# Sample what you can't compress

**Anonymous authors**
Paper under double-blind review

## Abstract

For learned image representations, basic autoencoders often produce blurry results. Reconstruction quality can be improved by incorporating additional penalties such as adversarial (GAN) and perceptual losses. Arguably, these approaches lack a principled interpretation. Concurrently, in generative settings diffusion has demonstrated a remarkable ability to create crisp, high quality results and has solid theoretical underpinnings (from variational inference to direct study as the Fisher Divergence). Our work combines autoencoder representation learning with diffusion and is, to our knowledge, the first to demonstrate *jointly learning a continuous encoder and decoder under a diffusion-based loss and showing that it can lead to higher compression and better generation.*. We demonstrate that this approach yields better reconstruction quality as compared to GAN-based autoencoders while being easier to tune. We also show that the resulting representation is easier to model with a latent diffusion model as compared to the representation obtained from a state-of-the-art GAN-based loss. Since our decoder is stochastic, it can generate details not encoded in the otherwise deterministic latent representation; we therefore name our approach "Sample what you can't compress", or SWYCC for short.

## 1 Introduction

Image autoencoders ultimately necessitate a pixel-level loss to measure and minimize distortion. A common choice is to use mean squared error (MSE). This is a problem for image and video models because MSE favors low frequencies over high frequencies (Rybkin, 2018). Although generalized robust loss functions have been developed (Barron, 2019), they are insufficient on their own for avoiding blurry reconstructions. A popular fix is to augment a pixel-level loss with additional penalties. Typically, MSE is still used because it is easy to optimize due to its linear gradient.

For example, Rombach et al. (2021) use a combination of MSE, perceptual loss, and adversarial loss. Esser et al. (2021) noted that an adversarial loss helps them get high-quality images with realistic textures. Unfortunately GANs remain challenging to train; which was most recently noted by Kang et al. (2023), when they couldn't naively scale up their architecture. The diversity of their outputs is also limited, because modern GAN based decoders are deterministic, and thus lack the capacity to sample multiple different possibilities.

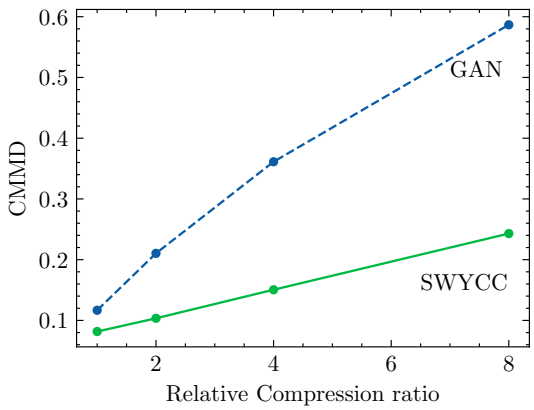



Figure 1: Reconstruction distortion (lower is better) as a function of compression for SWYCC and GAN based auto-encoders.

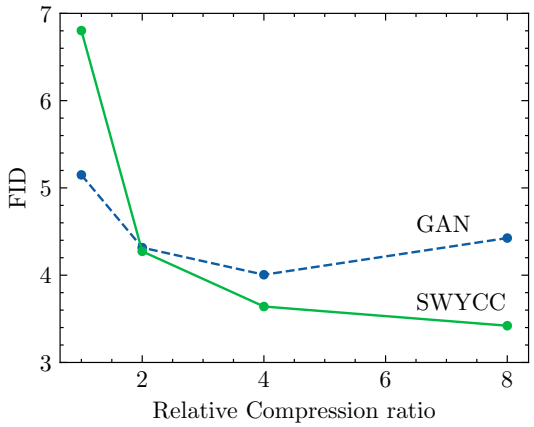



Figure 2: Class conditional generation quality (lower is better) as a function of compression for SWYCC and GAN based auto-encoders.

As an alternative, this paper describes a technique for using a diffusion loss to learn an autoencoder. The diffusion loss is sensible because it is a proper scoring rule with favorable theoretical properties such as being formally connected to KL divergence (Sohl-Dickstein et al., 2015). It has proven itself capable of generating crisp results with high perceptual quality as reflected by human evaluation studies (Hoogeboom et al., 2023b; Karras et al., 2024b).

To demonstrate its simplicity, we take a popular encoder architecture Chang et al. (2022) and marry it with a U-Net decoder, a popular denoising architecture for diffusion (Hoogeboom et al., 2023b). With some additional details outlined in section 2, we show that this approach achieves lower distortion at all compression levels as measured by the CMMD metric (Jayasumana et al., 2024). Because our decoder is able to sample details at test-time that are not encoded in the latents, we call our approach "Sample what you can't compress" or SWYCC for short. This work will show that,

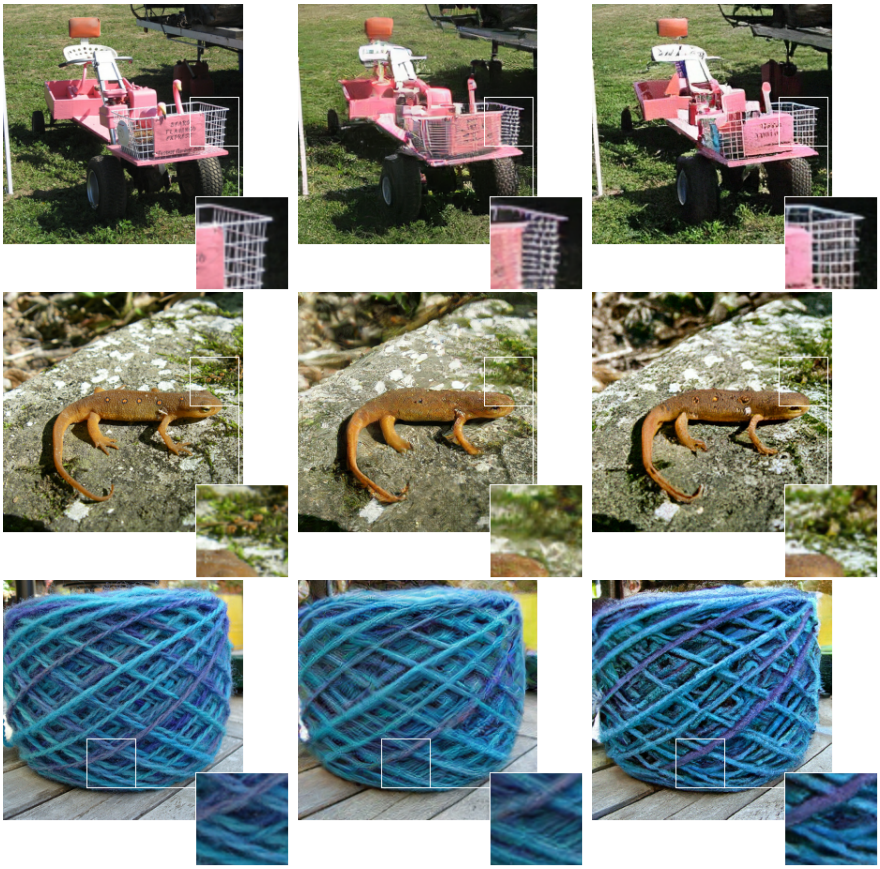
- SWYCC achieves lower reconstruction distortion at all tested compression levels vs SOTA GAN-based autoencoders (section 3).
- SWYCC representations enable qualitatively better latent diffusion generation results at higher compression levels vs SOTA GAN-based autoencoders (section 3.3).
- Splitting the decoder into two parts improves training dynamics (section 3.1 and 3.2).

## 2 Method

Eliding its various parametrizations for brevity, the standard diffusion loss is characterized by the Monte Carlo approximation of the following loss,

$$\ell(x) \stackrel{\text{def}}{=} \mathsf{E}_{\varepsilon\sim\text{MVN}(0,I_{h\cdot w\cdot 3}),t\sim\text{Uniform}[0,1]}\left[w_t \left\|x - D(\alpha_t x + \sigma_t\varepsilon, t)\right\|_2^2\right]. \tag{1}$$

Herein $x \in \mathbb{R}^{h\times w\times 3}$ denotes a natural image and $D$ is a neural network fit using gradient descent and which serves to denoise the corrupted input $x_t \stackrel{\text{def}}{=} \alpha_t x + \sigma_t\varepsilon$ at a given noise-level $t$. Let the corruption process be the cosine schedule (Hoogeboom et al., 2023a), $\sigma_t^2 \stackrel{\text{def}}{=} 1 - \alpha_t^2$ and $\alpha_t \stackrel{\text{def}}{=} \cos(at + b(1-t))$ where $a = \arctan(e^{10})$ and $b = \arctan(e^{-10})$.



(a) Groundtruth (b) GAN (c) SWYCC

Figure 3: Comparison of GAN versus SWYCC reconstructions at 8× relative compression level. You can see that the GAN based autoencoder loses a significant amount of detail in the highlighted portions, which SWYCC is able to sample effectively.

We extend this definition to the task of autoencoding by simply allowing the denoising function to take an additional argument, $D_{\text{Initial}}(E(x))$, itself having access to the uncorrupted input $x$ but only through the bottlenecking function $E$. The result is,

$$\ell_{\text{AE}}(x) \stackrel{\text{def}}{=} \mathsf{E}_{\varepsilon \sim \text{MVN}(0, I_{h \cdot w \cdot 3}), t \sim \text{Uniform}[0,1]} \left[ w_t \left\| x - D_{\text{Refine}}(\alpha_t x + \sigma_t \varepsilon, t, D_{\text{Initial}}(E(x))) \right\|_2^2 \right]. \tag{2}$$

As the notation suggests $E$ is an encoder which, notably, is learned jointly with "diffusion decoder" $D_{\text{Refine}}$ and secondary decoder $D_{\text{Initial}} : \mathcal{Z} \to \mathbb{R}^{h \times w \times 3}$. The specification of $D_{\text{Initial}}$ is largely a convenience but also merits secondary advantages. By mapping $z = E(x)$ back into $x$-space, we can simply concatenate the corrupted input $x_t$ and its "pseudo reconstruction," $D_{\text{Initial}}(z)$. Additionally, we find that directly penalizing $D_{\text{Initial}}(z)$, as described below, speeds up training.

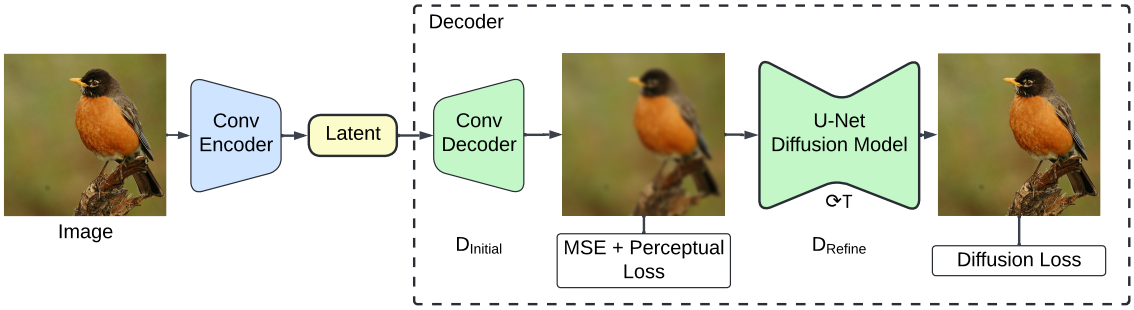



Figure 4: A block diagram of our auto-encoder architecture. Our "Diffusion Model" is a U-Net as defined by Hoogeboom et al. (2023b). During inference, the diffusion model is run in a loop to iteratively sample an image. All of these parameters jointly after being initialized from scratch, except for the weights of the network used in the perceptual loss.

### 2.1 Architectural Details

**Encoder:** We use a fully convolutional encoder in all of our experiments, whose specifics we borrow from MaskGIT(Chang et al., 2022). The encoder consists of multiple ResNet(He et al., 2016) blocks stacked on top of each other, with GeLU(Hendrycks & Gimpel, 2017) for its non-linearities and GroupNorm(Wu & He, 2018) for training stability. The ResNet blocks are interspersed with strided convolutions with stride 2 which achieves a $2\times$ downsampling by itself. To get the $8 \times 8$ patch size, we use 4 ResNet blocks with 3 downsampling blocks. The encoder architecture is common for all of our experiments, and we only change the number of channels at the output layer to achieve the desired compression ratio.

**Decoder:** For the decoder in the GAN baseline and $D_{\text{Initial}}$ we use an architecture that is the reverse of the encoder. For up-sampling, we use the depth-to-space operation. Just like the encoder, we have 4 ResNet blocks interspersed with 3 depth-to-space operations. For $D_{\text{Refine}}$ we use a U-Net as defined by Hoogeboom et al. (2023b). The U-Net has 4 ResNet blocks for downsampling and corresponding 4 ResNet blocks for upsampling with residual connections between blocks of the same resolution. After 4 downsampling stages, when resolution is $16 \times 16$, we use a self-attention block to give the network additional capacity.

### 2.2 Reducing distortion using additional distance metrics

We find that additional direct penalization of $D_{\text{Initial}}(E(x)$ leads to improved CMMD and FID and less distortion (see Figure 11). This was achieved by minimizing a composite loss containing terms with favorable Hessian (eq. (4)) and perceptual characteristics (eq. (5)),

$$\ell_{\text{Total}} \stackrel{\text{def}}{=} \ell_{\text{AE}} + \lambda_p \ell_{\text{Perceptual}} + \lambda_m \ell_{\text{MSE}} \tag{3}$$

where,

$$\ell_{\text{MSE}} \stackrel{\text{def}}{=} \|x - D_{\text{Initial}}(E(x))\|_2^2 \tag{4}$$

and,

$$\ell_{\text{Perceptual}} \stackrel{\text{def}}{=} \left\| f_{\text{Frozen}}(x) - f_{\text{Frozen}}\Big(D_{\text{Initial}}(E(x))\Big) \right\|_2^2. \tag{5}$$

| $\lambda_p$ | $\lambda_m$ | CMMD ↓ |
|---|---|---|
| 0 | 0 | 0.43 |
| 0 | 1 | 0.32 |
| 0.1 | 0 | 0.13 |
| 0.1 | 1 | 0.15 |

Table 1: Individual impact of each of our auxiliary losses. We note that the perceptual loss has a big impact, and is crucial to being competitive while training autoencoders. See image samples in Figure 11. The first row $\lambda_p = \lambda_m = 0$ is analogous to $D_{\text{Initial}}$ having no role to play.

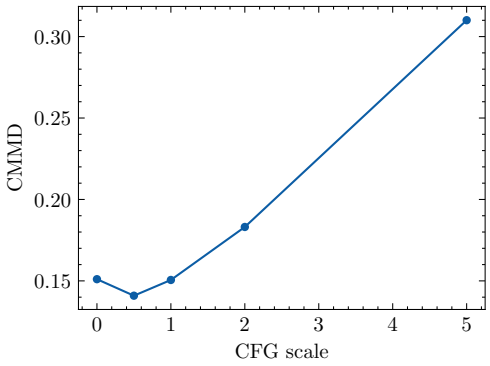



Figure 5: Effect of CFG scale in $D_{\text{Refine}}$.We use CFG scale of 0.5 in our experiments.

The function $f_{\text{Frozen}}$ is an unlearnable standard ResNet, itself trained on ImageNet and used for both the baseline and SWYCC. We found the best hyper-parameter setting for eq. (3) is $\lambda_m = 1$ and $\lambda_p = 0.1$ Setting $\lambda_p > 0$ was particularly important to be competitive at reconstruction with GAN based methods in Table 1. The visual impact of perceptual loss is shown in Figure 11.

For generating reconstructions (recall that the SWYCC decoder is stochastic) we used classifier-free guidance during inference (Ho & Salimans, 2021); for the unconditional model the U-Net was trained with $D_{\text{Initial}}(E(x))$ dropped-out, i.e., randomly zeroed out on 10% of training instances.

## 3 Experiments

In this section we explore how the GAN-based loss compares to our approach. Without loss of generality, we define the relative compression ratio of 1 to be a network that maps $8 \times 8$ RGB patches to an 8 dimensional latent vector. Effectively, this means for our encoder $E$ if and $x \in \mathbb{R}^{256\times256\times3}$, then $E(x) \in \mathbb{R}^{32\times32\times C}$ where $C = 8$. In general, to achieve a relative compression ratio of $K$ we set $C = \frac{8}{K}$. The effect of increasing the compression ratio is plotted in Figure 1. Observe that distortion degrades much more rapidly for the GAN based auto-encoder as measured by CMMD (Jayasumana et al., 2024) which Imagen-3 (Imagen-Team-Google et al., 2024) showed better correlates with human perception.

Not only is our approach better at all compression levels; the gap between the GAN based autoencoder and SWYCC widens as we increase the relative compression ratio. Using the much simpler diffusion formulation under Equation 3, we are able to reconstruct crisp looking images with detailed textures (See Figure 3). Our method has the added benefit that we do not need to tune any GAN related hyper-parameters, and can scale up effectively using the large body of diffusion literature (Karras et al. (2022), Karras et al. (2024a)).

### 3.1 Impact of $D_{\text{Initial}}$

Observing Equation 2 and Figure 4, we note that the output of $D_{\text{Initial}}$ is an intermediate tensor that is not strictly required for the diffusion loss or for generating the output. We show using Table 1 that this piece is crucial for achieving performance comparable to the GAN based autoencoder. The perceptual loss term in particular has a large impact in reducing distortion. Visual examples

2 steps 5 steps 10 steps 50 steps 150 steps

Figure 6: Example of how reconstructions look when changing the number of sampling steps for $D_{\text{Refine}}$. We can see that even with steps $\leq 5$, the high level structure of the image is preserved. When our method is used for interactive generation (for example, with a latent diffusion model), we can use fewer steps of $D_{\text{Refine}}$ to show the user multiple low-quality inputs, and use a high number of steps for the final generation that the user selects.

are shown in Figure 11. We found that separating the decoder into 2 parts was necessary. When we applied LPIPS and MSE losses without $D_{\text{Initial}}$, our method was not competitive with GAN based autoencoders.

### 3.2 Analysis of sampling in $D_{\text{Refine}}$

In Figure 6 we show the qualitative difference number of sampling steps makes to reconstructions. We can see that even with just 2 steps the high level structure of the image is present. In Figure 7 we study the impact of number of sampling steps by using the CMMD metric. Figure 9 shows what sampling in $D_{\text{Refine}}$ actually ends up changing. We can see that only regions with high-frequency components and detailed textures are changed between samples, while regions containing similar colors over large areas are left untouched.

Figure 5 studies the effect of classifier-free guidance (Ho & Salimans, 2021) as used in $D_{\text{Refine}}$. We ablate the guidance with a model trained at a relative compression factor of 4 (See Section 3 for definition) and find that a guidance scale of 0.5 works the best. This is not to be confused with the guidance scale of the latent diffusion model that may be trained on top of our autoencoder, which is a completely separate parameter to be tuned independently.

### 3.3 Modeling latents for diffusion

We use a DiT model (Peebles & Xie, 2023) to model the latent space of our models for the task of class-conditional image generation. In Figure 2 we compare our latents with those of a GAN based autoencoder. Our approach leads to 5% lower FID (Heusel et al., 2017) than the GAN baseline at the best $4\times$ compression ratio. Notably our approach achieves its best result at the highest compression ratio, where the task of modelling the latent representation is simplest, whereas the GAN autoencoder is unable to operate effectively in this regime.

### 3.4 Exploring better perceptual losses

In Table 1 we showed the large impact perceptual loss has on reconstruction quality. This begs the question; are there better auxiliary losses we can use? We compare the perceptual loss as described by in VQGAN (Esser et al., 2021; Johnson et al., 2016) and replace it with the DISTS (Ding et al., 2020). DISTS loss differs from perceptual loss in 2 important ways. a) It uses SSIM (Wang et al., 2004) instead of mean squared error and b) It uses features are multiple levels instead of

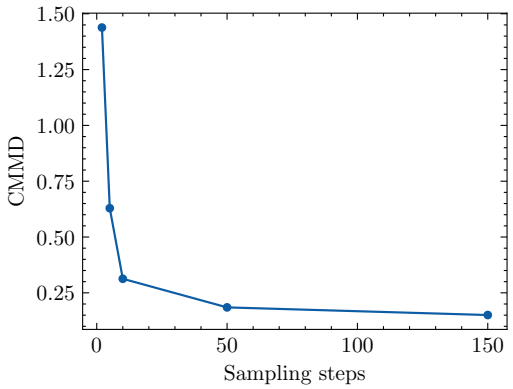



Figure 7: Impact of number of sampling steps on $D_{\text{Refine}}$ on CMMD.

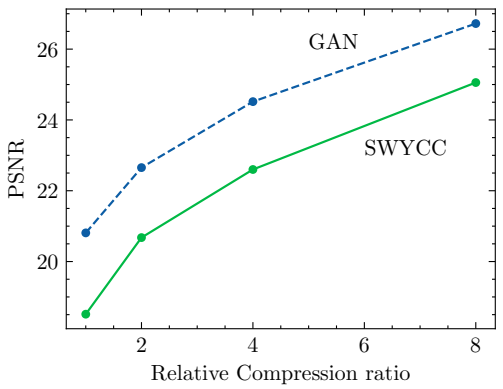



Figure 8: PSNR for GAN v/s SWYCC. Note that despite lower PSNR, SWYCC achieves less distortion in CMMD as shown in Figure 1.

Figure 9: Reconstructed images (top) and a heat-map of variance between 10 samples (bottom).

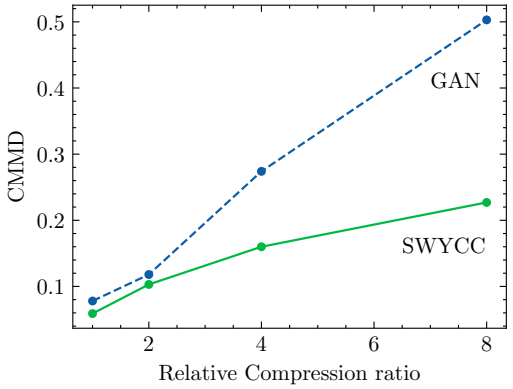



Figure 10: Effect of using DISTS loss on reconstruction quality.

using only the activation's from the last layer. The results are shown in Figure 10. At lower relative compression ratios, DISTS loss helps GANs and SWYCC. But at higher relative compression ratios, GAN based autoencoders perform even worse than the perceptual loss based baseline. We think this is an avenue for future exploration. We perform all other experiments with perceptual loss (Esser et al., 2021), since it is more prevalent in literature and it helps GAN based auto-encoders at higher relative compression ratios.

### 3.5 Architecture and hyper-parameters

**Autoencoder training hyper-parameters** We train all of our models on the ImageNet dataset resized at $256 \times 256$ resolution. During training, we resize the image such that the shorter side

measures 256 pixels and take a random crop in that image of size $256 \times 256$. For measuring reference statistics on the validation set, we take the largest possible center square crop. All of our models are trained at a batch size of 256 for $10^6$ steps which roughly equals 200 epochs.

**GAN baseline:** We use the popular convolutional encoder-decoder architecture popularized by MaskGIT (Chang et al., 2022) in our GAN-based baselines. This autoencoder design is used by many image models, including FSQ (Mentzer et al., 2024) and GIVT (Tschannen et al., 2023), and was extended to the video domain by MAGVIT-v2 (Yu et al., 2024), VideoPoet (Kondratyuk et al., 2024), and WALT (Gupta et al., 2023). We take advantage of decoder improvements developed by MAGVIT-v2 (see Section 3.2 in (Yu et al., 2024)) to improve reconstruction quality. Note that while the video models enhance the base autoencoder architecture with 3D convolution to integrate information across time, the discriminator and perceptual loss are still applied on a per-frame basis and thus are essentially unchanged in our model.

**SWYCC:** In our experiments we keep the architecture of $D_{\text{Initial}}$ identical to the decoder used in the GAN baseline. For $D_{\text{Refine}}$ we use the U-Net architecture as parameterized by Hoogeboom et al. (2023b). We borrow the U-Net 256 architecture and make the following modifications:

```
channel_multiplier = [1, 2, 4, 8]
num_res_blocks = [2, 4, 8, 8]
downsampling_factor = [1, 2, 2, 2]
attn_resolutions = [16]
dropout = 0.0
```

We train using $v$-parameterization (Salimans & Ho, 2022), which corresponds to $w_t = \sigma_t^{-2}$ in equation 1. We use the Adam optimizer to learn our parameters. The learning rate is warmed up for $10^4$ steps from 0 to a maximum value of $10^{-4}$ and cosine decayed to 0. We use gradient clipping with global norm set to 1.

**Latent Diffusion**: All of our experiments are done with the DiT-L architecture with $2 \times 2$ patching (Peebles & Xie, 2023) with the addition of SwiGLU (Shazeer, 2020) and 2D RoPE (Heo et al., 2024). We train for $4 \times 10^5$ steps with a batch size of 256, using a constant learning rate of $10^{-4}$ and dropout the class embedding 10% of the time during training. For inference we use a classifier-free guidance scale of 0.5 which gave optimal results in Peebles & Xie (2023).

## 4 Related Work

**Autoencoders for 2-stage generation:** For discrete representation learning, van den Oord et al. (2017) showed the usefulness of the 2-stage modeling approach. In this broad category, the first stage fits an autoencoder to the training data with the goal of learning a compressed representation useful for reconstructing images. This is followed by a second stage where the encoder is frozen and a generative model is trained to predict the latent representation based on a conditioning signal. This approach regained popularity when Ramesh et al. (2021) showed that it can be used for zero-shot text generation, and is now the dominant approach for image and video generation (Chang et al., 2022; Yu et al., 2024; Chang et al., 2023; Gupta et al., 2023; Kondratyuk et al., 2024).

**Adversarial losses:** Esser et al. (2021) extended the autoencoder from van den Oord et al. (2017) with two important new losses, the perceptual loss and the adversarial loss, taking inspiration from the works of Johnson et al. (2016) and Isola et al. (2017). The perceptual loss is usually defined as the L2 loss between a latent representation of the original and reconstructed image. The latent representation, for example, can be extract from the final layer activations of a ResNet optimized to classify ImageNet images. The adversarial loss is a patch-based discriminator that

uses a discriminator network to predict at a patch-level whether it is real or fake. This encourages the decoder to produce realistic looking textures.

**Latent Diffusion:** Rombach et al. (2021) popularized text-to-image generation using latent diffusion models. They kept the autoencoder from Esser et al. (2021) intact and simply removed the quantization layer. This accelerated diffusion model research in the community owing to the fact that the latent space was much smaller than the pixel space, which allows fast training and inference compared to diffusion models like Imagen that sample pixels directly (Saharia et al., 2022).

**2-stage diffusion autoencoders:** Pandey et al. (2022) and Preechakul et al. (2022) both train decoders with a diffusion loss. The crucial difference is that both these works train their autoencoder in 2-stages.

**Compression and diffusion:** Hoogeboom et al. (2023a) showed that diffusion models can be used for compression. Crucially, compared to our approach they use a frozen autoencoder, and do not train their autoencoder end-to-end. They also use an objective based on modified flow matching. In contrast, we did not modify the loss or the sampling algorithm. Pandey et al. (2022) use a similar approach with a 2-stage autoencoder training process for their autoencoder.

In a similar context, Yang & Mandt (2024) developed an end-to-end optimized compression model using a diffusion decoder. They show improved perceptual quality compared to earlier GAN-based compression methods at the expense of higher distortion (pixel-level reconstruction accuracy). Different from our approach, they use a discrete latent space, which is required for state-of-the-art compression rates achieved via entropy coding. This limits the reconstruction quality but is required for a compression model that ultimately seeks to minimize a rate-distortion objective, not just a reconstruction and sampling quality objective.

Würstchen architecture (Pernias et al., 2024) has shown that training a cascade of diffusion models improves training efficiency. Crucially Würstchen, does not apply a diffusion loss on pixels, instead resorting to a GAN loss. Whang et al. (2022) also pointed out that diffusion can fix a lot of pixel level artifacts, although they do not investigate training autoencoders.

Shi et al. (2022) learn an autoencoder using a diffusion loss for discrete vector quantized encodings. We differ from them in 2 crucial ways; (i) we learn a continuous representation (ii) We show that our architecture produces latents that are better for latent diffision by showing that we can achieve *higher compression and better generation performance (Figure 2)* .

## 5 Conclusion

We have described a general autoencoder framework that uses a diffusion based decoder. Compared to decoders that use GANs, our system is much more easier to tune and has the same theoretical underpinnings as diffusion models. We showed our method produces sigificantly less distortions as compared to GAN based autoencoders in Figure 1 and are better behaved as latent spaces for diffusion in Figure 2. In Section 3.1 and 3.2 we studied the hyper-parameter settings on the 2 major components of our decoder, $D_{\text{Initial}}$ and $D_{\text{Refine}}$.

**Possible extensions:** The autoencoder technique we describe is fairly general and can be extended to any other continuous modality like audio, video or 3D-point clouds. In addition, all improvements to diffusion algorithms like those by Karras et al. (2024a) can be carried over.

**Limitations:** The main limitation of our method is the increase in inference cost during decoding. This can be partly mitigated by using fewer steps like in Figure 6. In addition, techniques used to improve diffusion sampling time like Progressive distillation (Salimans & Ho, 2022) and Instaflow

(Liu et al., 2024) are also prudent. Because of $D_{\text{Refine}}$, our training time compute cost is also higher. Combining $D_{\text{Initial}}$ and $D_{\text{Refine}}$ in a clever way to reduce training time compute and memory could be a promising research direction.

## References


Jonathan T. Barron. A general and adaptive robust loss function. *CVPR*, 2019.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. Maskgit: Masked generative image transformer. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.

Huiwen Chang, Han Zhang, Jarred Barber, Aaron Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Patrick Murphy, William T. Freeman, Michael Rubinstein, Yuanzhen Li, and Dilip Krishnan. Muse: Text-to-image generation via masked generative transformers. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 4055–4075. PMLR, 23–29 Jul 2023.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE transactions on pattern analysis and machine intelligence*, 44(5):2567–2581, 2020.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883, 2021.

Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Li Fei-Fei, Irfan Essa, Lu Jiang, and José Lezama. Photorealistic video generation with diffusion models. *arXiv preprint arXiv:2312.06662*, 2023.

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, 2016. URL `https://api.semanticscholar.org/CorpusID:6447277`.

Dan Hendrycks and Kevin Gimpel. Bridging nonlinearities and stochastic regularizers with gaussian error linear units, 2017. URL `https://openreview.net/forum?id=Bk0MRI5lg`.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer, 2024. URL `https://arxiv.org/abs/2403.13298`.

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 I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper_files/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf`.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. URL `https://openreview.net/forum?id=qw8AKxfYbI`.

Emiel Hoogeboom, Eirikur Agustsson, Fabian Mentzer, Luca Versari, George Toderici, and Lucas Theis. High-fidelity image compression with score-based generative models. *arXiv preprint arXiv:2305.18231*, 2023a.

Emiel Hoogeboom, Jonathan Heek, and Tim Salimans. Simple diffusion: end-to-end diffusion for high resolution images. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023b.

Imagen-Team-Google, :, Jason Baldridge, Jakob Bauer, Mukul Bhutani, Nicole Brichtova, Andrew Bunner, Kelvin Chan, Yichang Chen, Sander Dieleman, Yuqing Du, Zach Eaton-Rosen, Hongliang Fei, Nando de Freitas, Yilin Gao, Evgeny Gladchenko, Sergio Gómez Colmenarejo, Mandy Guo, Alex Haig, Will Hawkins, Hexiang Hu, Huilian Huang, Tobenna Peter Igwe, Christos Kaplanis, Siavash Khodadadeh, Yelin Kim, Ksenia Konyushkova, Karol Langner, Eric Lau, Shixin Luo, Soňa Mokrá, Henna Nandwani, Yasumasa Onoe, Aäron van den Oord, Zarana Parekh, Jordi Pont-Tuset, Hang Qi, Rui Qian, Deepak Ramachandran, Poorva Rane, Abdullah Rashwan, Ali Razavi, Robert Riachi, Hansa Srinivasan, Srivatsan Srinivasan, Robin Strudel, Benigno Uria, Oliver Wang, Su Wang, Austin Waters, Chris Wolff, Auriel Wright, Zhisheng Xiao, Hao Xiong, Keyang Xu, Marc van Zee, Junlin Zhang, Katie Zhang, Wenlei Zhou, Konrad Zolna, Ola Aboubakar, Canfer Akbulut, Oscar Akerlund, Isabela Albuquerque, Nina Anderson, Marco Andreetto, Lora Aroyo, Ben Bariach, David Barker, Sherry Ben, Dana Berman, Courtney Biles, Irina Blok, Pankil Botadra, Jenny Brennan, Karla Brown, John Buckley, Rudy Bunel, Elie Bursztein, Christina Butterfield, Ben Caine, Viral Carpenter, Norman Casagrande, Ming-Wei Chang, Solomon Chang, Shamik Chaudhuri, Tony Chen, John Choi, Dmitry Churbanau, Nathan Clement, Matan Cohen, Forrester Cole, Mikhail Dektiarev, Vincent Du, Praneet Dutta, Tom Eccles, Ndidi Elue, Ashley Feden, Shlomi Fruchter, Frankie Garcia, Roopal Garg, Weina Ge, Ahmed Ghazy, Bryant Gipson, Andrew Goodman, Dawid Górny, Sven Gowal, Khyatti Gupta, Yoni Halpern, Yena Han, Susan Hao, Jamie Hayes, Amir Hertz, Ed Hirst, Tingbo Hou, Heidi Howard, Mohamed Ibrahim, Dirichi Ike-Njoku, Joana Iljazi, Vlad Ionescu, William Isaac, Reena Jana, Gemma Jennings, Donovon Jenson, Xuhui Jia, Kerry Jones, Xiaoen Ju, Ivana Kajic, Christos Kaplanis, Burcu Karagol Ayan, Jacob Kelly, Suraj Kothawade, Christina Kouridi, Ira Ktena, Jolanda Kumakaw, Dana Kurniawan, Dmitry Lagun, Lily Lavitas, Jason Lee, Tao Li, Marco Liang, Maggie Li-Calis, Yuchi Liu, Javier Lopez Alberca, Peggy Lu, Kristian Lum, Yukun Ma, Chase Malik, John Mellor, Inbar Mosseri, Tom Murray, Aida Nematzadeh, Paul Nicholas, João Gabriel Oliveira, Guillermo Ortiz-Jimenez, Michela Paganini, Tom Le Paine, Roni Paiss, Alicia Parrish, Anne Peckham, Vikas Peswani, Igor Petrovski, Tobias Pfaff, Alex Pirozhenko, Ryan Poplin, Utsav Prabhu, Yuan Qi, Matthew Rahtz, Cyrus Rashtchian, Charvi Rastogi, Amit Raul, Ali Razavi, Sylvestre-Alvise Rebuffi, Susanna Ricco, Felix Riedel, Dirk Robinson, Pankaj Rohatgi, Bill Rosgen, Sarah Rumbley, Moonkyung Ryu, Anthony Salgado, Sahil Singla, Florian Schroff, Candice Schumann, Tanmay Shah, Brendan Shillingford, Kaushik Shivakumar, Dennis Shtatnov, Zach Singer, Evgeny Sluzhaev, Valerii Sokolov, Thibault Sottiaux, Florian Stimberg, Brad Stone, David Stutz, Yu-Chuan Su, Eric Tabellion, Shuai Tang, David Tao, Kurt Thomas, Gregory Thornton, Andeep Toor, Cristian Udrescu, Aayush Upadhyay, Cristina Vasconcelos, Alex Vasiloff, Andrey Voynov, Amanda Walker, Luyu Wang, Miaosen Wang, Simon Wang, Stanley Wang, Qifei Wang, Yuxiao Wang, Ágoston Weisz, Olivia Wiles, Chenxia Wu, Xingyu Federico Xu, Andrew Xue, Jianbo Yang, Luo Yu, Mete Yurtoglu, Ali Zand, Han Zhang, Jiageng Zhang, Catherine Zhao, Adilet Zhaxybay, Miao Zhou, Shengqi Zhu, Zhenkai Zhu, Dawn Bloxwich, Mahyar Bordbar, Luis C. Cobo, Eli Collins, Shengyang Dai, Tulsee Doshi, Anca Dragan, Douglas Eck, Demis Hassabis, Sissie Hsiao, Tom Hume, Koray Kavukcuoglu, Helen King, Jack Krawczyk, Yeqing Li, Kathy Meier-Hellstern, Andras Orban, Yury Pinsky, Amar Subramanya, Oriol Vinyals, Ting Yu, and Yori Zwols. Imagen 3, 2024. URL `https://arxiv.org/abs/2408.07009`.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking fid: Towards a better evaluation metric for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9307–9315, 2024.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pp. 694–711. Springer, 2016.

Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10124–10134, 2023.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. *arXiv preprint arXiv:2406.02507*, 2024a.

Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. Analyzing and improving the training dynamics of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24174–24184, June 2024b.

Dan Kondratyuk, Lijun Yu, Xiuye Gu, Jose Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, Krishna Somandepalli, Hassan Akbari, Yair Alon, Yong Cheng, Joshua V. Dillon, Agrim Gupta, Meera Hahn, Anja Hauth, David Hendon, Alonso Martinez, David Minnen, Mikhail Sirotenko, Kihyuk Sohn, Xuan Yang, Hartwig Adam, Ming-Hsuan Yang, Irfan Essa, Huisheng Wang, David A Ross, Bryan Seybold, and Lu Jiang. Videopoet: A large language model for zero-shot video generation. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=LRkJwPIDuE`.

Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, and Qiang Liu. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *International Conference on Learning Representations*, 2024.

Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: VQ-VAE made simple. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=8ishA3LxN8`.

Kushagra Pandey, Avideep Mukherjee, Piyush Rai, and Abhishek Kumar. DiffuseVAE: Efficient, controllable and high-fidelity generation from low-dimensional latents. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL `https://openreview.net/forum?id=ygoNPRiLxw`.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Pablo Pernias, Dominic Rampas, Mats Leon Richter, Christopher Pal, and Marc Aubreville. Würstchen: An efficient architecture for large-scale text-to-image diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=gU58d5QeGv.

Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8821–8831. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/ramesh21a.html.

Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10674–10685, 2021. URL https://api.semanticscholar.org/CorpusID:245335280.

Oleg Rybkin. The reasonable ineffectiveness of pixel metrics for future prediction (and what to do about it). https://bit.ly/3LXB7SS, 2018. Accessed: Aug 7 2024.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo-Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=08Yk-n5l2Al.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=TIdIXIpzhoI.

Noam M. Shazeer. Glu variants improve transformer. *ArXiv*, abs/2002.05202, 2020. URL https://api.semanticscholar.org/CorpusID:211096588.

Jie Shi, Chenfei Wu, Jian Liang, Xiang Liu, and Nan Duan. Divae: Photorealistic images synthesis with denoising diffusion decoder. *CoRR*, abs/2206.00386, 2022. doi: 10.48550/ARXIV.2206.00386. URL https://doi.org/10.48550/arXiv.2206.00386.

Jascha Narain Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *ArXiv*, abs/1503.03585, 2015. URL https://api.semanticscholar.org/CorpusID:14888175.

Michael Tschannen, Cian Eastwood, and Fabian Mentzer. Givt: Generative infinite-vocabulary transformers. *arXiv preprint arXiv:2312.02116*, 2023.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 6309–6318, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. doi: 10.1109/TIP.2003.819861.

Jay Whang, Mauricio Delbracio, Hossein Talebi, Chitwan Saharia, Alexandros Dimakis, and Peyman Milanfar. Deblurring via stochastic refinement. In *CVPR*, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/papers/Whang_Deblurring_via_Stochastic_Refinement_CVPR_2022_paper.pdf.

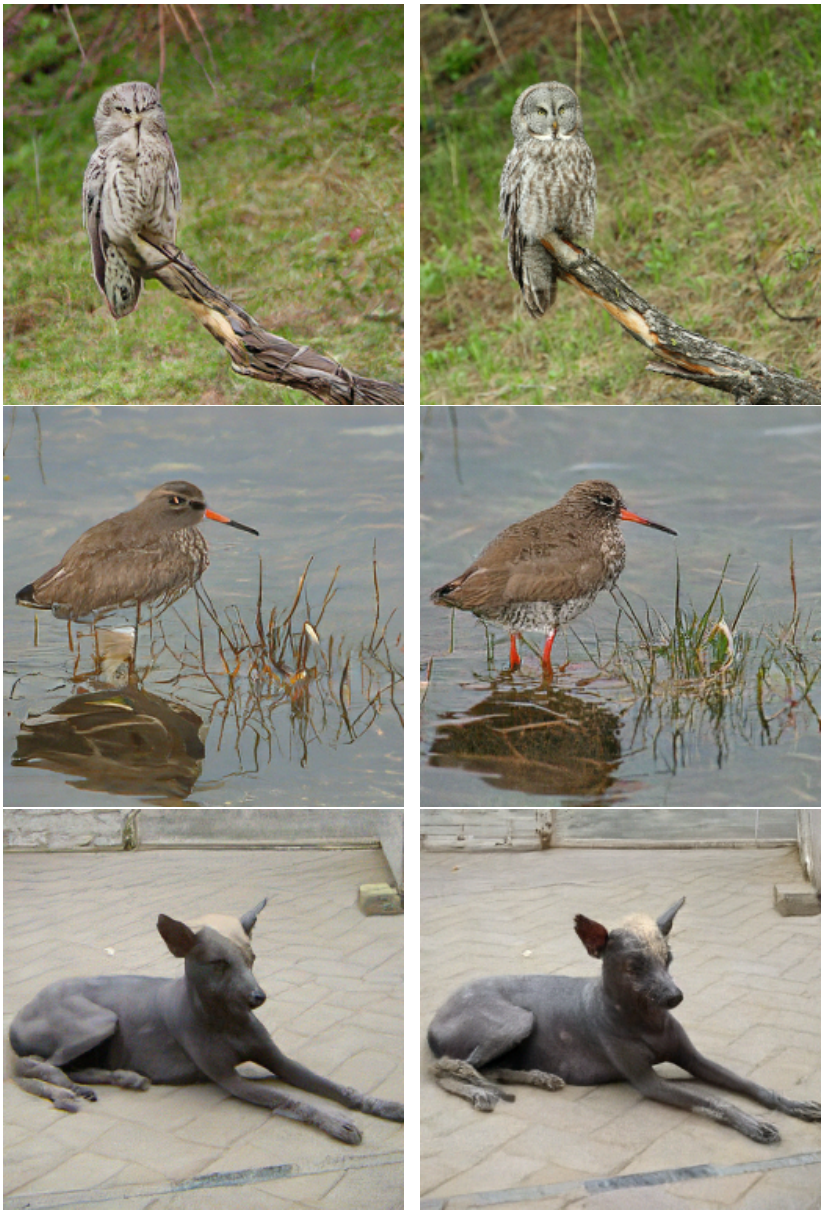
Yuxin Wu and Kaiming He. Group normalization. In *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIII*, pp. 3–19, Berlin, Heidelberg, 2018. Springer-Verlag. ISBN 978-3-030-01260-1. doi: 10.1007/978-3-030-01261-8_1. URL https://doi.org/10.1007/978-3-030-01261-8_1.

Ruihan Yang and Stephan Mandt. Lossy image compression with conditional diffusion models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2024. Curran Associates Inc.

Lijun Yu, Jose Lezama, Nitesh Bharadwaj Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, Boqing Gong, Ming-Hsuan Yang, Irfan Essa, David A Ross, and Lu Jiang. Language model beats diffusion - tokenizer is key to visual generation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=gzqrANCF4g.

# 6 Appendix

## 6.1 Visual examples



(a) No auxiliary losses. (b) Only perceptual auxiliary loss.

Figure 11: Visual impact of perceptual loss.