\times 3}$ into patch tokens $\mathbf{x}_i \in \mathbb{R}^{h\times w\times d}$ with patch size $(h, w) = (32, 32)$ and embedding dimension $d = 192$. A masking ratio of 0.6 is applied, and the visible patches are processed by 12 Transformer layers. The decoder prepends learnable mask tokens to the

encoded sequence and reconstructs the full image using another 12 Transformer layers. We impose a KL-divergence loss on the latent representations and employ both pixel-wise reconstruction loss and perceptual loss on the decoder outputs.

We train all autoencoders using the optimal hyperparameters for each model on 8 NVIDIA A100 GPUs (40GB). The base learning rate is set to 10^{-5} for convolution-based models (AEs, DAEs, VAEs, and SD-VAEs), while 10^{-4} for our VMAEs. Global batch size is set to 2048 except for the SD-VAEs, which require smaller batch size (256) for stable training of the adversarial network, following the implementation in VQGAN [10].

B.2. Experimental Setup for Image Generation

Datasets. We select the following datasets to test various aspects of generation performance. For unconditional image generation, we use 256×256 downscaled CelebA-HQ [18], a collection of 30,000 high-quality celebrity face images, commonly used for assessing face generation tasks. For class-conditioned image generation, we train the diffusion model on ImageNet-1K [7], a large-scale dataset containing over 1.2 million labeled images on 1,000 categories, providing a rigorous benchmark.

Implementation Details. We employ DiT-B/1 [31] as the diffusion model across all datasets, maintaining fixed hyperparameters for each dataset to ensure fair comparisons among autoencoders. The learning rate is set to 2×10^{-4} and the global batch size is fixed to 1024 for all datasets. To mitigate divergence caused by uncontrolled attention logit growth, we apply QK normalization [6]. We train for 100K iterations on ImageNet and 60K on CelebA-HQ. For all other configurations, we use the default settings in Yao and Wang [45]. During sampling, we use a 250-step Euler integrator and apply classifier-free guidance (CFG) with a consistent scale across all architectures for class-conditional generation.

Evaluation Metrics. Inception Score (IS) [35] measures how well a model captures the full class distribution while producing convincing class-specific samples. Generative Fréchet Inception Distance (gFID) [15] calculates the distance between two image distributions in the Inception-v3 [42] latent space, capturing both fidelity and diversity, and is widely regarded as more consistent with human judgment. sFID [28], a variation of FID using spatial features, better captures spatial relationships and high-level structure in image distributions. Improved Precision and Recall [21] both assess the fidelity of generated samples in different ways. Specifically, precision measures how realistic or high-quality the generated samples are, while recall evaluates whether the model captures the full diversity of the real dataset.

Figure I. **Additional Class-Conditional Generation Examples on ImageNet-1K.** We present additional examples of class-conditional generation on ImageNet-1K (256×256) across various classes.

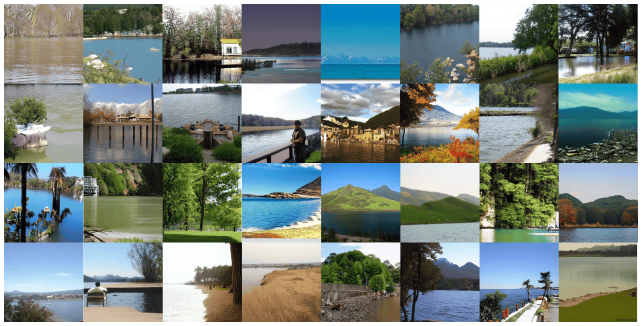
(a) Flamingo (130)

(b) Macaw (88)

(c) Siberian Husky (250)

(d) Passenger Car (705)

(e) Mushroom (947)



(f) Lakeside (975)

(g) Fig (952)

(h) Vase (883)

Figure II. **Uncurated Class-Conditional Generation on ImageNet-1K.** We present a collection of uncurated class-conditional generation examples on ImageNet-1K at a resolution of 256×256. Each subcaption indicates the class name along with the corresponding class index.