Generative Hierarchical Features from Synthesizing Images Supplementary Material

Yinghao Xu* Yujun Shen* Jiapeng Zhu Ceyuan Yang Bolei Zhou
The Chinese University of Hong Kong
{xy119, sy116, jpzhu, yc019, bzhou}@ie.cuhk.edu.hk

Table 1. **Encoder Structure**, which is based on ResNet-50 [4]. Fully-connected (FC) layers are employed to map the feature maps produced by the Spatial Alignment Module (SAM) to our proposed Generative Hierarchical Features (GH-Feat). GH-Feat exactly align with the multi-scale style codes used in StyleGAN [6]. The numbers in brackets indicate the dimension of features at each level.

Stage	Encoder Pathway	Output Size	SAM	FC Dimension	GH-Feat	Style Code in StyleGAN
input	–	3×256^2				
conv_1	7×7, 64 stride 2, 2	64×128^2				
pool_1	3×3, max stride 2, 2	64×64^2				
res_2	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	256×64^2				
res_3	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	512×32^2				
res_4	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 6$	1024×16^2	512×4^2	8192×1792	Level 1-2 Level 3-4 Level 5-6	Layer 14-13 ($128d \times 2$) Layer 12-11 ($256d \times 2$) Layer 10-9 ($512d \times 2$)
res_5	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$	2048×8^2	512×4^2	8192×4096	Level 7-8 Level 9-10	Layer 8-7 ($1024d \times 2$) Layer 6-5 ($1024d \times 2$)
res_6	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 1$	2048×4^2	512×4^2	8192×4096	Level 11-12 Level 13-14	Layer 4-3 ($1024d \times 2$) Layer 2-1 ($1024d \times 2$)

A. Overview

This supplementary material is organized as follows

- Sec. B describes the detailed structure of the proposed hierarchical encoder.
- Sec. C gives more discussion on the ImageNet experiment presented in the main paper.
- Sec. D conducts ablation study on training the generator together with the encoder.
- Sec. E visualizes some style mixing results on human faces.

B. Encoder Structure

Tab. 1 provides the detailed architecture of our hierarchical encoder by taking a 14-layer StyleGAN [6] generator as an instance. Recall that the design of GH-Feat treats the layer-wise style codes used in the StyleGAN model (*i.e.*, the code fed into the AdaIN module [5]) as generative features. Accordingly, GH-Feat consists of 14 levels that exactly align with the multi-scale style codes yet in a reverse order, as shown in the last two columns of Tab. 1. In particular, these features are projected from the feature maps produced by the last three stages via fully-connected layers.

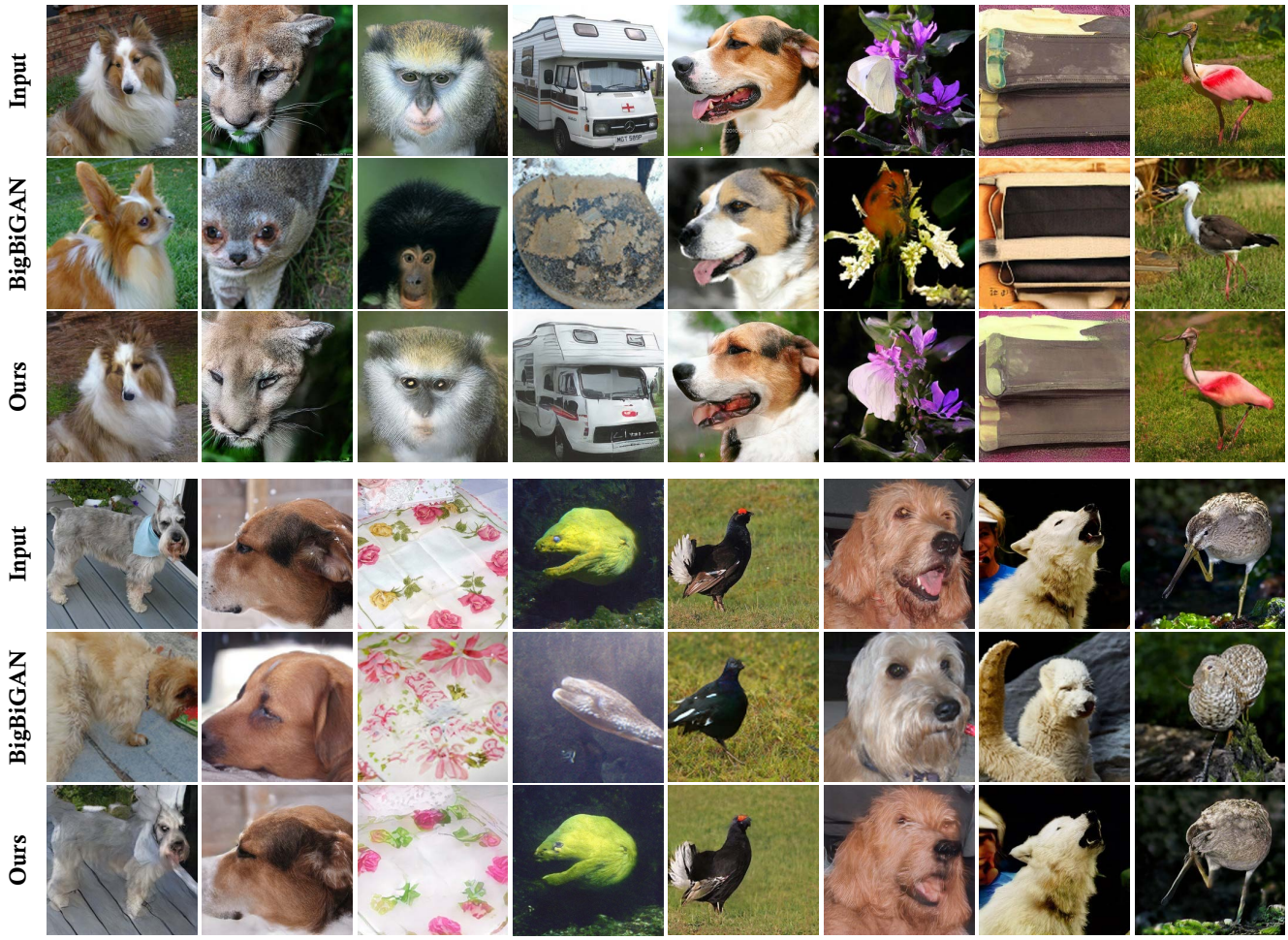

Figure 1. Qualitative comparison between BigBiGAN [3] and GH-Feat on reconstructing images from ImageNet [2].

C. More Details and Results on ImageNet

Training Details. During the training of the StyleGAN model on the ImageNet dataset [2], we resize all images in the training set such that the short side of each image is 256, and then centrally crop them to 256×256 resolution. All training settings follow the StyleGAN official implementation [7], including the progressive strategy, optimizer, learning rate, *etc*. The generator and the discriminator are alternatively optimized until the discriminator have seen $250M$ real images. After that, the generator is fixed and treated as a well-learned loss function to guide the training of the encoder. During the training of the hierarchical encoder, images in the training collection are pre-processed in the same way as mentioned above. After the encoder is ready (usually trained for 25 epochs), we treat it as a feature extractor. We use the output feature map at the "res_5" stage (with dimension 2048×8^2), apply adaptively average pooling to obtain 2×2 spatial feature, and vectorize it. A linear classifier, *i.e.*, with one fully-connected layer, takes these extracted features as the inputs to learn the image classification task. SGD optimizer, together with batch size 2048, is used. The learning rate is initially set as 1 and decayed to 0.1 and 0.01 at the 60-th and the 80-th epoch respectively. During the training of the final classifier, ResNet-style data augmentation [4] is applied.

Discussion. As shown in the main paper, GH-Feat achieves comparable accuracy to existing alternatives. Especially, among all of methods based on generative modeling, GH-Feat obtains second performance only to BigBiGAN [3], which requires incredible large-scale training.[1] However, our GH-Feat facilitates a wide rage of tasks besides image classification. Taking image reconstruction as an example, our approach can well recover the input image, significantly outperforming BigBiGAN [3]. As shown in Fig. 1, BigBiGAN can only reconstruct the input image from the category level (*i.e.*, dog or bird). By contrast, GH-Feat is able to recover more details, like shape and texture.

[1] As reported in [1], the model train on images of 256×256 resolution requires 256 TPUs running for 48 hours.

Table 2. Quantitative comparison on image reconstruction between training the generator from scratch together with the encoder and our GH-Feat that treats the well-learned StyleGAN generator as a loss function.

	MSE↓	SSIM↑	FID↓
Training $G(\cdot)$ from Scratch	0.429	0.301	46.20
GH-Feat (Ours)	0.0464	0.558	18.48

D. Ablation Study

Recall that, during the training of the encoder, we propose to treat the well-trained StyleGAN generator as a learned loss function. In this part, we explore what will happen if we train the generator from scratch together with the encoder. Tab. 2 and Fig. 2 show the quantitative and qualitative results respectively, which demonstrate the strong performance of GH-Feat. It suggests that besides higher efficiency, reusing the knowledge from a well-trained generator can also bring better performance.

E. Style Mixing

In this part, we verify the hierarchical property of GH-Feat on the task of style mixing and further make comparison with ALAE [8]. In particular, we use ALAE and our approach to extract features from same images (including both style images and content images) and then use the extracted features for style mixing at different levels.

Fig. 3 shows the comparison results. Note that all test images are selected following the original paper of ALAE [8]. We can see that when mixing high-level styles, the pose, age, and gender of mixed results are close to those of style images. By comparing with ALAE, results using GH-Feat better preserve the identity information (high-level feature) from style images as well as the color information (low-level feature) from content images. In addition, when mixing low-level styles (bottom two rows), both ALAE and GH-Feat can successfully transfer the color style from style images to content images, but GH-Feat shows much stronger identity preservation.

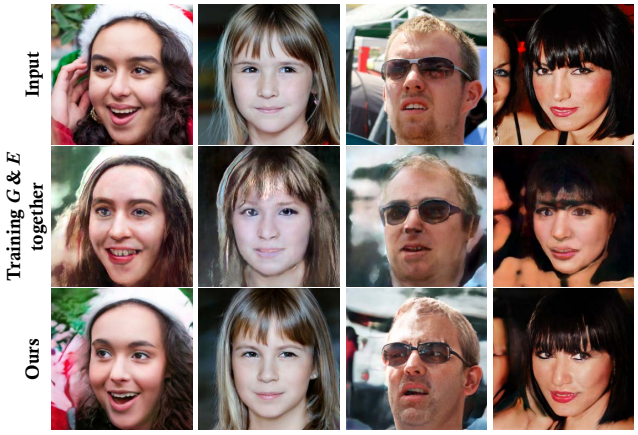

Figure 2. Qualitative comparison on image reconstruction between training the generator from scratch together with the encoder, and our GH-Feat that treats the well-learned StyleGAN generator as a loss function.

References

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

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2009. 2

[3] Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. In *Adv. Neural Inform. Process. Syst.*, 2019. 2

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 1, 2

[5] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Int. Conf. Comput. Vis.*, 2017. 1

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

[7] Tero Karras, Samuli Laine, and Timo Aila. Stylegan - official tensorflow implementation. https://github.com/NVlabs/stylegan, 2019. 2

[8] Stanislav Pidhorskyi, Donald Adjeroh, and Gianfranco Doretto. Adversarial latent autoencoders. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 3, 4

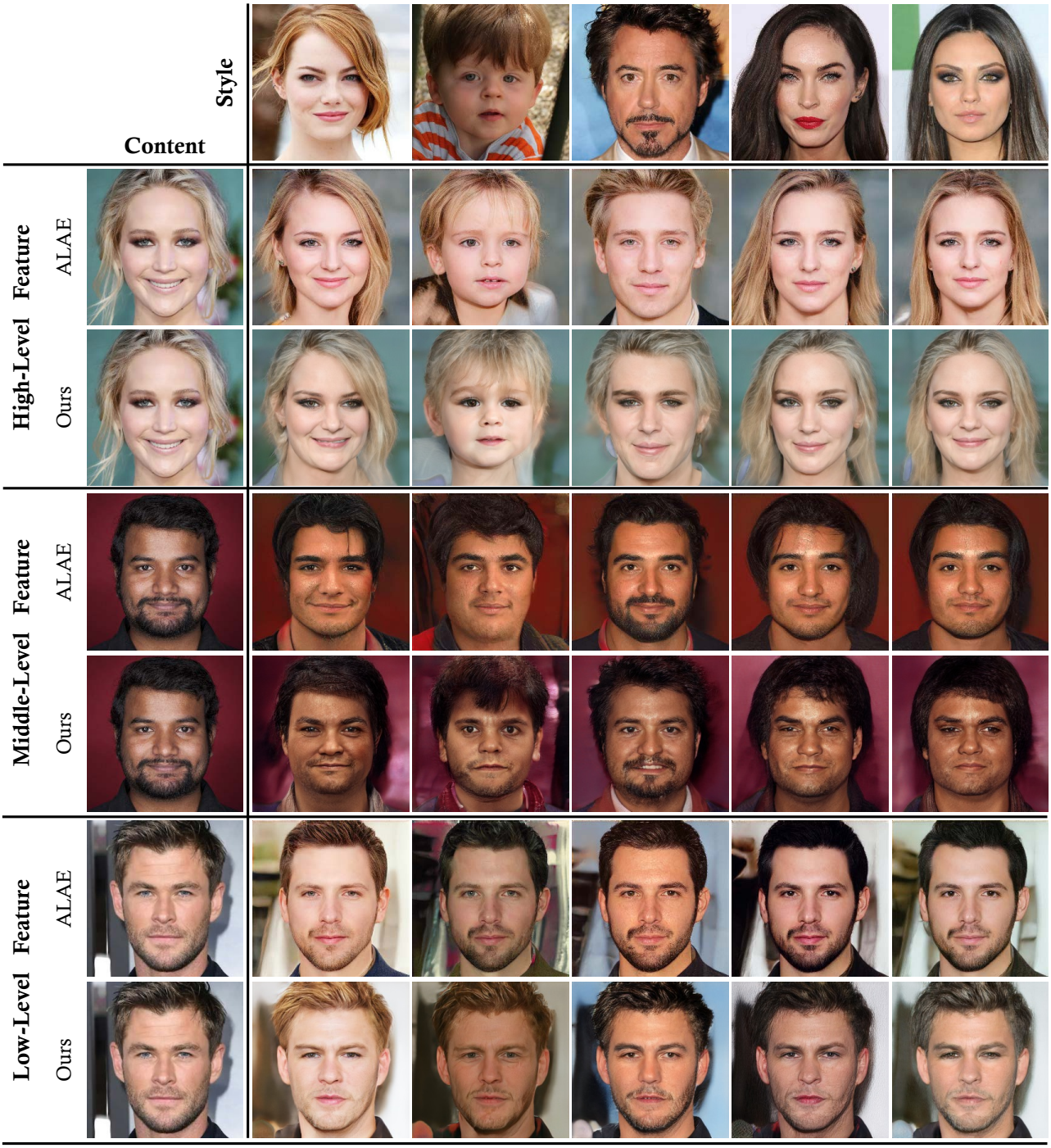

Figure 3. Qualitative comparison between our proposed GH-Feat and ALAE [8] on the style mixing task. After extracting features from both content images and style images, we replace different levels of features from content images with those from style images.