# Anime Character Creation with GAN


Holly Liang
*xuejiao@stanford.edu*

Rena Sha
*renasha@stanford.edu*

Stanford University


## Overview

### Motivation

In many scenarios, such as in game design and content creation, graphic artists are needed to draw cartoon or anime avatars. Besides being financially expensive to hire an artist, creating an avatar is also time-consuming work. If we could instead teach computers how to draw anime characters, then we can significantly lower the barrier of entry to avatar creation. Anyone with access to the program will be able to generate cute and unique avatars for their projects.

### Goal

Use Generative Adversarial Networks (GAN) to generate high quality anime avatars.

### Related Work

**Deep Convolutional GAN (DC-GAN) Models**

The most popular method of anime avatar generation uses vanilla DC-GAN and a highly diverse dataset of images from online image boards such as Danbooru, Pinterest and Safebooru. The avatars created using this model, however, were uniformly of low quality. The generated images were small in size and contained significant noise and pixilation with some outputs having unrecognizable facial features.

**Jin et al**

Their model consists of a modified SRResNet architecture combined with a novel gradient penalty scheme called DRAGAN (Deep Regret Analytic Generative Adversarial Networks. They also use a specific, high-quality dataset consisting of character illustrations obtained from a single Japanese ecommerce website selling video games. The avatars in their dataset are all frontal facing with an all-white background. Jin was able to produce aesthetically pleasing, high quality avatars using this method. However, a major disadvantage of their generated outputs is low variation - most of their avatars look very similar to each other.

## Methods

**Deep Convolutional GAN (DC-GAN) Baseline Model**

We will use DC-GAN as our baseline model. We will first fine-tune a DC-GAN model by trying out different architectures and varying the hyperparameters.

**Extensions**

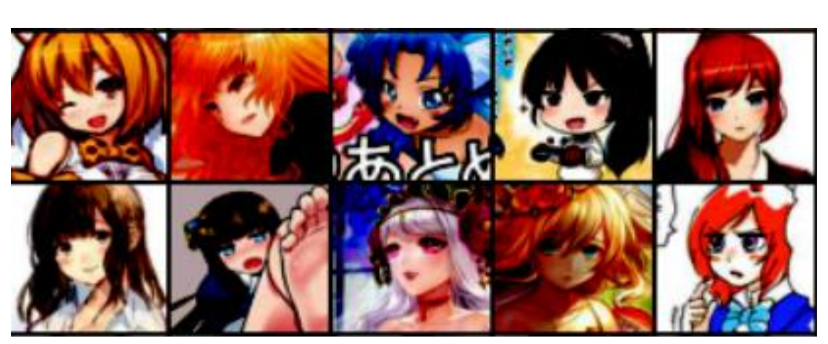
1. DC-GAN + Spectral Normalization
2. DC-GAN + Wasserstein GAN (W-GAN)

**Super Resolution**

We will use Super Resolution GAN (SR-GAN) to apply super resolution to the images generated by GAN.

## Dataset

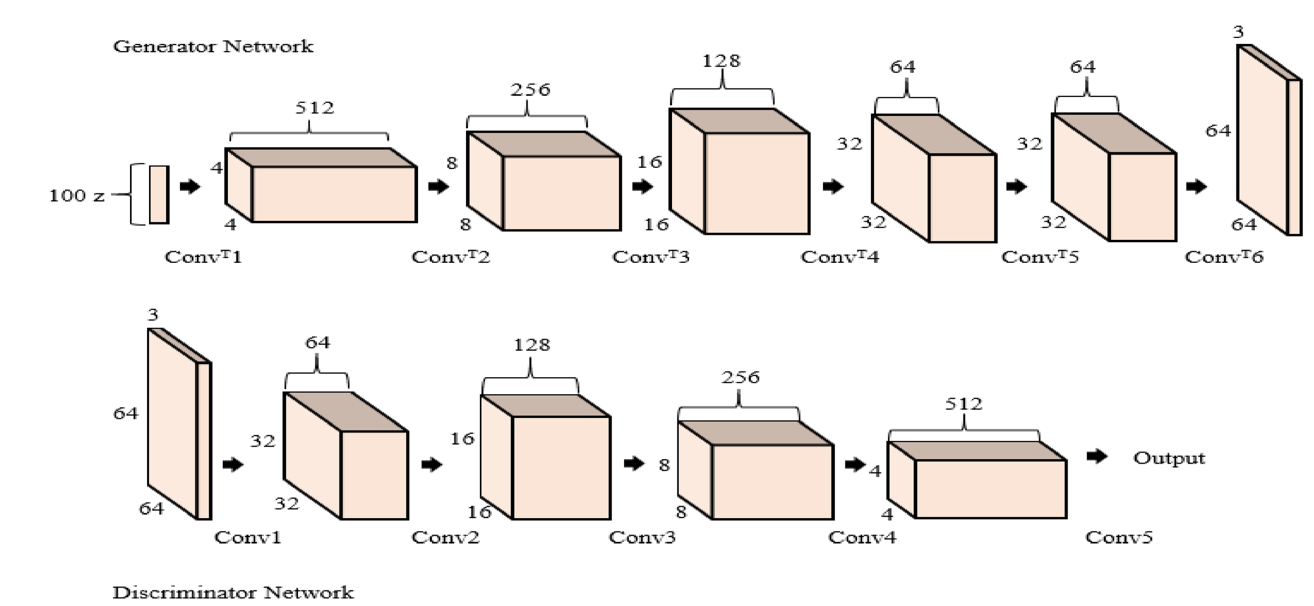
The dataset consists of ~143,000 images featuring anime characters with diverse facial attributes (hair style, eye color etc), facial orientation and background setting drawn in various artistic styles.
To obtain it we crawled Danbooru using the gallery-dl crawler. Then we processed the data by getting facial crops of these pictures using animeface to yield pictures with a uniform size 96x96 pixels.



*Sample images from dataset*

## Models

### DC-GAN




### Spectral Normalization

Constrain the spectral norm of each layer (hin → hout) to control the discriminator's Lipschitz constant. σ(A), the spectral norm of matrix A:

$$\sigma(A) := \max_{\boldsymbol{h}:\boldsymbol{h}\neq \boldsymbol{0}} \frac{\|A\boldsymbol{h}\|_2}{\|\boldsymbol{h}\|_2} = \max_{\|\boldsymbol{h}\|_2\leq 1} \|A\boldsymbol{h}\|_2,$$

### W-GAN

W-GAN attempts to stabilize training and address the mode collapse problem by minimizing an approximation of the Earth Mover (EM) distance:

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma\in\Pi(\mathbb{P}_r,\mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}\big[\,\|x-y\|\,\big]$$

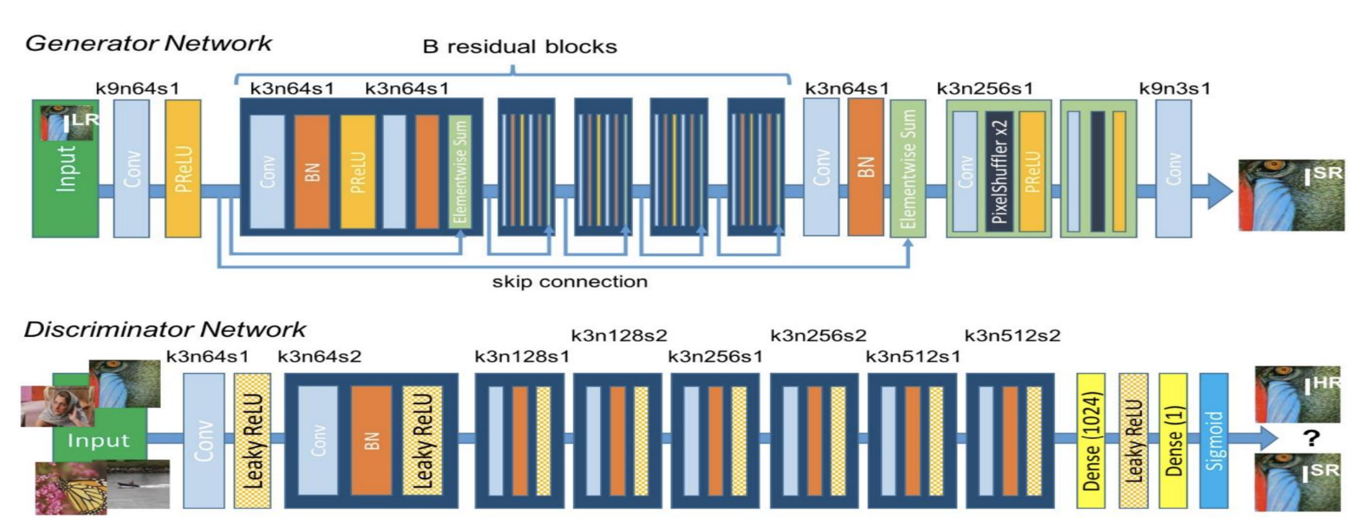
See right for complete implementation algorithm.

**Algorithm 1** WGAN, our proposed algorithm. All experiments in the paper used the default values $\alpha = 0.00005$, $c = 0.01$, $m = 64$, $n_{\text{critic}} = 5$.

**Require:** : $\alpha$, the learning rate. $c$, the clipping parameter. $m$, the batch size. $n_{\text{critic}}$, the number of iterations of the critic per generator iteration.
**Require:** : $w_0$, initial critic parameters. $\theta_0$, initial generator's parameters.

**while** $\theta$ has not converged **do**
  **for** $t = 0, ..., n_{\text{critic}}$ **do**
    Sample $\{x^{(i)}\}_{i=1}^m \sim \mathbb{P}_r$ a batch from the real data.
    Sample $\{z^{(i)}\}_{i=1}^m \sim p(z)$ a batch of prior samples.
    $g_w \leftarrow \nabla_w \left[\frac{1}{m}\sum_{i=1}^m f_w(x^{(i)}) - \frac{1}{m}\sum_{i=1}^m f_w(g_\theta(z^{(i)}))\right]$
    $w \leftarrow w + \alpha \cdot \text{RMSProp}(w, g_w)$
    $w \leftarrow \text{clip}(w, -c, c)$
  **end for**
  Sample $\{z^{(i)}\}_{i=1}^m \sim p(z)$ a batch of prior samples.
  $g_\theta \leftarrow -\nabla_\theta \frac{1}{m}\sum_{i=1}^m f_w(g_\theta(z^{(i)}))$
  $\theta \leftarrow \theta - \alpha \cdot \text{RMSProp}(\theta, g_\theta)$
**end while**

### SR-GAN




## Experimentation

### DC-GAN

| Variable | Experimentation |
|---|---|
| Optimizer | • Adam (beta1 = 0.5, beta2 = 0.999)<br>• SGD |
| Generator Conv Layers | • 6<br>• 7 |
| Learning Rate | • 0.0025<br>• 0.002<br>• 0.0015 |

### Spectral Normalization

| Variable | Experimentation |
|---|---|
| Adam Parameters | • **Beta 1 = 0**<br>• Beta 1 = 0.5<br>• **Beta 2 = 0.9**<br>• Beta 2 = 0.999 |
| Conv Layers | • G: 6 conv, D: 5 conv<br>• G: 5 conv, D: 5 conv<br>• G: 5 conv, D: 6 conv<br>• **G: 5 conv, D: 8 conv** |
| Output Size | • 64x64<br>• **32x32** |
| Learning Rate Decay | • **No decay**<br>• Decay from 0.002 to 0.0002 |

### SR-GAN

| Variable | Experimentation |
|---|---|
| Image Properties | • Brightness<br>○ 0.1<br>○ 0<br>• Contrast<br>○ 0.1<br>○ 0<br>• Saturation<br>○ 0.1<br>○ 0<br>• Hue<br>○ 0.1<br>○ 0 |
| Image Size | • 32x32 (low-res 16x16, high-res 32x32)<br>• 64x64 (low-res 32x32, high-res 64x64) |

## Results & Discussion

### DC-GAN

Almost all of the generated avatars have recognizable facial features, with hair and eyes being the most prominent. There are a couple of high quality images (~15%) where the avatar created is atheistically pleasing with smooth line and colour transitions (boxed in green). There are also a couple (~14%) of low quality images where the facial features are unrecognizable (boxed in red). There is a significant amount of blurring and noise distortion in all outputs as we expected. The avatars generated are quite diverse, with none looking very similar to another.

### Spectral Normalization

For the 64x64 outputs, the facial features are barely recognizable, there is significant distortion and noise in the generated images and the output is not diverse. Most of the images appear to have the same underlying facial structure.
For the 32x32 outputs however, we can see that performance hit its peak at around 30 epochs of training. At ~30 epochs the avatars have distinguishable facial features and there is great diversity in outputs. After 30 epochs however, performance starts to worsen. We conjecture this is because spectral normalization performs well when producing different classes of images such as horse, car etc, where one class is significantly different from each other in terms of image structure (i.e. a horse has four legs, a car is rectangular shaped). For anime faces however, the overall structure of the image is pretty similar. As thus, it only generates images from one or two classes resulting in monotonic images.

### W-GAN

For W-GAN, all of the generated avatars have recognizable facial features. There are a lot more high-quality images (~40%) where the avatar created is atheistically pleasing with smooth line and colour transitions.

### SR-GAN

SR-GAN gave us mixed results. While it performed well on the training dataset (our original dataset from Danbooru), it performed poorly when we inputted images generated from our GAN models. We hypothesized that the reason our model performed poorly on GAN generated images is that there is significantly more noise and irregularity in these images

Thus, we tampered with our ground truth images in an effort to increase their noise levels to be on par with the generated avatars. Training on a dataset with higher noise (as a result of tuning up brightness, saturation etc) however, decreased the model's training performance against real anime images but increased its performance at super resolution of our GAN generated images. Our results were still poor because generated images are significantly more noisy in a way that mere color jittering cannot replicate.

## Conclusions & Future Work

Best Model: **DC-GAN + WGAN**
Lessons Learned:
- GANs are extremely hard to train. Discriminator to generator layers ratio can often influence the result a lot.
- Having a good dataset is very important. Our results suffered a lot from the noisy dataset. When we looked into the dataset, we found out that almost 10% of the dataset do not contain a clear face.
- SR-GAN learns the transition from a blurry image to a high-res image, however, our GANs generated images are blurry in a different way than just being low-res and jittered. Thus, direct application of SR-GAN is not a good idea

Future Work:
- Try different network structures such as conditional GAN.
- Select a high quality dataset or manually prune dataset.

## Reference


Jin et al, Towards the High-quality Anime Characters Generation with Generative Adversarial Networks, 2017, https://arxiv.org/abs/1708.05509
Goodfellow et al, Generative Adversarial Nets, 2014, https://arxiv.org/pdf/1406.2661.pdf
Radford et al, Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, 2016, https://arxiv.org/pdf/1511.06434.pdf
Arjovsky et al, Wasserstein GAN, 2017, https://arxiv.org/pdf/1701.07875.pdf
Miyato et al, Spectral Normalization for Generative Adversarial Networks, 2018, https://arxiv.org/pdf/1802.05957.pdf
Ledig et al, Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, 2016, https://arxiv.org/pdf/1609.04802.pdf


* indicates the best model from all trails