Evolution 2.0 - Generating, Fine tuning and Evaluating Artificial Human Faces from StyleGANs

Akhil Sai Peddireddy, Matthew Bielskas
Department of Computer Science, University of Virginia
[ap3ub, mb6xn]@virginia.edu

Abstract

StyleGANs, proposed by Nvidia, is a state-of-the-art model that allows for a surprising amount of control over generating new unseen images and can be leveraged to other applications like semantic face editing and easy fine-tuning. In our project we demonstrate StyleGANs by generating artificial yet high-quality human faces, and further customizing these faces to create more variation in facial features that a human would pick up on. We also demonstrate how to convert a real image into the latent space of StyleGANs and use it for facial editing of features like smile, age, gender, pose etc. We then evaluate the generated images from various pre-trained models by proposing a novel concept of 'Dataset evaluation using models' in contrast to 'Model evaluation using datasets' which is typical in ML and Deep Learning.

1. Introduction

Generative Adversarial Networks are a breakthrough in Computer Vision and Deep Learning. It has helped AI to reach a point where it can create new things which are unseen, by learning from the existing dataset. While GANs have a wide variety of applications, it's ability to generate new unseen images (especially human faces) is particularly interesting and largely sought after by the industry. However, they are often termed as black box models - as any neural network, and leave us with little control over the training/synthesis process. Typically, this is not ideal since customizing and fine tuning the generated images are preferred to match the diverse needs of the target population. Style GANs - 'A Style-Based Generator Architecture for Generative Adversarial Networks' by Nvidia [1] presents a model which adopts the literature from style transfer, with an unsupervised learning of the variation in scale of the attributes.

This solves the above stated problem of less control over training, and helps to generate images at very high quality

Figure 1. Sample artificial human faces generated from the FFHQ pre-trained model at a resolution of 1024x1024

by using many other techniques and heuristics. It allows a greater control over the images generated and also learns the features at various levels - from stochastic variation at freckles, hair etc to variation at higher level features like pose and shape of the face. It generates the artificial image in a step by step fashion starting from a low resolution and going forward to a resolution as high as 1024*1024 pixels. At each step it changes the input gradually to achieve the tuning of coarse, mid and high level features. The Style GANs generate a latent space of the input vectors which is a great innovation as it allows us to manipulate features in a realistic way which can also be leveraged to applications like Deep Face editing.

2. Related Work

StyleGANs are based out of GANs - as the name suggests. GANs consist of a generator and a discriminator. The generator takes a new typically random input and the discriminator picks one input from the training data along with the input from generator. It then classifies whether the generator input is real or fake by looking at the training input and computing a loss which is minimized over time by the feedback to generator. There have been many attempts to

generate artificial human faces using GANs, like with DCGANs [8], but Progressive GANs [2] by Nvidia stand out for their ability to generate high quality and high resolution images. Progressive GANs are novel in that they introduced the first step-wise training from lower to higher resolution. However the drawback is that the features are entangled, so modification to any part of the input might cause the entire image to be distorted, which is what StyleGANs try to address.

StyleGANs also draws some literature from Neural Style Transfer [3] - which employs a pretrained Convolutional Neural Network to transfer the styles from other images. Instead of relying on progressive layers as seen in ProGAN, StyleGANs have a Mapping network which encodes the input vector into an intermediate vector. This is the step where the distribution in the training data is decoupled and the feature entanglement is eliminated. The Mapping network as defined in StyleGANs has 8 fully connected layers. Another component is the Adaptive Instance Normalization (AdaIN), which transfers the encoded information from Mapping Network to the generated image. The initial input is replaced with constant values, yet the same AdaIN mechanism is used to add noise. There have been many application of StyleGANs such as InterFaceGAN [9] which among other similar work, tries to edit facial features in the images using the latent space concepts of StyleGANs.

3. Generating, Face Editing and Style Mixing

We first demonstrate generation of various artificial human faces from the pre-trained models, as seen in Fig 1. Since the latent space is what it makes StyleGANs unique, our interest was to explore this on facial features like smile, age, hair texture, pose etc.

3.1. Face Editing with Latent Directions

This latent space in StyleGANs is a novel concept which opens the door for many applications. One of them is Semantic Face Editing. StyleGANs learns the latent boundaries for different facial features and attributes, allowing us to edit both artificially generated and real faces. The latent vector for the artificially generated faces can be directly used to combine them with the latent directions whereas for a real face, it must first be converted into the latent representation and then edited with these latent directions like smile, age, gender, pose, accessories etc. Fig 2 and Fig 3 shows a real image which is first converted into latent space and then imposed with smile and age with varying coefficients that represents the degree of the encoding. These latent directions [5] and the generator are taken from a pre-trained FFHQ 1024x1024 model from the official Tensorflow implementation of StyleGANs.

Figure 2. Edited Real Image by converting it into latent representation and adding the latent direction for 'smile'

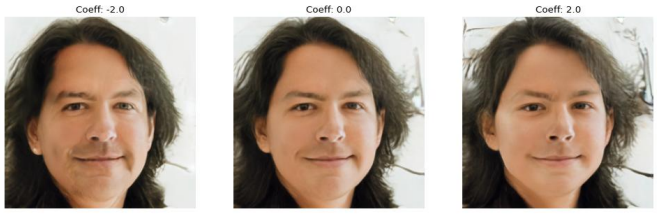

Figure 3. Edited Real Image by converting it into latent representation and adding the latent direction for 'age'

3.2. Style Mixing

Style Mixing is an important aspect because other works in this area i.e DCGANs don't allow much control over the generated images as StyleGANs do. Typically these artificial humans are extensively used in entertainment and fashion industries where fine-tuning them according to demographics, age and ethnic taste is an expected feature. This led our way to explore Style mixing - an interesting idea in the StyleGANs paper which can help to fine-tune the images as demonstrated in Fig 4. It gives the ability to choose source and target images and generate an artificial face by combining these two on different levels (coarser to finer) as desired. This can be particularly useful when we want to change finer feature of a generated image (aka source image in style mixing) like hair color, skin color etc to that of another (target) image which has these features, while still preserving the actual look and coarse features of the generated image. We also demonstrate an application of Style Mixing in Section 4.

4. Evaluation of the Generated Images

Image quality is an important metric for these artificial and fake images. These images are intended to be used in place of real humans yet not compromising on quality, which means the people who see it should not be able to distinguish between a real face and an artificial face. The quality of generated artificial faces depends upon factors like (a) the source dataset the GAN model is trained on (b) the resolution of the source images and (c) the training time of the model. To assess this image quality, we did experiments with images generated from pre-trained models - FFHQ 1024x1024, FFHQ 512x512 [10], FFHQ

Figure 4. Style Mixing on a 5x3 grid where Source Images fill the top row and Destination Images fill the left column. By our process in Section 4, we have gender array [10110 10110 10110] and age array [11000 11000 11000].

256x256 and CelebA 1024x1024, where each pre-trained model represents the dataset and resolution of images the GAN is trained on. Our approach is to generate automatically labeled images (for age and gender) from the pre-trained models by augmenting the dataset using Style Mixing and then trying to calculate the accuracy for each face-set by predicting gender and age using a state-of-the-art pre-trained model for age and gender prediction. This is a different approach where instead of evaluating model on different datasets, we try to evaluate different artificial face datasets using the same model. Another reason behind these experiments is to demonstrate how a large-scale dataset of faces can be categorised for industrial application.

4.1. Style Transfer and Labeling

With m source and n destination images, we would like to manually label the source ones so that coarse-grained style transfer accurately distributes labels to an mn-image dataset. With FFHQ datasets we label gender (M/F) and age (child/adult), while with Celeb we only label gender due to lack of child faces. The goal of this task is to create a sufficiently large dataset that we can apply pre-trained gender/age prediction models on. Below we describe our approach for $m = n = 50$ based on StyleGAN code.

1. Generate 100 unique integers to be used as StyleGAN seeds. Split (and ideally order) them in separate Source and Destination lists.

2. Concatenate these lists, and use them with StyleGAN to paste generated faces on a grid image. In our case Source seeds were first in the concatenation, and the faces were ordered top-down along each column from left to right.

3. Check this image for faces that stand out as poor-quality, change the corresponding seeds (i.e add 1), and repeat Step 2 until you are satisfied. Label gender or age for the 50 source faces in an array.

4. Turn this into a length 2500 array for gender or age. At this point we already modified existing Style Transfer code and thus knew the "order" of image output. Essentially with coarse-grained transfer, the Destination image is fixed for 50 images while Source images are iterated so that gender/age of the final image reflects the Source. This corresponds to us simply copying our original array many times to create the final array.

5. Run Style Transfer iteratively using Source and Destination seeds and save the output in an ordered manner i.e "example400.png". With Step 4, you have a guarantee that the number of your saved image corresponds to an array index with the correct gender or age label.

4.2. Gender and Age Prediction

Next we use a pretrained gender or age prediction model on our images to predict labels, which we evaluate with the ground-truth array. Recall that we ensure array index and picture number are the same for easy retrieval. We decided to apply a high-performing network from Savachenko that is compatible with both tasks, as described in [6] and implemented in [7]. This network is denoted as *age_gender_tf2_224_deep-03-0.13-0.97*, where essentially MobileNet is pretrained on the VGGFace dataset for face recognition. Freezing weights allows this to transfer to gender and age prediction. Unlike in our case, where we know there is exactly one face, Savachenko's methodology generalizes to multiple faces via bounding boxes.

To elaborate on our labeling: we previously assigned 0 to Female and 1 to Male for Gender, and 0 to Child and 1 to Adult for Age (see Figure 4). We maintain the same standard for labeling predictions. Gender results from the implementation are sigmoid outputs where values under 0.5 round to Female (0) and results over 0.5 round to Male (1). Age results include a single predicted value along with a Prediction Interval. Using just the value, we round to Child (0) if age is under 14 and Adult (1) otherwise. When first running this experiment, we ran into a rare circumstance where no bounding box was created. Thus we assign 2 to an image in such cases. Note that this creates an additional kind of error, but it may indicate that Style Transfer produced an image too far removed from what a face should look like. See our prediction accuracies below.

Prediction on Style Mixed Images		
Model	Gender Acc.	Age Acc.
Celeb-1024	0.9036	N/A
FFHQ-1024	0.8404	0.9504
FFHQ-512	0.8088	0.9012
FFHQ-256	0.8636	0.8984

There are minor limitations that impact our results. One is the absence of bounding box as previously described. Another is the possibility of generating androgynous faces because our Source and Destination images are StyleGAN output. Of course we see such faces in real life, but here we're referring to minor artifacts from StyleGAN. Rather than discard the seed like we do for artifacts that completely ruin the face, we decide to label with our best guess. Ultimately this impacts very few labels but it is still a source of uncertainty. Can we overcome this with StyleGAN pretrained for just male or female faces? Finally, another limitation is the generation of significant artifacts through Style Transfer despite our previous 'filtering'. We suspect this may reasonably impact gender and age prediction after looking at our images. Can we overcome this by experimenting with the new StyleGAN2?

5. Future Work

We have manually labeled the 50 source images and as described above, the labeling for some images is ambiguous to the point where human perception and model prediction can easily differ. One extension and fix for this is to use the accurately predicted images (the roughly 90 percent of Celeb1024) to further repeat the experiment and generate many new labelled images with a "double confirmation". These roughly 2250 images are labelled the same both by humans (through the 50 source images) and models thereby removing all ambiguity of human assessment, and thus can be used to generate 2250x2250 images with more confidence.

Another interesting extension for our Age Prediction is to rely on crowdsourcing for integer-valued ground truths. Our motivation is that we've noticed some unexpectedly good predictions before binarization. The issue here is that this cannot be done on Source and Destination images because Style Transfer poorly transfers age when compared to gender. For example, we could not even form an "Older Adult" category because we saw too much change from source to output image. Thus this line of work goes hand-in-hand with developments in transferring age-related characteristics.

6. Conclusion

We have demonstrated that StyleGANs are ideal for image generation since they allow for greater control over generated images using the latent space concept, which is very important in generating artificial human faces. We have also outlined the applications of StyleGANs for editing even faces of real people for smile, age, gender etc. Style mixing is then demonstrated along with its applications pertaining to the fine-tuning of generated images for various interests. Afterwards we evaluated our automatically-labelled face datasets with a state-of-the-art gender and age prediction model, thereby proposing a unique and simple approach to "evaluation of datasets using a model". We conclude that this also helps in automatically eliminating bad images generated from StyleGANs as explained in our first proposed extension in Section 5.

References

[1] Tero Karras, Samuli Laine, Timo Aila. *A Style-Based Generator Architecture for Generative Adversarial Networks*

[2] Tero Karras, Timo Aila, Samuli Laine, Jaakko Lehtinen. *Progressive Growing of GANs for Improved Quality, Stability, and Variation*

[3] Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, Yizhou Yu, Mingli Song. *Neural Style Transfer: A Review*

[4] StyleGAN: Official Tensorflow Implementation `https://github.com/NVlabs/stylegan`

[5] StyleGAN Encoder - converts real images to latent space `https://github.com/Puzer/stylegan-encoder`

[6] Andrey V. Savchenko. *Efficient Facial Representations for Age, Gender and Identity Recognition in Organizing Photo Albums using Multi-output CNN*

[7] Face Recognition: Official Tensorflow Implementation `https://github.com/HSE-asavchenko/HSE_FaceRec_tf`

[8] Alec Radford, Luke Metz, Soumith Chintala. *Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks*

[9] Yujun Shen, Jinjin Gu, Xiaoou Tang, Bolei Zhou. *Interpreting the Latent Space of GANs for Semantic Face Editing*

[10] Implementation A Style-Based Generator Architecture for Generative Adversarial Networks in PyTorch `https://github.com/rosinality/style-based-gan-pytorch`