

# SUPPLEMENTARY MATERIAL FOR “A STYLEMAP-BASED GENERATOR FOR REAL-TIME IMAGE PROJECTION AND LOCAL EDITING”

**Anonymous authors**

Paper under double-blind review

## A IMPLEMENTATION DETAILS

**Architecture.** We follow StyleGAN2 (Karras et al., 2020) regarding the discriminator architecture and the feature map counts in the convolution layers of the synthesis network. Our mapping network is an MLP with eight fully connected layers followed by a reshape layer. The channel sizes are 64 except the last being 4,096. Our encoder adopts the discriminator architecture until the  $8 \times 8$  layer and without minibatch discrimination.

**Training.** We jointly train the generator, the encoder and the discriminator. It is simpler and leads to more stable training and higher performance than separately training the adversarial networks and the encoder. For the rest, we mostly follow the settings of StyleGAN2, *e.g.*, the discriminator architecture, Adam optimizer with the same hyperparameters, exponential moving average of the generator and the encoder, leaky ReLU, equalized learning rate for all layers, random horizontal flip for augmentation, and reducing the learning rate by two orders of magnitude for the mapping network. Our code is based on unofficial PyTorch implementation of StyleGAN2<sup>1</sup>. All StyleMapGAN variants on comparison are trained for two weeks on 5M images with two Tesla V100 GPUs using minibatch size of 16. We note that most cases keep slowly improving until 10M images. Our code will be publicly available online for reproduction<sup>2</sup>.

**Losses.** Here we use G, D and E as short forms of the generator, the discriminator and the Encoder. The adversarial loss for G and D are non-saturating loss (Goodfellow et al., 2014).  $R_1$  regularization term (Mescheder et al., 2018) is computed every 16 steps for D. G and E are trained with image reconstruction loss (MSE) and perceptual loss (Johnson et al., 2016). Domain-guiding loss (Zhu et al., 2020a) is applied to all networks. Table 1 summarizes the losses.

Loss	Generator	Discriminator	Encoder
Adversarial loss	✓	✓	
$R_1$ regularization		✓	
Latent reconstruction			✓
Image reconstruction	✓		✓
Perceptual loss	✓		✓
Domain-guided loss	✓	✓	✓

Table 1: Losses for training each network.

## B MAPPING NETWORK DESIGN FOR THE STYLEMAP

There are several choices when designing a mapping network. We can easily think of convolutional layers due to the spatial dimensions of the stylemap. Alternatively, we can remove the mapping

<sup>1</sup><https://github.com/rosinality/stylegan2-pytorch>

<sup>2</sup><http://publicurl.com>

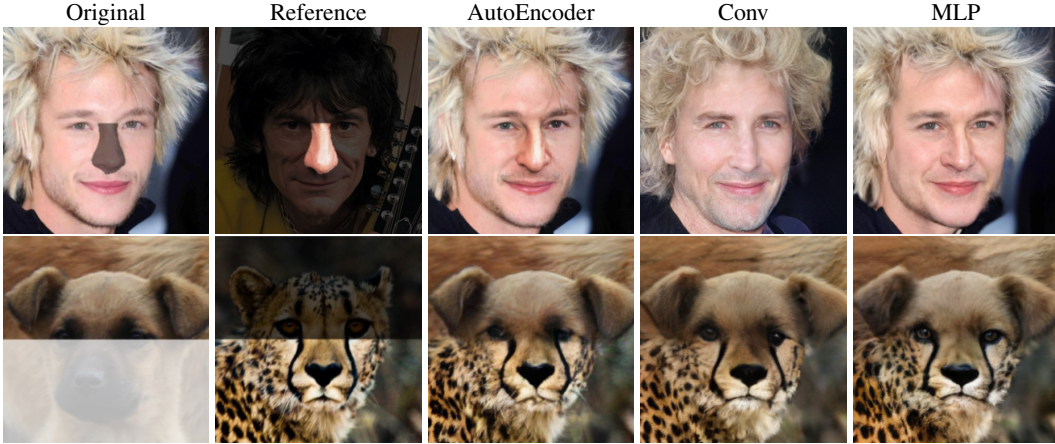


Figure 1: Local editing comparison across different mapping network structures in the generator. The autoencoder method without a mapping network is most unnatural in a modified image. The mapping network with convolutional layers has more natural results than the autoencoder. Nevertheless, due to its bad reconstruction quality, it suffers from preserving the characteristics of original images. Our MLP mapping network is natural in local editing and preserves the original image well. Also, even if the eye part is not properly inserted like the animal image, it naturally creates it.

network so that our method does not generate images from the standard Gaussian distribution, and uses only real images for training like autoencoder (Hinton & Salakhutdinov, 2006). As shown in Figure ??, autoencoder fails to produce realistic images using the projected stylemap. It seems to copy and paste between two images on RGB space. We give continuous input to the generator from the standard Gaussian distribution using a mapping network, letting the network generate seamless images in image space. However, the autoencoder only gives discrete input, which is projected from the encoder. On the other hand, the mapping network with convolutional layers often struggles in reconstruction so that the edited results images are quite different from the original images. We assume that there is such a limit because the convolutional layer’s mapping is bounded to the local area. In MLP, each weight and input are fully-connected so that it can make a more plausible latent space.

## C RELATED WORK

### C.1 LATENT-BASED IMAGE EDITING

There are active studies (Abdal et al., 2019; Collins et al., 2020; Zhu et al., 2020a) on image editing using latent vector arithmetic where well-trained GANs (Karras et al., 2019; 2020) are adopted for real-world applications. These studies aim to find a latent vector to reconstruct an original image. In general, there are two approaches to embed images into latent vectors, learning and optimization-based ones. The learning-based approach (Zhu et al., 2020a; Perarnau et al., 2016; Zhu et al., 2016) trains an encoder that maps a given image to a latent vector. This method has a potential of projecting an image in real time. However, the existing methods suffer from low quality of the reconstructed images, which indicates the difficulty of embedding real images. The optimization-based approach (Creswell & Bharath, 2018; Lipton & Tripathi, 2017; Ma et al., 2018; Abdal et al., 2019), given an input image, aims at optimizing the latent vector to minimize the pixel-wise reconstruction loss. Though it is not feasible to project images in real time due to its iterative nature, it exhibits high quality of the reconstructed images while enabling edits include global changes in semantic attribute, e.g. smiling, beard, etc. Compared with these approaches, our StyleMapGAN can project images in real time while offering high quality of reconstruction images.

## C.2 LOCAL EDITING

Several methods (Collins et al., 2020; Alharbi & Wonka, 2020; Zhu et al., 2020b) tackle locally editing specific parts (e.g., nose, background) as opposed to the most GAN-based image editing methods modifying global appearance. Editing in style (Collins et al., 2020) tries to identify components in the style vector which are responsible for specific parts and achieves local editing. It requires preceding component analysis and the correspondence between the components and regions is loose. Structured noise injection (Alharbi & Wonka, 2020) replaces the learned constant from StyleGAN with an input tensor which has spatial dimensions and is a combination of local and global codes. Though it learns some sense of spatial disentanglement, its applicability is limited due to the separate source of variation, the style vector. These two methods are limited to editing fake images while editing real images with them requires projecting the images to the latent space. SEAN (Zhu et al., 2020b) facilitates editing real images by encoding images into the per-region style codes and manipulating them. However, per-region style codes do not capture details and it requires semantic segmentation masks for training. On the other hand, our StyleMapGAN captures and controls fine details of images with a stylemap which has explicit spatial correspondence with images. Our method does not require segmentation masks for training.

## C.3 CONDITIONAL IMAGE SYNTHESIS

Conditional image synthesis models, such as image-to-image translation (Isola et al., 2017; Zhu et al., 2017; Kim et al., 2020), learn to synthesize an output image given an original image. Thanks to this framework, many applications have been successfully built, including colorization (Kim et al., 2019; Larsson et al., 2016; Zhang et al., 2016), image inpainting (Liu et al., 2018; Pathak et al., 2016; Yang et al., 2017), semantic image synthesis and editing (Wang et al., 2018; Chen & Koltun, 2017; Park et al., 2019; Portenier et al., 2018). Recent models extend it to multi-domain and multi-modal (Huang et al., 2018; Lee et al., 2018; Choi et al., 2020). Image-to-image translation and local edit have been separately studied since they target different objectives, *i.e.*, regarding global and local levels of detail in image generation. However, our method can be applied to the both tasks by semantic manipulation of stylemap for image-to-image translation and local manipulation of stylemap. For example, our StyleMapGAN can make only the eyes laugh or the mouth laugh via local editing as well as change the domain of generated image via global semantic manipulation.

## REFERENCES

- Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *CVPR*, 2019.
- Yazeed Alharbi and Peter Wonka. Disentangled image generation through structured noise injection. In *CVPR*, 2020.
- Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 1511–1520, 2017.
- Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *CVPR*, 2020.
- Edo Collins, Raja Bala, Bob Price, and Sabine Susstrunk. Editing in style: Uncovering the local semantics of gans. In *CVPR*, 2020.
- Antonia Creswell and Anil Anthony Bharath. Inverting the generator of a generative adversarial network. *IEEE transactions on neural networks and learning systems*, 30(7):1967–1974, 2018.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. In *NeurIPS*, 2014.
- Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.
- Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018.

- Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial nets. In *CVPR*, 2017.
- Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pp. 694–711. Springer, 2016.
- Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019.
- Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *CVPR*, 2020.
- Hyunsu Kim, Ho Young Jho, Eunhyeok Park, and Sungjoo Yoo. Tag2pix: Line art colorization using text tag with secat and changing loss. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9056–9065, 2019.
- Junho Kim, Minjae Kim, Hyeonwoo Kang, and Kwang Hee Lee. U-gat-it: Unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. In *International Conference on Learning Representations*, 2020.
- Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *European conference on computer vision*, pp. 577–593. Springer, 2016.
- Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Kumar Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *ECCV*, 2018.
- Zachary C Lipton and Subarna Tripathi. Precise recovery of latent vectors from generative adversarial networks. *arXiv preprint arXiv:1702.04782*, 2017.
- Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 85–100, 2018.
- Fangchang Ma, Ulas Ayaz, and Sertac Karaman. Invertibility of convolutional generative networks from partial measurements. In *NeurIPS*, 2018.
- Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. Which training methods for gans do actually converge? In *ICML*, 2018.
- Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2337–2346, 2019.
- Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2536–2544, 2016.
- Guim Perarnau, Joost Van De Weijer, Bogdan Raducanu, and Jose M Álvarez. Invertible conditional gans for image editing. *arXiv preprint*, 2016.
- Tiziano Portenier, Qiyang Hu, Attila Szabo, Siavash Arjomand Bigdeli, Paolo Favaro, and Matthias Zwicker. Faceshop: Deep sketch-based face image editing. *arXiv preprint arXiv:1804.08972*, 2018.
- Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8798–8807, 2018.
- Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6721–6729, 2017.
- Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European conference on computer vision*, pp. 649–666. Springer, 2016.

Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain gan inversion for real image editing. 2020a.

Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A Efros. Generative visual manipulation on the natural image manifold. In *ECCV*, 2016.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *ICCV*, 2017.

Peihao Zhu, Rameen Abdal, Yipeng Qin, and Peter Wonka. Sean: Image synthesis with semantic region-adaptive normalization. In *CVPR*, 2020b.