Closed-Form Factorization of Latent Semantics in GANs

Yujun Shen Bolei Zhou
The Chinese University of Hong Kong
{sy116, bzhou}@ie.cuhk.edu.hk

Figure 1. **Versatile interpretable directions of the latent space unsupervisedly discovered in different GAN models** including PGGAN [16], StyleGAN [17], BigGAN [4], and StyleGAN2 [18]. For each set of images, the middle one is the original output, while the left and the right are the output images by moving the latent code toward and backward the interpretable direction found by SeFa.

Abstract

A rich set of interpretable dimensions has been shown to emerge in the latent space of the Generative Adversarial Networks (GANs) trained for synthesizing images. In order to identify such latent dimensions for image editing, previous methods typically annotate a collection of synthesized samples and train linear classifiers in the latent space. However, they require a clear definition of the target attribute as well as the corresponding manual annotations, limiting their applications in practice. In this work, we examine the internal representation learned by GANs to reveal the underlying variation factors in an unsupervised manner. In particular, we take a closer look into the generation mechanism of GANs and further propose a closed-form factorization algorithm for latent semantic discovery by directly decomposing the pre-trained weights. With a lightning-fast implementation, our approach is capable of not only finding semantically meaningful dimensions comparably to the state-of-the-art supervised methods, but also resulting in far more versatile concepts across multiple GAN models trained on a wide range of datasets.[1]

1. Introduction

Generative Adversarial Networks (GANs) [8] have achieved tremendous success in image synthesis [16, 17, 4, 18]. It has been recently found that when learning to

[1]Project page is at https://genforce.github.io/sefa/.

synthesize images, GANs spontaneously represent multiple interpretable attributes in the latent space [7, 15, 24, 22, 27], such as gender for face synthesis [24] and lighting condition for scene synthesis [27]. By properly identifying these semantics, we can reuse the knowledge learned by GANs to reasonably control the image generation process, enabling a wide range of editing applications, like face manipulation [25, 9] and scene editing [27, 29].

The crux of interpreting the latent space of GANs is to find the meaningful directions in the latent space corresponding to the human-understandable concepts [7, 15, 24, 22, 27]. Through that, moving the latent code towards the identified direction can accordingly change the semantic occurring in the output image. However, due to the high dimensionality of the latent space as well as the large diversity of image semantics, finding valid directions in the latent space is extremely challenging.

Existing supervised approaches typically first randomly sample a large amount of latent codes, then synthesize a collection of images and annotate them with some pre-defined labels, and finally use these labeled samples to learn a classifier in the latent space. To get the labels for training, they either employ pre-trained attribute predictors [7, 24, 27] or utilize some simple statistical information of the image (*e.g.*, object position and color tone) [15, 22]. Several limitations rise from the above supervised training process. Firstly, relying on pre-defined classifiers hinders the algorithm from being applied to the case where the classifiers are not available or difficult to train. On the other hand, sampling is both time-consuming and unstable, *e.g.*, a different collection of synthesized data may lead to a different training result. Some very recent studies explore the unsupervised discovery of interpretable GAN semantics [26, 10], but they also require model training [26] or data sampling [10].

In this work, we propose a novel algorithm to discover the latent semantic directions learned by GANs, which is *independent of any kind of training or sampling*. We call it *SeFa* as the short for *Semantic Factorization*. Instead of relying on the synthesized samples as an intermediate step, SeFa takes a deep look into the generation mechanism of GANs to examine the relation between the image variation and the internal representation. In fact, GANs project a latent code to a photo-realistic image step by step (or say layer by layer), where each step learns a projection from one space to another. Many explanatory factors originate in such process. Thus we investigate the first projection step that directly acts on the latent space we want to study. We propose a *closed-form* method that can identify versatile semantics from the latent space by *merely using the pre-trained weights of the generator*. More importantly, these variation factors, unsupervisedly found by SeFa, are accurate and in a wider range compared to the state-of-the-art supervised approaches. We demonstrate some interesting manipulation results using the discovered semantics in Fig. 1. For instance, we can rotate the object in an image without knowing its underlying 3D model or pose label. Extensive experiments suggest that our approach is efficient and applicable to most popular GAN models (*e.g.*, PGGAN [16], StyleGAN [17], BigGAN [4], and StyleGAN2 [18]) that are trained on different datasets.

1.1. Related Work

Generative Adversarial Networks. GAN [8] has significantly advanced image synthesis in recent years [23, 2, 16, 4, 17, 18]. The generator in GANs can take a randomly sampled latent code as the input and output a high-fidelity image through adversarial learning. Existing GAN models are commonly built on deep convolutional neural networks where the latent code is fed into the first convolution layer using an affine transformation [23, 2, 16]. Recently, this idea is improved by the style-based generator [17, 18] where the latent code is mapped to layer-wise style codes and then fed into each convolution layer through Adaptive Instance Normalization (AdaIN) [14] operation.

Latent Semantic Interpretation. Generative models show great potential in learning variation factors from observed data. Chen *et al.* [5] and Higgins *et al.* [13] propose to add regularizers into the training process to explicitly learn an interpretable factorized representation. Recent work has found that the native GANs, without any constraints or regularizers, are able to automatically encode various semantics in the intermediate feature space [3] and the initial latent space [7, 15, 24, 27]. However, these methods are usually performed in a supervised fashion, which requires sampling a collection of images and labeling them to train a classifier. Thus they heavily rely on the attribute predictors or human annotators to get the label. Some concurrent work studies unsupervised semantic discovery in GANs. Voynov and Babenko [26] jointly learn a candidate matrix and a classifier such that the semantic directions in the matrix can be properly recognized by the classifier. Härkönen *et al.* [10] perform PCA on the sampled data to find primary directions in the latent space. However, they still require model training [26] and data sampling [10]. Differently, we study the generation mechanism of GANs and propose a *closed-form* factorization method, which is independent of any kind of training or sampling.

2. Method

We introduce SeFa, a closed-form method to discover latent interpretable directions in GANs. By taking a close look into the generation mechanism of GANs, SeFa can identify semantically meaningful directions in the latent space efficiently by decomposing the model weights.

2.1. Preliminaries

Generation Mechanism of GANs. The generator $G(\cdot)$ in GANs learns the mapping from the d-dimensional latent space $\mathcal{Z} \subseteq \mathbb{R}^d$ to a higher dimensional image space $\mathcal{I} \subseteq \mathbb{R}^{H \times W \times C}$, as $\mathbf{I} = G(\mathbf{z})$. Here, $\mathbf{z} \in \mathcal{Z}$ and $\mathbf{I} \in \mathcal{I}$ denote the input latent code and the output image respectively. State-of-the-art GAN models [23, 16, 4, 17, 18] typically adopt convolutional neural networks as the generator architecture. Consisting of multiple layers, $G(\cdot)$ projects the starting latent space to the final image space step by step. Each step learns a transformation from one space to another. We focus on examining the first step, which directly acts on the latent space we would like to explore. In particular, it can be formulated as an affine transformation, like most GANs [23, 16, 4, 17, 18] have done, as

$$G_1(\mathbf{z}) \triangleq \mathbf{y} = \mathbf{A}\mathbf{z} + \mathbf{b}, \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^m$ is the m-dimensional projected code. $\mathbf{A} \in \mathbb{R}^{m \times d}$ and $\mathbf{b} \in \mathbb{R}^m$ denote the weight and bias used in the first transformation step $G_1(\cdot)$ respectively.

Manipulation Model in GAN Latent Space. The latent space of GANs has recently been shown to encode rich semantic knowledge [7, 15, 24, 27]. These semantics can be further applied to image editing with the vector arithmetic property [23]. More concretely, prior work [7, 24, 27, 26, 10] proposed to use a certain direction $\mathbf{n} \in \mathbb{R}^d$ in the latent space to represent a semantic concept. After identifying a semantically meaningful direction, the manipulation can be achieved via the following model

$$\texttt{edit}(G(\mathbf{z})) = G(\mathbf{z}') = G(\mathbf{z} + \alpha\mathbf{n}), \tag{2}$$

which is commonly used in the existing approaches [7, 24, 27, 26, 10]. Here, $\texttt{edit}(\cdot)$ denotes the editing operation. In other words, we can alter the target semantic by linearly moving the latent code $\mathbf{z}$ along the identified direction $\mathbf{n}$. α indicates the manipulation intensity.

2.2. Unsupervised Semantic Factorization

Our goal is to reveal the explanatory factors (*i.e.*, the direction $\mathbf{n}$ in Eq. (2)) from the latent space of GANs. As discussed above, the generator in GANs can be viewed as a multi-step function that gradually projects the latent space to the image space. Let us take a closer look into the first projection step, as suggested in Eq. (1). Under its formulation of affine transformation, the manipulation model in Eq. (2) can be simplified as

$$\begin{aligned}\mathbf{y}' \triangleq G_1(\mathbf{z}') &= G_1(\mathbf{z} + \alpha\mathbf{n}) \\ &= \mathbf{A}\mathbf{z} + \mathbf{b} + \alpha\mathbf{A}\mathbf{n} = \mathbf{y} + \alpha\mathbf{A}\mathbf{n}.\end{aligned} \tag{3}$$

We observe from Eq. (3) that the manipulation process is instance independent. In other words, given any latent code $\mathbf{z}$ together with a certain latent direction $\mathbf{n}$, the editing can be always achieved by adding the term $\alpha\mathbf{A}\mathbf{n}$ onto the projected code after the first step. From this perspective, the weight parameter $\mathbf{A}$ should contain the essential knowledge of the image variation. Thus we aim to discover important latent directions by decomposing $\mathbf{A}$.

To this end, we propose an *unsupervised* approach, which is *independent of data sampling and model training*, for semantic factorization by solving the following optimization problem

$$\mathbf{n}^* = \underset{\{\mathbf{n} \in \mathbb{R}^d:\ \mathbf{n}^T\mathbf{n}=1\}}{\arg\max} ||\mathbf{A}\mathbf{n}||_2^2, \tag{4}$$

where $||\cdot||_2$ denotes the l_2 norm. This problem aims at finding the directions that can cause large variations after the projection of $\mathbf{A}$. Intuitively, if some direction $\mathbf{n}'$ is projected to a zero-norm vector, *i.e.*, $\mathbf{A}\mathbf{n}' = \mathbf{0}$, the editing operation in Eq. (3) turns into $\mathbf{y}' = \mathbf{y}$, which will keep the output synthesis unchanged, let clone alter the semantics occurring in it.

When the case comes to finding k most important directions $\{\mathbf{n}_1, \mathbf{n}_2, \cdots, \mathbf{n}_k\}$, we expand Eq. (4) into

$$\mathbf{N}^* = \underset{\{\mathbf{N} \in \mathbb{R}^{d \times k}:\ \mathbf{n}_i^T\mathbf{n}_i = 1\ \forall i = 1, \cdots, k\}}{\arg\max} \sum_{i=1}^{k} ||\mathbf{A}\mathbf{n}_i||_2^2, \tag{5}$$

where $\mathbf{N} = [\mathbf{n}_1, \mathbf{n}_2, \cdots, \mathbf{n}_k]$ correspond to the top-k semantics. To solve this problem, we introduce the Lagrange multipliers $\{\lambda_i\}_{i=1}^k$ into Eq. (5) as

$$\begin{aligned}\mathbf{N}^* &= \underset{\mathbf{N} \in \mathbb{R}^{d \times k}}{\arg\max} \sum_{i=1}^{k} ||\mathbf{A}\mathbf{n}_i||_2^2 - \sum_{i=1}^{k} \lambda_i(\mathbf{n}_i^T\mathbf{n}_i - 1) \\ &= \underset{\mathbf{N} \in \mathbb{R}^{d \times k}}{\arg\max} \sum_{i=1}^{k} (\mathbf{n}_i^T\mathbf{A}^T\mathbf{A}\mathbf{n}_i - \lambda_i\mathbf{n}_i^T\mathbf{n}_i + \lambda_i).\end{aligned} \tag{6}$$

By taking the partial derivative on each $\mathbf{n}_i$, we have

$$2\mathbf{A}^T\mathbf{A}\mathbf{n}_i - 2\lambda_i\mathbf{n}_i = 0. \tag{7}$$

All possible solutions to Eq. (7) should be the eigenvectors of the matrix $\mathbf{A}^T\mathbf{A}$. To get the maximum objective value and make $\{\mathbf{n}_i\}_{i=1}^k$ distinguishable from each other, we choose columns of $\mathbf{N}$ as the eigenvectors of $\mathbf{A}^T\mathbf{A}$ associated with the k largest eigenvalues.

2.3. Implementation on GAN Models

In Sec. 2.2, we propose a closed-form algorithm, termed as SeFa, to factorize the latent semantics learned by GANs. Our algorithm can be performed *in a completely unsupervised fashion by efficiently investigating the weights of a pre-trained GAN generator*. In this part, we introduce how our approach is applied to the state-of-the-art GAN models, such as PGGAN [16], StyleGAN [17], and BigGAN [4].

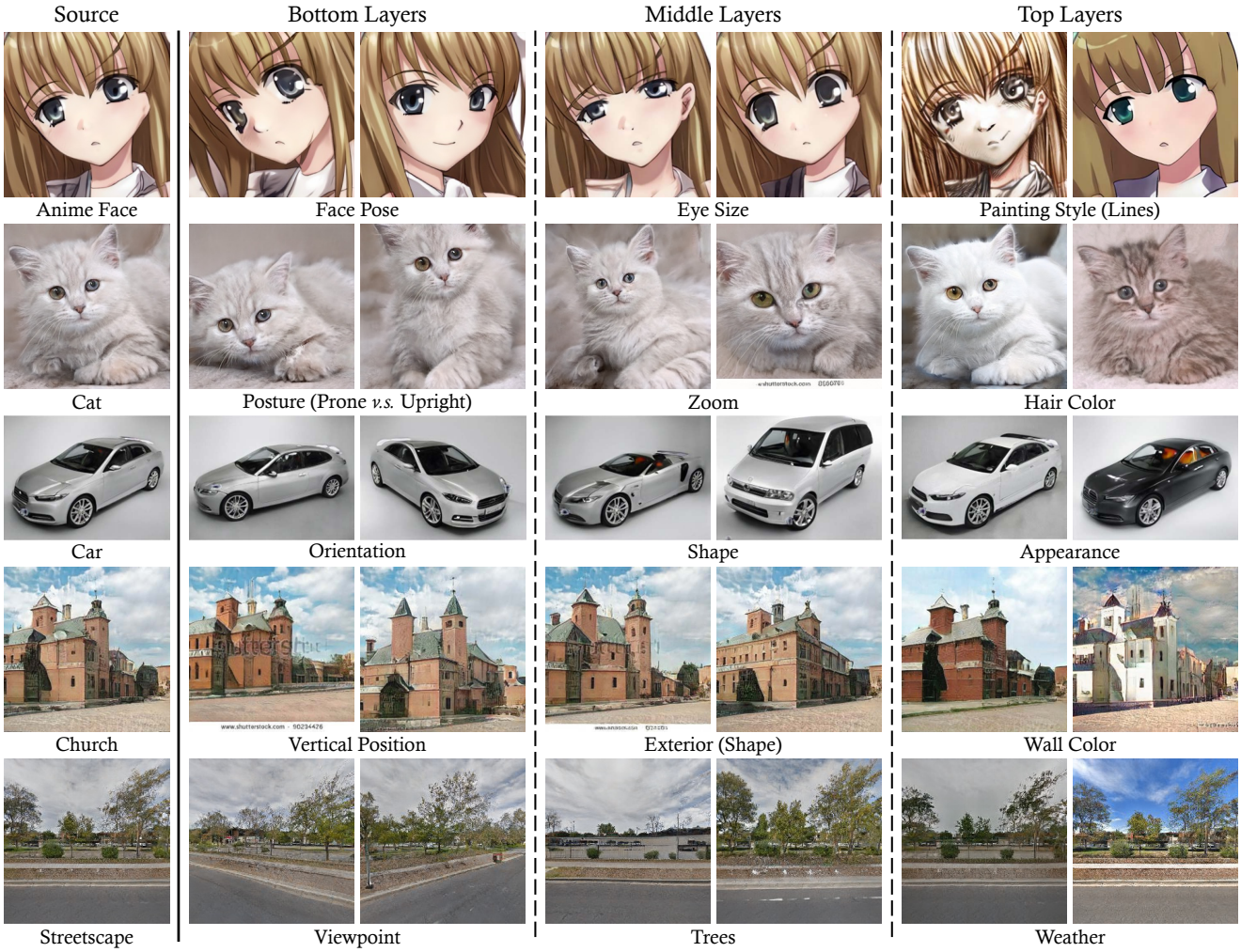

Figure 2. **Hierarchical interpretable directions** discovered in the style-based generators, *i.e.*, StyleGAN [17] and StyleGAN2 [18]. Among them, the streetscapes model is trained with StyleGAN2, while the others are using StyleGAN.

PGGAN. PGGAN [16] is a representative of the conventional generator, where the input latent code is firstly mapped into a spatial feature map and then projected to an image with a sequence of convolution layers. For this kind of generator structure, SeFa studies the transformation from the latent code to the feature map.

StyleGAN. StyleGAN [17] proposes the style-based generator, which feeds the latent code into each convolution layer. In particular, for each layer, the latent code is transformed to a style code, which is used to alter the channel-wise mean and variance of the feature map through Adaptive Instance Normalization (AdaIN) [14]. For this GAN type, we investigate the transformation from the latent code to the style code. Note that our algorithm is flexible such that it supports interpreting all or any subset of layers. For this purpose, we concatenate the weight parameters (*i.e.*, $\mathbf{A}$ in Eq. (1)) from all target layers along the first axis, forming a larger transformation matrix.

BigGAN. BigGAN [4] is a large-scale GAN model primarily designed for conditional generation. The latent code is both mapped to the initial feature map and fed into each convolution layer. Hence, the analysis on BigGAN can be viewed as a combination of the above two types of GANs.

3. Experiments

We evaluate our closed-form algorithm on a wide range of models to discover interpretable directions. We also compare SeFa with existing supervised and unsupervised alternatives to demonstrate its effectiveness.

3.1. Results on Diverse Models and Datasets

We conduct experiments on the state-of-the-art GAN models, such as StyleGAN [17], BigGAN [4], and StyleGAN2 [18]. They are trained on different datasets, including human faces (FF-HQ [17]), anime faces [1], scenes and

Figure 3. **Diverse interpretable directions** found in the BigGAN [4], which is conditionally trained on ImageNet [6]. These semantics are further used to manipulate images from different categories.

objects (LSUN [28]), streetscapes [20], and ImageNet [6].[2]

Interactive Editing by Tuning Interpretable Directions. Our algorithm is performed in a completely unsupervised manner, hence we do not rely on any auxiliary predictors. After discovering the important directions by decomposing the model weights, we can interact with the GAN model for collaborative content editing. Thus we develop an interface to facilitate human-model interaction, as shown in Fig. 4. Meanwhile, with the help of this interface, users can easily annotate the identified semantics.

Results on StyleGAN. As described in Sec. 2.3, our algorithm can interpret a subset of layers in the style-based generators [17, 18]. We evaluate SeFa on the models trained on a wide range of datasets, including anime faces, objects, scenes, and streetscapes. In particular, we interpret a target model at the levels of bottom layers, middle layers, and top layers respectively. Fig. 2 shows the versatile semantic directions found in these models. We noticeably find that they are organized as a hierarchy, which is consistent with the observations from prior work [17, 27]. Taking cars as an example, bottom layers tend to control the rotation, middle layers determine the shape, while top layers correspond to the color. We further conduct a user study to see how the variation factors found by SeFa align with human perception. Here, questions are asked to 10 annotators. As suggested in Tab. 1, SeFa can indeed find human-understandable concepts, even from some particular layers in GAN models.

[2]We have collected a model zoo consisting of various types of GANs. SeFa can be easily applied to interpreting these models benefiting from its efficient implementation (*i.e.*, less than 1 second for one model). Please refer to the demo video for diverse and continuous manipulation results.

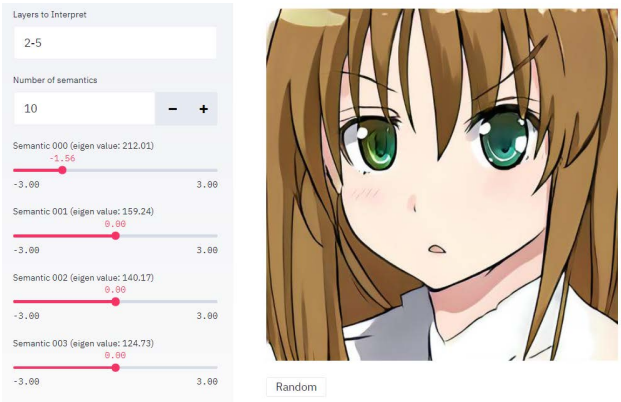

Figure 4. **Interface** for interactive editing.

Table 1. User study. We randomly generate $2K$ images for each dataset, and use the Top-50 eigen directions from each level of layers to manipulate these images. Numbers in brackets indicate the index of the layers to interpret. Users are asked how many directions result in *obvious* content change (numerator) and how many directions are semantically meaningful (denominator).

Dataset	Bottom (0-1)	Middle (2-5)	Top (6-)
Anime Face [1]	12/12	26/26	38/50
LSUN Cat [28]	14/15	21/28	47/50
LSUN Car [28]	10/10	16/22	22/34
LSUN Church [28]	15/15	18/26	48/50
Streetscape [20]	9/9	12/18	15/36

Results on BigGAN. We also interpret the large-scale BigGAN [4] model that is conditionally trained on ImageNet [6]. BigGAN extends the latent code with a category-derived embedding vector to achieve conditional synthesis. Here, we only focus on the latent code part for semantic

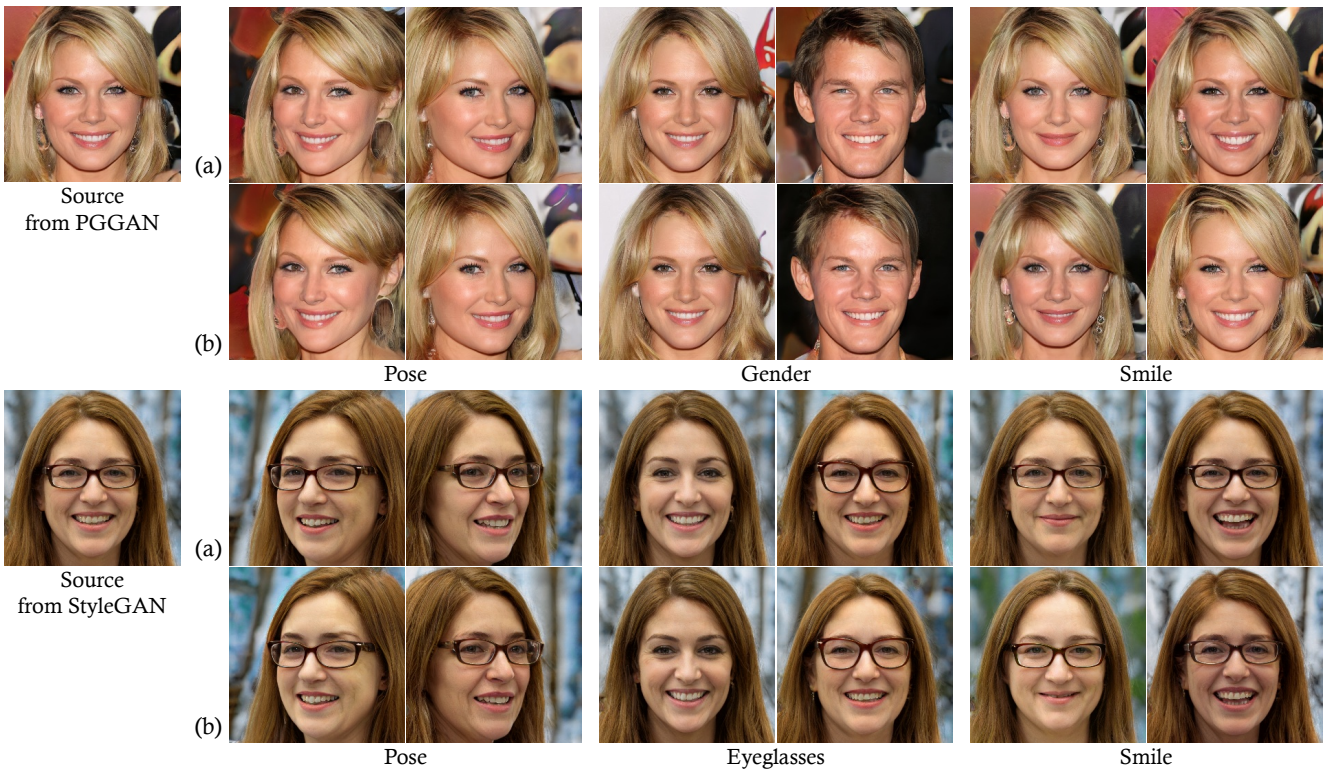

Figure 5. Qualitative comparison of the latent semantics found by (a) the supervised method, InterFaceGAN [24] and (b) our *closed-form* solution, SeFa, where SeFa achieves similar performance to InterFaceGAN. PGGAN trained on CelebA-HQ [16] and StyleGAN trained on FF-HQ [17] are used as the target models to interpret.

Table 2. **Re-scoring analysis** of the semantics identified by InterFaceGAN [24] and SeFa from the PGGAN model trained on CelebA-HQ dataset [16]. Each row evaluates how the semantic scores change after moving the latent code along a certain direction.

(a) InterFaceGAN [24], which is supervised.

	Pose	Gender	Age	Glasses	Smile
Pose	0.53	-0.06	-0.09	-0.01	0.05
Gender	-0.02	0.59	0.20	0.08	-0.07
Age	-0.03	0.35	0.50	0.08	-0.03
Glasses	-0.01	0.37	0.19	0.24	0.00
Smile	-0.01	-0.07	0.03	-0.01	0.60

(b) SeFa, which is unsupervised.

	Pose	Gender	Age	Glasses	Smile
Pose	0.51	-0.11	-0.07	0.02	0.06
Gender	0.02	0.55	0.46	0.09	-0.13
Age	-0.07	-0.25	0.34	0.10	0.10
Glasses	0.02	0.55	0.46	0.09	-0.13
Smile	0.03	-0.03	0.15	-0.16	0.42

discovery. Fig. 3 provides some examples. We can tell that the semantics found by our algorithm can be applied to manipulating images from different categories. This verifies the generalization ability of SeFa.

3.2. Comparison with Supervised Approach

We compare our closed-form algorithm with the state-of-the-art supervised method, InterFaceGAN [24]. We conduct experiments on face synthesis models due to the well definition of facial attributes. In particular, we make comparison between SeFa and InterFaceGAN on both the conventional generator (*i.e.*, PGGAN [16]) and the style-based generator (*i.e.*, StyleGAN [17]).

Qualitative Results. Fig. 5 visualizes some manipulation results by using the identified semantics. We can tell that SeFa achieves similar performance as InterFaceGAN from the perspective of editing pose, gender, eyeglasses, and expression (smile), suggesting its effectiveness. More importantly, InterFaceGAN requires sampling numerous data and pre-training attribute predictors. By contrast, SeFa is completely independent of data sampling and model training, which is more efficient and generalizable.

Re-scoring Analysis. For quantitative analysis, we train an attribute predictor on CelebA dataset [19] with ResNet-50 structure [11], following [24]. With this predictor, we are able to perform re-scoring analysis to quantitatively evaluate whether the identified directions can properly represent the corresponding attributes. In particular, we randomly sample $2K$ images and manipulate them along a certain discovered direction. We then use the prepared

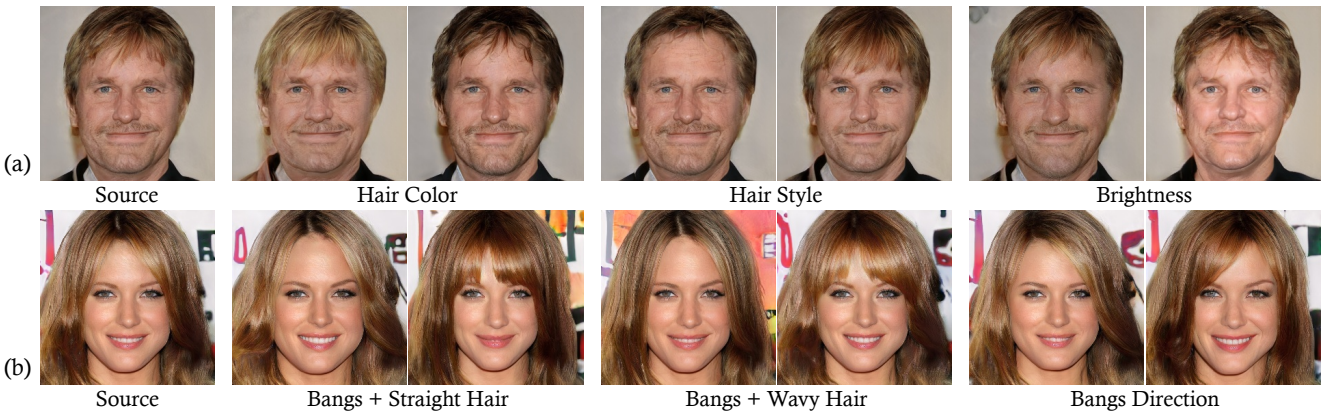

Figure 6. (a) Diverse semantics, which can *not* be identified by InterFaceGAN [24] due to the lack of semantic predictors. (b) Diverse hair styles, which can *not* be described as a binary attribute. The PGGAN model trained on CelebA-HQ dataset [16] is used.

Figure 7. Qualitative comparison between (a) GANSpace [10] and (b) SeFa. The StyleGAN model trained on FF-HQ dataset [17] is used.

predictor to check how the semantic score varies in such manipulation process. Tab. 2 shows the results where we have three observations. (i) SeFa can adequately control some attribute, such as pose and gender, similar to InterFaceGAN. (ii) When altering one semantic, InterFaceGAN shows stronger robustness to other attributes, benefiting from its supervised training manner. For example, the age and eyeglasses corresponding to the same latent direction identified by SeFa. That is because the training data is somewhat biased (*i.e.*, older people are more likely to wear eyeglasses), as pointed out by [24]. By contrast, involving labels as the supervision can help learn a more accurate direction to some extent. (iii) SeFa fails to discover the direction corresponding to eyeglasses. The reason is that the presence of eyeglasses is not a large variation and hence does not meet the optimization objective in Eq. (4).

Diversity Comparison. Supervised approach highly depends on the available attribute predictors. By contrast, our method is more general and can find more diverse semantics in the latent space. As shown in Fig. 6 (a), we successfully identify the directions corresponding to hair color, hair style, and brightness. This surpasses InterFaceGAN since predictors for these attributes are not easy to acquire in practice. Also, supervised methods are usually limited by the training objective. For example, InterFaceGAN is proposed to handle binary attributes [24]. In comparison, our method can identify more complex attributes, like the different hair styles shown in Fig. 6 (b).

Table 3. Quantitative comparison with GANSpace [10].

	FID	Re-scoring	User Study
GANSpace [10]	7.43	0.33	41%
SeFa (Ours)	**7.36**	**0.38**	**59%**

3.3. Comparison with Unsupervised Baselines

We compare our method with some unsupervised alternatives, including the sampling-based method [10] and the learning-based method [5]. The major difference is that SeFa works as a closed-form solution, which is independent of any kind of data sampling or model training.

Comparison with Sampling-based Baseline. GANSpace [10] proposes to perform PCA on a collection of sampled data to find principal directions in the latent space. In this part, we compare SeFa with GANSpace on the StyleGAN model trained on FF-HQ dataset [17]. Fig. 7 visualizes some qualitative comparison results, where the semantics

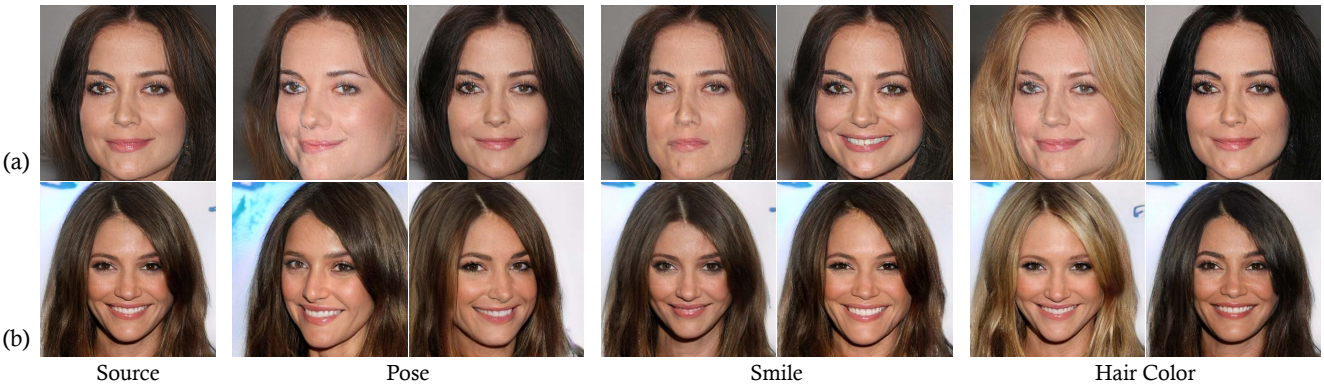

Figure 8. Qualitative comparison between (a) Info-PGGAN [21, 5] and (b) SeFa. The result of the Info-PGGAN model is extracted directly from [21], and the official PGGAN model trained on CelebA-HQ dataset [16] is used for SeFa.

found by SeFa lead to a more precise control. For example, when changing face pose, SeFa better preserves the identity and skin color. We also quantitatively compare these two approaches with FID [12], re-scoring analysis, and user study. Here, users are asked which approach changes a particular attribute more adequately on $2K$ manipulations. Results are shown in Tab. 3. SeFa and GANSpace show close FID score since this is mostly determined by the generator itself as well as the manipulation model in Eq. (2), which are shared by these two methods. But SeFa outperforms GANSpace on attribute re-scoring and user study.

Comparison with Learning-based Baseline. InfoGAN [5] proposed to explicitly learn a factorized representation by introducing a regularizer to maximize the mutual information between the output image and the input latent code. We compare our method with the Info-PGGAN model [21], which trains the native PGGAN [16] with the information regularizer [5]. Fig. 8 shows the comparison results. We can tell that the semantics identified by SeFa through a closed-form factorization on pre-trained weights are more accurate than those learned from Info-PGGAN. Taking pose manipulation as an example, the hair color varies when using Info-PGGAN for editing. By contrast, SeFa achieves a more precise control.[3]

3.4. Real Image Editing

In this part, we verify that the latent semantics revealed by SeFa is applicable for real image editing. Since the generator lacks the inference ability to take a real image as the input, we involve GAN inversion [9, 29] approaches into our algorithm. More concretely, given a target image to edit, we first project it back to the latent space, and then use the variation factor found by SeFa to modulate the inverted code. Fig. 9 shows some examples, where SeFa shows satisfying performance. For example, we manage to remove eyeglasses from the input images and also alter the face pose. It suggests that SeFa is capable of discovering the directions of the latent space which are generalizable for real image editing.

[3]We use different samples for Info-PGGAN [21] and SeFa because Info-PGGAN requires model retraining, leading to a different model from the one that is officially released by [16]. As a result, it is hard to produce the same face with these two different models.

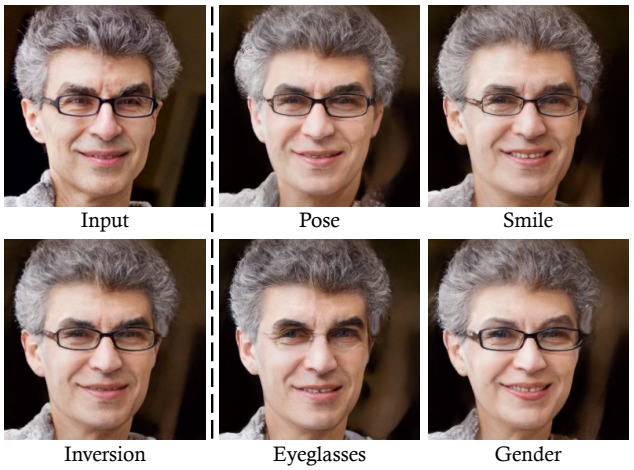

Figure 9. **Real image editing** with respect to various facial attributes. All semantics are found with the proposed SeFa. GAN inversion [29] is used to project the target real image back to the latent space of StyleGAN [17].

4. Conclusion

In this work we propose a closed-form solution to factorizing the latent semantics learned by GANs. Extensive experiments demonstrate the great power of our algorithm in identifying versatile semantics from different types of GAN models in an unsupervised manner.

Acknowledgements: This work is supported in part by the Early Career Scheme (ECS) through the Research Grants Council (RGC) of Hong Kong under Grant No.24206219, CUHK FoE RSFS Grant, SenseTime Collaborative Grant and Centre for Perceptual and Interactive Intelligence (CPII) Ltd under the Innovation and Technology Fund.

References

[1] Anonymous, Danbooru community, and Gwern Branwen. Danbooru2019: A large-scale crowdsourced and tagged anime illustration dataset. https://www.gwern.net/Danbooru2019, 2020. 4, 5

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Int. Conf. Mach. Learn.*, 2017. 2

[3] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B. Tenenbaum, William T. Freeman, and Antonio Torralba. Gan dissection: Visualizing and understanding generative adversarial networks. In *Int. Conf. Learn. Represent.*, 2019. 2

[4] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *Int. Conf. Learn. Represent.*, 2019. 1, 2, 3, 4, 5

[5] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Adv. Neural Inform. Process. Syst.*, 2016. 2, 7, 8

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2009. 5

[7] Lore Goetschalckx, Alex Andonian, Aude Oliva, and Phillip Isola. Ganalyze: Toward visual definitions of cognitive image properties. In *Int. Conf. Comput. Vis.*, 2019. 2, 3

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Adv. Neural Inform. Process. Syst.*, 2014. 1, 2

[9] Jinjin Gu, Yujun Shen, and Bolei Zhou. Image processing using multi-code gan prior. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2, 8

[10] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable gan controls. In *Adv. Neural Inform. Process. Syst.*, 2020. 2, 3, 7

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 6

[12] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Adv. Neural Inform. Process. Syst.*, 2017. 8

[13] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *Int. Conf. Learn. Represent.*, 2017. 2

[14] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Int. Conf. Comput. Vis.*, 2017. 2, 4

[15] Ali Jahanian, Lucy Chai, and Phillip Isola. On the "steerability" of generative adversarial networks. In *Int. Conf. Learn. Represent.*, 2020. 2, 3

[16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *Int. Conf. Learn. Represent.*, 2018. 1, 2, 3, 4, 6, 7, 8

[17] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 1, 2, 3, 4, 5, 6, 7, 8

[18] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1, 2, 3, 4, 5

[19] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Int. Conf. Comput. Vis.*, 2015. 6

[20] Nikhil Naik, Jade Philipoom, Ramesh Raskar, and César Hidalgo. Streetscore-predicting the perceived safety of one million streetscapes. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2014. 5

[21] Jonasz Pamuła. Progressive training of gans with mutual information penalty. https://github.com/jonasz/progressive_infogan, 2018. 8

[22] Antoine Plumerault, Hervé Le Borgne, and Céline Hudelot. Controlling generative models with continuous factors of variations. In *Int. Conf. Learn. Represent.*, 2020. 2

[23] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Int. Conf. Learn. Represent.*, 2016. 2, 3

[24] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2, 3, 6, 7

[25] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. Interfacegan: Interpreting the disentangled face representation learned by gans. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020. 2

[26] Andrey Voynov and Artem Babenko. Unsupervised discovery of interpretable directions in the gan latent space. In *Int. Conf. Mach. Learn.*, 2020. 2, 3

[27] Ceyuan Yang, Yujun Shen, and Bolei Zhou. Semantic hierarchy emerges in deep generative representations for scene synthesis. *Int. J. Comput. Vis.*, 2020. 2, 3, 5

[28] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, 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. 5

[29] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain gan inversion for real image editing. In *Eur. Conf. Comput. Vis.*, 2020. 2, 8