OPTIMIZING LATENT SPACE DIRECTIONS FOR GAN-BASED LOCAL IMAGE EDITING

Ehsan Pajouheshgar, Tong Zhang, and Sabine Süsstrunk

School of Computer and Communication Sciences, EPFL, Switzerland

Fig. 1: Examples of localized edits performed by our method on GAN-generated images. The images to the left and right of the middle one are results of moving the latent code in directions discovered by our method to edit each semantic part.

ABSTRACT

Generative Adversarial Network (GAN) based localized image editing can suffer from ambiguity between semantic attributes. We thus present a novel objective function to evaluate the locality of an image edit. By introducing the supervision from a pre-trained segmentation network and optimizing the objective function, our framework, called Locally Effective Latent Space Direction (LELSD), is applicable to any dataset and GAN architecture. Our method is also computationally fast and exhibits a high extent of disentanglement, which allows users to interactively perform a sequence of edits on an image. Our experiments on both GAN-generated and real images qualitatively demonstrate the high quality and advantages of our method.

Index Terms— GANs, Latent Space Directions, Local Image Editing, Semantic Attribute Editing, StyleGAN

1. INTRODUCTION

Generative Adversarial Networks, like StyleGAN [1, 2] and BigGAN [3], are capable of generating diverse high-resolution images, sometimes indistinguishable from real photos. In addition, substantial semantic meaning has been found in the latent space of trained GANs, which makes high-level semantic image editing possible. Semantic image editing using GANs [4] has found a broad range of applications in Digital Art [5], Fashion [6], Interior Design [7], and Face Editing [8].

Several recent works control the semantics of the GAN-generated images by tweaking the latent code to perform global [9, 10, 11, 12, 13, 14] or localized [7, 15, 16, 17] image editing. Although a lot of progress has been made in global image editing, it remains challenging to disentangle the semantic attributes and thus control the local semantics of the image. Therefore, in this paper, we specifically focus on localized semantic image editing, where the goal is to control one semantic attribute of the image without changing other image parts.

State-of-the-art methods for finding semantically localized latent space directions rely on the first-order Taylor expansion of the generator network [16, 17], and thus assume a linear relation between the latent code and the generated image. Since this assumption is only valid in a close proximity to the original latent code, these methods [16, 17] are limited in the range of local editing they can achieve. Meanwhile, [16] proposes a method to achieve disentanglement by exhaustively searching the latent space of StyleGAN [1, 2], and consequently cannot be applied to other GAN architectures. However, we are specifically interested in designing a

GAN-based Image Editing Algorithms	Supervision	A	B	C	D	E	F	G	
[10, 11, 12]	Unsupervised	✓	✓	✓	✓	✓	✗	✗	Latent Space Traversal
Zhu et al. [17]	Unsupervised	✓	✗	✓	✓	✗	✓	✗	
[19, 14]	Self-Supervised	✓	✓	✓	✓	✓	✗	✗	
InterFaceGAN [9]	Supervised	✗	✓	✓	✓	✓	✗	✗	
GANALYZE [13]	Supervised	✓	✓	✓	✓	✓	✗	✗	
StyleSpace [16]	Supervised	✓	✓	✗	✓	✗	✓	✓	
LELSD (Ours)	Supervised	✓	✓	✓	✓	✗	✓	✓	
Bau et al. [20]	Unsupervised	✓	✗	✗	✗	✗	✓	✓	Image Composition
Chai et al. [21]	Unsupervised	✓	✓	✓	✗	✗	✓	✓	
Editing in Style [15]	Unsupervised	✓	✓	✗	✗	✗	✓	✗	
Zhang et al. [7]	Supervised	✗	✓	✗	✗	✗	✓	✓	
Barbershop [22]	Supervised	✓	✓	✗	✗	✗	✓	✗	

Table 1: Comparison of GAN-based image editing algorithms by their characteristics. **(A)** Works on any Dataset, **(B)** Does not need test-time optimization, **(C)** Works on any GAN architecture, **(D)** Can perform the edit using a single image, **(E)** Allows global semantic editing, **(F)** Allows localized semantic editing, **(G)** Allows editing any object in the image.

framework which is not only agnostic to the GAN architecture, but also is able to effectively disentangle the semantic attributes. To this end, we propose *Locally Effective Latent Space Directions* (LELSD), a framework to find latent space directions that affect local regions of the output image. We introduce a novel objective function to evaluate the localization and disentanglement of an image edit by incorporating supervision from a pre-trained semantic segmentation model. Note that, the supervision could also come from unsupervised [15] or weakly-supervised [18] models that use the intermediate featuremaps of the generator network to achieve semantic segmentation. As a result, our method is not limited to any specific dataset. Figure 1 shows some of the semantic edits that our method can perform. Since we apply optimization instead of exhaustive search, our training time is three orders of magnitude faster than [16]. Meanwhile, unlike [17] we do not perform test-time optimization and thus allow interactive image editing.

2. RELATED WORKS

Table 1 summarizes the strengths and weaknesses of of different GAN-based image editing methods. Some works use an unsupervised approach to discover meaningful latent space directions, and then manually attribute a semantic meaning to each of the found directions [10, 11, 12, 19]. However, the discovered directions are semantically entangled and usually change more than one attribute simultaneously. Hence, they are not suitable for localized image editing.

To solve this problem, [9, 13, 14] use an external supervision and find latent space directions that yield the desired change in the generated images. This is done by finding the latent space direction that maximizes a designed objective function.

There are two distinct approaches to GAN-based localized semantic image editing: 1) *Latent Space Traversal*, and 2) *Image Composition*. In the former, the goal is to discover latent space directions that yield localized changes in the output image. The later aims to combine different parts from two images to achieve localized editing, e.g., transferring the nose from one face image to another. Our method falls in the first category. The disadvantage of the Image Composition methods is that they require a second image to transfer the parts from (See Table 1, Column D). In this paper we thus focus on the Latent Space Traversal methods as they can perform single-image editing, which is more user-friendly.

InterFaceGAN [9] finds latent space directions that maximally change the score of a pre-trained SVM classifier for face attributes, and therefore is only applicable on face images. On the other hand, [16, 17] use the gradient of the output image w.r.t the latent code to find subspaces of the latent code that highly correlate with local regions in the generated image. Wu et al. [16] use a pretrained semantic segmentation network to find channels in StyleGAN's style code that have a high overlap with a semantic category, e.g., eyes. Zhu et al [17] perform test-time optimization to find latent space directions that mostly affect the regions of the image outline in the user's query. The upper part of Table 1 compares the existing Latent Space Traversal methods in the literature.

Similar to [7, 16, 22], we use the supervision from a pretrained semantic segmentation network, and propose a novel scoring function that encourages the latent space directions that mainly affect the desired semantic part to edit, e.g., eyes in a face image. Meanwhile, we allow both coarse and fine-grained semantic changes by adopting the layer-wise editing approach of [10] for both style-based GANs and BigGAN. GAN Inversion is complementary to our method and advancement in GAN Inversion research [23, 24] also enhances the quality of semantic editing of real photos when combined with our method.

3. METHOD

Figure 2 provides an overview of our method. The generator network $G(.)$ in a GAN generates an image starting from a latent code $\omega \in \Omega$, i.e. $\mathbf{x} = G(\boldsymbol{\omega}) = f(h(\boldsymbol{\omega}))$ where $\mathbf{r} = h(\boldsymbol{\omega})$ is a tensor representing the activation of an intermediate layer in the network. The latent space Ω can be any of $\mathcal{Z}, \mathcal{W}, \mathcal{W}+, \mathcal{S}$ for the StyleGAN generator as in [16], and $\mathcal{Z}, \mathcal{Z}+$ for BigGAN generator as in [10]. Semantic editing of an image is done by moving its latent code along a specific direction

$$\mathbf{x}^{edit}(\mathbf{u}) = f(\mathbf{r}^{edit}(\mathbf{u})) = G(\boldsymbol{\omega} + \alpha\mathbf{u}) \tag{1}$$

where α controls the intensity of the change, and the latent direction $\mathbf{u}$ determines the semantic of the edit.

Our goal is to find an editing direction $\mathbf{u}_c$ that mostly changes parts of the generated image corresponding to a binary mask given by a pretrained semantic segmentation model $\mathbf{s}_c(\mathbf{x})$ where c indicates the desired object to edit in the image.

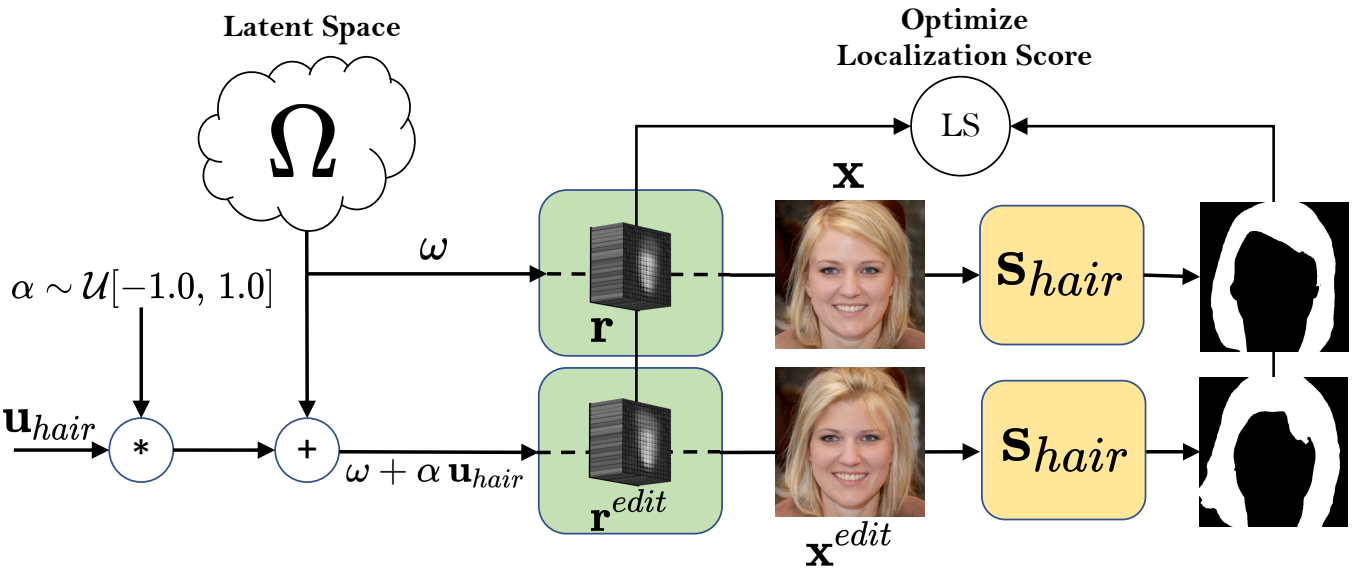

Fig. 2: Scheme of our method. The green boxes indicate the pretrained generator network and the yellow box shows the pretrained semantic segmentation model. We draw random samples from the latent space Ω and optimize the latent space direction to maximize the localization score.

Based on this, we can write the localization score as

$$LS(\mathbf{u}) = \frac{\sum_{i,j} \downarrow_{\mathbf{s}_c}(\mathbf{x}, \mathbf{x}^{edit}) \odot |\mathbf{r} - \mathbf{r}^{edit}(\mathbf{u})|^2}{\sum_{i,j} |\mathbf{r} - \mathbf{r}^{edit}(\mathbf{u})|^2} \qquad (2)$$

where i, j iterate over the spatial dimensions and $\downarrow_{\mathbf{s}_c}(\mathbf{x}, \mathbf{x}^{edit})$ is the average of the two semantic segmentation masks downsampled to the resolution of the corresponding layer. This objective function measures the proportion of the change in the featuremap that happens inside the semantic segmentation mask. Our final objective function is calculated by simply summing up the localization scores for all intermediate layers in the generator network. Unlike [17] that only aims to achieve localized change in the generated image, we also encourage the intermediate featuremaps to only change locally. This allows us to achieve a larger variety of edits than [17]. For example, we can change both hairstyle and hair color, while [17] cannot manipulate hairstyle.

4. EXPERIMENTS

For the pretrained semantic segmentation model, we use Face-BiSeNet [25] for face/portrait images and DeepLabV3 [26] for other images. We use the Adam optimizer to find a latent space direction that maximizes the localization score defined in Equation (2)[1]. We train on 800 randomly sampled latent codes with a batch size of 4, starting from a learning rate of 0.001 and halving it every 50 steps. Optimizing a latent space direction for each semantic part takes approximately two minutes on a single Tesla V100 GPU. We observe that our method is robust to the choice of the mask aggregation method in Equation (2) and works as well with the union or the intersection of the two masks.

[1]Our code can be found at https://github.com/IVRL/LELSD

4.1. Finding multiple directions

In order to find two or more distinct directions for editing the same semantic part such as hair, we add $R(\mathbf{u}_1, \ ..., \mathbf{u}_k) = \frac{-1}{2}||\text{Corr}(\mathbf{u}_1, \ ..., \mathbf{u}_K) - \text{I}_K||_\text{F}$ as a regularization term to our objective function, where Corr(.) is the correlation matrix of a set of vectors, $||.||_\text{F}$ is the Frobenius Norm, and I_K is the $K \times K$ identity matrix. Our final objective can thus be written as

$$J(\mathbf{u}_1, \ ..., \mathbf{u}_k) = \sum_k LS(\mathbf{u}_k) + cR(\mathbf{u}_1, \ ..., \mathbf{u}_k) \qquad (3)$$

where c is the regularization coefficient. The added regularization term encourages the editing directions to be mutually perpendicular, and carry distinct semantics as can be seen in Figure 3. We linearly increase the number of training samples w.r.t K and alternate between each $\mathbf{u}_k$ during the optimization process.

4.2. Comparison with First-Order methods

Both [16, 17] rely on the first-order Taylor expansion of the generator network and assume a linear relationship between the generated image and the latent code. This causes them to perform poorly as the editing strength α increases. Since α in Equation (1) has a different scale for each GAN-based image editing method, we use the LPIPS distance [27] for the comparison. For each method we find the value of α such

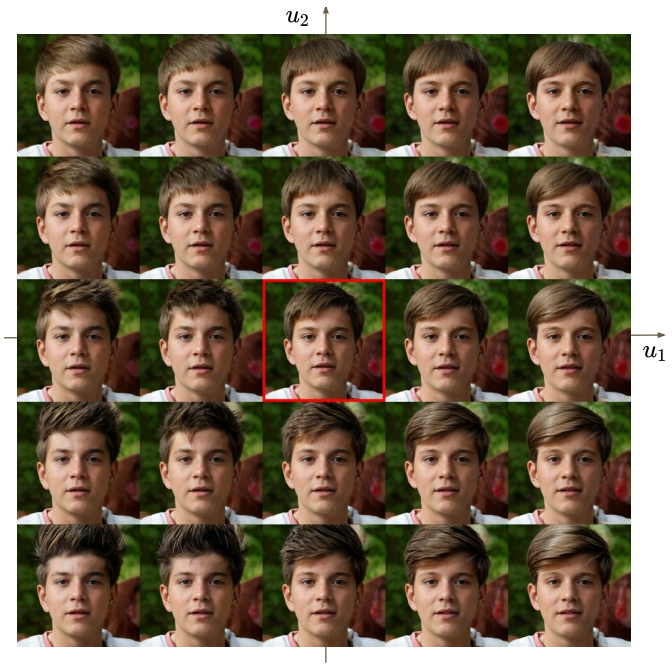

Fig. 3: Visualization of two latent space directions found by our method for editing hairstyle. Linearly combining different directions gives limitless image editing possibilities to users.

that $LPIPS(\mathbf{x}, \mathbf{x}^{edit}) = d$, and show that for higher values of d where the linearity assumption is not valid, our method outperforms [16]. Note that there are two values of α that yield the same LPIPS distance d, where one is positive and the other one is negative. Figure 4 compares the edits performed on mouth and hair by our method and StyleSpace [16], for different values of editing strength. As d and subsequently absolute value of α increase, our method performs coherently while StyleSpace distorts the semantics of the image.

4.3. Editing Real Images

Combined with a GAN Inversion model, our method allows editing real images. We use e4e [24] trained on StyleGAN2 FFHQ to project real face photos into the latent space of the StyleGAN. More importantly, we can perform sequential editing by simply adding up the discovered latent space directions for each semantic. Figure 5 shows a series of edits applied to the inversion of real photos. As can be seen, the semantics of the edits are consistent across different images. The quality of the GAN inversion is beyond the scope of this paper.

4.4. Performance Comparison

Interactiveness and performance are two very important factors in the image editing experience. The method in [17] requires test-time optimization and hence is not interactive. Although the approach of [16] allows interactive editing, it requires a lot of training time as it needs to separately back-propagate through all 6080 channels in the style code of the StyleGAN. As we use the same number of training samples as [16], we estimate that our method is three orders of magnitude faster than [16].

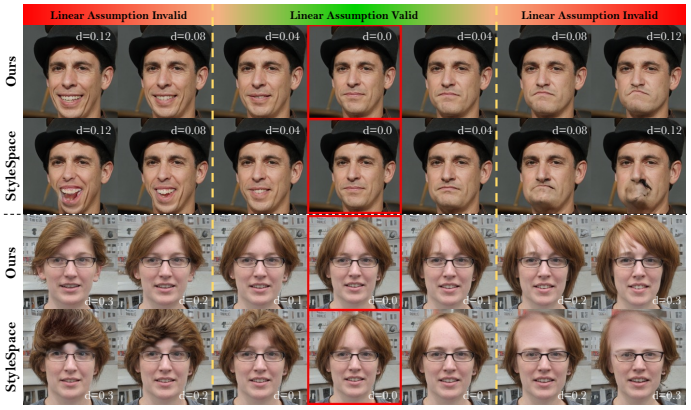

Fig. 4: Comparison of our method with StyleSpace [16]. The first-order approximation of the generator network is only valid in a close proximity of the original latent code. Hence, gradient-based methods like StyleSpace [16] perform poorly as the editing strength increases. The d in the figure shows the LPIPS distance between the edited and original images.

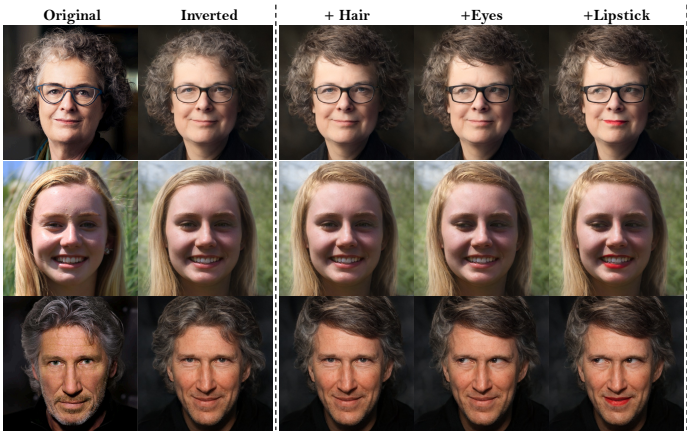

Fig. 5: Sequential editing applied to real images. The edits are semantically consistent between different images and each edit only changes the desired part without affecting previous changes.

5. CONCLUSION

In this work, we have presented LELSD, our computationally friendly framework that uses the supervision from a pre-trained semantic segmentation network to maximize a novel objective function that encourages local image edits and can be applied to any GAN architecture and dataset. Our experiments in different setting qualitatively show the advantage of our method, especially in the extent of disentanglement achieved between local attributes.

6. REFERENCES

[1] Tero Karras, Samuli Laine, and Timo Aila, "A style-based generator architecture for generative adversarial networks," in *CVPR*, 2019, pp. 4401–4410.

[2] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila, "Analyzing and improving the image quality of stylegan," in *CVPR*, 2020, pp. 8110–8119.

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan, "Large scale gan training for high fidelity natural image synthesis," *arXiv preprint arXiv:1809.11096*, 2018.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," *NeurIPS*, vol. 27, 2014.

[5] Aaron Hertzmann, "Aesthetics of neural network art," *arXiv preprint arXiv:1903.05696*, 2019.

[6] Amir Hossein Raffiee and Michael Sollami, "Garmentgan: Photo-realistic adversarial fashion transfer," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 3923–3930.

[7] Chen Zhang, Yinghao Xu, and Yujun Shen, "Decorating your own bedroom: Locally controlling image generation with generative adversarial networks," *arXiv preprint arXiv:2105.08222*, 2021.

[8] FaceApp, "https://www.faceapp.com/," 2021.

[9] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou, "Interfacegan: Interpreting the disentangled face representation learned by gans," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[10] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris, "Ganspace: Discovering interpretable gan controls," *arXiv preprint arXiv:2004.02546*, 2020.

[11] Andrey Voynov and Artem Babenko, "Unsupervised discovery of interpretable directions in the gan latent space," in *ICML*. PMLR, 2020, pp. 9786–9796.

[12] Yujun Shen and Bolei Zhou, "Closed-form factorization of latent semantics in gans," in *CVPR*, 2021, pp. 1532–1540.

[13] Lore Goetschalckx, Alex Andonian, Aude Oliva, and Phillip Isola, "Ganalyze: Toward visual definitions of cognitive image properties," in *CVPR*, 2019, pp. 5744–5753.

[14] Ali Jahanian, Lucy Chai, and Phillip Isola, "On the" steerability" of generative adversarial networks," *arXiv preprint arXiv:1907.07171*, 2019.

[15] Edo Collins, Raja Bala, Bob Price, and Sabine Susstrunk, "Editing in style: Uncovering the local semantics of gans," in *CVPR*, 2020, pp. 5771–5780.

[16] Zongze Wu, Dani Lischinski, and Eli Shechtman, "Stylespace analysis: Disentangled controls for stylegan image generation," in *CVPR*, 2021, pp. 12863–12872.

[17] Jiapeng Zhu, Ruili Feng, Yujun Shen, Deli Zhao, Zhengjun Zha, Jingren Zhou, and Qifeng Chen, "Low-rank subspaces in gans," *arXiv preprint arXiv:2106.04488*, 2021.

[18] Jianjin Xu and Changxi Zheng, "Linear semantics in generative adversarial networks," in *CVPR*, 2021, pp. 9351–9360.

[19] Antoine Plumerault, Hervé Le Borgne, and Céline Hudelot, "Controlling generative models with continuous factors of variations," *arXiv preprint arXiv:2001.10238*, 2020.

[20] David Bau, Steven Liu, Tongzhou Wang, Jun-Yan Zhu, and Antonio Torralba, "Rewriting a deep generative model," in *ECCV*. Springer, 2020, pp. 351–369.

[21] Lucy Chai, Jonas Wulff, and Phillip Isola, "Using latent space regression to analyze and leverage compositionality in gans," *arXiv preprint arXiv:2103.10426*, 2021.

[22] Peihao Zhu, Rameen Abdal, John Femiani, and Peter Wonka, "Barbershop: Gan-based image compositing using segmentation masks," *arXiv preprint arXiv:2106.01505*, 2021.

[23] Rameen Abdal, Yipeng Qin, and Peter Wonka, "Image2stylegan++: How to edit the embedded images?," in *CVPR*, 2020, pp. 8296–8305.

[24] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or, "Encoding in style: a stylegan encoder for image-to-image translation," in *CVPR*, 2021, pp. 2287–2296.

[25] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *ECCV*, 2018, pp. 325–341.

[26] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[27] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018, pp. 586–595.