Image Neural Field Diffusion Models

Supplementary Material

Figure 1. Comparison of using LIIF (left) and CLIF (right, ours) as the renderer in our framework (Mountains and FFHQ dataset). We find CLIF can produce more photorealistic details than LIIF with the convolutional formulation.

A. Comparison to implementing neural field autoencoder with LIIF [4]

LIIF [4] is an image neural field defined on a feature map, which can also be used as the renderer in the framework. However, we observe that directly implementing our framework with LIIF does not produce photorealistic details, as shown in Figure 1. LIIF was originally proposed for super-resolution with L1 loss. We find the generator with LIIF struggles to learn photorealistic details and the adversarial training quickly collapses with the discriminator as the winner in the adversarial game. Our proposed CLIF renderer addresses this issue by decoding the patch as a whole and incorporating a larger context with convolution layers. Besides having higher capacity, we also find CLIF can be learned to be scale-consistent even with LPIPS and GAN loss and without point-independent decoding.

B. Scale-consistency of CLIF

Our CLIF neural field renderer does not assume pixel independence and is learned with LPIPS and GAN loss. However, instead of synthesizing different contents for different output scales as a conditional GAN, we observe that CLIF is learned to be scale-consistent, as shown in Figure 3. We observe that the object boundary precisely aligns when we render the same latent representation to different resolutions (while in AnyResGAN [3], the contents and object boundary may obviously change for different scales). This property enables multi-scale supervision on the same latent representation in training and solving inverse problems.

Figure 2. Samples of the method where the latent neural field space is replaced by low-resolution images with the same spatial dimension (FFHQ dataset). Even at 256 resolution, these images are overly smooth and lack details, which supports the effectiveness of having a latent space.

Data	Method	FID-256@5K
All HR	LR-DM + SR	36.01
	INFD	9.26
6K-Mix	LR-DM + SR	37.97
	INFD	9.81

Table 1. Comparison to low-resolution diffusion model with any-resolution upsampler. The upsampling network has the same architecture as our decoder-renderer for comparison.

C. Ablation on without latent space

To evaluate the benefits of having a latent space for neural image fields, we compare our method, to a baseline ablation where we remove the encoder and latent space. Instead, in this baseline, we first learn an any-scale super-resolution model for low-resolution images. This super-resolution model has the same architecture as the composition of our decoder and renderer $R \circ D$. The difference is that it acts on low-resolution images, rather than latent codes. We then train a low-resolution diffusion model. At inference time, we first generate a low-resolution image by diffusion, then upsample it with $R \circ D$. For this baseline, the low-resolution image has the same spatial dimensions 64×64 as the latent representation in our main model, and all other training settings are kept identical.

Figure 2 shows this baseline generates overly smooth im-

Figure 3. Scale consistency of our CLIF renderer. In each pair, the left is rendering at resolution 1K, the right is rendering at resolution 2K then downsampling to 1K. Yellow/green boxes show two examples of inconsistent areas in AnyResGAN (besides the boxes, the object boundary also does not align for 1K/2K of AnyResGAN).

Figure 4. Results using LDM [10] trained for 256×256 faces to generate higher-resolution faces at 512×512 by making convolutional samples over a larger noise map.

w/ random downsampling GT w/o random downsampling GT

Figure 5. Comparison of training with/without random downsampling the ground-truth images (Mountains dataset, outputs in the autoencoder stage). Training without random downsampling produces samples with worse quality.

ages. This is already visible at 256×256. The quantitative comparison in Table 1, confirms our model outperforms the baseline on FID. We hypothesize this is because: (i) the latent representation contains much richer information compared to a simple low-resolution RGB image with the same spatial dimension. Intuitively, with the latent representation, our first training stage can encode information relevant to our final goal to synthesize a high-resolution image, which is impossible with a plain RGB low-resolution image. This information is preserved by the diffusion model in the second training stage. (ii) The latent space (with VQ or KL regularization) is much more robust than RGB space to the domain shift from real to generated sample.

D. Convolutional samples of LDM [10] for higher-resolution generation

In LDM [10], the iterative denoising process is operating over a noise map with a UNet, where the UNet contains convolution and attention layers. Since both types of layers can be directly applied to higher-resolution input, a potential method to generate higher-resolution images is to process a noise map with higher resolution without changing the network. Despite this method increasing the computation cost linearly to the number of pixels, we observe that it fails to generate images with correct global structures, as shown in Figure 4 for the FFHQ unconditional generation task. We hypothesize this is because the model trained for 256×256 images learns to generate faces at a specific scale, when making convolutional samples at 512×512, it might try to generate multiple 256×256 faces at different spatial locations and fails to preserve the global structure.

E. Effect of scale-varied training on Mountains dataset

Scale-varied training, i.e., training with randomly downsampled ground-truth images, allows the same latent representation to get supervision from multiple scales with the fixed-resolution patch, which helps the performance even if all ground-truth images are at fixed and high resolution. Besides results on the FFHQ dataset shown in the main pa-

Figure 6. **2K results on faces.** When training data is available, our method can be used to generate images at resolutions above 1K. Here we show 2K generated images trained on datasets of high-resolution portraits.

per, we observe that scale-varied training is more important on the Mountains dataset which contains more complex images. As shown in Figure 5, without random downsampling of the ground-truth images during training, the model produces samples with worse quality.

F. Image generation beyond 1K

Our method uses patchwise supervision at arbitrary coordinates and can be trained on higher-resolution images without changing the architecture. We explore going beyond the 1024 resolution and train our model on a collected dataset of faces, which contains images at varied resolutions between 1K to 2K (about 84% images have 2K resolution). We show some qualitative results in Figure 6. We observe that, with no change to the architecture, our method learns to generate highly detailed textures on skin and hairs for 2K faces, which suggests the potential for pushing our method further, for ultra-high resolution image generation.

G. Comparison to any-resolution learning in GANs

The idea of using neural fields for training with any-resolution images was explored for GANs in ScaleParty [9] and AnyResGAN [3]. We take AnyResGAN [3], which worked for more diverse high-resolution images, as an example for comparison. The pros and cons of AnyResGAN and image neural field diffusion models largely derive from the difference between GANs and diffusion models. GANs are still state-of-the-art on FID, especially for the single-class generation. For example, for typical fixed-resolution synthesis on FFHQ, StyleGANv3 [6] (the backbone of AnyResGAN) reports FID 2.79 while LDM [10] (the backbone of our implementation) has FID at 4.98.

Model	pFID@50K		
	256/1K	512/1K	1K/1K
AnyRes-GAN [3]	6.17	4.02	3.25
INFD	7.53	6.84	5.13

Table 2. FID comparison between any-resolution GAN and diffusion model on Mountains dataset. While GANs are state-of-the-art at single class FID, the diffusion-based method achieves competitive FID and does not have the GAN artifacts shown in Figure 7, and shows better visual quality on actual images. We refer to Sec. G for random samples and detailed discussions.

However, FID is counting for the statistics of deep features and is not a perfect metric yet for image generation [1, 2]. Image neural field diffusion model inherits the advantages of diffusion models over GANs. We detail below.

Sample quality. While our method achieves competitive FID to AnyResGAN, as a diffusion model, we find it shows better quality in actual samples. From the samples, we observe that AnyResGAN commonly shows artifacts of regular patterns (e.g. grid in the sky and water as shown in Figure 7, lines on the mountains, and array-like trees). We also find the best samples from our method have better quality than the best samples from AnyResGAN.

Sample diversity. Besides the results we discussed above, we demonstrate random samples from ground-truth images in Figure 12, AnyResGAN in Figure 13, and image neural field diffusion model's in Figure 14. Overall, we observe that AnyResGAN's samples are more in flat and simple layouts (typically a front view of a mountain, with a horizontal sky-mountain line), while the diffusion-based method has better sample diversity (usually more layers and contents along the depth in the layout).

Besides, unlike AnyResGAN, INFD is scale-consistent as shown in Figure 3. Finally, INFD can be used for text-to-image synthesis (more samples are in Figure 10,11), which remains a challenge for any-resolution GANs.

H. Discussion on dataset scale-consistency

Our model assumes the images in the dataset to be scale-consistent, i.e., downsampled high-resolution images follow the same distribution as low-resolution images. Datasets that severely violate this assumption would hurt the performance of our model. This is because if dataset scale consistency is violated, the latent code encoded from low-resolution images might follow a different distribution than the latent code encoded from downsampled high-resolution images. The high-resolution supervision is only applied to the latter type of latent code during training, therefore, the former type of latent code might not have a guaranteed quality when rendered at high resolution. For text-to-image synthesis, our current model is finetuned with clean high-resolution data from the Stable Diffusion model,

Figure 7. Examples of artifacts in AnyResGAN [3] on FFHQ (black dots) and Mountains (grid-like patterns). Image neural field diffusion model is based on diffusion models and avoids these artifacts (see random examples in Figure 8) and the main paper's Figure 4.

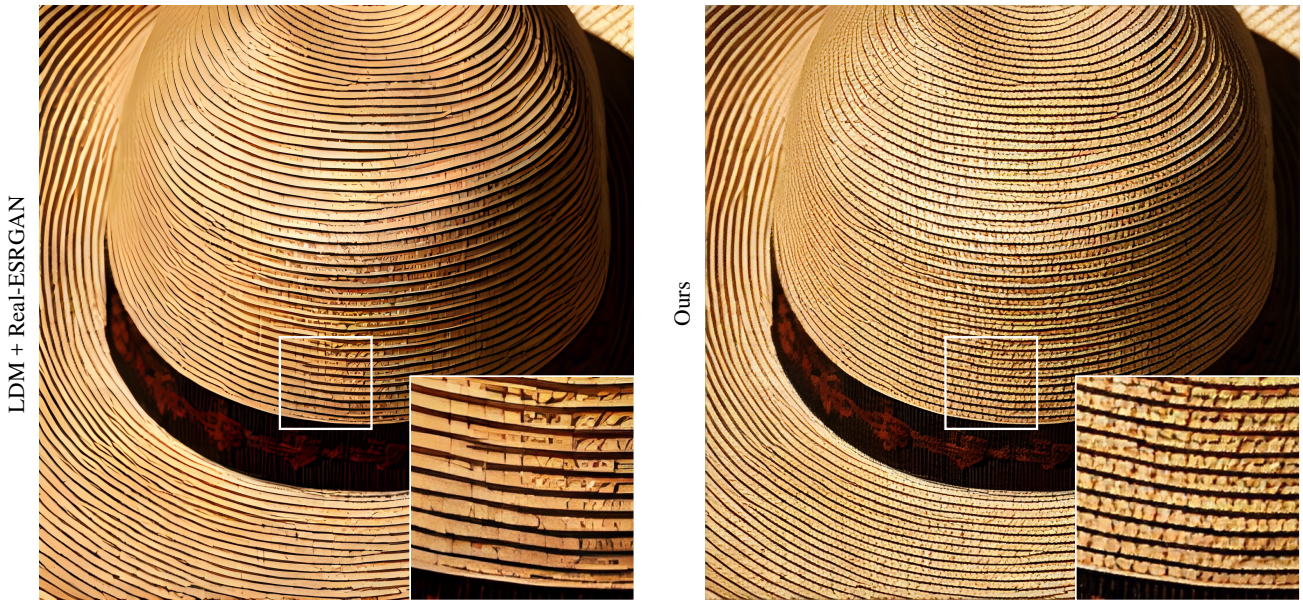
while the Stable Diffusion model could have seen many noisy low-resolution images in its pre-training, appending text prompts like "high definition" or "4k" reduces such distribution shift and thus improves the quality. Training from scratch with scale-consistent data will not have a distribution shift in the latent space and might thus avoid this issue.

I. Additional Generated Samples

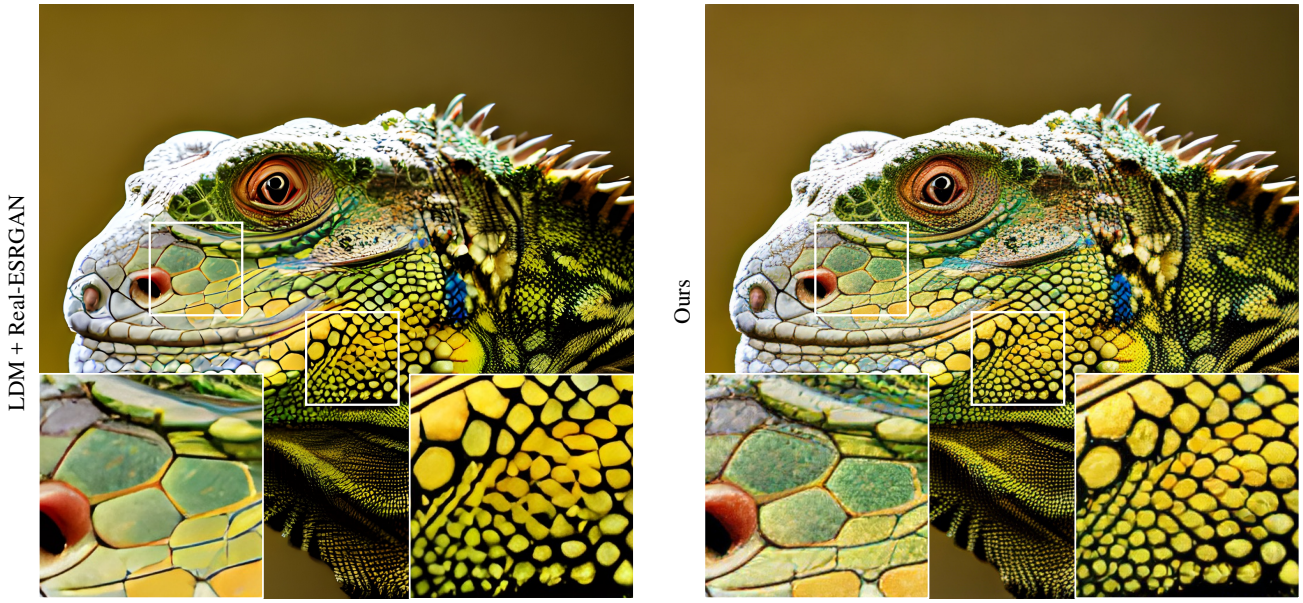
We show additional generated samples in Figure 8,9,10,11 for FFHQ-1024, 2K portrait dataset, and text-to-image generation at 2K resolution, where the experimental settings are the same as the main paper. In text-to-image generation, we observe that while the output resolution is at 2K, appending the prompt "high definition" or "4k" after the text description is helpful to generate high-quality high-resolution images.

Figure 8. Additional generated samples of our method on FFHQ-1024.

Figure 9. Additional generated samples of our method on 2K portrait dataset.

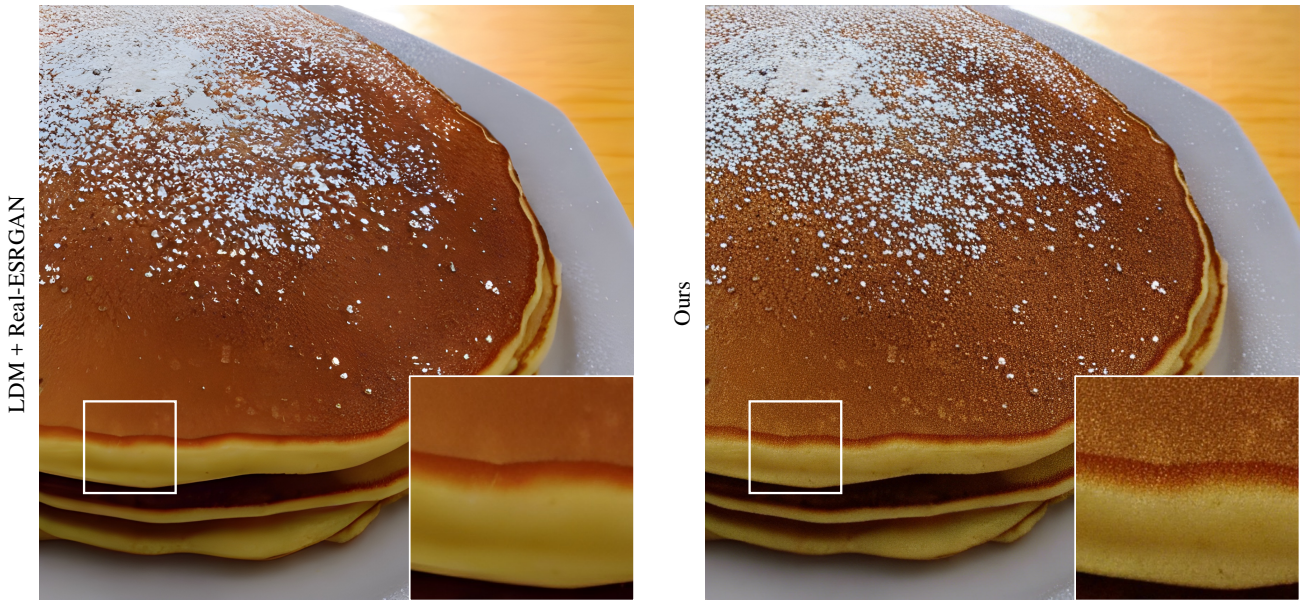


"A straw hat, high definition, 4k"

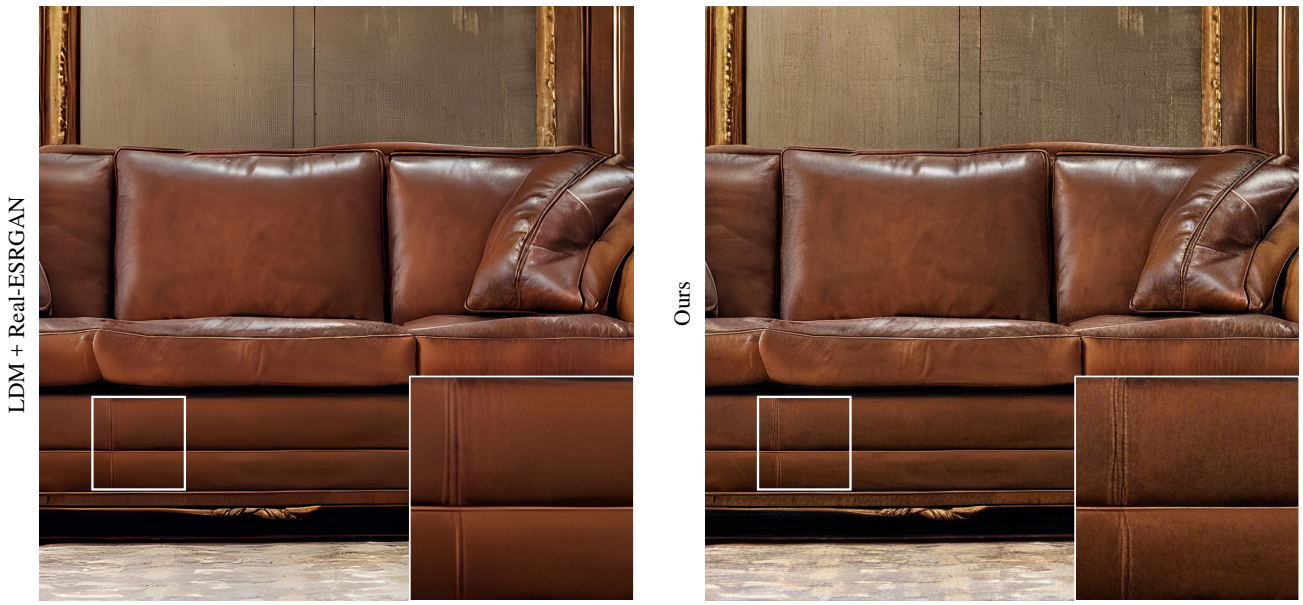


"Portrait of a colored iguana, high definition, 4k"

Figure 10. Additional generated samples of our method on text-to-image generation (resolution at 2K).



"A pancake, high definition, 4k"



"A leather sofa, high definition, 4k"

Figure 11. Additional generated samples of our method on text-to-image generation (resolution at 2K).

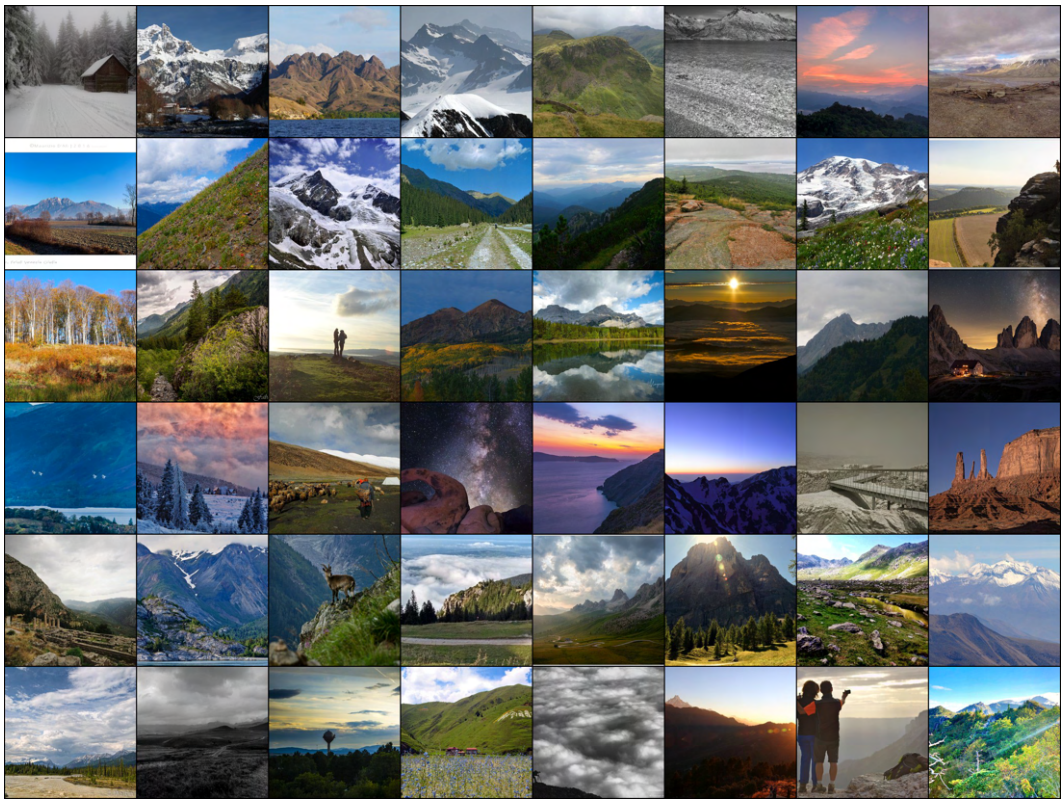


Figure 12. Sample diversity of ground-truth images on Mountains dataset (shown in 256×256).

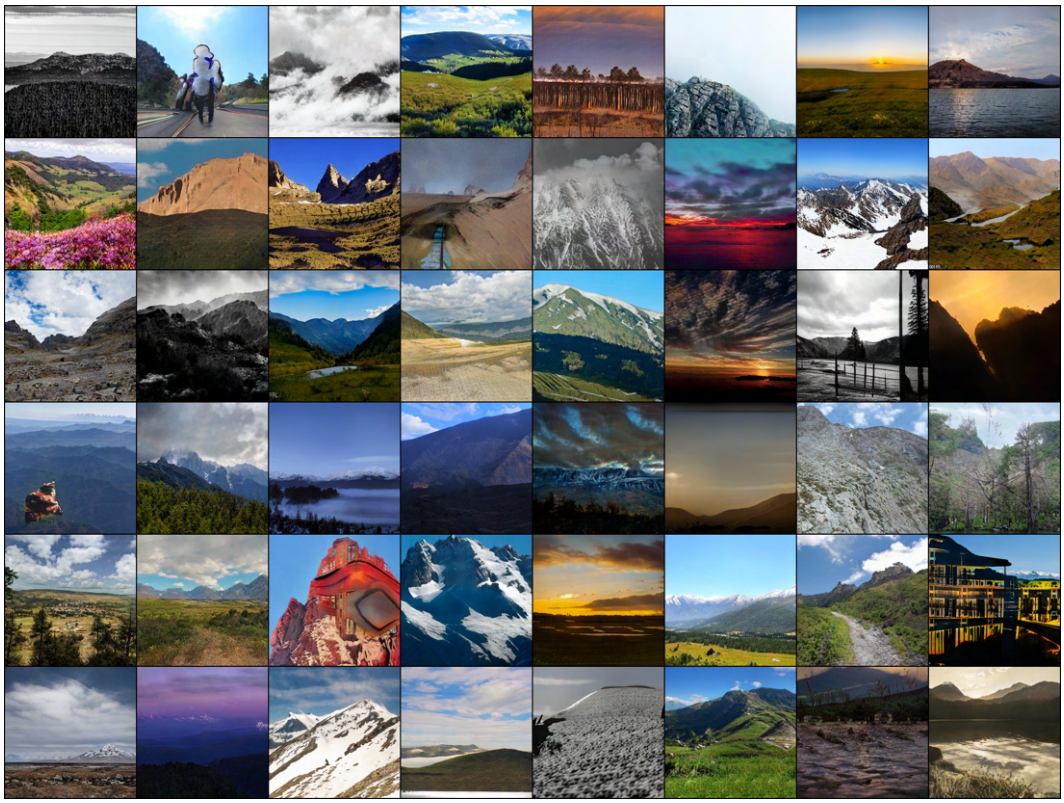


Figure 13. Sample diversity of AnyResGAN [3] on Mountains dataset (shown in 256×256).

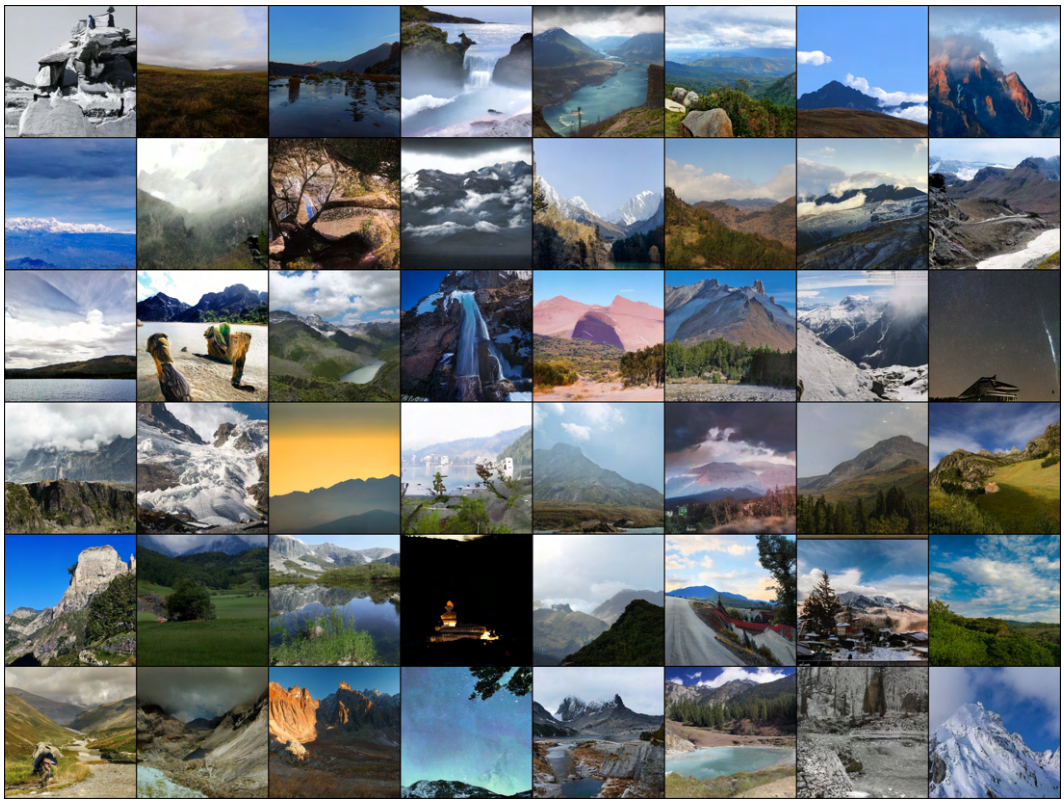


Figure 14. Sample diversity of image neural field diffusion model on Mountains dataset (shown in 256×256).

J. Implementation Details

J.1. Model architecture

Our encoder and decoder follow the architecture used in LDM [10]. They are modified from a UNet's encoder and decoder by removing the connections that skip the bottleneck latent representation. The encoder and decoder are symmetric, each has 3 levels. Downsampling/upsampling happens after each level. The base channel is 128, the channel multiplication factors are 1,2,4 for different levels in the encoder. There are 2 ResNet blocks within each level. The feature map of latent representation has a downsampling rate of 4 compared to the input and has 3 channels. The CLIF renderer is a convolutional neural network with one convolution layer, and two ResNet blocks, followed by another convolution layer, convolution kernel sizes are all 3.

Our diffusion model in latent space follows the implementation in ADM [5] (also used in LDM [10]). The encoder and decoder have base channels 224 and channel multiplication factors are 1,2,3,4 at different encoder levels, with 2 ResNet blocks at every level. At the downsampling rates of 2,4,8, multi-head self-attention with 32 channels per head is applied on the feature map.

J.2. Training setting

For the first stage, the encoder, decoder, and renderer are end-to-end trained jointly. We use Adam [7] with $\beta_1 = 0.5, \beta_2 = 0.9$ and optimize for 1M iterations. The learning rate is $3.6 \cdot 10^{-5}$ for a batch size of 8. The discriminator for GAN loss is adversarially trained with the same optimizer specifications. On Mountains and text-to-image tasks, we keep the ground-truth images at their original resolution with a probability of 0.5 in the last 400K iterations.

For the second stage, the latent space diffusion model is trained with AdamW [8] for 600K iterations on FFHQ, for 1.7M iterations on Mountains, with $\beta_1 = 0.9, \beta_2 = 0.999$ and weight decay of 0.01. The learning rate is $9.6 \cdot 10^{-5}$ for a batch size of 48. For either the first stage or the second stage, it takes about 4 days every 1M iterations to train our model on 4 NVIDIA A100 GPUs.

References

[1] Ali Borji. Pros and cons of gan evaluation measures. *Computer Vision and Image Understanding*, 179:41–65, 2019. 3

[2] Ali Borji. Pros and cons of gan evaluation measures: New developments. *Computer Vision and Image Understanding*, 215:103329, 2022. 3

[3] Lucy Chai, Michael Gharbi, Eli Shechtman, Phillip Isola, and Richard Zhang. Any-resolution training for high-resolution image synthesis. *arXiv preprint arXiv:2204.07156*, 2022. 1, 3, 4, 10

[4] Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8628–8638, 2021. 1

[5] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 12

[6] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34:852–863, 2021. 3

[7] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 12

[8] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 12

[9] Evangelos Ntavelis, Mohamad Shahbazi, Iason Kastanis, Radu Timofte, Martin Danelljan, and Luc Van Gool. Arbitrary-scale image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11533–11542, 2022. 3

[10] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 2, 3, 12