Efficient Scale-Invariant Generator with Column-Row Entangled Pixel Synthesis — Supplementary Material —

Thuan Hoang Nguyen* Thanh Van Le* Anh Tran
VinAI Research, Hanoi, Vietnam
{v.thuannh5,v.thanhlv19,v.anhtt152}@vinai.io

Abstract

In this supplementary PDF, we first provide implementation details, including network architecture and training config in Sec. 1. Then we include extra quantitative results, including the transfer learning result on AFHQ-Dog, an extra comparison of CREPS with AnyresGAN and ScaleParty, and an additional ablation study in Sec. 2. Finally, we show additional qualitative results in Sec. 3. Besides, we include two extra image samples at high resolution, a qualitative video, and our code at here.

1. CREPS Implementation Details

1.1. Architecture details

In this section, we describe in detail the implementation of each component in our proposed method.

Synthesis Block As reported in the main paper, this block is largely identical to the blocks in [4] with some minor modifications. First, we change the kernel size of the convolution operator from 3×3 to 1×1 one. Second, we dismiss the use of small injection noise to the feature output as it is against our objective of scale-invariant generation. Third, we double the number of channels in this block compared to [4] to improve the capacity of the bi-line features. This block both receives and outputs bi-line features.

Refinements Block consists of two synthesis blocks with the hidden width of 128 and 64, respectively (Tab. 1). Instead of bi-line features, this block input and output are both 2D features; the input feature map is decoded from previous bi-line features. The residual output of each synthesis block will be an RGB image in the shape of $3 \times R \times R$.

Decoder Block is a stack of fully-connected layers with LeakyReLU activations in-between. The structure of this block is illustrated in Tab. 2.

[1]Equal contribution.

1.2. Transfer learning details

Similar to Karras et al. [3], we train MetFaces and AFHQ-Dog (next section) with adaptive discriminator augmentation (ADA) [3] using weights trained on FFHQ-512. Even though our FFHQ was trained with resolution 512×512 only, we can easily train on resolution 1024×1024 simply by doubling the length of row and column coordinates e^r and e^c. The transfer learning results are reported in Sec. 2.

1.3. Training config

To train our models, we start with the batch size of 128 and gamma of 0.5 for resolution 128×128. For higher resolution, we decrease the batch size and increase gamma to further stablilize the training. Specifically, for resolution 512×512, we use 32 and 10 for batch size and gamma respectively. Lastly, we set the batch size and gamma as 8 and 32 for resolution 1024×1024

Layer	Input Shape	Output Shape
SynthesisBlock(32, 128)	$32 \times R \times R$	$128 \times R \times R$
ToRGB(128, 3)	$128 \times R \times R$	$3 \times R \times R$
SynthesisBlock(128, 64)	$32 \times R \times R$	$64 \times R \times R$
ToRGB(64, 3)	$64 \times R \times R$	$3 \times R \times R$

Table 1. Structure of Refinements Block.

Layer	Input Shape	Output Shape
Fusion	$32 \times R \times 2D$	$R \times R \times 32$
Linear(32, 64)	$R \times R \times 32$	$R \times R \times 64$
Linear(64, 128)	$R \times R \times 64$	$R \times R \times 128$
Linear(128, 64)	$R \times R \times 128$	$R \times R \times 64$
Linear(64, 32)	$R \times R \times 64$	$R \times R \times 32$
Permute	$R \times R \times 32$	$32 \times R \times R$

Table 2. Structure of Decoder Block.

2. Additional Quantitative Results

2.1. Transfer Learning Results on AFHQ-Dog

Besides MetFaces, we conduct a further experiment to verify the adaptability of our model from FFHQ to AFHQ-Dog. AFHQ-Dog consists of 4677 facial images of various dog breeds at resolution 1024×1024. Following prior works [3], we directly use the weight of CREPS trained on FFHQ and continue the training on AFHQ-Dog. Our model achieved an FID score of 9.7, which is slightly higher than the FID score of StyleGAN2-ADA (7.4). However, qualitatively, the images generated by this model are of good quality as illustrated in Fig. 5.

2.2. Comparison With AnyresGAN and ScaleParty

We provide an additional comparison in terms of FID score with two prior works that support any-scale image synthesis, including AnyresGAN [2] and ScaleParty [5], in Tab. 3. Note that both of them make use of spatial convolution, so they are not scale-consistent. Here, the FID scores of AnyresGAN are taken directly from the paper, while those for ScaleParty are re-computed using their publicly available code and pre-trained model.

2.3. Additional Ablation Study

In this section, we measure the influences of the decoder in our proposed asymmetric fusion on CREPS's performance. We omit the decoder π between synthesis blocks and simplify the fusion scheme to $E^{(l+1)} = E^{(l)} + F^{(l+1)}$. We ran this config on FFHQ resolution 128×128 and reported the result in Tab. 4. As can be seen, the FID score of this variant is even worse than the smaller config with d=4, which proves that adding decoders to aggregate information across channels can largely improve the generation quality.

3. Additional Qualitative Results

3.1. Super-resolution Comparison

By scaling the length of row and column coordinates e^r and e^c, CREPS can not only generate higher output resolution but also produce finer details. As shown in Fig. 1, the crop of an image generated by scaling the coordinate of CREPS from 512 to 2048 has more details than directly applying Lanczos upsampling on the corresponding image generated at resolution 512×512.

Additionally, we also provide two images of resolution 6144×6144 at here. Even though the images are not as sharp as real ultra-high-resolution ones, it can be seen that our produced images are much better than ones produced by classical upsampling methods.

3.2. Scale Consistency Comparison

We provide a scale-consistency comparison video of CREPS with previous any-scale synthesis architectures, including AnyresGAN [2], ScaleParty [5], and CIPS [1] at here. Note that, for a fair comparison, we use the provided codes from each method to produce the video except for ScaleParty, where we obtain the video directly from their GitHub codebase [1]. For clearer visualization, we highlight the crop with the largest changes in each method's picture with a blue square.

As can be seen in the video, ScaleParty performs the worst among the four methods, while AnyresGAN still maintains a good degree of consistency since it inherits all the advantages of StyleGAN3. However, it is still behind INR-GANs, CIPS, and CREPS since it shows more prominent changes in the highlighted crop compared with the latter.

3.3. Additional Geometric Transformation Results

Apart from scaling, our model can be used to sample other types of geometric transformation, such as translation, rotation, and distortion. Even though it should be stressed that our model is geometry-consistent by nature with the use of fully-connected layers, we also present some qualitative results in the video at here to further verify our claim.

3.4. Additional Image Generation Results

We provide additional results generated by CREPS on FFHQ and LSUN-Church in Figs. 2 and 3. We further verify that our proposed bi-line representation does not limit the capacity of our models by performing transfer learning from FFHQ-512 to MetFaces and AFHQ-Dog. The results are shown in Fig. 4 and Fig. 5, respectively.

References

[1] Ivan Anokhin, Kirill Demochkin, Taras Khakhulin, Gleb Sterkin, Victor Lempitsky, and Denis Korzhenkov. Image generators with conditionally-independent pixel synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[2] Lucy Chai, Michael Gharbi, Eli Shechtman, Phillip Isola, and Richard Zhang. Any-resolution training for high-resolution image synthesis. In *European Conference on Computer Vision (ECCV)*, 2022. 2, 3

[3] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 1, 2

[4] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[1] https://github.com/vglsd/ScaleParty

Generator	FFHQ-512	FFHQ-1024	LSUN Church-256
ScaleParty	6.23†	10.91†	N/A
AnyresGAN	3.71*	4.06*	3.84*
CREPS (ours)	4.43	4.09‡	5.50

Table 3. Comparison of our method against other works in FID metric. '*' means the result is taken from original paper [2]. '†' means the result is obtained by re-computing the score using the code from author. '‡' means the result is obtained by scaling the output resolution of the FFHQ-512 model. N/A means the pretrained weight for this dataset is not released.

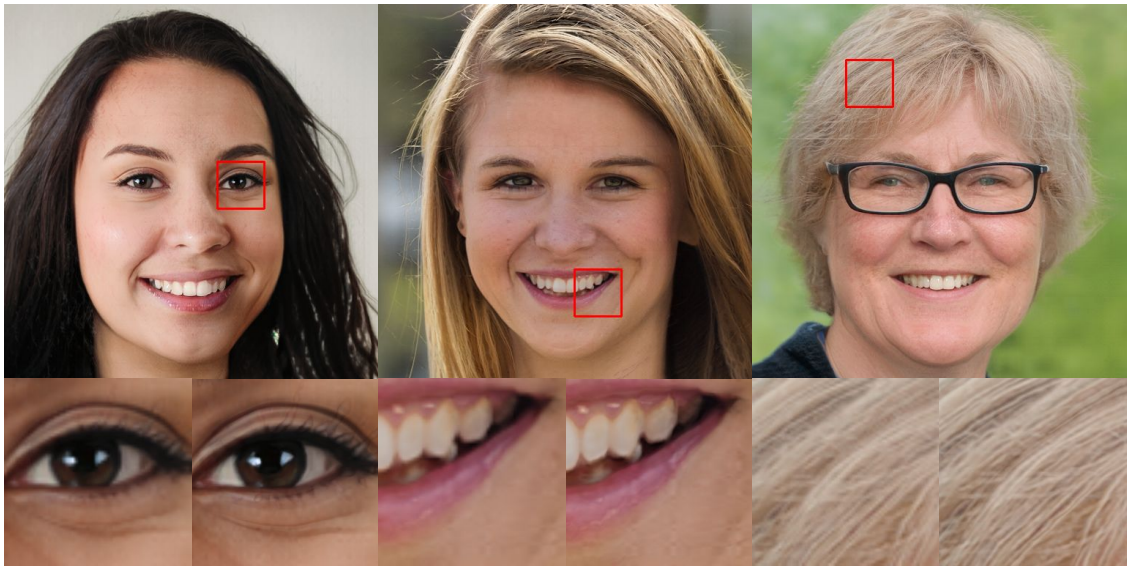

Figure 1. Comparison of CREPS high-resolution image synthesis with Lanzcos upsampling on FFHQ. Top: Images synthesized by CREPS at resolution 512×512. Bottom left: the crop at resolution 512×512, upscaled with Lanczos upsampling. Bottom right: the corresponded crop of CREPS at resolution 2048×2048.

Configuration	FID	Memory	Time
+ bi-line and d=8	8.23	1.6GB	0.03s
+ no decoders and d=8	6.91	1.7GB	0.03s
+ multiple decoders and d=4	6.46	1.6GB	0.03s
+ multiple decoders and d=8	4.66	1.7GB	0.04s

Table 4. Effects of the modifications of CREPS on the FFHQ dataset in terms of FID score, memory usage, and running time.

[5] Evangelos Ntavelis, Mohamad Shahbazi, Iason Kastanis, Radu Timofte, Martin Danelljan, and Luc Van Gool. Arbitrary-scale image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022. 2

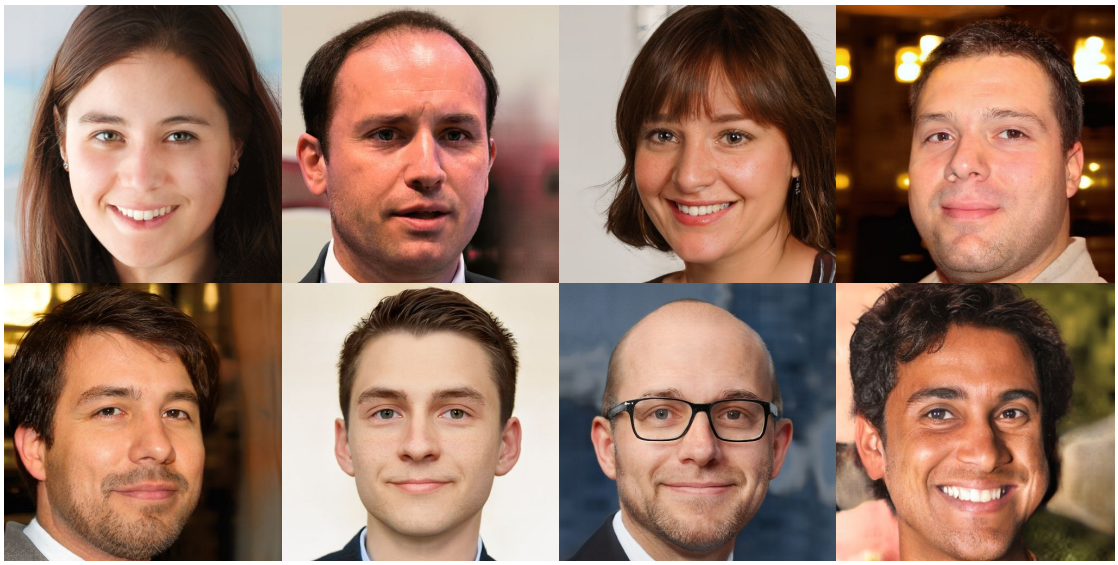

Figure 2. Sample images generated by our models on FFHQ resolution 512×512

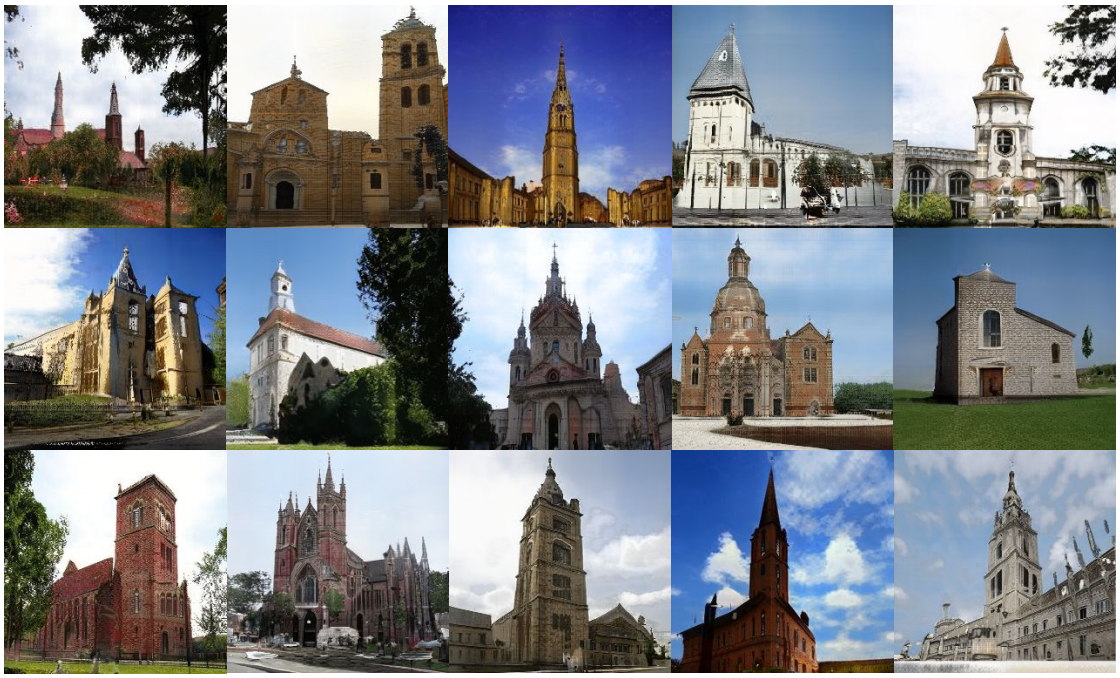

Figure 3. Sample images generated by our models on LSUN Church resolution 256×256

Figure 4. Sample images generated by our models on MetFaces resolution 1024×1024

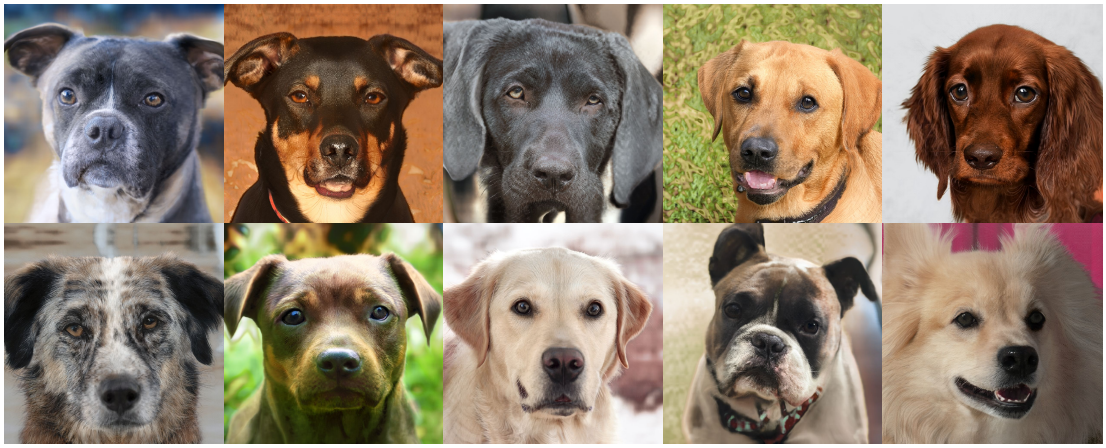

Figure 5. Sample images generated by our models on AFHQ-Dog resolution 1024×1024