🙃GAN inversion through latent codes of a pretrained encoder

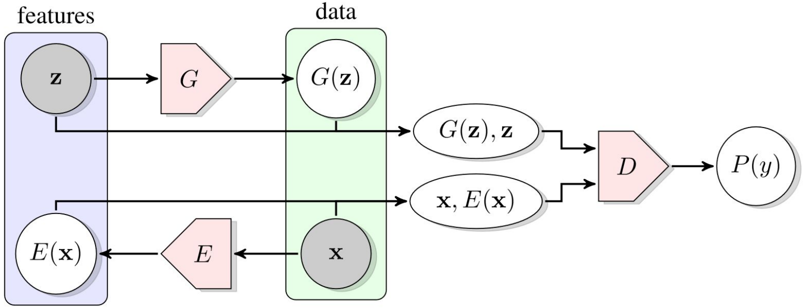
Nikita Khoroshavtsev, Nikita Kurdiukov, Sergey Karpukhin

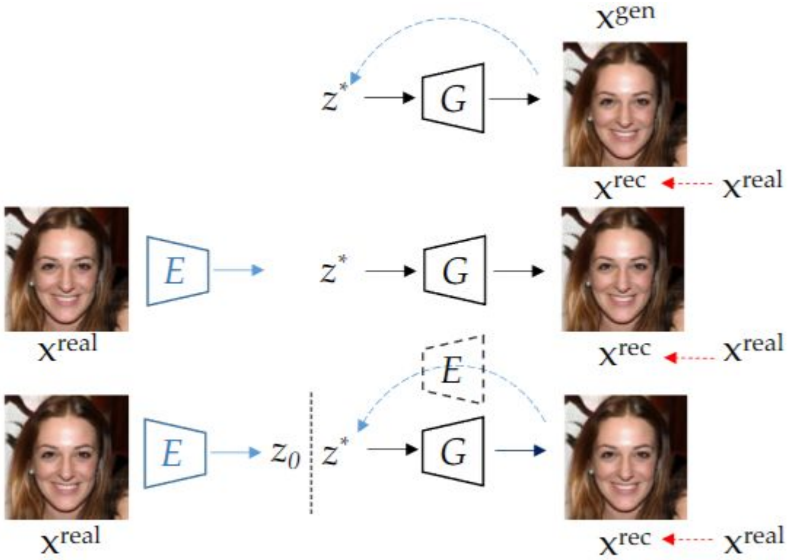
Problem statement

GAN inversion - want to obtain latent code corresponding to the image *x*

Main approaches:

- optimization based
- encoder based
- hybrid
- (*) joint training [2]

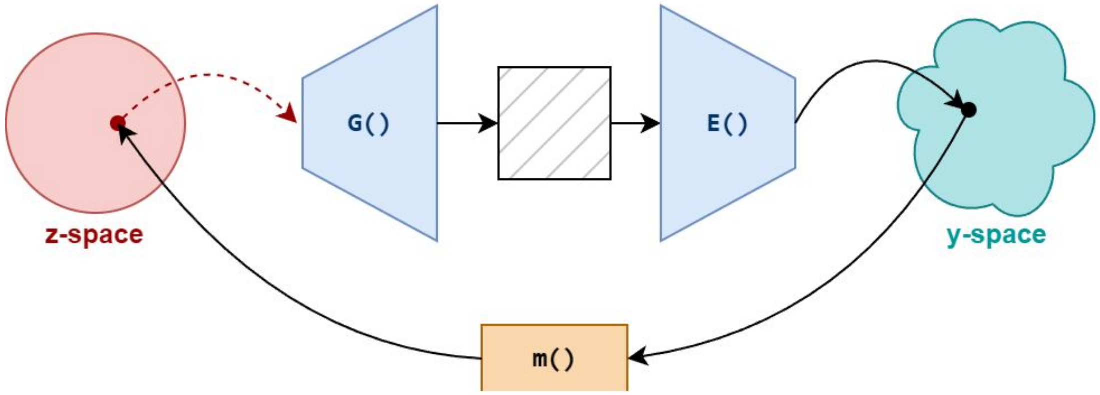




Our approach

Instead of learning encoder from zero let's utilize pretrained encoder feature-space and learn additional map between it and GAN's latent space.

Pros: less compute, no need for real data (only for validation), simple



$$\hat{z} = m_\theta(E(G(z))))$$

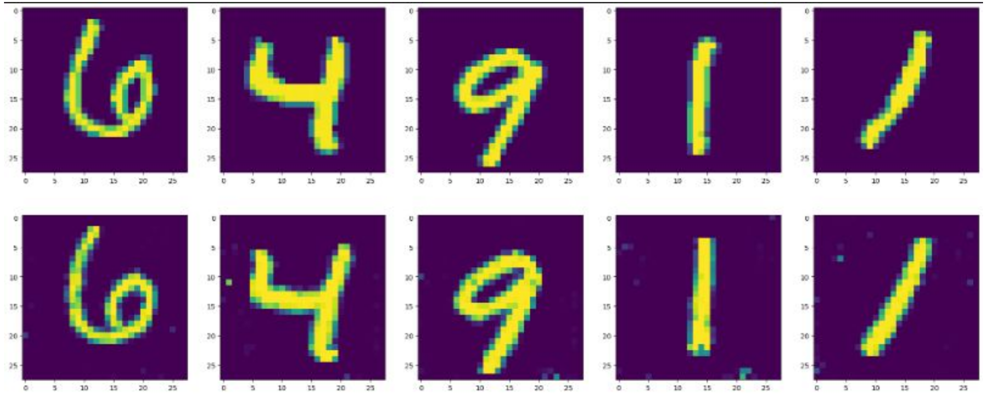
Experiments with MNIST: Setup 1

Generator: DCGAN pretrained on MNIST

Encoder: SimCLR backbone, 250 epochs on MNIST

M(): 5 layer MLP with GELU activations

In all setups **M()** was trained for 5000 iterations, batch size of 256 and Adam optimizer with lr between 1e-4 and 3e-3; visual validation is done on hold-out set

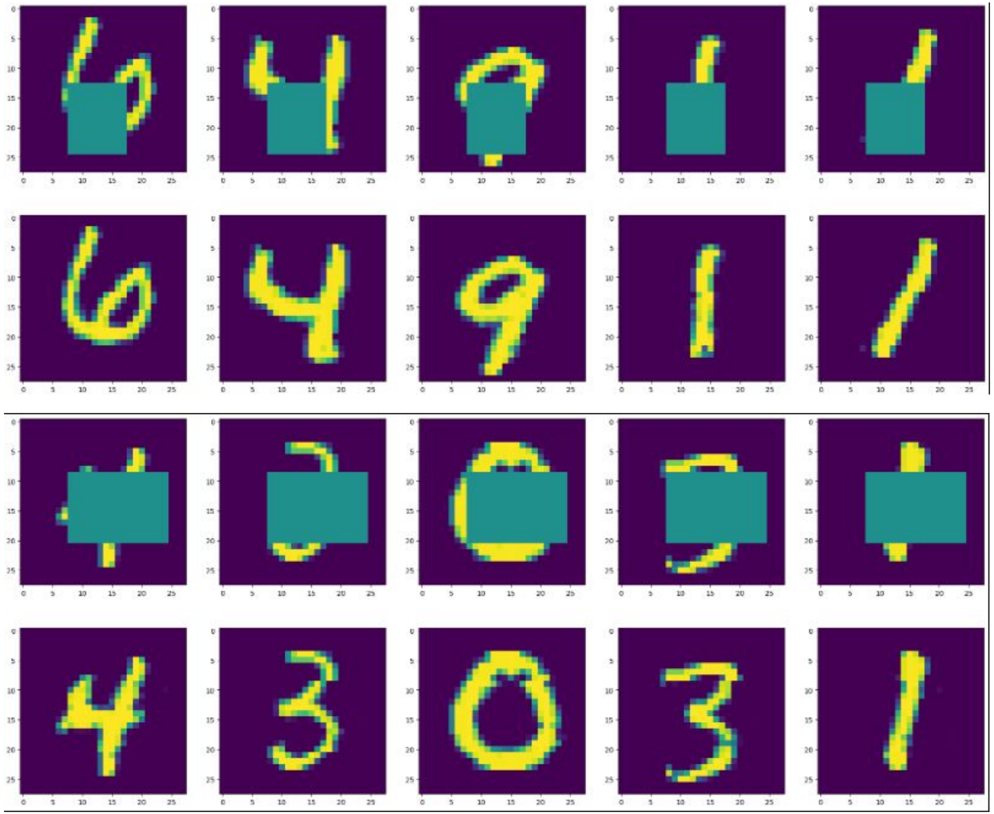


inversion result from one of the setups.
training took 5 minutes!

Experiments with MNIST: Setup 1

We used additional tricks, such as: noising encoder latents, masking generated images, additional reconstruction loss, EMA.

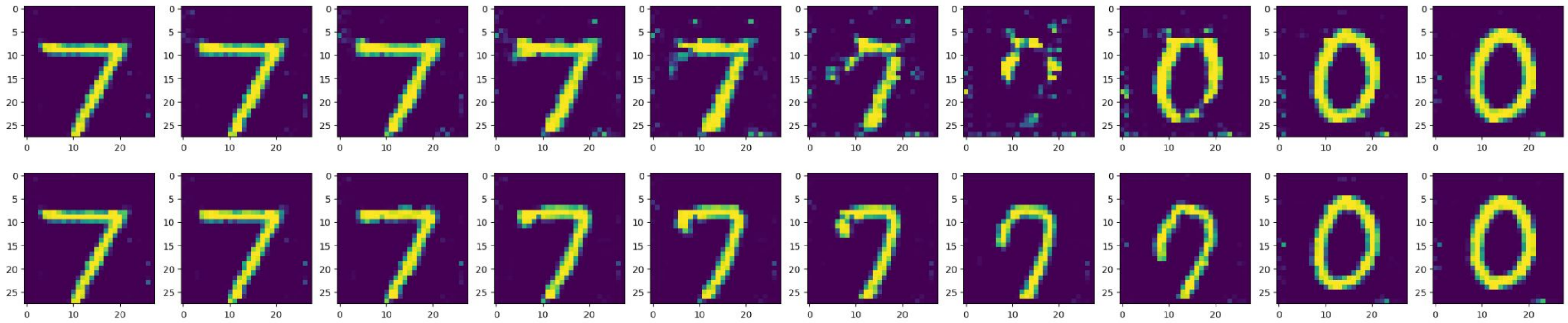
In setup with masking we are also able to obtain a robust map **M()**, which allows to perform inpainting during inference.



Experiments with MNIST: Setup 1

Additionally, we explored interpolation through map projection of two image representations.

We observed that for empty regions in z-space of DCGAN this interpolation allows to bypass them (this is not always true).



Experiments with MNIST: Setup 2

Generator: DCGAN pretrained on MNIST

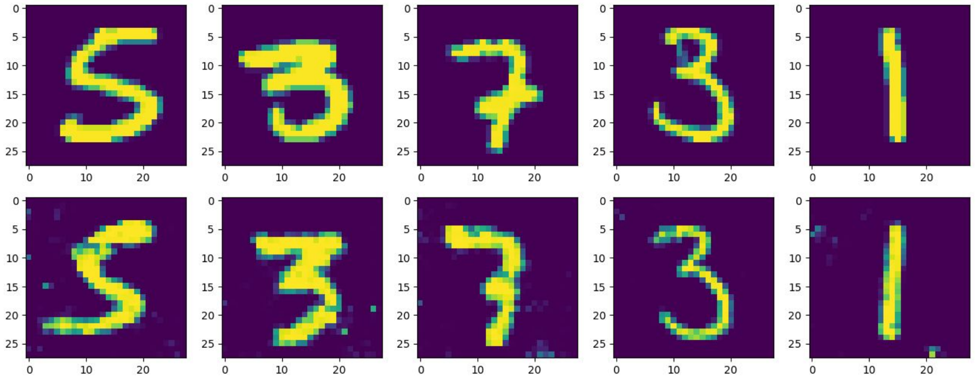
Encoder: **CLIP/ViT-16 Image Encoder**

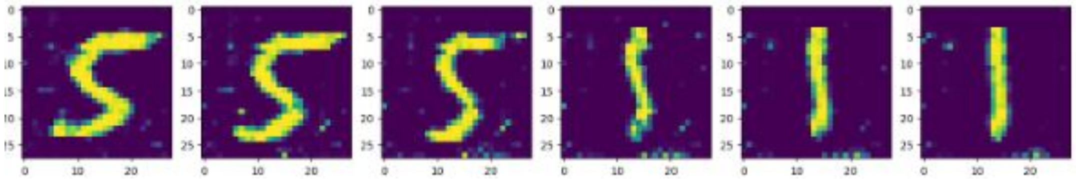
M(): 5 layer MLP with GELU activations

Trained for 2000 iterations, no masking, latent noising, high lr=3e-3, reconstruction l2-loss

CLIP turned out to be good enough encoder for our dataset!

Interpolation results also hold





Experiments with CIFAR10

Generator: DCGAN pretrained on CIFAR10.

Encoder: **CLIP/ViT-16 Image Encoder**

M(): 6 layer MLP with GELU activations

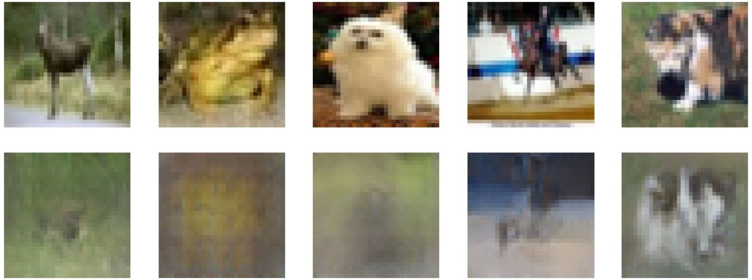
Trained for 3000 iterations, no masking, high lr=3e-3, latents reconstruction l2-loss

The reconstructions are generally poor with both generated and real images (**this is probably due to the poor GAN Generator**)

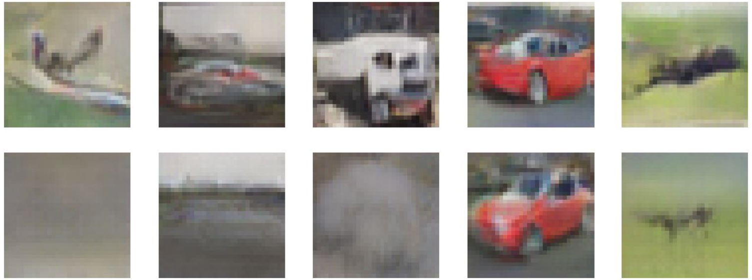
Additional experiments with losses and architectures did not yield reasonable improvements.

	FID ↓	Inception Score ↑	LPIPS ↓
Real Images	121.56	**0.014**	0.585
Generated Images	**106.68**	0.006	**0.455**

Real images reconstruction



Generated images reconstruction



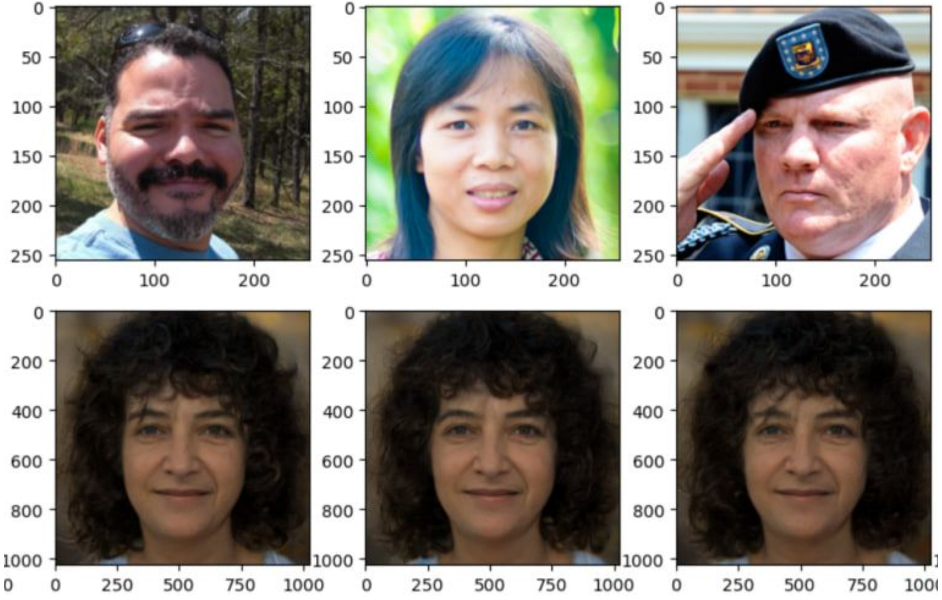
Experiments with FFHQ. Setup 1

Generator: StyleGAN2 pretrained on FFHQ 256x256

Encoder: CLIP with VIT-16 backbone

M(): 5 layer MLP with GELU activations

In all setups **M()** was trained for 5000 iterations, batch size of 4 and Adam optimizer with lr between 5e-5 and 5e-4; visual validation is done on hold-out set



Experiments with FFHQ. Setup 1

What went wrong?

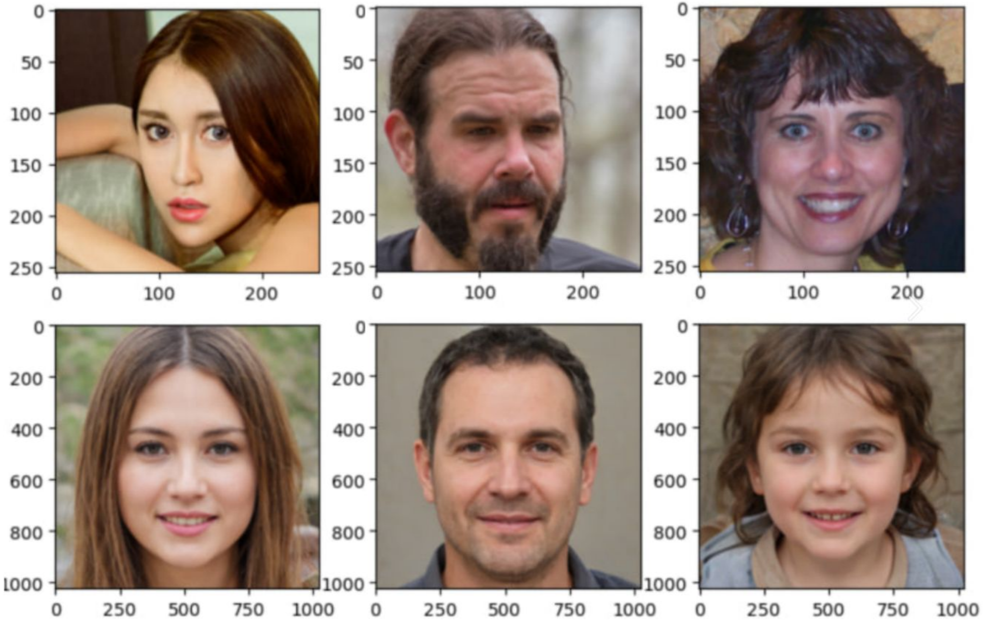
- Map doesn’t get different modes

Ideas to fix?

- Try more different losses on reconstruction and latent space

Result?

- Model understands more features about people



Experiments with FFHQ. Setup 2

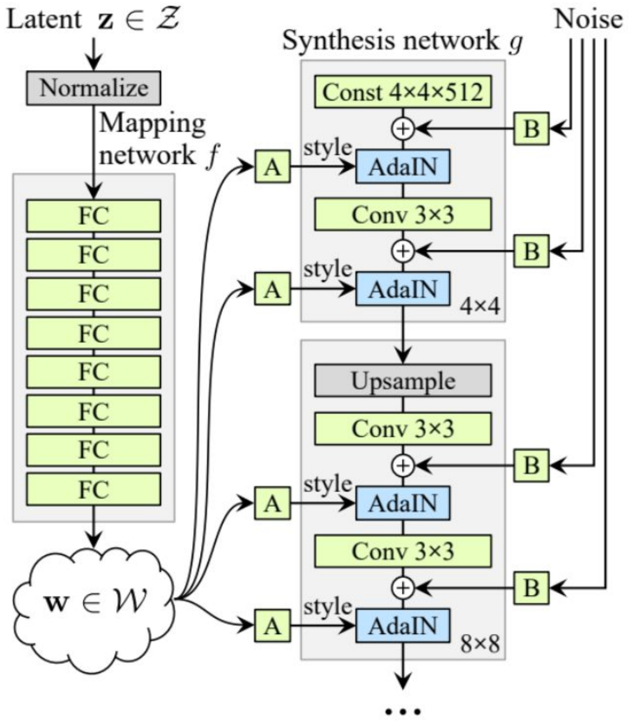
Generator: StyleGAN1 pretrained on FFHQ 256x256

Encoder: CLIP/VIT-16 Image Encoder

M(): to z-space/w-space

- 6 layer MLP with dropouts and skip-connection (**MLPv1**)
- 6 layer MLP with additional projection on inner representations from CLIP Encoder (**MLPv2**)

2500 - 5000 iters, 32 batch size, Adam optimizer with lr = 3e-4



Experiments with FFHQ. Setup 2

Learning a map to z-space turned out to be not feasible.

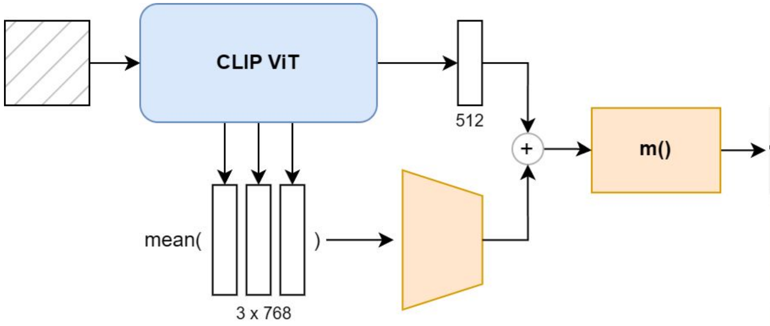
As for map to w-space, both architectures of **M()** are able to capture meaningful semantic features of faces, but a lot of details are lost.

inversion with MLPv1 into w-space

Experiments with FFHQ. Setup 2

Idea of MLPv2: let’s utilize low-level image information from intermediate encoder layers

MLPv2



Narek Tumanyan, Omer Bar-Tal, Shir Amir, Shai Bagon, Tali Dekel, “Disentangling Structure and Appearance in ViT Feature Space.” arXiv, Nob. 20, 2023. [10.48550/arXiv.2101.05278](https://doi.org/10.48550/arXiv.2101.05278).

Experiments with FFHQ. Setup 2

We also experimented with training a robust map (with masking augmentations) in order to try inpainting. Results are not that great - features make sense, but there is no spatial consistency w.r.t. face position and background.

Experiments with FFHQ. Setup 2

Quantitative comparison of results on FFHQ 256x256

	FID ↓	**Inception Score ↑**	**LPIPS ↓**
MLPv1	**30.503**	**4.12**	0.52
MLPv2	47.110	3.037	**0.488**
MLPv2 (with masking)	53.735	2.803	0.492

Discussion

- with simple data like MNIST approach demonstrates very strong results compared to other works ([2], [3])
- for high resolution image data and StyleGAN, current approach with w-space might be bottlenecked - need to go to w+-space + utilize advanced losses for details
- CLIP shared text/image embedding space can be utilized - but our latest experiments with them weren’t successful
- our results comply with the ones in [8]: using only final layer latents for image reconstruction tends to preserve high-level features of an input image in detriment to more local details

References

[1] W. Xia, Y. Zhang, Y. Yang, J.-H. Xue, B. Zhou, and M.-H. Yang, “GAN Inversion: A Survey.” arXiv, Mar. 22, 2022. doi: 10.48550/arXiv.2101.05278.

[2] J. Donahue, P. Krähenbühl, and T. Darrell, “Adversarial Feature Learning.” arXiv, Apr. 03, 2017. doi: 10.48550/arXiv.1605.09782.

[3] Y. Wang, W. Zhou, J. Bao, W. Wang, L. Li, and H. Li, “CLIP2GAN: Towards Bridging Text with the Latent Space of GANs.” arXiv, Nov. 27, 2022. doi: 10.48550/arXiv.2211.15045.

[4] Stanislav Pidhorskyi, Donald Adjeroh, Gianfranco Doretto, “Adversarial Latent Autoencoders”, arxiv, Apr. 9 2020. doi: 10.48550/arXiv.2004.04467

[5] J. Zhu, D. Zhao, B. Zhang, and B. Zhou, “Disentangled Inference for GANs with Latently Invertible Autoencoder.” arXiv, Mar. 01, 2022. Accessed: May 27, 2024. [Online]. Available: http://arxiv.org/abs/1906.08090

[6] J. Zhu, Y. Shen, D. Zhao, and B. Zhou, “In-Domain GAN Inversion for Real Image Editing.” arXiv, Jul. 16, 2020. doi: 10.48550/arXiv.2004.00049.

[7] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, and D. Cohen-Or, “Designing an Encoder for StyleGAN Image Manipulation.” arXiv, Feb. 04, 2021. doi: 10.48550/arXiv.2102.02766.

[8] G. Perarnau, J. van de Weijer, B. Raducanu, and J. M. Álvarez, “Invertible Conditional GANs for image editing.” arXiv, Nov. 19, 2016. Accessed: May 22, 2024. [Online]. Available: http://arxiv.org/abs/1611.06355