Final Project Team 33:

Criminal Face Generator

Advisors: Dr. Chia-Wen Lin, Dr. Chih-Chung Hsu

Students: 106091221 Sophia Yang, 106091228 Jun-Teng Zhang

*Abstract*—In this work, we propose a criminal face generator based on GANs to replace the problematic process of traditional crime investigation. The proposed method consisted of two crucial components: DCGAN and StyleGAN which is for generations and manipulation. Our model can solve many problems of crime investigation by first generating a face and inversing the face to different features or expressions for unlimited times in an objective and efficient way.

Keywords—Datasets, Generative Adversarial Networks, Text to Image, Facial Attributes, Face Generation Introduction

# Introduction

How to reconstruct the face of the suspect is a difficult but meaningful task. Suppose police are trying to reconstruct suspect's face to help them solve a case, the most common method adopted by these polices is to hire an artist to create a portrait according to the descriptions provided by the eyewitnesses. Then, the police use this portrait to search for anyone who looks like it. However, this method is usually not practical in the real scenario. We conclude that there exist four main problems that need to be solved: (1) Firstly, the portrait largely depends on painter art style, so the same person may look totally different in different art style. (2) Moreover, it's well-known that art is subjective, so it's difficult to tell the similarity between the portrait and the owner of the face. (3) Facial expression also affects the result a lot, so we need a generator which can deal with facial expressions as well. (4) Finally, the sketching process is too time-consuming and not interactive enough. All these 4 problems will lead to difficulties to find the real criminal; therefore, we need a new suspect face generator which is not only fast and objective but also user-friendly which has an ability to create interactive atmosphere to reconstruct suspect face.

In this project, for the first time, we propose a method focusing on specific scenario-crime investigation. To achieve this goal, we need our model not just general face generator, but a generator has some special attributes. As compared to original way, our idea is to transfer the facial features provided by eyewitnesses to the machine and let machine generate a suspect's photo. To be More specific, our method consists of two parts, DCGAN for face generation, and StyleGAN for feature-guided face manipulation. By our method, we can first generate a face and inverse the face to different features or expressions for unlimited times so eyewitnesses can continually revise the generated image to fit the suspect face better, which can solve the problem in modern criminal investigation.

To sum up, this work has the following contributions:

1. The art style problem in real investigation scenario will be solved because our model generates photo-like results not portraits.
2. And the subjective problem will also be solved because the generated result will be relatively objective because the results are generated by machine not artists.
3. We can apply a kind of Generative Adversarial Network called StyleGAN to generated picture to change the facial expression of the face which increase the diversity of the same face.
4. Our model can not only be utilized to criminal investigation but also to generate fictional characters and to customize game characters.

# Methods

## Deep Convolutional GAN

We want to use DCGAN to generate some model for StyleGAN step and further usage. DCGAN focus on improving GAN in network architecture, replacing the generator and discriminator with CNN [1]. Figure 1 is the construction detail and capabilities of DCGAN.

**Constructure.** The structure contains following features:

1. Remove pooling, use strided convolution in discriminator, use transpose convolution in generator.
2. Use batchnorm on generator and discriminator.
3. Remove fully connected layers.
4. In the generator, output layer use tanh, and all other layers use ReLU.

figure 1. Generator network constructure

**Training Detail.** The training detail of our DCGAN is composed of deconvolution and batch normalization:

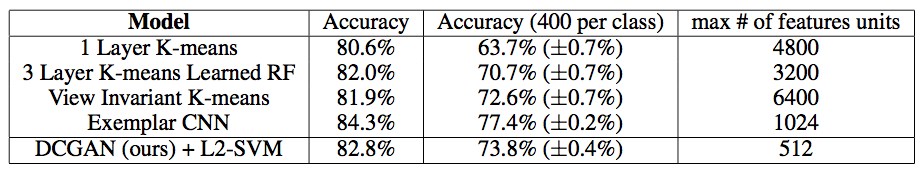
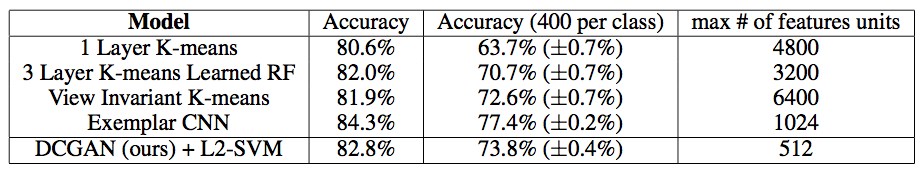
1. Deconvolution (Transposed Convolution)

The normal convolution maps a 4 x 4 image to a 2 x 2 image, deconvolution do the reverse.

1. Batch Normalization

Speed up training process and prevent gradient vanishing. Reduce the influence produced by data initialize of tanh or sigmoid. Eventually improve the accuracy of training.

**Capabilities.** To verify capabilities, put features into L2-SVM and compare the result with other unsupervised-learning.

1. Comparison on Minist
2. Comparison on SVNH

## Style-based Generator Step

We would want to construct a neural network which can translate the work into corresponding latent code. Then we could use the latent code to modify the original image with different look. Compared with traditional GAN we use StyleGAN as our basic model. Here is the overall architecture of StyleGAN and how it works with evaluation method. [2]

**AdaIN.**  The AdaIN operation is defined as equation (1).

一張含有 文字, 手錶 的圖片

自動產生的描述 (1)

Each feature map is normalized separately, and then scaled and biased using the corresponding scalar components from style y. Thus, the dimensionality of y is twice the number of feature maps on that layer.

**Stochastic Variation.** Thus Finally, the generator generates stochastic detail by introducing explicit noise inputs. They feed a dedicated noise image to each layer of the synthesis network. The noise image is broadcasted to all feature maps using learned per-feature scaling factors and then added to the output of the corresponding convolution. Stochastic aspects in human portraits: hairs, stubble, freckles, or skin pores. Traditional generator implements stochastic variation: Add noise through input latent code, which consumes network capacity. StyleGAN architecture: Add per-pixel noise after each convolution.

**Style Mixing.**  StyleGAN use latent code at each layer of neural networks, which leads to learning process to be more relevant. To reduce the correlation, our model randomly use two kinds of input vector and generate the latent code. While training an input code, we would switch to another input code at random. By doing so, we could reduce the correlation. However, it could not increase the performance. But it could generate the images in coherence.

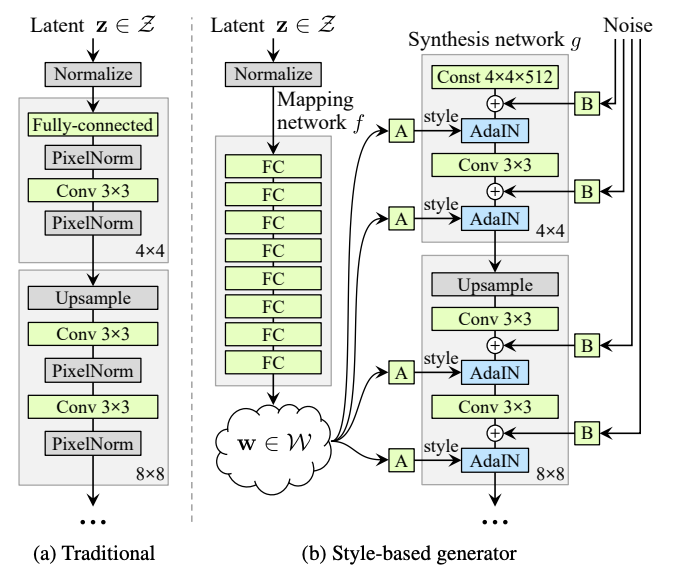


figure 2. Image by StyleGAN paper [2]: Style-based Generator

**Perceptual Path Length.** A perceptually based pairwise image distance: a weighted difference between two VGG16 embeddings. If they subdivide a latent space interpolation path into linear segments, they can define the total perceptual length of this segmented path as the sum of perceptual differences over each segment. In conclusion, we use Perceptual Path Length to measure how smooth the latent code can be. In other words, if two image have similar quality of image. They should have similarity in latent space. In general, the bad quality of image would be restricted to a small region. The difference between good from bad is the perceptual distance. If perceptual distance is long than it means the bad quality in latent space which leads to fail generate nice one.

**Linear Separability.** Linear separability is an important concept in neural networks. The idea is to check if you can separate points in an n-dimensional space using only n-1 dimensions. Lost it? Here's a simpler explanation. We start by generating 200K images with z ∼ P ( z) and classify them using an auxiliary classification network with label Y. We keep 50% of samples with the highest confidence score. This results in 100k high-score auto-labeled (Y) latent-space vectors z for progressive GAN and w for the Style-GAN. We fit a linear SVM to predict the label X based only on the latent-space point ( z and w for the Style-GAN) and classify the points by this plane. We compute the conditional entropy H( Y | X) where X represents the classes predicted by the SVM and Y are the classes determined by the classifier. We calculate the separability score as exp( Σ (H(Y| X) ) ), summing for all the given attributes of the dataset. We basically fit a model for each attribute. Note that the CelebA dataset contains 40 attributes such as gender info.

一張含有 桌 的圖片

自動產生的描述

figure 3 Image by StyleGAN paper [2]

# Experiment

## Dataset

In our implementation, we choose CelebA as our training dataset. CelebA, a.k.a CelebFaces Attribute, collects lots of images of celebrities and their face attribute. The data set contains 10177 distinct celebrities and 202599 images in total, each image has 40 attribute annotations. It is provided by the Chinese University of Hong Kong and is widely used in some tasks, including face attribute recognition, face recognition, face detection, landmark localization…etc. For our task, we will use it to train a model, which can generate fake images according to the input feature. We used the Align and cropped images it provided as our training data and extract 20000 images for each attribute. By doing this, then we can train different model to generate specific image.

## Generation

In the first part, we use DCGAN to generate a fake image. After we extract images for each attribute, we will get 40 corresponding folders. In our implementation, it will create a data loader to load the dataset for the specified feature. As for our DCGAN training setting, we use batch size 128, image\_size 64, num\_epoch 100, learning\_rate = 0.0002, beta1 for Adamoptimizer 0.05 and weight initialization from N(0,0.02). Each model will take about 3~4 hours to train, and we can use the trained model to generate fake images.

一張含有 朦朧 的圖片

自動產生的描述

figure 4. Image generated by our DCGAN

## Style Transfer

In the second part, the image produced in previous stage will be transform based on which feature we want. We use an open-source method called TediGAN to achieve our goal. According to the original paper, the TediGAN contains 3 parts: StyleGAN inversion module, visual-linguistic similarity learning, and instance-level optimization. It uses StyleGAN as a method to transfer facial features of an image, and the authors design an inversed model as an image encoder, which can map the input image into the latent space of the pretrained StyleGAN. To learn text-to-image matching, it uses a pretrained text encoder, which is called CLIP (Contrastive Language-Image Pre-Training), and the encoder will map the image and text into a common embedding space. The overall flow is first giving an input image, we will get a encoding result by inversion process, then perform the StyleGAN and CLIP model to get the final image.

一張含有 男人, 室內, 個人, 尋找 的圖片

自動產生的描述

figure 5. The result of style transferring. The left image is generated by our DCGAN, after performing two inversions, we can get two images.

## Demo

In this part, we will show how to perform our program. First, we need to select a certain feature, which is contained in the CelebA. Then our DCGAN model will generate a fake image which were shown in figure 6.

After we get the image, then we can select any feature to transfer the image to what we expect. The styleGAN will transfer the image based on the input, and we can get the edited image. The process can be repeat until the ideal image is produced which were shown in figure 7.

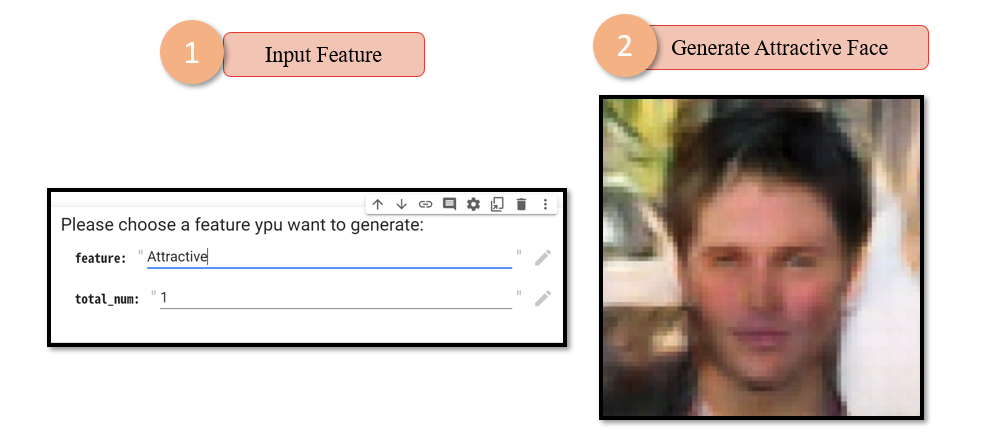


figure 6. The process of DCGAN

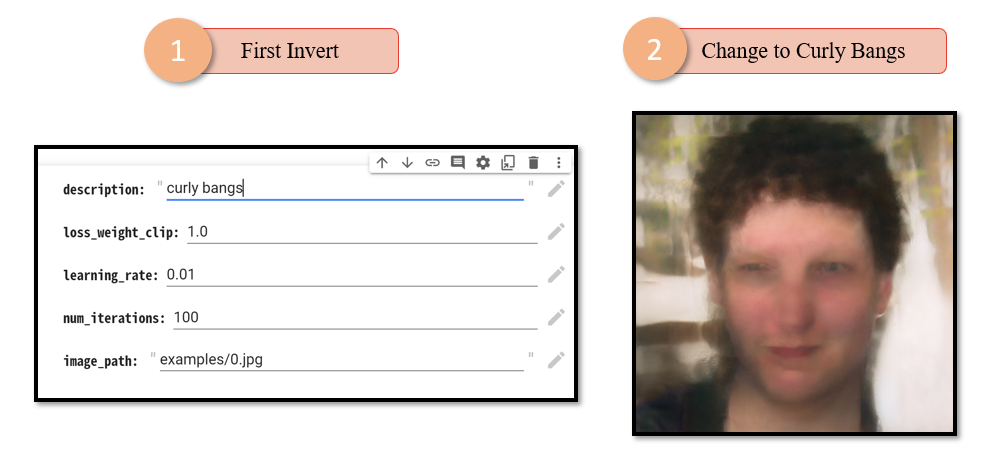


figure 7. The process of StyleGAN

## Comparison

## In this part, we compare our generated image with the images generated by other GAN including CGAN, attrGAN and progressive GAN which were shown in figure 8. From the comparison, we can find that the quality of the images we generated is not very good, and even far less than the images generated by progressive GAN. In addition to these models, there are many SOTA models can perform the same task, which means that we still have long way to go.

一張含有 文字, 室內 的圖片

自動產生的描述

figure 8. The comparison of images generated by DCGAN, cDCGAN, attrGAN and progressive GAN (From left to right)

## Adding PULSE

## From the last experiment, we found that we have serious problem about resolution. It’s not only having low resolution but also affect the performance of transfering of style. We tried to solve the problem by adding PULSE [7]. PULSE can generate a high-generation image by a self-supervised method.

一張含有 箭 的圖片

自動產生的描述

figure 9. adding PULSE to improve quality of image

Then we trained the model with higher epoch.

一張含有 個人, 朦朧 的圖片

自動產生的描述

figure 10 Our style transfering process before adding PULSE

一張含有 個人, 室內, 擺姿勢, 笑臉 的圖片

自動產生的描述

figure 11 Our style transfering process after adding PULSE

As you can see, the quality of image got much better performance. In conclusion, out model is highly dependent on the input image. By adding pulse, our image can tell the difference after the transfer of each layer easily. That say that our approach is working successfully.

We also observe that the distortion of generated pictures will be greater when the times of inversion increase. To solve this problem, we add PULSE right after DCGAN let the resolution increase from 64x64 to 256x256. And as picture shows in figure, the distortion became smaller.

一張含有 文字 的圖片

自動產生的描述Our Final Overall Architecture

## Evaluation

Although we can use GAN to generate a fake image, but there is not a perfect criterion to evaluate the model. Because training GAN do not need an object function, it is hard to measure the difference between real and fake image. There are still some methods which can judge the quality or diversities of GAN. Because we don’t have much time, we only generate images without evaluation. But we will still introduce some evaluation method, and maybe we can conduct the evaluation in the future.

**Inception Score (IS).** Inception Score is calculated based on the inceptionNet, it considers the quality and diversities of the generated images. For the quality, it will put a generated image into an inceptionNet to performing classification, if the quality of the image is high, then would have high probability to be label a certain class. On the other hand, the measure of diversity is to generate a large enough number of images by the model. Ideally, the number of each category should be similar, or it would be lack of diversity. The formula of Inception Score is equation (2).

The actual meaning is the difference between the entropy of the distribution function of all generated samples in each category and the entropy of the probability distribution of each sample in each category. But the Inception Score has some limitation. First, high quality images can get a high IS value, but high IS value doesn’t mean they would be high images. Second, the original calculation needs lots of images to achieve a much precise result. Third, the inceptionNet is trained on the ImageNet, so the GAN should also use ImageNet to train, or the score has no meaning. Last, inception score would not tell us whether the GAN has the overfitting problem.

**Frechet Inception Distance score (FID).** FID is a new method proposed as an improvement of IS, it will calculate the mean and covariance of the images based on the comparison of the statistics of a set of generated images with the statistics of real images from the target domain. The output of the function is generalized to a multivariate Gaussian distribution. These statistics are then used to calculate the activation functions on the real and generated image sets, and the distance between these two distributions is calculated using the Frechet distance (aka Wasserstein-2 distance). The formula of FID is equation (3).

The smaller the FID, the closer the generated distribution is to the real image.

# Discussion

Regarding the method we used, we present some advantages and disadvantages for discussion.

## Advantages

First, we used DCGAN to train the model. Compared with traditional GAN, DCGAN has the following characteristics [6]:

1. Use strided convolutions in the discriminator model instead of pooling layers; In the generator model, use fractional convolutions to complete the generation process from random noise to images.
2. In the network structure, in addition to the output layer of the generator and the input layer of the corresponding discriminator, Batch Normalization is used on other layers. In this way, fixes weak initializations and resolves gradient propagation. To each layer, it also prevents the generator from converging all samples to the same point.
3. DCGAN uses various activation functions, such as Adam optimization, Rectified Linear Unit (ReLU) activation function, and Leaky ReLU.

Second, we used StyleGAN to transfer image style. StyleGAN has the following characteristics:

1. It generates artificial images step-by-step, starting from very low resolutions and going all the way up to high resolutions. By modifying the input at each level in the network separately, it can control the visual features represented at that level, from coarse features such as pose, face shape to fine details such as hair color, then without affecting others' levels.
2. StyleGAN can not only generate high-quality and realistic images, but also enables better control and understanding of the generated images, and even makes it easier than ever to generate high-confidence fake images.

## Disadvantages

About DCGAN, we also discuss its shortcoming:

1. Since removing the fully connected layer, and directly using the convolution layer to connect the input and output layers of the generator and discriminator. It increases the stability of the model but slows down the convergence speed.
2. The generator produces a limited variety of samples and is highly sensitive to the hyperparameter selections.
3. About the parameters of the model, they destabilize and never converge.

Then, about StyleGAN also has some disadvantages that need to improve:

1. Obviously, using StyleGAN to generate images sometimes contains artifacts which like speckled.
2. Many details such as teeth, eyes, and other parts are not clear enough.

## Discussion of Our Work

In this work, we propose the shortcomings and difficulties encountered by the method used in our implementation:

1. The original expected goal was to generate corresponding images through a full sentence of text description, but after actual research, it was found that it was difficult for us to implement, so we changed it to input a single feature to implement.
2. The DCGAN used in the first part has no way to generate fake pictures of the corresponding features. Although there is a way such as Conditional GAN that can achieve similar goals, the implementation is difficult, so we finally chose to separate the datasets according to the features to train the model and generate fake images.
3. The environment we use is Colab and Colab has hardware limitations. Hence, there is no way to train so much data, and the effect of the output is relatively limited.

But on the other hand, we also discuss where our proposed work has advantages in the application.

1. When asking about the face characteristics of a crime, the process is often highly interactive. Therefore, our proposed work, in the beginning, can first propose the most critical characteristics of the crime, generate a fake image. Then, step by step according to the detailed description of the victim, the facial features of the crime are continuously corrected in real-time to achieve an efficient investigation process.
2. Our proposed work not only can be a means of investigating criminals, but also in games, animations, and novels, this technology can be used to generate several possible faces as a reference to design the appearance of characters.

# Conclusion

We have proposed a novel method for suspect face synthesis based on facial attribute, which unified two kinds of GAN into the same framework and achieves high controlability and user-friendly. Through the proposed method, we can effectively synthesize images with selectable features, and the following face-guided manipulation can be completed by the utilization of StyleGAN which can solve the problems of crime investigation in real scenario. In the future, we want to generate images of higher quality and resolution by adopting a high-quality dataset such as CelebA HQ. By doing this, we expect we can generate 1028x1028 images with little distortion compared with our methods at present. Moreover, we will try to integrate our work at present with the NLP method. By doing this, we can not only use a single feature as input but a long sentence or description which will be more like a real scenario.

# References

1. Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." arXiv preprint arXiv:1511.06434 (2015).
2. Karras, Tero, Samuli Laine, and Timo Aila. "A style-based generator architecture for generative adversarial networks." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019.
3. Xia, Weihao, et al. "TediGAN: Text-Guided Diverse Face Image Generation and Manipulation." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021
4. Barratt, Shane, and Rishi Sharma. "A note on the inception score." *arXiv preprint arXiv:1801.01973* (2018).
5. Heusel, Martin, et al. "Gans trained by a two time-scale update rule converge to a local nash equilibrium." *Advances in neural information processing systems* 30 (2017).
6. Dewi, Christine, et al. "Various Generative Adversarial Networks Model for Synthetic Prohibitory Sign Image Generation." *Applied Sciences* 11.7 (2021): 2913.
7. Menon, Sachit, et al. "Pulse: Self-supervised photo upsampling via latent space exploration of generative models." *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*. 2020