Unsupervised Discovery of Interpretable Directions in the GAN Latent Space

Andrey Voynov [1] **Artem Babenko** [1 2]

Abstract

The latent spaces of GAN models often have semantically meaningful directions. Moving in these directions corresponds to human-interpretable image transformations, such as zooming or recoloring, enabling a more controllable generation process. However, the discovery of such directions is currently performed in a supervised manner, requiring human labels, pretrained models, or some form of self-supervision. These requirements severely restrict a range of directions existing approaches can discover. In this paper, we introduce an **unsupervised** method to identify interpretable directions in the latent space of a pretrained GAN model. By a simple model-agnostic procedure, we find directions corresponding to sensible semantic manipulations without any form of (self-)supervision. Furthermore, we reveal several non-trivial findings, which would be difficult to obtain by existing methods, e.g., a direction corresponding to background removal. As an immediate practical benefit of our work, we show how to exploit this finding to achieve competitive performance for weakly-supervised saliency detection. The implementation of our method is available online[1].

1. Introduction

Nowadays, generative adversarial networks (GANs) (Goodfellow et al., 2014) have become a leading paradigm of generative modeling in the computer vision domain. The state-of-the-art GANs (Brock et al., 2019; Karras et al., 2019) are currently able to produce good-looking high-resolution images often indistinguishable from real ones. The exceptional generation quality paves the road to ubiquitous usage of GANs in applications, e.g., image editing (Isola et al., 2017; Zhu et al., 2017), super-resolution (Ledig et al., 2017), video generation (Wang et al., 2018) and many others.

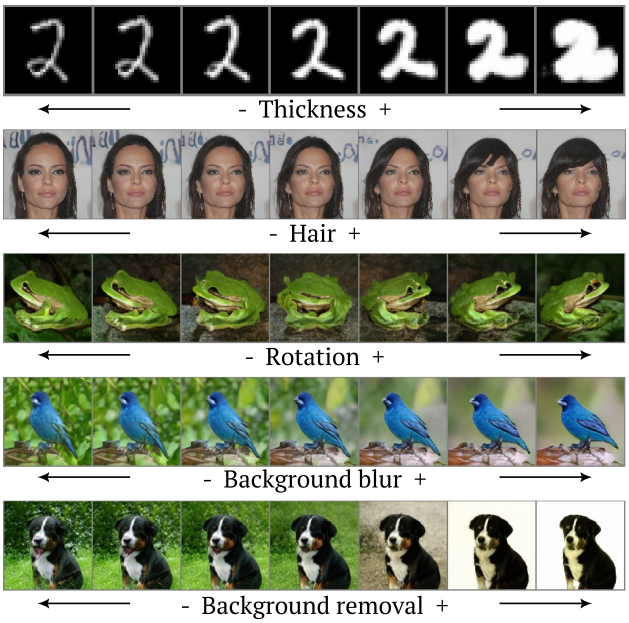

Figure 1. Examples of interpretable directions discovered by our unsupervised method for several datasets and generators.

However, in most practical applications, GAN models are typically used as black-box instruments without a complete understanding of the underlying generation process. While several recent papers (Bau et al., 2019; Voynov & Babenko, 2019; Yang et al., 2019; Karras et al., 2019; Jahanian et al., 2020; Plumerault et al., 2020) address the interpretability of GANs, this research area is still in its preliminary stage.

An active line of study on GANs interpretability investigates the structure of their latent spaces. Namely, several works (Jahanian et al., 2020; Plumerault et al., 2020; Goetschalckx et al., 2019; Shen et al., 2019) aim to identify semantically meaningful directions, i.e., corresponding to human-interpretable image transformations. At the moment, prior works have provided enough evidence that a wide range of such directions exists. Some of them induce domain-agnostic transformations, like zooming or translation (Jahanian et al., 2020; Plumerault et al., 2020), while others correspond to domain-specific transformations, e.g., adding smile or glasses on face images (Radford et al., 2015).

[1]Yandex, Russia [2]National Research University Higher School of Economics , Moscow, Russia. Correspondence to: Andrey Voynov <an.voynov@yandex.ru>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

[1]http://github.com/anvoynov/GanLatentDiscovery

While the discovery of interpretable directions has already been addressed by prior works, all these works require some form of supervision. For instance, (Shen et al., 2019; Goetschalckx et al., 2019; Karras et al., 2019) require explicit human labeling or pretrained supervised models, which can be expensive or even impossible to obtain. Recent works (Jahanian et al., 2020; Plumerault et al., 2020) develop self-supervised approaches, but they are limited only to directions, corresponding to simple transformations achievable by automatic data augmentation.

In this paper, we propose a **completely unsupervised** approach to discover the interpretable directions in the latent space of a pretrained generator. In a nutshell, the approach seeks a set of directions corresponding to diverse image transformations, i.e., it is easy to distinguish one transformation from another. Intuitively, under such formulation, the learning process aims to find the directions corresponding to the independent factors of variation in the generated images. For several generators, we observe that many of the obtained directions are human-interpretable, see Figure 1.

As another significant contribution, our approach discovers new practically important directions, which would be difficult to obtain with existing techniques. For instance, we discover the direction corresponding to the background removal, see Figure 1. In the experimental section, we exploit it to generate high-quality synthetic data for saliency detection and achieve competitive performance for this problem in a weakly-supervised scenario. We expect that exploitation of other directions can also benefit other computer vision tasks in the unsupervised and weakly-supervised niches.

As our main contributions we highlight the following:

1. We propose the first unsupervised approach for the discovery of semantically meaningful directions in the GAN latent space. The approach is model-agnostic and does not require costly generator re-training.

2. For several common generators, we managed to identify non-trivial and practically important directions. The existing methods from prior works are not able to identify them without expensive supervision.

3. We provide an example of immediate practical benefit from our work. Namely, we show how to exploit the background removal direction for weakly-supervised saliency detection.

The paper is organized as follows. Section 2 discusses the relevant ideas from prior literature. Section 3 formally describes our approach, Section 4 reports the results and Section 5 applies our finding to the weakly-supervised saliency detection. In Section 6 we ablate hyperparameters. Section 7 concludes the paper.

2. Related Work

In this section, we describe the relevant research areas and explain the scientific context of our study.

Generative adversarial networks (Goodfellow et al., 2014) currently dominate the generative modeling field. In essence, GANs consist of two networks – a generator and a discriminator, which are trained jointly in an adversarial manner. The role of the generator is to map samples from the latent space distributed according to a standard Gaussian distribution to the image space. The discriminator aims to distinguish the generated images from the real ones. More complete understanding of the latent space structure is an important research problem as it would make the generation process more controllable.

Interpretable directions in the latent space. Since the appearance of earlier GAN models, it is known that the GAN latent space often possesses semantically meaningful vector space arithmetic, e.g., there are directions corresponding to adding smiles or glasses for face image data (Radford et al., 2015). Since exploitation of these directions would make image editing more straightforward, the discovery of such directions currently receives much research attention. A line of recent works (Goetschalckx et al., 2019; Shen et al., 2019; Karras et al., 2019) employs explicit human-provided supervision to identify interpretable directions in the latent space. For instance, (Shen et al., 2019; Karras et al., 2019) use the classifiers pretrained on the CelebA dataset (Liu et al., 2015) to predict certain face attributes. These classifiers are then used to produce pseudo-labels for the generated images and their latent codes. Based on these pseudo-labels, the separating hyperplane is constructed in the latent space, and a normal to this hyperplane becomes a direction that captures the corresponding attribute. Another work (Plumerault et al., 2020) solves the optimization problem in the latent space that maximizes the score of the pretrained model, predicting image memorability. Thus, the result of the optimization is a direction corresponding to the increase of memorability. The crucial weakness of supervised approaches above is their need for human labels or pretrained models, which can be expensive to obtain. Two recent works (Jahanian et al., 2020; Plumerault et al., 2020) employ self-supervised approaches and seek the vectors in the latent space that correspond to simple image augmentations such as zooming or translation. While these approaches do not require supervision, they can be used to find only the directions capturing simple transformations that can be obtained automatically.

All these approaches are able to discover only directions, which researchers expect to identify. In contrast, our unsupervised approach often identifies surprising directions, corresponding to non-trivial image manipulations.

Disentanglement learning. An alternative line of research

on the model interpretability aims to train generators with disentangled latent spaces (Chen et al., 2016; Higgins et al., 2017; Liu et al., 2019; Lee et al., 2020; Ramesh et al., 2018). In particular, the seminal InfoGAN model (Chen et al., 2016) enforces the generated images to preserve information about the latent code coordinates by maximizing the corresponding mutual information. Another notable work proposes the β-VAE (Higgins et al., 2017) model, which puts more emphasis on the KL-term in the standard VAE's ELBO objective. This objective modification requires the latent codes to be more "efficient", which is shown to result in disentangled representations.

While these models do achieve disentanglement of their latent spaces, they are often inferior in terms of generation quality and diversity. Several recent papers address these issues by improving the original architectures and training protocols. For instance, (Liu et al., 2019) forces the code vector c to be one-hot, simplifying the task for a GAN discriminators' head to predict the code. The authors of (Lee et al., 2020) combine VAE and GAN to achieve a disentanglement images representation by the VAE and then pass the discovered code to the GAN model. At the moment, it is unclear if disentangled generative models can be competitive to the state-of-the-art generators, e.g. BigGAN. In contrast, our method does not affect the pretrained generator distribution.

Jacobian decomposition. Probably, the closest to ours is a recent work (Ramesh et al., 2018) that also investigates the latent space of a pretrained GAN model. They note that the left eigenvectors of the generators' Jacobian matrix can serve as the most disentangled directions. The authors also propose an iterative algorithm that constructs an "interpretable curve" starting from a latent point z_0 and moving it in a direction of the Jacobians' k-th left eigenvector at each point. Once the latent vector moves along that curve, the generated image appears to be transformed by a human-meaningful transformation. Nevertheless, while the constructed curves often capture interpretable transformations, their effects are typically entangled (i.e. lighting and geometrical transformations appear simultaneously). This method also requires an expensive (in terms of both memory and runtime) iterative process computing the Jacobian matrix on each step of the curve construction and has to be applied for each latent code independently. On the contrary, we propose a lightweight approach that identifies a set of the directions at once. The method from (Ramesh et al., 2018) is also limited with the maximal number of discovered directions equal to the latent space dimensionality, while our approach can be applied for a higher number of directions.

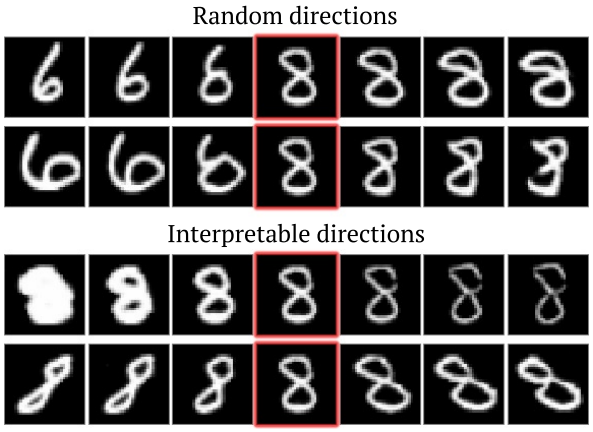

Figure 2. Image transformations obtained by moving in random (top) and interpretable (bottom) directions in the latent space.

3. Method

3.1. Motivation

Before a formal description of our method, we explain its underlying motivation by a simple example on Figure 2. Figure 2 (top) shows the transformations of an original image (in a red frame) obtained by moving in two random directions of the latent space for the Spectral Norm GAN model (Miyato et al., 2018) trained on the MNIST dataset (LeCun, 1989). As one can see, moving in a random direction typically affects several factors of variations at once, and different directions "interfere" with each other. This makes it difficult to interpret these directions or to use them for semantic manipulations in image editing.

The observation above provides the main intuition behind our method. Namely, we aim to learn a set of directions inducing "orthogonal" image transformations that are easy to distinguish from each other. We achieve this via jointly learning a set of directions and a model to distinguish the corresponding image transformations. The high quality of this model implies that directions do not interfere; hence, hopefully, affect only a single factor of variation and are easy-to-interpret.

3.2. Learning

The learning protocol is schematically presented on Figure 3. Our goal is to discover the interpretable directions in the latent space of a pretrained GAN generator $G : z \longrightarrow I$, which maps samples from the latent space $z \in \mathbb{R}^d$ to the image space. G is a non-trainable component of our method, and its parameters do not change during learning. Two trainable components of our method are:

1. A matrix $A \in \mathbb{R}^{d \times K}$, where d equals to the dimensionality of the latent space of G. A number of columns K determines the number of directions our method will

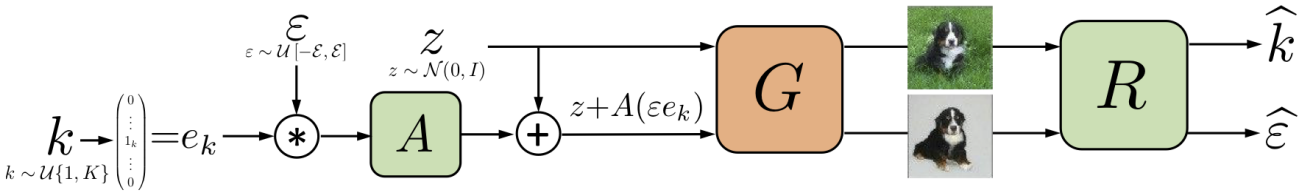

Figure 3. Scheme of our learning protocol, which discovers interpretable directions in the latent space of a pretrained generator G. A training sample in our protocol consists of two latent codes, where one is a shifted version of another. Possible shift directions form a matrix A. Two codes are passed through G and an obtained pair of images go to a reconstructor R that aims to reconstruct a direction index k and a signed shift magnitude ε.

discover. It is a hyperparameter of our method, and we discuss its choice in the next section. In essence, the columns of A correspond to the directions we aim to identify.

2. A *reconstructor* R, which obtains an image pair of the form $(G(z),\ G(z + A(\varepsilon e_k)))$, where the first image is generated from a latent code $z \sim \mathcal{N}(0, I)$, while the second one is generated from a *shifted* code $z+A(\varepsilon e_k)$. Here e_k denotes an axis-aligned unit vector $(0, \ldots, 1_k, \ldots, 0)$ and ε is a scalar. In other words, the second image is a transformation of the first one, corresponding to moving by ε in a direction, defined by the k-th column of A in the latent space. The reconstructor's goal is to reproduce the shift in the latent space that induces a given image transformation. In more details, R produces two outputs $R(I_1, I_2) = \left(\widehat{k}, \widehat{\varepsilon}\right)$, where $\widehat{k}$ is a prediction of a direction index $k \in \{1, \ldots, K\}$, and $\widehat{\varepsilon}$ is a prediction of a shift magnitude ε. More formally, the reconstructor performs a mapping $R : (I_1,\ I_2) \longrightarrow (\{1, \ldots, K\},\ \mathbb{R})$.

Optimization objective. Learning is performed via minimizing the following loss function:

$$\min_{A,R} \mathop{\mathbb{E}}_{z,k,\varepsilon} L(A, R) = \min_{A,R} \mathop{\mathbb{E}}_{z,k,\varepsilon} \left[L_{cl}(k, \widehat{k}) + \lambda L_r(\varepsilon, \widehat{\varepsilon}) \right] \tag{1}$$

For the classification term $L_{cl}(\cdot, \cdot)$ we use the cross-entropy function, and for the regression term $L_r(\cdot, \cdot)$ we use the mean absolute error. In all our experiments we use a weight coefficient λ=0.25.

As A and R are optimized jointly, the minimization process seeks to obtain such columns of A that the corresponding image transformations are easier to distinguish from each other, to make the classification problem for reconstructor simpler. In the experimental section below, we demonstrate that these "disentangled" directions often appear to be human-interpretable.

The role of the regression term L_r is to force shifts along discovered directions to have the continuous effect, thereby preventing "abrupt" transformations, e.g., mapping all the images to some fixed image. See Figure 4 for a latent direction example that maps all the images to a fixed one.

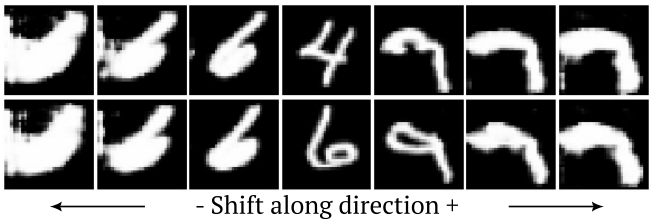

Figure 4. Direction of collapsing variation. The regression term in our objective prevents discovery of such directions.

3.3. Practical details

Here we describe the practical details of the pipeline and explain our design choices.

Reconstructor architecture. For reconstructor models R we use the LeNet backbone (LeCun et al., 1998) for the MNIST and AnimeFaces and the ResNet-18 model (He et al., 2016) for Imagenet and CelebA-HQ. In all experiments, a number of input channels is set to six (two for MNIST), since we concatenate the input image pair along channels dimension. We also use two separate "heads", predicting a direction index and a shift magnitude, respectively.

Distribution of training samples. The latent codes z are always sampled from the normal distribution $\mathcal{N}(0, I)$ as in the original G. The direction index k is sampled from a uniform distribution $\mathcal{U}\{1, K\}$. A shift magnitude ε is sampled from the uniform distribution $\mathcal{U}[-6, 6]$. We also found a minor advantage in forcing ε to be separated from 0 as too small shifts almost do not affect the generated image. Thus, in practice after sampling we take ε equal to $\text{sign}(\varepsilon) \cdot \max(|\varepsilon|, 0.5)$. We did not observe any difference from using other distributions for ε.

Choice of K. The number of directions K is set to be equal to the latent space dimensionality for Spectral Norm GAN

(Miyato et al., 2018) with Anime Faces dataset and BigGAN (Brock et al., 2019) (which are 128 and 120). For Spectral Norm GAN with MNIST dataset, we use K=64, since its latent space is 128-dimensional, and it is too difficult for our model to obtain so many different interpretable directions for simple digit images. Following the same considerations for ProgGAN (Karras et al., 2018) we use K=200 since its latent space is 512-dimensional.

Choice of A. We experimented with three options for A:

- A is a general linear operator;
- A is a linear operator with all matrix columns having a unit length;
- A is a linear operator with orthonormal matrix columns.

The first option appeared to be impractical as during the optimization, we frequently observed the columns of A to have very high l_2-norms. The reason is that for a constant latent shift c of a high norm, most of the generated samples $G(z+c)$ with $z \sim \mathcal{N}(0, I)$ appears to be almost the same for all z. Thus the classification term in the loss (1) pushes A to have columns of a high norm to simplify classification.

In all experiments, we use either A with columns of length one (the second option) either with orthonormal columns (the third option). To guarantee that A has unit-norm columns, we divide each column by its length. For the orthonormal case, we parametrize A with a skew-symmetric matrix S (that is $S^T = -S$) and define A as the first K columns of the exponent of S, see details in supplementary.

In experiments, we observe that both two options discover similar sets of interpretable directions. In general, using matrix A with unit-norm columns is more expressive and is able to find more directions. However, on some datasets, the option with orthonormal columns discovered more interesting directions. In the experiments, we use orthonormal A for AnimeFaces and BigGAN and unit length columns for MNIST and ProgGAN.

4. Experiments

Here we evaluate our approach on several datasets in terms of both quantitative and qualitative results. In all experiments, we do not exploit any form of external supervision and operate in a completely unsupervised manner.

Datasets and generator models. We experiment with four common datasets and generator architectures:

1. MNIST (LeCun, 1989), containing 32×32 images. Here we use Spectral Norm GAN (Miyato et al., 2018) with ResNet-like generator of three residual blocks.
2. AnimeFaces dataset (Jin et al., 2017), containing 64×64 images. For AnimeFaces we use Spectral Norm GAN (Miyato et al., 2018) with ResNet-like generator of four residual blocks.
3. CelebA-HQ dataset (Liu et al., 2015), containing 1024×1024 images. We use a pretrained ProgGAN generator (Karras et al., 2018), available online[2].
4. BigGAN generator (Brock et al., 2019) trained on ILSVRC dataset (Deng et al., 2009), containing 128×128 images. We use the BigGAN, available online[3].

Optimization. In all the experiments, we use the Adam optimizer to learn both the matrix A and the reconstructor R. We always train the models with a constant learning rate 0.0001. We perform $2 \cdot 10^5$ gradient steps for ProgGAN and 10^5 steps for others as the first has a significantly higher latent space dimension. We use a batch size of 128 for Spectral Norm GAN on the MNIST, and Anime Faces datasets, a batch size of 32 for BigGAN, and a batch size of 10 for ProgGAN. All the experiments were performed on the NVIDIA Tesla v100 card.

Evaluation metrics. Since it is challenging to measure interpretability and disentanglement directly, we propose two evaluation measures described below.

1. *Reconstructor Classification Accuracy (RCA)*. As described in Section 3, the reconstructor R aims to predict what direction in the latent space produces a given image transformation. In essence, the reconstructor's classification "head" solves a multi-class classification problem. Therefore, high RCA values imply that directions are easy to distinguish from each other, i.e., corresponding image transformations do not "interfere" and influence different factors of variations. While it does not mean interpretability directly, in practice, transformations affecting a few factors of variation are easier to interpret. RCA allows us to compare the directions obtained with our method with random directions or with directions corresponding to coordinate axes. To obtain RCA values for random or standard coordinate directions, we set A to be equal random or identity matrix and do not optimize it during learning.
2. *Individual interpretability (mean-opinion-score, MOS)*. To quantify the interpretability of individual directions, we perform human evaluation. For assessment, we employ eleven human assessors, all having ML/CV background. The evaluation protocol is the following:

[2] http://github.com/ptrblck/prog_gans_pytorch_inference

[3] http://github.com/ajbrock/BigGAN-PyTorch

- For each assessor we sample ten random $z \sim \mathcal{N}(0, I)$;
- For each direction h we plot a chart similar to Figure 5. Namely, we plot $G(z + s \cdot h)$, varying s from -8 to 8 for all z sampled on the previous step).

The assessor is then asked two questions:

- Does h operate consistently for different z?
- Does h affect a single factor of variation, which is easy-to-interpret?

If h meets both requirements, it is treated as "interpretable" and is marked as 1. Otherwise it is marked as 0. To obtain a final MOS value for a set of directions, we average the marks across all assessors and all directions from the set. For a fair comparison, we evaluate different sets of directions on completely the same z.

While MOS measures the quality of directions independently, high RCA values indicate that discovered directions are substantially different, so both metrics are important. Therefore, we report MOS and RCA for directions discovered by our method for all datasets. We compare to directions corresponding to coordinate axes and random orthonormal directions in Table 2. Along with quantitative comparison, we provide the qualitative results for each dataset below.

4.1. MNIST

Qualitative examples of transformations induced by directions obtained with our method are shown on Figure 5. The variations along learned directions are easy to interpret and transform all samples in the same manner for any z.

Evolution of directions. On Figure 6 we illustrate how the image variation along a given direction evolves during the optimization process. Namely, we take five snapshots of the matrix A: $A_{step=0}, \dots A_{step=10^5}$ from different optimization steps. Hence $A_{step=0}$ is the identity transformation and $A_{step=10^5}$ is the final matrix of directions. Here we fix a direction index k and latent $z \in \mathbb{R}^{128}$. The i-th row on Figure 6 are the images $G(z + A_{step=25 \cdot 10^3 \cdot (i-1)}(\varepsilon \cdot e_k))$. As one can see, in the course of optimization the direction stops to affect digit type and "focuses" on thickness.

4.2. Anime Faces

On this dataset, we observed advantage of orthonormal A compared to A with unit-norm columns. We conjecture that the requirement of orthonormality can serve as a regularization, enforcing diversity of directions. However, we do not advocate the usage of orthonormal A for all data since it did not result in practical benefits for MNIST/CelebA. On the Figure 7, we provide examples discovered by our approach.

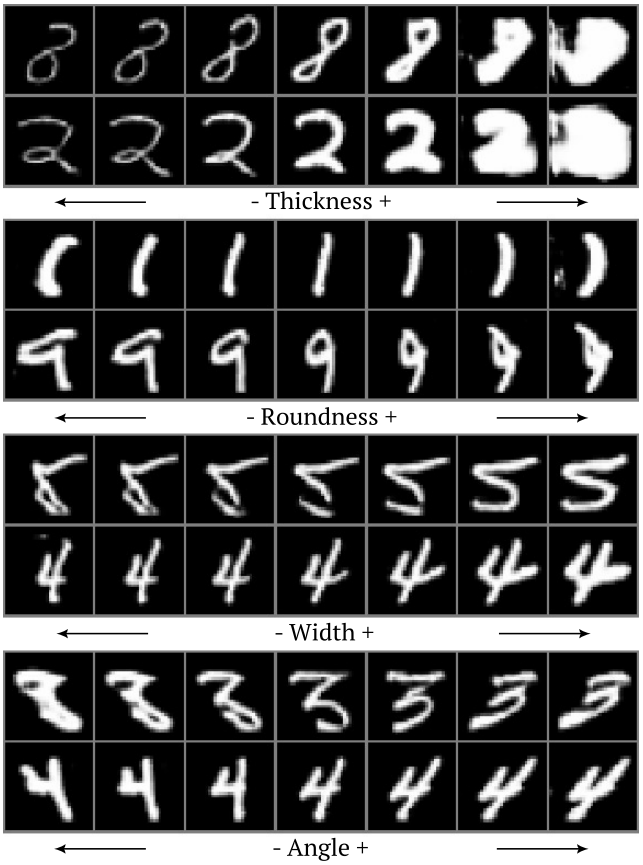

Figure 5. Examples of interpretable directions for Spectral Norm GAN and MNIST.

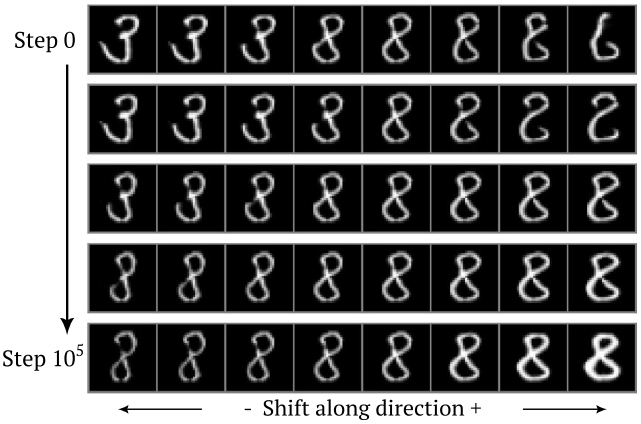

Figure 6. Image variation along a particular column of A during the optimization process. Before optimization, the corresponding transformation affects several factors of variation, and gradually "concentrates" only on the digit thickness as optimization proceeds.

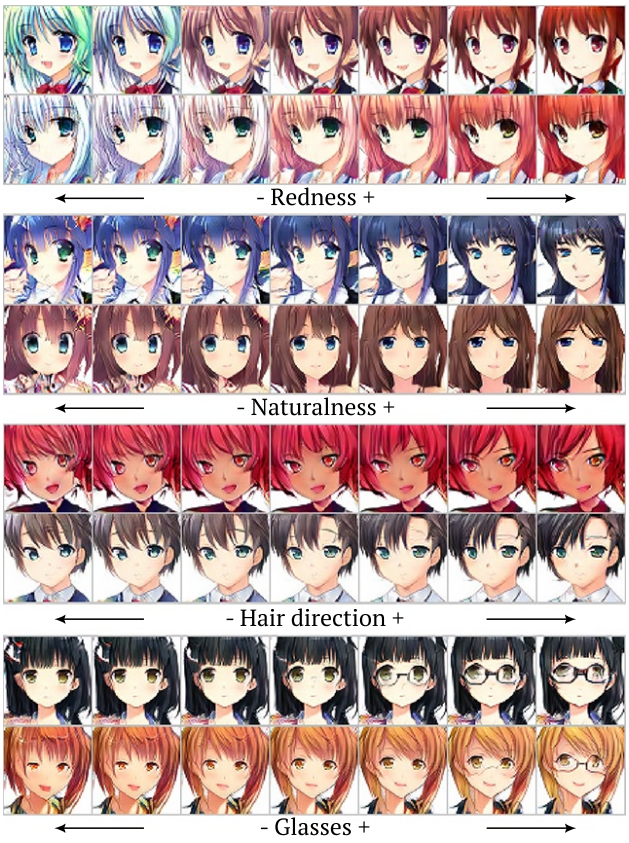

Figure 7. Examples of directions discovered for Spectral Norm GAN and AnimeFaces dataset.

4.3. ProgGAN

Since the latent space dimensionality for ProgGAN equals 512 and is remarkably higher compared to other models, we observed that the reconstructor RCA values notably degrade with $K{=}512$ and we set $K{=}200$ in this experiment. See Figure 8 for examples of discovered directions for ProgGAN. These directions are likely to be useful for face image editing and are challenging to obtain without supervision.

4.4. BigGAN

Several examples of directions discovered by our method are presented on Figure 9. In this dataset, our method reveals several interesting directions, which can be of significant practical importance. For instance, we discover directions, corresponding to background blur and background removal, which can serve as a valuable source of training data for various computer vision tasks, as we show in the following section. Here we also use orthonormal A since it results in a more diversified set of directions.

For BigGAN we also perform more detailed analysis by asking the assessors to categorize the interpretable directions into three types:

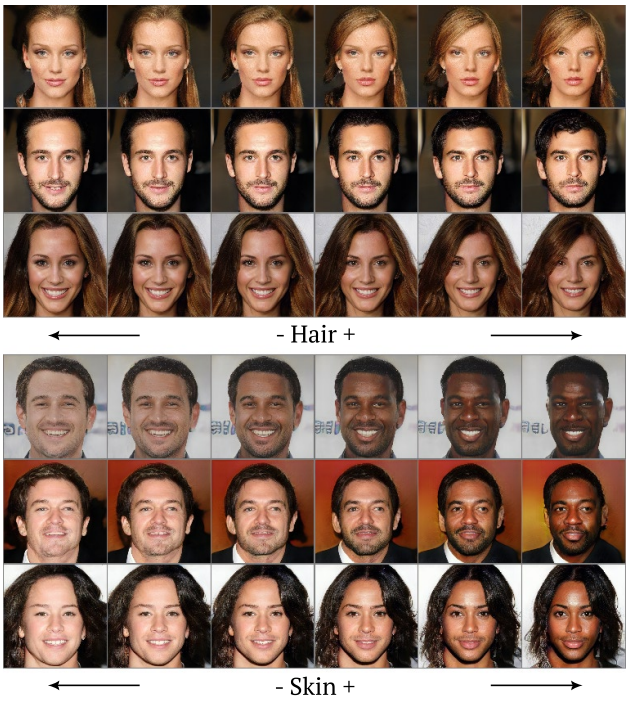

Figure 8. Examples of directions discovered for ProgGAN and CelebA dataset.

Category \ Direction	Random	Coordinate	Ours
Geometry	0.84	0.17	0.45
Coloring	0.16	0.45	0.2
Textural	0	0.38	0.35

Table 1. Rates of different types of transformations among interpretable directions in the BigGAN latent space.

- Geometry (e.g. zoom / shift / rotation);
- Texture (e.g. background blur / add grass / sharpness);
- Color (e.g. lighting / saturation).

Results are presented in Table 1. Notably, interpretable coordinate directions mostly belong to the color or texture types, while interpretable random directions mostly affect geometry (all corresponding to zooming).

5. Weakly-Supervised Saliency Detection

In this section, we provide a simple example of practical usage of directions discovered by our method. Namely, we describe a straightforward way to exploit the background removal direction h_{bg} from the BigGAN latent space for a problem of weakly supervised saliency detection. In a nutshell, this direction can be used to generate high-quality synthetic data for this task. Below we always explicitly

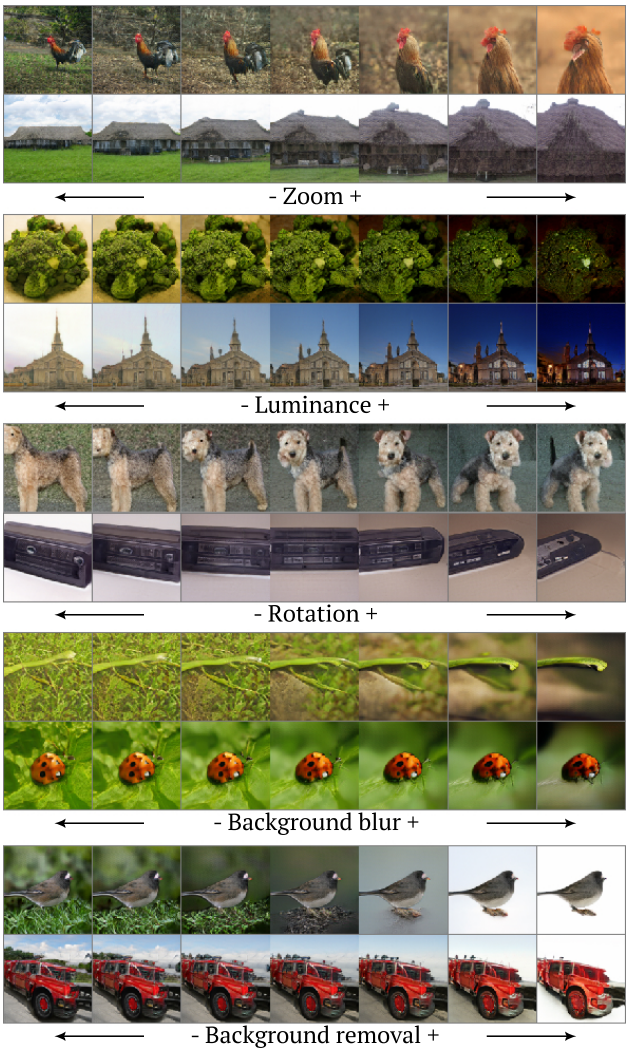

Figure 9. Examples of directions discovered for BigGAN.

specify an Imagenet class passed to the BigGAN generator, i.e., $G(z, c),\ 1 \leq c \leq 1000$.

Figure 9 shows that h_{bg} is responsible for the background opacity variation. After moving in this direction, the pixels of a foreground object remain unchanged, while the background pixels become white. Thus, for a given BigGAN sample $G(z, c)$, one can label the white pixels from the corresponding shifted image $G(z + h_{\text{bg}}, c)$ as background, see Figure 10. Namely, to produce labeling, we compare an average intensity over three color channels for the image $G(z + h_{\text{bg}}, c)$ to a threshold θ:

$$\text{Mask}(G(z, c)) = [\, G(z + h_{\text{bg}}, c) < \theta \,] \qquad (2)$$

Assuming that intensity values lie in the range $[0, 1]$, we set $\theta = 0.95$.

Table 2. Quantitave comparison of our method with random and coordinate axes directions in terms of RCA and individual directions interpretability.

Directions	MNIST	Anime	CelebA	BigGAN
Reconstructor classification accuracy				
Random	0.46	0.85	0.6	0.76
Coordinate	0.48	0.89	0.82	0.66
Ours	**0.88**	**0.99**	**0.9**	**0.85**
Individual interpretability (mean-opinion-score)				
Random	0.06	0.00	0.29	0.19
Coordinate	0.00	0.00	0.22	0.66
Ours	**0.47**	**0.26**	**0.30**	**0.69**

Figure 10. Segmentation masks for BigGAN samples used to train a saliency detection model. *Line 1:* original samples $G(z)$; *Line 2:* samples with reduced background $G(z + h_{\text{bg}})$; *Line 3:* generated binary masks obtained by thresholding.

Given such synthetic masks, it is possible to train a model that achieves high quality on real data. Let us have an image dataset $\mathcal{D}$. Then one can train a binary segmentation model on the samples $[G(z, c),\ \text{Mask}(G(z, c))]$ with classes c that frequently appear in images from $\mathcal{D}$. While the images of $\mathcal{D}$ can be unlabeled, we perform the following trick. We take an off-the-shelf pretrained Imagenet classifier (namely, ResNet-18) M. For each sample $x \in \mathcal{D}$ we consider five most probable classes from the prediction $M(x)$. Thus, for each of 1000 ILSVRC classes, we count a number of times it appears in the top-5 prediction over $\mathcal{D}$. Then we define a subset of classes $\mathcal{C}_{\mathcal{D}}$ as the top 25% most frequent classes. Finally, we form a pseudo-labeled segmentation dataset with the samples $[G(z, c),\ \text{Mask}(G(z, c))]$ with $z \sim \mathcal{N}(0, I), c \in \mathcal{C}_{\mathcal{D}}$. We exclude samples with mask area below 0.05 and above 0.5 of the whole image area. Then we train a segmentation model U on these samples and apply it to the real data $\mathcal{D}$.

Note that the only supervision needed for the saliency detection method described above is image-level ILSVRC class labels. Our method does not require any pixel-level or dataset-specific supervision.

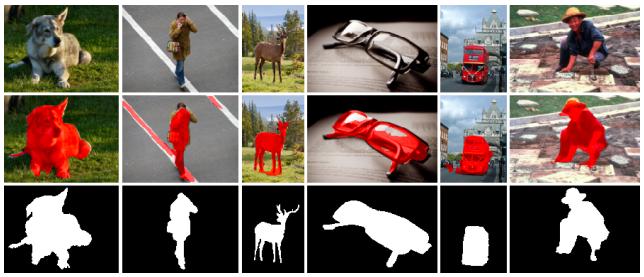

Figure 11. Results of saliency detection provided by our method. *Line 1:* ECSSD images; *Line 2:* predicted masks; *Line 3:* groundtruth masks.

5.1. Experiments on the ECSSD dataset

We evaluate the described method on the ECSSD dataset (Yan et al., 2013), which is a standard benchmark for weakly-supervised saliency detection. The dataset has separate train and test subsets, and we obtain the subset of classes $\mathcal{C}_\mathcal{D}$ from the train subset and evaluate on the test subset. For the segmentation model U, we take a simple U-net architecture (Ronneberger et al., 2015). We train U on the pseudo-labeled dataset with Adam optimizer and the per-pixel cross-entropy loss with the temperature 10.0. We perform 15000 steps with the initial rate of 0.005 and decrease it by 0.2 every 4000 steps and a batch size equal to 128. During inference, we rescale an input image to have a size 128 along its shorter side.

We measure the model performance in terms of the mean average error (MAE), which is an established metric for weakly-supervised saliency detection. For an image x and a groundtruth mask m, MAE is defined as:

$$\text{MAE}(U(x), m) = \frac{1}{W \cdot H} \sum_{i,j} |U(x)_{ij} - m_{ij})| \qquad (3)$$

where H and W are the image sizes. Our method based on BigGAN achieves MAE equal to 0.099, which is a competitive performance on ECSSD across the methods using the same amount of supervision (Wang et al., 2019) (i.e., image-level class labels from the ILSVRC dataset). Figure 11 demonstrates several examples of saliency detection, provided by our method.

6. Ablation

Here we present an ablation of the number of latent directions K and the shift loss term. We ablate K on MNIST and ILSVRC, see the results in Table 3 and Table 4 in terms of individual interpretability (MOS) and RCA (see Section 4 for metrics details). For each K we also report the total number of interpretable directions according to the human evaluation. Notably, small values of K are inferior since the classification task becomes easier and the model does not enforce directions to be "disentangled". On the other hand, higher K does not harm interpretability but often results in duplicate directions.

Table 3. Number of directions K ablation for Spectral Norm GAN pretrained on MNIST dataset.

metrics	$K = 16$	32	64	128
MOS	0.5	0.58	0.47	0.46
MOS (absolute)	8	19	30	59
RCA	0.98	0.95	0.88	0.79

Table 4. Number of directions K ablation for BigGAN.

metrics	K = 15	30	60	90	120
MOS	0.3	0.3	0.38	0.75	0.69
MOS (absolute)	5	9	23	68	83
RCA	0.99	0.98	0.92	0.9	0.85

We also perform ablation of the shift loss term of the reconstructor R by varying its multiplier. The ablation results for MNIST are presented in Table 5. Notably, the extreme values of the scaling factor λ lead to quality degradation. In particular, $\lambda = 0$ leads to "collapse" directions, see Figure 4. With high lambda the directions mostly become similar (e.g. all perform zoom).

Table 5. Number of directions K ablation for Spectral Norm GAN pretrained on MNIST dataset.

metrics	$\lambda = 0$	0.125	0.25	0.5	2
MOS	0.27	0.35	0.47	0.42	0.25
RCA	0.88	0.90	0.88	0.87	0.75

7. Conclusion

In this paper, we have addressed the discovery of interpretable directions in the GAN latent space, which is an important step to an understanding of generative models required for researchers and practitioners. Unlike existing techniques, we have proposed a completely unsupervised method, which can be universally applied to any pretrained generator. On several standard datasets, our method reveals interpretable directions that have never been observed before or require expensive supervision to be identified. Finally, we have shown that one of the revealed directions can be used to generate high-quality synthetic data for the challenging problem of weakly supervised saliency detection. We expect that other interpretable directions can also be used to improve the performance of machine learning in existing computer vision tasks.

References

Bau, D., Zhu, J.-Y., Strobelt, H., Bolei, Z., Tenenbaum, J. B., Freeman, W. T., and Torralba, A. Gan dissection: Visualizing and understanding generative adversarial networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

Brock, A., Donahue, J., and Simonyan, K. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pp. 2172–2180, 2016.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Goetschalckx, L., Andonian, A., Oliva, A., and Isola, P. Ganalyze: Toward visual definitions of cognitive image properties. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5744–5753, 2019.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. *Iclr*, 2(5):6, 2017.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

Jahanian, A., Chai, L., and Isola, P. On the"steerability" of generative adversarial networks. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

Jin, Y., Zhang, J., Li, M., Tian, Y., and Zhu, H. Towards the high-quality anime characters generation with generative adversarial networks. In *Proceedings of the Machine Learning for Creativity and Design Workshop at NIPS*, 2017.

Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

LeCun, Y. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1989.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4681–4690, 2017.

Lee, W., Kim, D., Hong, S., and Lee, H. High-fidelity synthesis with disentangled representation. *arXiv preprint arXiv:2001.04296*, 2020.

Liu, B., Zhu, Y., Fu, Z., de Melo, G., and Elgammal, A. Oogan: Disentangling gan with one-hot sampling and orthogonal regularization. *arXiv preprint arXiv:1905.10836*, 2019.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

Plumerault, A., Le Borgne, H., and Hudelot, C. Controlling generative models with continuos factors of variations. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

Ramesh, A., Choi, Y., and LeCun, Y. A spectral regularizer for unsupervised disentanglement. *arXiv preprint arXiv:1812.01161*, 2018.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and*

computer-assisted intervention, pp. 234–241. Springer, 2015.

Shen, Y., Gu, J., Tang, X., and Zhou, B. Interpreting the latent space of gans for semantic face editing. *arXiv preprint arXiv:1907.10786*, 2019.

Voynov, A. and Babenko, A. Rpgan: Gans interpretability via random routing. *arXiv preprint arXiv:1912.10920*, 2019.

Wang, T.-C., Liu, M.-Y., Zhu, J.-Y., Liu, G., Tao, A., Kautz, J., and Catanzaro, B. Video-to-video synthesis. In *Advances in Neural Information Processing Systems*, pp. 1144–1156, 2018.

Wang, W., Lai, Q., Fu, H., Shen, J., and Ling, H. Salient object detection in the deep learning era: An in-depth survey. *arXiv preprint arXiv:1904.09146*, 2019.

Yan, Q., Xu, L., Shi, J., and Jia, J. Hierarchical saliency detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1155–1162, 2013.

Yang, C., Shen, Y., and Zhou, B. Semantic hierarchy emerges in deep generative representations for scene synthesis. *arXiv preprint arXiv:1911.09267*, 2019.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.