

FaceSpex

Alexander Aeschbacher and Claude Messner
University of Bern

Abstract

In their paper "Deep Models of Superficial Face Judgments" [1], Joshua C. Peterson et al. employ Nvidia's StyleGAN technology to manipulate facial images based on surface-level attributes such as attractiveness or age. They conducted an exhaustive investigation to capture these attributes. More than 1000 synthetic facial images were generated, accompanied by over a million human judgments across 34 attributes. The resultant model has the capability to produce facial images that evoke specific impressions aligned with the modeled attributes. Both the evaluated images and the corresponding judgments have been publicly shared. Leveraging this dataset, we have developed our own customizable face generation model. Our model is publicly available on Github.

1 Introduction

In their paper "Deep Models of Superficial Face Judgments", Joshua C. Peterson et al. employ Nvidia's StyleGAN technology to manipulate facial images based on surface-level attributes such as attractiveness or age. They conducted an exhaustive investigation to capture these attributes. More than 1000 synthetic facial images were generated, accompanied by over a million human judgments across 34 attributes. The resultant model has the capability to produce facial images that evoke specific impressions aligned with the modeled attributes.

Both the evaluated images and the corresponding judgments have been publicly shared. Leveraging this dataset is the base for developing an alternative customizable face generation model: the FaceSpex. FaceSpex is publically available on Github.

This document aims to describe the steps for developing FaceSpex, showcase what FaceSpex can do, and discuss its limitations.

2 StyleGAN

In this project we use stylegan2-ada-pytorch.

When generating latent vectors, we use a normal distribution on \mathcal{Z} , and then map them to \mathcal{W} . From \mathcal{W} we can take vectors and synthesize images using the synthesis network.

StyleGANs \mathcal{W} is higher dimensional than \mathcal{Z} so by using a neural network, styleGAN achieves better disentanglement of attributes and more meaningful dimensions. This makes manipulation easier.

The key to manipulating superficial attributes of faces is to calculate so-called attribute directions in \mathcal{W} for every attribute. The further a latent vector $w \in \mathcal{W}$ lies in some attribute direction $d \in \mathcal{W}$, the more this attribute will be present on the generated image $g(w)$.

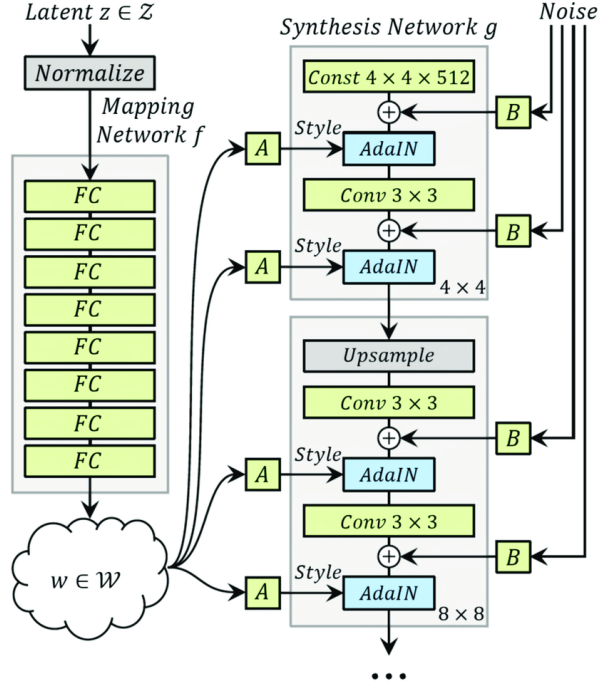


Figure 1: Summary of the StyleGAN Generator Model Architecture
 Taken from: A Style-Based Generator Architecture for Generative Adversarial Networks.

3 Finding attribute directions

Let a be any attribute (e.g. age). We use the data provided by Joshua C. Peterson et al. [1] to do linear regressions. Let

$$X = \begin{bmatrix} w_1^t \\ w_2^t \\ \vdots \\ w_n^t \end{bmatrix},$$

where w_k is the latent vector such that $g(w_k) = I_k$ for any image $I_k \in \{I_1, I_2, \dots, I_n\}$ used in the study by Peterson et al. [1]. Furthermore, let

$$y = \begin{pmatrix} s_{a_1} \\ s_{a_2} \\ \vdots \\ s_{a_n} \end{pmatrix},$$

where s_{a_k} is the mean human judgment of I_k concerning attribute a for any $k \in \{1, \dots, n\}$.

Using this we do a ridge regression with cross validation (as described in [1]):

```
from sklearn.linear_model import RidgeCV
clf = RidgeCV().fit(X, y)
```

Now, the attribute direction d_a we get by normalizing the coefficients of our linear model.

```
attr_dir = normalize(clf.coef_)
```

3.1 Getting the latent vectors

Peterson et al. have provided the images they used for the study $\{I_1, \dots, I_n\}$ as well as the corresponding ratings $\{s_{a_1}, \dots, s_{a_n}\}$ for every attribute a . However, they did not provide the latent vectors $\{w_1, \dots, w_n\}$ used to generate the images $\{I_1, \dots, I_n\}$. StyleGAN provides a projection utility that maps images to their corresponding latent vector. Since these images were generated by styleGAN to begin with, the projection worked very accurately.

3.2 Stabilized directions

Some attributes are highly dependent from one another. Notably age and attractiveness. When making old faces more attractive a common side effect is decreasing age. To remedy this we stabilize directions respectively. So we find a direction that makes faces more attractive without changing their age.

Let a be the attribute we want to change and s be the attribute we want to keep stable. Then we can calculate a stabilized direction by normalizing $d_a - \langle d_a, d_s \rangle d_s$

4 Manipulating the images

In order to manipulate images using meaningful units we calculate mean and standard deviation along any direction.

4.1 Mean and standard deviation

We generate a big enough sample from a normal distribution on \mathcal{Z} and the map it to \mathcal{W} . From this sample we then empirically calculate the mean face μ_F as well as the covariance matrix Σ . These values do not represent any real population, but at least they are not random. Regarding any direction d (normalized vector) we can then calculate the directional mean

$$\mu_d = \langle \mu_F, d \rangle$$

and the directional standard deviation

$$\sigma_d = \sqrt{d^t \Sigma d}.$$

Note that here we could use a more curated sample to get more meaningful results.

4.2 Face Generation

Now we can generate a face f by taking a sample from a normal distribution on \mathcal{Z} and then mapping it to \mathcal{W} . We then normalize it with respect to some direction d by translating along this direction such that the scalar projection equals μ_d :

$$n(f, d) := f - \langle f, d \rangle d + \mu_d d = f - (\langle f, d \rangle - \mu_d) d$$

This ensures that for example if d_a is the age attribute direction, then for any face f , $n(f, d_a)$ appears to have average age. Similarly we can calculate a direction such that its magnitude is its directional standard deviation: $s(d) := d \cdot \sigma_d$.

So we can manipulate faces in standard deviations as units. We can for example generate a face that is m standard deviations older than average by first sampling a face f and then calculating

$$n(f, d_a) + m \cdot s(d_a).$$

5 References

1) Peterson, J. C., Uddenberg, S., Griffiths, T. L., Todorov, A., Suchow, J. W. (2022). Deep models of superficial face judgments. *Proceedings of the National Academy of Sciences*, 119(17), <https://doi.org/10.1073/pnas.2115228119>