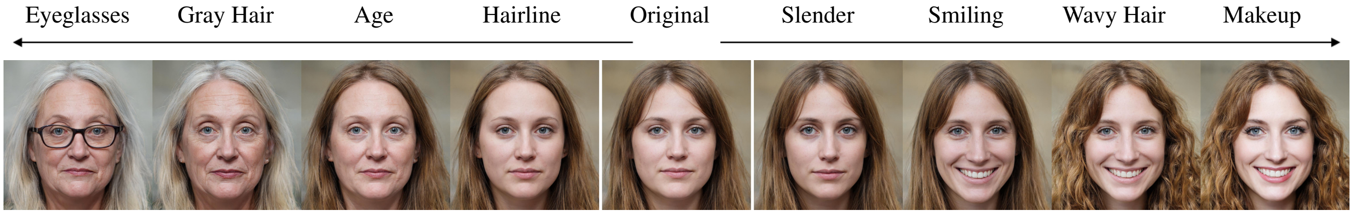
LEARNING NON-LINEAR DISENTANGLED EDITING FOR STYLEGAN

Xu Yao[⋆†], *Alasdair Newson*[⋆], *Yann Gousseau*[⋆], *Pierre Hellier*[†]

[⋆]LTCI, Télécom Paris, Institut Polytechnique de Paris, France
[†]InterDigital R&I, 975 avenue des Champs Blancs, Cesson-Sévigné, France



Sequential disentangled attribute manipulation. We show in this example how to achieve realistic, controllable, disentangled face editing. From the original image (center), we propose two opposite editing directions where only one attribute is manipulated at a time. To the right: 'slender', 'smiling', 'wavy hair' and 'makeup' and to the left: 'receding hairline', 'age', 'gray hair' and 'eyeglasses'. All results are obtained at resolution 1024^2.

ABSTRACT

Recent work has demonstrated the great potential of image editing in the latent space of powerful deep generative models such as StyleGAN. However, the success of such methods relies on the assumption that a linear hyperplane may separate the latent space into two subspaces for a binary attribute. In this work, we show that this hypothesis is a significant limitation and propose to learn a non-linear, regularized and identity-preserving latent space transformation that leads to more accurate and disentangled manipulations of facial attributes.

Index Terms— Face editing, disentangled representation, GAN

1. INTRODUCTION

Facial attribute manipulation has many real-world applications, such as film post-production or photo retouching. Previous works addressing this problem have mainly focused on encoder-decoder based architectures. Recently, there has been an increasing interest in exploring this problem in the latent space of generative networks, due to their impressive results in high quality images synthesis [1, 2, 3, 4]. Current approaches assume that a linear interpolation of latent codes lead to smooth variations of a visual attribute [5, 6]. However, the discovered directions often modify several attributes simultaneously; they are *entangled*. For example, wearing eyeglasses is often correlated with age. Furthermore, since no attribute has been used in the training of the generative models, there is no reason to assume that the transformation associated to one attribute should be linear. This work proposes a transformation network to navigate the StyleGAN latent space in a disentangled and non-linear manner, so that each attribute can be manipulated independently and smoothly.

To achieve this, we train a transformation network in the latent space to modify each attribute independently, explicitly preserving the other attributes. To ensure better disentanglement, we propose a correlation-wise attribute regularization term in the objective function during the training. In addition our model handles the sequential modification of several attributes with great photo-realism, which, to the best of our knowledge, we are the first to achieve. We conduct experiments on the state-of-the-art generative model StyleGAN2 [3] and perform qualitative and quantitative evaluations.

Our contributions are summarized as follows: (1) We propose a transformation network to navigate the StyleGAN latent space in a non-linear manner. (2) We realize controllable and disentangled facial attribute manipulations without any manual intervention, contrary to previous approaches [5, 6]. (3) We propose a correlation-wise attribute regularization term, to better preserve the attributes which we do not wish to modify. (4) We illustrate experimentally that our approach achieves a sequential modification of several attributes, allowing artists to edit images in a controlled, smooth and stable manner for real-world applications.

2. RELATED WORKS

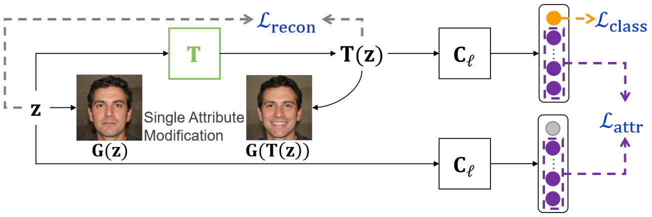
Facial Attribute Manipulation. Upchurch *et al.* [7] proposed an optimization-based method using linear interpolation of deep features from pretrained convnets to realize highlevel semantic transformations. Lample *et al.* [8] and Choi *et al.* [9] trained encoder-decoder based architectures that are able to disentangle attributes in the latent space and generate manipulated images by varying the attribute values. He *et al.* [10] and Liu *et al.* [11] further improved the attribute manipulation accuracy by introducing additional attribute manipulation loss. These works are limited to low resolution images. Recently, Chen *et al.* [12] proposed a pixel translation framework for high resolution facial image editing. Viazovetskyi *et al.* [13] used generated high resolution images to train the pix2pixHD [14] for facial attribute manipulation.

Latent Space Exploration. Thanks to the style mixing capacity of StyleGAN [2], various studies have assessed the possibility of manipulating images via latent spaces of image generators. Collins *et al.* [15] applied k-means clustering to the hidden layer activations of StyleGAN and discovered a decomposition of semantic objects, which allow local editing. Abdal *et al.* [16] proposed an optimization-based method to project real images to the latent space of StyleGAN, which is combined with a scribbled image to obtain a realistic image modified with StyleGAN. Goetschalckx *et al.* [17] learned a manifold in the latent space of BigGAN [4] to modify the memorability of generated images. The work of Shen *et al.* [5] shows that GANs trained on high quality images learn various semantics in some linear subspaces of the latent space. Tewari *et al.* [18] introduced an approach that provides a face rig-like control on generated images by training a rigging network between 3D morphable face model's semantic parameters and StyleGAN's input. Other approaches have attempted to imitate or directly carry out Principal Component Analysis (PCA) in the latent space of generative networks [6, 19]. Recent work of Nie *et al.* [20] showed that disentanglement of StyleGAN's latent space can be improved by adding a mutual information loss and a limited supervision at training time.

3. METHOD

Let $\mathbf{z} \in \mathcal{Z}$ be a random vector drawn from the latent space $\mathcal{Z}$ of a given generator $\mathbf{G}$. Typically in our work, we refer to the latent space $\mathcal{W}$ of StyleGAN2 generator [3]. Let $\mathbf{x} = \mathbf{G}(\mathbf{z}) \in \mathcal{X}$ be the corresponding image generated with $\mathbf{G}$. Our goal is to learn a transformation network $\mathbf{T}$, such that $\mathbf{G}(\mathbf{T}(\mathbf{z}))$ is an image with the same identity as $\mathbf{x}$, but where a single facial attribute has been modified, all other attributes being unchanged. This transformation provides the non-linear navigation of the latent space for modifying facial attributes, which is the final goal of this work. See Figure 1 for an illustration.

High quality generators contain many parameters, so to reduce training time we compute the objective functions in the latent space instead of the image space. We first train a classifier $\mathbf{C}_\ell$ in the latent space to measure the attributes of latent codes. During the training of $\mathbf{T}$, $\mathbf{C}_\ell$ is freezed. To modify the target attribute, we apply a classification loss. To preserve the other attributes and maintain the identity, we use, respectively, a correlation-wise attribute regularization term and a reconstruction loss on the latent codes.



$\mathcal{L}_{\text{recon}}$
T
T(z)
$\mathbf{C}_\ell$
$\mathcal{L}_{\text{class}}$
z
G(z)
Single Attribute Modification
G(T(z))
$\mathbf{C}_\ell$
$\mathcal{L}_{\text{attr}}$

Fig. 1. **Latent vector transformation model for single attribute modification.** Given a latent vector $\mathbf{z}$, $\mathbf{T}(\mathbf{z})$ is the modified latent vector obtained through our transformation network $\mathbf{T}$, which is trained to modify one target attribute in the generated image, without affecting the others. $\mathbf{C}_\ell$ is a pretrained attribute classifier in the latent space.

3.1. Models

Latent Classifier. In order to train an attribute classifier $\mathbf{C}_\ell$ in the latent space, we first generate training data by randomly sampling $500,000$ latent vectors and passing them through the generator $\mathbf{G}$. We then automatically label the corresponding images using a classifier pretrained on CelebA-HQ dataset [1]. The pretrained classifier is implemented using ResNet-50 network [21] and trained to predict simultaneously all the attributes of CelebA-HQ. Thus we obtain the label for each latent vector. A latent code classifier $\mathbf{C}_\ell$ is further trained on the latent vector-label pairs. Our implementation uses three linear layers with ReLU activations, which achieves higher precision, recall and F_1 compared to a simple linear layer. This shows that the linear hypothesis of the latent space is not appropriate.

Latent Transformer. We express the transformation $\mathbf{T}$ as a deformation network $\mathbf{H}$ modeled with two fully connected layers with LeakyReLU activations. The scaling factor α is used to control the direction and the amount of change of the target attribute:

$$\mathbf{T}(\mathbf{z}, \alpha) = \mathbf{z} + \alpha \cdot \mathbf{H}(\mathbf{z}), \text{with } \alpha \in [-1, 1]. \tag{1}$$

3.2. Training Objectives

For each target attribute index j, we train a separate network $\mathbf{T_j}$. To simplify notation, we omit the index j in $\mathbf{T_j}$ and the losses. Compared to $\mathbf{x}$, the modified image $\mathbf{G}(\mathbf{T}(\mathbf{z}))$ should differ in the target attribute only, the others being preserved. For this, we use the following losses:

Classification. We denote the pretrained latent classifier as $\mathbf{C}_\ell$. The modified latent vector $\mathbf{T}(\mathbf{z})$ should have the desired label β for the j^{th} target attribute:

$$\mathcal{L}_{\text{class}} = \phi(\mathbf{C}_\ell(\mathbf{T}(\mathbf{z}))[j], \beta), \tag{2}$$

where ϕ denotes the binary cross-entropy loss.

Correlation-wise attribute regularization. In order to preserve other attributes, we use Euclidean distance between the classification features, extracted from $\mathbf{C}_\ell$. The distance of each attribute is weighted according to its correlation with the target attribute. For the attributes strongly correlated with the target, the coefficient is close to zero so as not to block the transform, while for the unrelated attributes it is close to one, fully regularized. The regularization term is defined as:

$$\mathcal{L}_{\text{attr}} = \sum_{i \neq j} (1 - |\gamma_{ij}|)\, \mathbb{E}[||\mathbf{C}_\ell(\mathbf{T}(\mathbf{z}))[i] - \mathbf{C}_\ell(\mathbf{z})[i]||_2], \quad (3)$$

where γ_{ij} is the correlation of the attribute i^{th} and the target attribute j^{th} computed on CelebA-HQ.

Reconstruction. To enforce identity preservation of the modified image, regularize the difference between $\mathbf{z}$ and $\mathbf{T}(\mathbf{z})$:

$$\mathcal{L}_{\text{recon}} = \mathbb{E}[||\mathbf{T}(\mathbf{z}) - \mathbf{z}||_2]. \quad (4)$$

The full objective can be expressed as:

$$\mathcal{L} = \mathcal{L}_{\text{class}} + \lambda_{\text{attr}}\mathcal{L}_{\text{attr}} + \lambda_{\text{recon}}\mathcal{L}_{\text{recon}}, \quad (5)$$

where λ_{attr} and λ_{recon} are weights balancing each loss.

4. EXPERIMENTS

4.1. Implementation Details

We conduct experiments on the state-of-the-art image generator StyleGAN2 [3] pretrained on FFHQ 1024^2 dataset. Our training data is the latent vector-label pairs presented in Section 3.1. For each facial attribute, we train a separate transformation network in the latent space $\mathcal{W}$ of StyleGAN2. We set $\lambda_{\text{attr}} = 1$ and $\lambda_{\text{recon}} = 100$. At training time, the scaling factor α is set according to the target attribute classification probability p of the original latent code ($1-p$ for $p < 0.5$, $-p$ for $p > 0.5$). At test time, α is suggested to be sampled from $[-1.5, 1.5]$. Our implementation is built in PyTorch [22]. The batch size is set to 16 and each model is trained on $10K$ random latent vector-label pairs for 10 epochs. We use the Adam [23] optimizer with learning rate set to 10^{-4}, $\beta_1 = 0.9$ and $\beta_2 = 0.999$.

4.2. Qualitative Results

Attribute manipulation. We compare our manipulation results with two recent state-of-the-art methods: GANSpace [6] and InterFaceGAN [5]. GANSpace explores latent directions using PCA and manually identifies the attributes corresponding to each direction. InterFaceGAN randomly samples latent codes, annotates the generated images, and finds a hyper-plane for each attribute. We train InterFaceGAN in StyleGAN2 latent space for all the attributes of CelebA-HQ, using the $500K$ latent vector-label pairs. To compare with these methods, we are limited to the attributes that both can manage. In Figure 2, each row shows the manipulation results of one attribute. In each row, the 1st column is the original generated image. The 2nd column is the result of GANSpace, where the attributes are not well disentangled. This method has good performance on local features like 'narrow eyes' or 'beard'. However, 'age' is correlated with gender (4th row). The 3rd column corresponds to the result of InterFaceGAN, where face identity undergoes visible modifications and attributes are not well disentangled. For example, modifying 'gray hair' also changes age; modifying 'narrow eyes enhances smiling. This is because when training the hyper-plane for one attribute, no regularization was applied to other attributes. The last column shows our result. Thanks to the attribute regularization term, our method preserves better the identity and generates more disentangled manipulations.

Scaling Factor. In order to ensure that the scaling factor (amount of change in each attribute direction) is not the

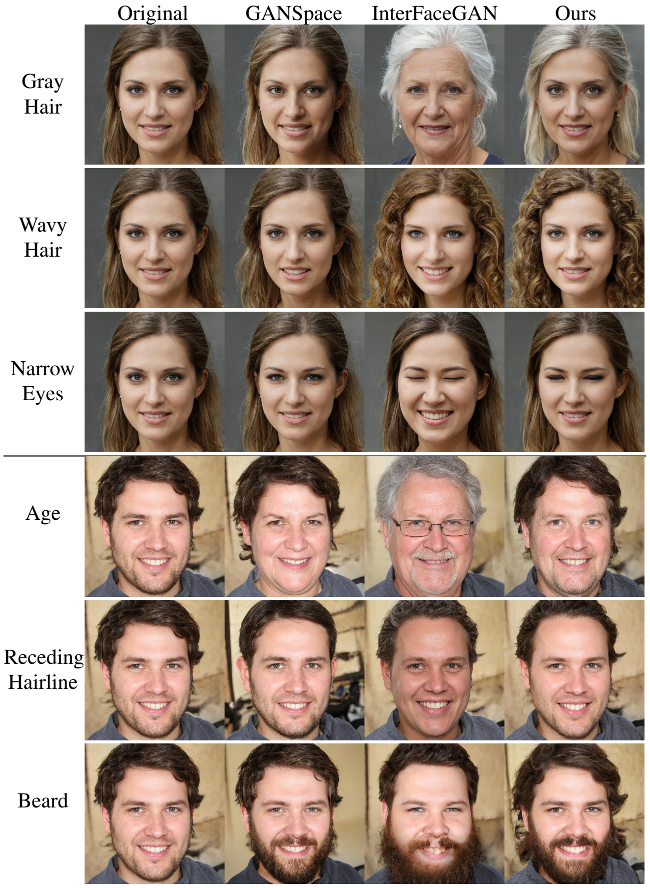


Fig. 2. **Disentangled facial attribute manipulation on StyleGAN2 generated images**. Each Line corresponds to an attribute manipulation. In each subfigure, from left to right are the original image, the manipulation result of GANSpace [6], that of InterFaceGAN [5] and ours. Compared to recent approaches, our method achieves a controllable, disentangled and realistic editing, where the person's identity is preserved.

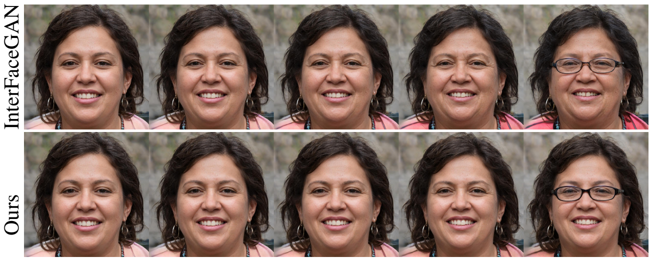


Fig. 3. **Progressive latent manipulation**. From left to right, the results of progressively moving a latent code in the 'Eyeglasses' direction. Compared to InterFaceGAN, our results preserve the identity and other attributes better.

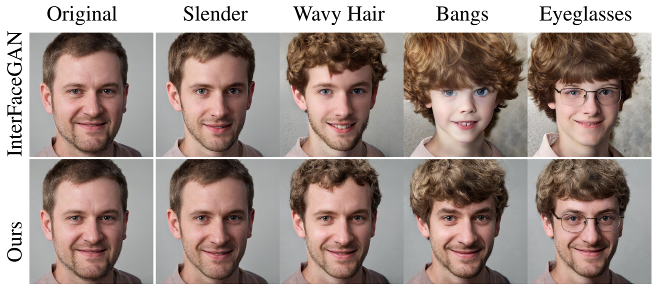


Fig. 4. **Sequential disentangled attribute manipulation**. The 2nd to 5th columns correspond to a sequential modification of: 'slender', 'wavy hair', 'bangs' and 'eyeglasses'. Compared to InterFaceGAN, our approach achieves better disentangled manipulation and allows sequential attribute modification.

source of the previous approaches shortcomings, in Figure 3 we show a progressive manipulation result on the 'eyeglasses' attribute. Compared to InterFaceGAN, our manipulation performs better in identity and attribute preservation.

Sequential attribute manipulation. Another advantage of our approach is the ability to modify several attributes sequentially, in a disentangled manner. In Figure 4 we show a sequential modification of several attributes: 'slender', 'wavy hair', 'bangs' and 'eyeglasses'. The top row presents the results of InterFaceGAN, where we observe that when adding bangs, age is reduced. Their method is able to manage two consecutive attribute modifications but fails on more attributes as the error of each manipulation accumulates. On the contrary, our method is able to realize several sequential attribute manipulations without influencing the identity of the original person. As shown in the bottom row, between each two columns, only the indicated attribute is changed.

4.3. Quantitative Evaluation

We perform a quantitative comparison of our approach and InterFaceGAN. For each target attribute, 1000 random latent codes are manipulated to reverse the target attribute, such as 'w/ bangs' to 'w/o bangs'. We label the input and manipulated latent codes by passing them to StyleGAN2 and annotating the output images. We analyze the manipulation results with three metrics: (1) *Identity Preservation,* percentage of identity-preserved manipulations, measured with the cosine similarity between VGG-Face [24] embeddings at 0.1% False Acceptance Rate; (2) *Attribute Preservation,* average of percentage of unchanged attributes; (3) *Classification Accuracy,* percentage of successful manipulations on the target attribute. (1) and (2) are calculated on successful manipulations.

Attribute	Identity Preservation(%)		Attributes Preservation(%)		Classification Accuracy(%)	
	InterFace	Ours	InterFace	Ours	InterFace	Ours
Age	76.28	99.69	81.79	90.65	99.90	98.10
Eyeglasses	99.53	99.69	86.22	90.25	85.50	97.10
Gender	91.28	99.23	76.91	86.25	98.60	90.90
Smiling	99.70	100.00	86.87	89.58	100.00	100.00
Average	91.70	99.65	82.95	89.19	96.00	96.53

Table 1. **Quantitative evaluation on attribute manipulations**. For each target attribute, we manipulate 1000 latent codes and compare attributes of input and output images. The identity preservation rate is the percentage of identity-preserved manipulations. The attribute preservation rate is the percentage of unchanged ones on other attributes. The classification accuracy is the percentage of successful manipulations on the target attribute. Our approach outperforms InterfaceGAN in identity and attribute preservation on successful manipulations, while achieving similar classification accuracy on the target attribute.

To have a comparable classification accuracy, for our method we set the scaling factor to 1.25, while for InterFaceGAN we set the distance to 3, maximum value recommended by the authors. As shown in Table 1, our method outperforms InterFaceGAN in identity and attribute preservation. We also observe that InterFaceGAN displays better classification accuracy on some attributes, probably taking advantage of the entanglement of attributes. In the case of 'gender', for example, adding 'beard' and changing 'hairstyle' when transforming women to men helps the classification.

5. CONCLUSION

In this paper, we presented a simple framework for disentangled attribute manipulation on StyleGAN2 generated images. Compared to state-of-the-art approaches, our method achieves realistic manipulations with better disentanglement and higher identity preservation. In addition, our method is able to generate sequential disentangled attribute manipulations, which offers great flexibility and controllability to users. This property is favorable for artists in imaging industry, with a potential in more real-world applications. Further work will study the extension to natural images.

6. REFERENCES

[1] 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.

[2] Tero Karras, Samuli Laine, and Timo Aila, “A style-based generator architecture for generative adversarial networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410.

[3] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila, “Analyzing and improving the image quality of stylegan,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8110–8119.

[4] Andrew Brock, Jeff Donahue, and Karen Simonyan, “Large scale GAN training for high fidelity natural image synthesis,” in *International Conference on Learning Representations*, 2019.

[5] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou, “Interpreting the latent space of gans for semantic face editing,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9243–9252.

[6] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris, “Ganspace: Discovering interpretable gan controls,” in *Proc. NeurIPS*, 2020.

[7] Paul Upchurch, Jacob Gardner, Geoff Pleiss, Robert Pless, Noah Snavely, Kavita Bala, and Kilian Weinberger, “Deep feature interpolation for image content changes,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7064–7073.

[8] Guillaume Lample, Neil Zeghidour, Nicolas Usunier, Antoine Bordes, Ludovic Denoyer, and Marc’Aurelio Ranzato, “Fader networks: Manipulating images by sliding attributes,” in *Advances in Neural Information Processing Systems*, 2017, pp. 5967–5976.

[9] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo, “Stargan: Unified generative adversarial networks for multi-domain image-to-image translation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8789–8797.

[10] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen, “Attgan: Facial attribute editing by only changing what you want,” *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5464–5478, 2019.

[11] Ming Liu, Yukang Ding, Min Xia, Xiao Liu, Errui Ding, Wangmeng Zuo, and Shilei Wen, “Stgan: A unified selective transfer network for arbitrary image attribute editing,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3673–3682.

[12] Xuanhong Chen, Bingbing Ni, Naiyuan Liu, Ziang Liu, Yiliu Jiang, Loc Truong, and Qi Tian, “Coogan: A memory-efficient framework for high-resolution facial attribute editing,” in *European Conference on Computer Vision*. Springer, 2020, pp. 670–686.

[13] Yuri Viazovetskyi, Vladimir Ivashkin, and Evgeny Kashin, “Stylegan2 distillation for feed-forward image manipulation,” in *European Conference on Computer Vision*. Springer, 2020, pp. 170–186.

[14] 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*, 2018, pp. 8798–8807.

[15] Edo Collins, Raja Bala, Bob Price, and Sabine Susstrunk, “Editing in style: Uncovering the local semantics of gans,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5771–5780.

[16] 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*, 2020, pp. 8296–8305.

[17] Lore Goetschalckx, Alex Andonian, Aude Oliva, and Phillip Isola, “Ganalyze: Toward visual definitions of cognitive image properties,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 5744–5753.

[18] Ayush Tewari, Mohamed Elgharib, Gaurav Bharaj, Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhofer, and Christian Theobalt, “Stylerig: Rigging stylegan for 3d control over portrait images,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6142–6151.

[19] Chi-Hieu Pham, Saïd Ladjal, and Alasdair Newson, “Pcaae: Principal component analysis autoencoder for organising the latent space of generative networks,” *arXiv preprint arXiv:2006.07827*, 2020.

[20] Weili Nie, Tero Karras, Animesh Garg, Shoubhik Debhath, Anjul Patney, Ankit B Patel, and Anima Anandkumar, “Semi-supervised stylegan for disentanglement learning,” *arXiv preprint arXiv:2003.03461*, 2020.

[21] 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*, 2016, pp. 770–778.

[22] 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*, pp. 8024–8035. 2019.

[23] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv:1412.6980*, 2014.

[24] Omkar M Parkhi, Andrea Vedaldi, and Andrew Zisserman, “Deep face recognition,” *British Machine Vision Conference*, 2015.