One Model to Reconstruct Them All: A Novel Way to Use the Stochastic Noise in StyleGAN

Christian Bartz*
christian.bartz@hpi.de

Joseph Bethge*
joseph.bethge@hpi.de

Haojin Yang
haojin.yang@hpi.de

Christoph Meinel
christoph.meinel@hpi.de

Hasso Plattner Institute
University of Potsdam
Potsdam, Germany

* Equal contribution.

Abstract

Generative Adversarial Networks (GANs) have achieved state-of-the-art performance for several image generation and manipulation tasks. Different works have improved the limited understanding of the latent space of GANs by embedding images into specific GAN architectures to reconstruct the original images. In this paper, we investigate the capabilities of the stochastic noise inputs of StyleGAN. We show that the stochastic noise inputs of a StyleGAN model can be used to transfer content and encode color information by presenting an encoder architecture that, together with a pre-trained and fixed StyleGAN model, is able to faithfully reconstruct images from virtually any domain. Thus, we demonstrate a previously unknown grade of generalizablility by training the encoder and decoder independently and on different datasets. Our proposed architecture processes up to 45 images per second on a single GPU, which is approximately 32× faster than previous approaches. Finally, as one example application, our approach also shows promising results compared to the state of the art on image denoising tasks.

1 Introduction

Generative Adversarial Networks (GANs) are applied in various computer vision areas, *e.g.*, image-to-image translation [15, 16, 31, 44], image superresolution [21, 30, 39], or unconditional generation of various image types [5, 10, 18, 23]. Over time, image quality, resolution, and realism of synthesized images were improved by a large margin [10, 18, 19, 23]. The StyleGAN architecture [19, 20] is one of the current state-of-the-art models for unconditional image generation. The architecture of StyleGAN with its projection into a semantically meaningful latent space $\mathcal{W}$ and the usage of noise inputs for stochastic variation not only allows to generate a diverse range of images but also enables meaningful image editing operations. Using recent GAN inversion techniques, it is also possible to perform similarly

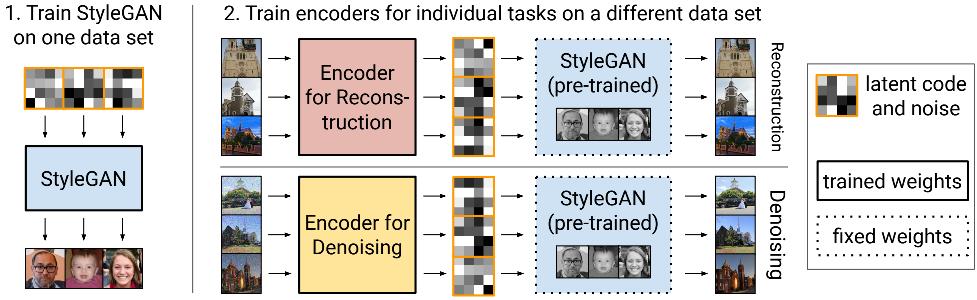


Figure 1: Our approach: a StyleGAN generator is trained on a dataset, *e.g.*, FFHQ [19]. Then, we train an encoder for reconstruction and denoising tasks on a *different* dataset, without updating the pre-trained StyleGAN which is used as a decoder.

meaningful edit operations on embeddings of real images in the latent space [1, 2, 26] (for further details on related work see Section 2). However, to the best of our knowledge, previous work has not provided complete answers for the following open research questions in the domain of GAN inversion and reconstruction: (1) Can the stochastic noise inputs provide more than stochastic variations of the generated image? (2) Can we use a generator model (*i.e.* StyleGAN) trained in one domain to effectively reconstruct images for a different domain? (3) Can we use an encoder model trained in one domain to reconstruct images for a different domain effectively?

In this work, we strive to provide answers to these questions. To this end, we train an encoder decoder architecture in multiple steps. First, we train a StyleGAN model for unconditional image generation on one dataset. Second, we freeze the trained model and train an encoder model with the task to find such inputs to StyleGAN that the inputs gets reconstructed. See Figure 1 and Section 3 for further details. In Section 4, we show that StyleGAN models can use the stochastic noise inputs to faithfully reconstruct input images from the domain the generator was trained on. We further show that encoders trained on a fixed StyleGAN model can effectively reconstruct images from domains unseen during training. For example, we show that a StyleGAN generator pre-trained on the FFHQ dataset [19] and an encoder trained to reconstruct images from the FFHQ dataset can faithfully reconstruct images from other datasets, such as the LSUN Church dataset [35]. We provide in-depth insights into why our encoder can use the pre-trained StyleGAN generator to reconstruct images from arbitrary domains. We conclude our work in Section 5. Overall, our contributions can be summarized as follows: (1) The first approach for faithful cross-domain image reconstruction, based on a fixed generator model, tested on a large variety of images from several domains. (2) Novel insights about the capabilities of the stochastic noise inputs in StyleGAN. (3) A fast method that allows reconstruction of up to 45 images per second on a single GPU (NVIDIA Tesla V100), which is much faster than other recent GAN inversion models, *e.g.*, [12], which can process approximately 1.4 images per second. (4) A practical application of our model in the area of image denoising, where we achieve competitive results. Our code and trained models are available online[1].

[1] https://github.com/Bartzi/one-model-to-reconstruct-them-all

2 Related Work

GANs have first been proposed by Goodfellow *et al.* [10] in 2014. Since then, they have been improved through different measures, such as training at different scales [18], adding novel weight normalization techniques [23] or generating high-resolution images over a diverse set of classes [5].

A recent work by Karras *et al.* [19] proposes a novel architecture inspired by recent work on neural style transfer [14]. Karras *et al.* train their StyleGAN architecture on the FFHQ dataset to generate high-quality, realistic images of human faces. Moreover, Karras *et al.* propose several improvements regarding architecture and normalization methods for StyleGAN in a more recent work [20]. In the following, we describe the prior art related to the two tasks covered in our work, image reconstruction and image denoising.

2.1 Image Reconstruction (Embedding, Inversion)

Generative models usually operate on a latent code and/or random noise as an input to generate new images [10, 19, 20]. Previous work on GAN inversion attempts to understand and interpret the underlying mechanisms of GANs by embedding existing images into a GAN architecture. These works can be roughly divided into two categories.

On the one hand, a given image can be embedded into the latent space of a trained GAN on a per-image basis [1, 2, 7, 43]. These methods achieve very faithful reconstructions but require optimization or training of a model for each image, making image embeddings of large-scale datasets infeasible.

On the other hand, there are works where an encoder network for embedding an image is learned. Such an approach is used, *e.g.*, in [4, 12, 25, 32]. It is computationally efficient since the learned encoder can retrieve an encoding for a given image. However, learning a code with semantic meaning proves to be a challenge, as stated by Zhu *et al.* [42].

2.2 Image Denoising

A typical application area for image reconstruction is image denoising, where the task is to remove noise to restore the original image. Here, we focus on image denoising techniques based on deep neural networks; for more detailed information about image denoising research, please refer to the following survey papers [9, 11].

Several neural-network-based image denoising systems have been proposed in the past [6, 17, 21, 34, 36, 37, 38, 39]. Some works have been trained for image denoising at fixed noise level [6], while others are able to denoise noisy images with various noise levels. Over time, the most common approach shifted from directly predicting the denoised image to predicting residual/noise images which are then subtracted from the input image, returning the denoised image [21, 36]. Based on this residual prediction strategy, further enhancements have been proposed. Zhang *et al.* propose multiple extensions, *i.e.*, using multiple denoising networks and model based optimization [37], providing noise level maps as auxiliary input to the network [38, 39], or creating specific network architectures for image denoising [39].

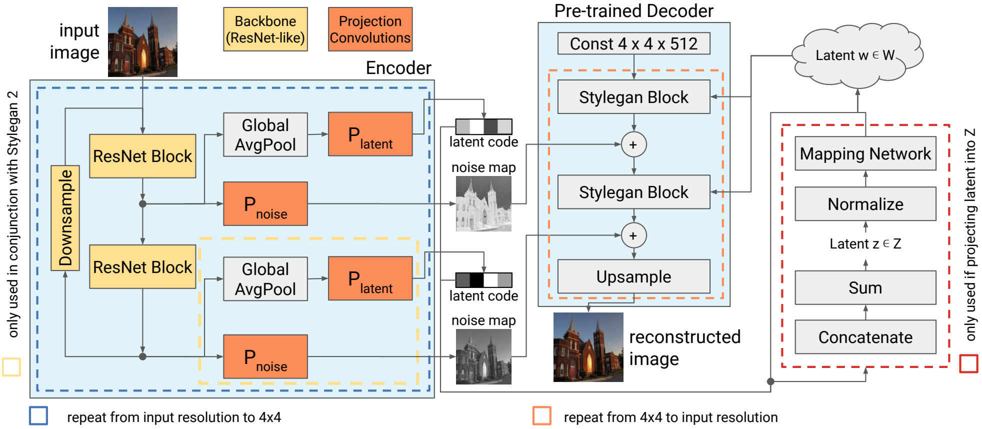


Figure 2: The structure of our proposed model. The encoder consists of multiple ResNet blocks, each followed by convolutional layers that predict a part of the latent code or a noise input. These outputs are used by the decoder (a pre-trained StyleGAN) to reconstruct the input image. Certain parts are only used for StyleGAN 2 or when using $\mathcal{Z}$, respectively.

3 Method

In this section, we describe our method that reconstructs arbitrary input images (not limited to the training domain) using a generative model that has only been trained for unconditional generation in one domain, *e.g.*, face images (see Figure 1). We introduce the architecture of our proposed encoder for arbitrary image reconstruction (see Figure 2). Finally, we describe the training details used in our experiments.

3.1 StyleGAN

For our experiments, we use generators based on StyleGAN [19] and the improved version of StyleGAN [20] as our decoder network. In the following we refer to the models based on the first version of StyleGAN [19] as "StyleGAN 1" and models based on the improved version of StyleGAN [20] as "StyleGAN 2".

StyleGAN currently sets the state-of-the-art in unconditional high-resolution image generation for many different natural image categories such as faces or buildings.

In our work, we focus on the latent code and stochastic noise inputs. The latent code defines the content of the image to generate and can also be used for semantic editing operations [1, 32]. We embed into two latent spaces. First, we embed into $\mathcal{Z}$, which refers to the input space of StyleGAN. Vectors from $\mathcal{Z}$ are transformed by a mapping network to the latent space $\mathcal{W}$, which is the second space we embed to.

We also focus on the stochastic noise inputs, which are originally tasked with generating stochastic details. It has been found that optimizing the noise inputs can lead to better image embeddings/inversions [2, 20]. We introduce a novel view of the role of these noise inputs and show that they can be used to encode the content and color of an input image.

We use the proposed decoder architectures, StyleGAN 1 and StyleGAN 2, as is and without any changes.

3.2 Encoder Architecture

Our architectural contribution is the encoder architecture (see Figure 2). Our encoder is a fully convolutional network that predicts latent vectors either in $\mathcal{Z}$ or in $\mathcal{W}$ and noise maps for each resolution of the generated images. Our network is a combination of a ResNet [13] and a U-Net architecture [27]. We predict latent vectors and noise inputs (see the supplementary material for further details). We use the predicted latent vectors and noise inputs as input to a pre-trained StyleGAN 1 or StyleGAN 2 based generator, which is *not* updated during the training process of the encoder.

We found that directly using our proposed encoder encourages the encoder to embed all information into the stochastic noise inputs (see Figure 5(a)). To mitigate this, we propose a two-stage training scheme that maximizes the latent code's semantic meaning. Here, we train the network in two stages. First, we disable learning (or usage) of the stochastic noise inputs, forcing the model to only rely on the latent code for reconstruction. Second, we train only the layers responsible for predicting the stochastic noise inputs to improve the reconstruction quality.

3.3 Training Details and Loss Function

We use two different loss functions for the training of our models. These loss functions are only used to update the weights of the *encoder*, the weights of the decoder (a pre-trained StyleGAN) are *fixed*. On the one hand, we use Mean Squared Error (MSE) between the pixels of the generated image and the reconstructed image. On the other hand, we utilize the Learned Perceptual Image Patch Similarity (LPIPS) [40] metric for judging the reconstruction quality. The resulting loss function is the following: $\mathcal{L}(x,y) = \mathcal{L}_{\text{mse}}(x,y) + \mathcal{L}_{\text{lpips}}(x,y)$. With $x,y \in \mathbb{R}^{[3,H,W]}$ being the input image and desired output image with three channels, height H and width W, respectively. $\mathcal{L}_{\text{mse}}$ and $\mathcal{L}_{\text{lpips}}$ denote MSE and LPIPS loss, respectively.

4 Results and Discussion

This section shows the experimental results of our approach on different datasets and two different tasks, image reconstruction and image denoising. First, we show that we can faithfully reconstruct images with our presented encoder architecture using a pre-trained and fixed StyleGAN decoder. Second, we show the results for cross-domain reconstruction, where the encoder is trained on a different dataset than the fixed StyleGAN decoder (trained on FFHQ). Third, we investigate how such high-quality reconstructions are possible by examining the role of the noise inputs in StyleGAN. Afterward, we present the results for our two-stage training method to increase the semantic meaning of the latent code. Finally, we show the capabilities of our model when applied for the task of image denoising.

4.1 Experimental Setup

We implement our model using PyTorch [24]. We use the human faces dataset FFHQ [19] and the two LSUN datasets Church and Bedroom [35], which only contain images of churches or bedrooms, respectively. We follow related work and evaluate our model using the following metrics on the given validation datasets: (1) The Frechet Inception Distance (FID) [29] of

(a) Reconstruction Results using our Approach

Model	Dataset and Metric for Evaluation			
	FFHQ		Church	
	FID$^\downarrow$	MSE$^\downarrow$	FID$^\downarrow$	MSE$^\downarrow$
FFHQ, 1, $\mathcal{Z}$	9.85	0.004	7.17	0.01
FFHQ, 1, $\mathcal{W}$	**0.64**	0.004	1.37	0.009
FFHQ, 2, $\mathcal{Z}$	3.92	0.005	4.66	0.01
FFHQ, 2, $\mathcal{W}$	0.75	*0.002*	1.23	0.006
Church, 1, $\mathcal{Z}$	17.28	0.01	3.17	0.007
Church, 1, $\mathcal{W}$	3.30	0.01	0.26	0.005
Church, 2, $\mathcal{Z}$	12.24	0.01	3.17	0.007
Church, 2, $\mathcal{W}$	2.33	0.008	**0.21**	*0.004*

(b) Reconstruction Results Reported in Related Work

Method	Dataset and Metric for Evaluation			
	FFHQ		Church	
	FID$^\downarrow$	MSE$^\downarrow$	FID$^\downarrow$	MSE$^\downarrow$
Zhu *et al.* [42]	42.64	0.03	44.77	0.052
Pidhorsky *et al.* [26]	16.52	-	-	-
Abdal *et al.* [1]*	65.80	0.01	66.29	0.02
Abdal *et al.* [2]*	13.92	**0.0003**	10.92	**0.0004**
Tov *et al.* [32]	25.17	0.03	26.96	0.09

* We calculated the FID and MSE for these approaches based on the reconstruction of 500 random images due to limited compute time (10 minutes needed per image).

Table 1: Our experimental results on image reconstruction (a) compared to other approaches (b). We denote the dataset our models are trained on (FFHQ, or LSUN Church), the StyleGAN variant, and the projection target ($\mathcal{Z}, \mathcal{W}$). The best overall results are indicated in **bold** and our own best result in *italic*.

reconstructed images with the original images (using a sample size of 50 000 images). Furthermore, we calculate (2) the MSE between each input and its corresponding reconstructed image to measure the reconstruction quality. Further details on our system, number of iterations, optimizer, learning rate, and data pre-processing can be found in the supplementary material.

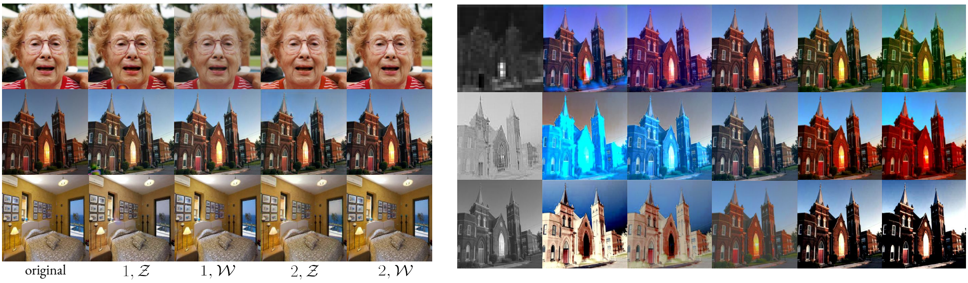
4.2 FFHQ-based Image Reconstruction

In our first set of experiments, we determined how well our architecture (introduced in Section 3) is able to reconstruct images of the FFHQ [19] and LSUN Church dataset [35], when using a StyleGAN model pre-trained on the FFHQ dataset and compare it to results found in related work. In this line, we trained a range of different encoders, using both StyleGAN 1 and StyleGAN 2 decoders. Furthermore, we examined the influence of different latent code projection strategies. On the one hand, we project into $\mathcal{Z}$. On the other hand, we project into $\mathcal{W}$.

The quantitative results (see the first block of Table 1(a)) show, that our encoder is able to perform reconstruction for different datasets with high quality. Even though both encoder and decoder were *only trained on FFHQ*, they show a high reconstruction quality when evaluated on the LSUN Church dataset. Compared to the results reported in related work (see Table 1(b)), our method outperforms almost all other methods. The exception is the MSE achieved by Abdal *et al.* [2], but our method is magnitudes faster than their approach (see Section 4.4).

The qualitative results also show nearly no perceptual differences (see the first row of Figure 3(a)). However, we can see that the models based on StyleGAN 1 exhibit the "bubble" artifacts typical for images produced by StyleGAN 1 [20]. The absence of these artifacts in the reconstructed images of the StyleGAN 2 based models is most likely the reason for the better quantitative results.

Cross-Domain Image Reconstruction Intrigued by our results on the FFHQ dataset, we trained a different set of encoder models that use the same pre-trained and fixed StyleGAN



(a) Reconstruction results based on a StyleGAN pre-trained on the FFHQ dataset. Each row shows reconstructions with images from the FFHQ dataset, LSUN Church dataset, and LSUN Bedroom dataset, respectively. The columns show the reconstruction result produced by: StyleGAN variant, latent projecting strategy.

(b) Shifting the noise maps shows that they can not only encode the content of an image, but also color and contrast. The columns represent "shifting" the noise map shown in the first column by multiplying it with -2, -0.75, 0.5, 1.75, and 3.

Figure 3: Figure (a) shows our reconstruction results, Figure (b) our color shift experiment.

generator but use different LSUN datasets for training the encoder part of our model. The quantitative results of experiments on the LSUN Church dataset are shown in the bottom block of Table 1(a). Further, we show the qualitative results of the encoders trained on churches and bedrooms in rows 2 and 3 of Figure 3(a), respectively. These results show that our cross-domain models can reconstruct images with high perceptual quality and scores.

The Significance of Noise for Image Reconstruction We examined the latent code and the stochastic noise inputs predicted by our model to understand how our reconstruction model can produce such high-quality results. First, we directly visualize the (normalized) stochastic noise inputs predicted by our encoder (see Figure 4). It is visible that the encoder learns to use the stochastic noise inputs to retain the input image's content, especially in the stochastic noise inputs of higher resolution. Although Karras *et al.* [20] made a brief note that it is required to regularize the noise inputs to prevent the retaining of image content, the possible effects were not discussed in detail. To the best of our knowledge, other works also have not examined this phenomenon.

Therefore, we further analyzed the noise inputs by shifting the value of each pixel in a noise input with a factor from the interval $[-2, 3]$ and examined the reconstructed image. The results (see Figure 3(b)) show that the encoder uses the noise inputs not only to capture the content of the image but can also (at least to some degree) encode the colors of each pixel in these noise inputs. We found a similar behavior when shifting the value of the noise inputs on unconditionally generated images and provide details about this in our supplementary material.

4.3 Semantic Image Reconstruction

We examined the semantic meaningfulness of the latent code with sample interpolations between two images (see Figure 5(a)) based on a StyleGAN 2 decoder trained on FFHQ. This visualization shows that a model trained for reconstruction using our method is not able to perform semantic interpolation. It is visually more similar to an alpha blending between two images. Thus, it seems the influence of the latent code is degraded in such a

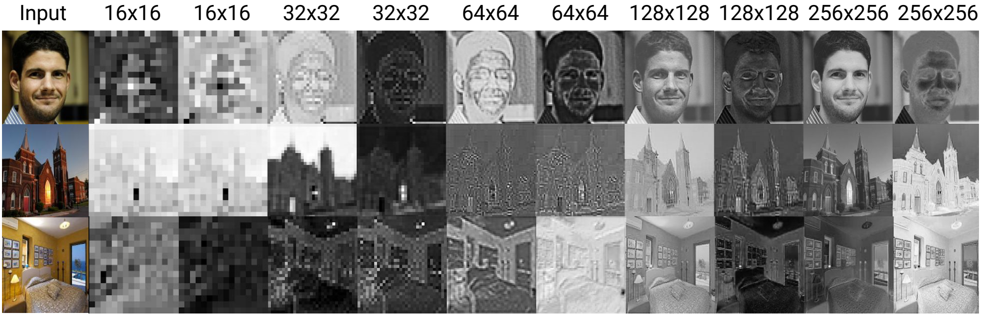
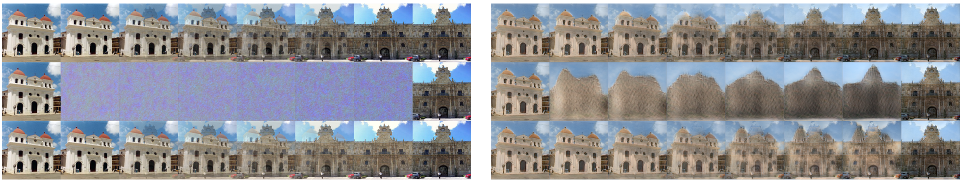


Figure 4: The stochastic noise inputs predicted by our model show that all content information is saved in the noise inputs (normalized individually, input image on the left). The progressive level of detail increases with each layer, corresponding to the architecture of StyleGAN.



(a) Regular training approach.

(b) Two-stage training with two networks.

Figure 5: The behaviour of our models when interpolating latent code and noise inputs between two reconstruction images (embedded using $\mathcal{W}$ of a StyleGAN 2 pre-trained on the FFHQ dataset). The rows show the interpolation of: (1) predicted latent code and noise at the same time, (2) only latent code (with fixed random noise), and (3) only noise inputs (with fixed latent code of the left image).

way that it is only used to provide some basic colors for the resulting image and the resulting reconstruction mostly depends on the predicted noise maps.

We also tested our improved two-stage training strategy (see Section 3.2) to find a more meaningful (semantic) latent code and enable meaningful semantic interpolations. The results (see Figure 5(b)) show that the semantic latent code captures the coarse structure of the content, but the predicted noise inputs still add fine details. We also observe that the interpolations seem to be more reasonable, but the visual quality of images reconstructed by using only the latent code is visibly lower than the original images. Nevertheless, we can see that the intermediate latent space $\mathcal{W}$ seems to be a general latent space that could be used to generate any kind of image. However, the StyleGAN 2 generator trained on the FFHQ dataset is not general enough to faithfully generate images from another domain using only the latent code. When incorporating the predicted stochastic noise inputs on top of the semantic latent codes, the quality increases but can not achieve the same level as our other experiments (metrics and further interpolations for can be found in the supplementary material).

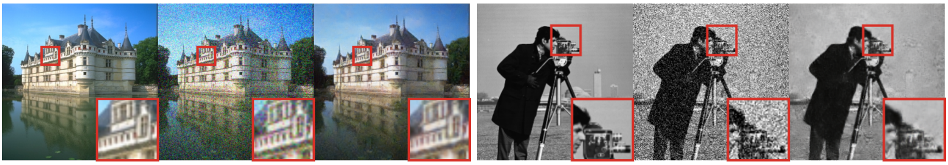


Figure 6: Qualitative results of image denoising tasks (BSD68 on the left, Set12 on the right). The noisy image is created with additive gaussian white noise ($\sigma = 50$). We use an encoder embedding into $\mathcal{W}$ of a StyleGAN 2 pre-trained on FFHQ.

Dataset	Set12			BSD68		
σ	15	25	50	15	25	50
Liu *et al.* [21]	33.15/**0.90**	30.79/**0.87**	27.74/**0.81**	31.86/0.90	29.41/0.84	26.53/0.74
Zhang *et al.* [38]	32.75/ -	30.43/ -	27.32/ -	31.63/ -	29.19/ -	26.29/ -
Zhang *et al.* [39]	**33.25/ -**	**30.94/ -**	**27.90/ -**	**31.91/ -**	**29.48/ -**	**26.59/ -**
StyleGAN 1, $\mathcal{Z}$	24.67/0.82	24.13/0.76	22.27/0.60	24.59/0.86	24.48/0.83	23.71/0.73
StyleGAN 1, $\mathcal{W}$	26.42/0.87	25.93/0.81	24.18/0.64	25.17/0.91	25.02/0.87	24.78/0.77
StyleGAN 2, $\mathcal{Z}$	26.02/0.84	25.58/0.80	23.88/0.67	24.70/0.85	24.76/0.83	24.30/0.75
StyleGAN 2, $\mathcal{W}$	27.46/0.88	27.08/0.85	24.94/0.71	27.57/**0.92**	27.46/**0.89**	26.22/**0.81**

Table 2: Average PSNR/SSIM results of our model compared to state-of-the-art image denoising models. **Bold font** indicates the best performing result.

4.4 Reconstruction Speed

We reproduced the approach of Abdal *et al.* [1, 2] which individually optimizes the latent code and noise maps iteratively. We measured its speed on a Tesla V100 GPU and found it needs approximately 10 minutes for a *single* image. The approach of Guan *et al.* [12] is much faster but still needs about 0.71 seconds per image on a Tesla V100 GPU, meaning they can process about 1.4 images per second. For comparison, our model can process approximately 45/30 (StyleGAN 1/StyleGAN 2) images per second on a Tesla V100 GPU, while still producing reconstructions with very high perceptual quality. Such throughput is possible since we only need to compute a forward pass of an image (or a batch of images) through our encoder network, which directly predicts the latent code and noise maps needed to reconstruct the original image(s).

4.5 Image Denoising

As an example task, we tested the capabilities of our approach on the application of image denoising.

For image denoising, we trained multiple models and compare them to the state of the art in image denoising on the BSD68 [28] and the SET 12 [22] benchmark datasets. We trained models based on StyleGAN 1 and StyleGAN 2 (both pre-trained on the FFHQ dataset), with latent code embedding into $\mathcal{Z}$ and $\mathcal{W}$, and use the ImageNet dataset [8] for training the encoder. We report the average PSNR and SSIM [33] of our model on different noise levels $\sigma = 15, 25$, and 50 on the benchmark datasets in Table 2. The qualitative results can be seen in Figure 6. The results show that our model is able to set new state-of-the-art results on the BSD68 dataset in terms of SSIM, even though our network has not been designed with the application of image denoising in mind. The results also show that our model works best on color images, as the denoising results on Set12, a grayscale dataset, show. We think our model performs better on colored images because our model uses the values of the stochastic noise inputs (see Section 4.2) to encode the color information of images. Some ideas mentioned in related work, *e.g.*, network design decisions, similar to [21] or [39], could be used to boost the performance of our model, but such improvements are out of the scope of this work.

The results are nonetheless quite interesting, considering that the generator has never been trained for image reconstruction and also never for the creation of images apart from faces of the FFHQ dataset. We also observe that our model removes noise from an image and performs a slight shift in colors similar to a color correction operation. This behavior is an interesting property and opens up further application possibilities of our proposed network in future work.

5 Conclusion

In this work, we examined the capabilities of the stochastic noise inputs of StyleGAN models. We showed that it is possible to design simple encoders for high-quality cross-domain image reconstruction. Our model is highly efficient and can be used to reconstruct images from virtually any dataset, even if trained on only one specific dataset, such as the FFHQ [19] dataset. In this paper, we mainly focused on an in-depth analysis of the reasons why StyleGAN models are able to reconstruct images from virtually any data distribution. We found that the stochastic noise inputs, which are only meant to produce stochastic variations, can capture tiny details and manipulate the colors of images generated by StyleGAN without using the latent code. We further found that the intermediate latent space ($\mathcal{W}$), where the mapping network of StyleGAN projects to, might be used to generate images from data distributions other than the original training distribution of a pre-trained StyleGAN model. We also provided one practical example of our architecture, where we showed that our architecture can be used for practical applications, such as image denoising.

We think that there are three more suitable applications for our approach based the stochastic noise inputs of a StyleGAN model: (1) We assume that super-resolution is a viable task for the application of our proposed approach, thanks to the capabilities of StyleGAN to produce high-quality images combined with the stochastic noise inputs providing details. (2) Based on the results of our image denoising experiments it should be possible to use our findings for models working on video restoration. This task should be a good usecase for our model because we can train a generator model directly on video images and optimize an encoder on other synthetically degraded images and then apply the resulting encoder-decoder model on the original degraded images. (3) Our encoder architecture should also be able to perform quite well on the task of image segmentation. In this case, we assume the model can be combined with semantic segmentation approaches incorporating StyleGAN [3, 41] with our proposed encoder architecture to directly perform pixel-wise semantic segmentation of images using a generative model.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International*

Conference on Computer Vision (ICCV), October 2019.

[2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan++: How to edit the embedded images? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[3] Christian Bartz, Hendrik Rätz, Haojin Yang, Joseph Bethge, and Christoph Meinel. Synthesis in Style: Semantic Segmentation of Historical Documents using Synthetic Data. *arXiv:2107.06777 [cs]*, July 2021.

[4] David Bau, Jun-Yan Zhu, Jonas Wulff, William Peebles, Hendrik Strobelt, Bolei Zhou, and Antonio Torralba. Seeing What a Gan Cannot Generate. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019.

[5] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large Scale GAN Training for High Fidelity Natural Image Synthesis. In *Proceedings of the 7th International Conference on Learning Representations, (ICLR)*, 2019.

[6] Yunjin Chen and Thomas Pock. Trainable Nonlinear Reaction Diffusion: A Flexible Framework for Fast and Effective Image Restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1256–1272, June 2017. ISSN 1939-3539. doi: 10.1109/TPAMI.2016.2596743.

[7] 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. doi: 10.1109/TNNLS.2018.2875194.

[8] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009. doi: 10.1109/CVPR.2009.5206848.

[9] Linwei Fan, Fan Zhang, Hui Fan, and Caiming Zhang. Brief review of image denoising techniques. *Visual Computing for Industry, Biomedicine, and Art*, 2(1):7, July 2019. ISSN 2524-4442. doi: 10.1186/s42492-019-0016-7.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems 27*, 2014.

[11] Bhawna Goyal, Ayush Dogra, Sunil Agrawal, B. S. Sohi, and Apoorav Sharma. Image denoising review: From classical to state-of-the-art approaches. *Information Fusion*, 55:220–244, March 2020. ISSN 1566-2535. doi: 10.1016/j.inffus.2019.09.003.

[12] Shanyan Guan, Ying Tai, Bingbing Ni, Feida Zhu, Feiyue Huang, and Xiaokang Yang. Collaborative Learning for Faster StyleGAN Embedding. *arXiv preprint arXiv:2007.01758*, 2020.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[14] Xun Huang and Serge Belongie. Arbitrary Style Transfer in Real-Time With Adaptive Instance Normalization. In *Proceedings of the IEEE International Conference on Computer Vision (CVPR)*, 2017.

[15] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal Unsupervised Image-to-image Translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[16] 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 (CVPR)*, 2017.

[17] Viren Jain and Sebastian Seung. Natural Image Denoising with Convolutional Networks. In *Advances in Neural Information Processing Systems 21*, 2009.

[18] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive Growing of GANs for Improved Quality, Stability, and Variation. In *International Conference on Learning Representations*, 2018.

[19] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[20] 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)*, June 2020.

[21] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo. Multi-Level Wavelet-CNN for Image Restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018.

[22] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423 vol.2, July 2001. doi: 10.1109/ICCV.2001.937655.

[23] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral Normalization for Generative Adversarial Networks. In *6th International Conference on Learning Representations, (ICLR)*, 2018.

[24] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, 2019.

[25] Guim Perarnau, Joost Van De Weijer, Bogdan Raducanu, and Jose M Álvarez. Invertible Conditional Gans for Image Editing. *arXiv preprint arXiv:1611.06355*, 2016.

[26] Stanislav Pidhorskyi, Donald A. Adjeroh, and Gianfranco Doretto. Adversarial latent autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, 2015.

[28] Stefan Roth and Michael J. Black. Fields of Experts. *International Journal of Computer Vision*, 82(2):205–229, April 2009. ISSN 0920-5691. doi: 10.1007/s11263-008-0197-6.

[29] Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved Techniques for Training GANs. In *Advances in Neural Information Processing Systems 29*, 2016.

[30] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. SinGAN: Learning a Generative Model From a Single Natural Image. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019.

[31] Zhiqiang Shen, Mingyang Huang, Jianping Shi, Xiangyang Xue, and Thomas S. Huang. Towards Instance-Level Image-To-Image Translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[32] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an Encoder for StyleGAN Image Manipulation. *arXiv:2102.02766 [cs]*, 2021.

[33] 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, April 2004. ISSN 1941-0042. doi: 10.1109/TIP.2003.819861.

[34] Junyuan Xie, Linli Xu, and Enhong Chen. Image Denoising and Inpainting with Deep Neural Networks. In *Advances in Neural Information Processing Systems 25*, 2012.

[35] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. *arXiv preprint arXiv:1506.03365*, 2015.

[36] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, July 2017. ISSN 1941-0042. doi: 10.1109/TIP.2017.2662206.

[37] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning Deep CNN Denoiser Prior for Image Restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[38] Kai Zhang, Wangmeng Zuo, and Lei Zhang. FFDNet: Toward a Fast and Flexible Solution for CNN-Based Image Denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, September 2018. ISSN 1941-0042. doi: 10.1109/TIP.2018.2839891.

[39] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. Plug-and-Play Image Restoration with Deep Denoiser Prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021. ISSN 1939-3539. doi: 10.1109/TPAMI.2021.3088914.

[40] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[41] Yuxuan Zhang, Huan Ling, Jun Gao, Kangxue Yin, Jean-Francois Lafleche, Adela Barriuso, Antonio Torralba, and Sanja Fidler. DatasetGAN: Efficient Labeled Data Factory With Minimal Human Effort. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10145–10155, 2021.

[42] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain Gan Inversion for Real Image Editing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[43] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A Efros. Generative Visual Manipulation on the Natural Image Manifold. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[44] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.