COMP4471 Final Report
ANIMEGAN: Face Manipulation on Anime Illustrations

Huang Jiaxin
HKUST
jhuangbo@connect.ust.hk

Shao Yuheng
HKUST
yshaoam@connect.ust.hk

Zhao Yizhe
HKUST
yzhaocj@connect.ust.hk

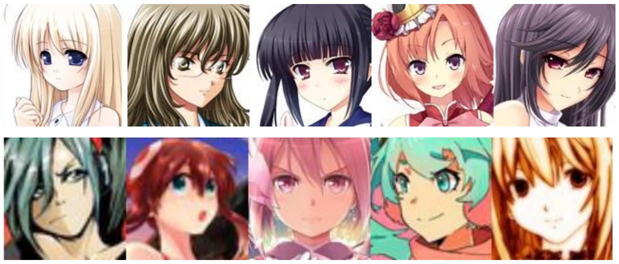

Figure 1: Random samples from our dataset (above) and existing anime dataset (below).

Abstract

Image attribute editing is well studied after GAN came out. Nice models exist on human facial editing works, but none of them are built specially for anime illustrations' attribute editing to give a promising result. In this work, we propose our ANIMEGAN which is trained on an anime illustration facial image dataset. Specifically, from the data aspect, we build a cleaner anime face dataset of higher quality. From the model aspect, we explore and conclude a feasible network architecture, ANIMEGAN that makes use of gated convolution and Selective Transfer Unit (STU). With quantitative evaluation and qualitative visual image presentation, we demonstrate that our effort leads to a well performed model for face manipulation on anime illustrations.

1. Introduction

Face attribute manipulation is an interesting and challenging task. CNN is good at extracting features from images. However, the nature of face manipulation determines the great difficulty to do supervised training. It's extremely difficult, or impossible, to get samples with different attributes from exactly the same person. Hence people have turned to unsupervised learning models, for example, GANs.

Various GAN structures have been proposed. Among the most popular structures is encoder-decoder structure, which generates high-level abstraction of the original image through downsampling. However, the downsampling often leads to relatively bad image quality.

STGAN [1] alleviates the problem by introducing Selective Transfer Units. This project is largely based on STGAN.

Japanese-style animation is also called Anime. Anime fans often imagine how their favorite characters would change at their will. However, no proper face manipulation application for Anime is available to date, probably due to insufficient and low-quality datasets. As deep learning has succeeded in human face manipulation, it's time to introduce a way by which everyone can enjoy "dressing up" their ideal anime girls.

In order to generate user-defined features from any anime girl illustration with decent accuracy and quality, we first build our own high quality and labeled anime face dataset. Then we will modify and train Generative Adversarial Network on our dataset to implement a stable and high-quality model learning from the empirical result of IcGan [2], FaderNet [3], AttGAN [4], StarGAN [5] and STGAN.

2. Related Work

Generative Adversarial Network. Since its introduction, GAN [6] has been applied in various tasks. It suffers from the model collapse problem. Efforts are made to stabilize its training, like Wasserstein GAN [7] from Ishaan et al.

Image-to-image translation. Image-to-image translation aims to build a mapping between images. Isola et al. [8] proposed pix2pix taking advantage of conditional GANs. Improvements like CycleGAN [9] uses bidirectional transfer models between two images, implementing cycle consistency loss. This loss is adopted by later face attribute manipulation GANs.

Face attribute manipulation. Face attribute manipulation has been a popular research area. cGAN [10] develops a method that generates attribute-corresponding images from noise. However, it cannot do image-to-image translation, hence is not applicable to manipulate images. Chen et al. [11] proposed an encoder-decoder CNN framework for face attribute manipulation. Li et al. [12] mixes the CNN and GAN methods to obtain separate attribute distribution. However, both [11] and [12] train individual models for different attributes, making the models resource-heavy and inextensible. IcGAN [2] was an early trial to integrate all attribute manipulation into one model. It adopts an encoder to encode image pixels, later uses a cGAN to decode and generate desired attributes. It requires that the encoder samples follow a distribution independent of attributes. FaderNet [3] also bases on the independent distribution assumption, setting focus on learning the attribute-invariant latent representations. However, this assumption is not necessary and can lead to oversmoothing. AttGAN [4] and StarGAN [5] are two similar improvements that remove the latent invariance constraint and put a output classification constraint instead. STGAN [1] introduces Selective Transfer Units (STU), which we will also be using in ANIMEGAN.

3. Data

3.1. Dataset

It is noticed that a well-organized anime face dataset is currently unavailable. Hence, we build our own dataset as follows:

Setting Image Source. Images on commercial game websites are typically of higher quality than those uploaded by individual anime fans. Higher quality is evaluated as: 1. Stable illustration standard. 2. Uniform white background. 3. Diversified illustration style. A comparison is shown in Figure 1.

Image Scraping. We scripted and downloaded over 47,000 images from over 12,000 individual websites.

Image Preprocessing. lbpcascade_animeface and OpenCV is used to identify character faces and crop them into 128 × 128 px head portraits.

Filtering and Labeling. Illustration2vec is used to label the dataset. We also move on to utilize the same tool for filtering. About 37,000 images are left after this step. random samples from our final dataset shown in Figure 1.

4. Method

This section details the network architecture of ANIMEGAN, while several design tradeoffs are also presented. Then, we introduce both quantitative and qualitative methods that we applied to evaluate the feasibility of ANIMEGAN.

4.1. Network Architecture

4.1.1. Gated Convolutional Layers

While a vanilla convolution involves all pixels in a feature map to the computation, gated convolution, proposed by Yu et al [13], adaptively select pixels that contain useful information. The formulation of gated convolutions is as follows:

$$O(x) = \phi\left(Feature(x)\right) \odot \sigma\left(Gate(x)\right)$$

Observed that anime pictures, compared to realistic photos, are typically made up of larger color blocks, each of which contains only a single color information, we adopt gated convolutions in ANIMEGAN, so that its capacity can be fully utilized in identifying useful texture information rather than colors. However, when all vanilla convolutions are replaced by gated convolutions, the output color variates in each color block, indicating that the network "forgets" that colors should be fixed in a block, which reduces the authenticity of generated images. Thus, we only apply gated convolutions to the encoder, while retaining vanilla convolutions in other parts of the network.

4.1.2. Selective Transfer Unit

Inspired by STGAN, where Selective Transfer Unit is proposed, we use a similar structure to skip-connects the encoder and the decoder. STUs adopt LSTM-like gates to selectively update its hidden state:

$$r = \sigma(W_r * [x, s_i])$$
$$z = \sigma(W_z * [x, s_i])$$
$$s_{i+1} = r \circ s_i$$
$$\hat{f} = \tanh(W_h * [x, s_{i+1}])$$
$$f = (1 - z) \circ s_{i+1} + z \circ \hat{f}$$

Where $*$ denotes a convolution and $\circ$ is an element wise multiplication. The selective transfer unit takes an encoder feature map and an attribute vector as input and provides a hint to the decoder on possible details and directions of attribute generation.

4.2. Loss Model

The loss function comprises image authenticity loss and attribute editing loss.

4.2.1. Authenticity Loss

This metric measures the generator's ability in fooling the discriminator in believing that the generated (and attributed edited) images come from the raw dataset. Authenticity loss is compatible with all GAN networks. Various loss models, including Least-Square-GAN,

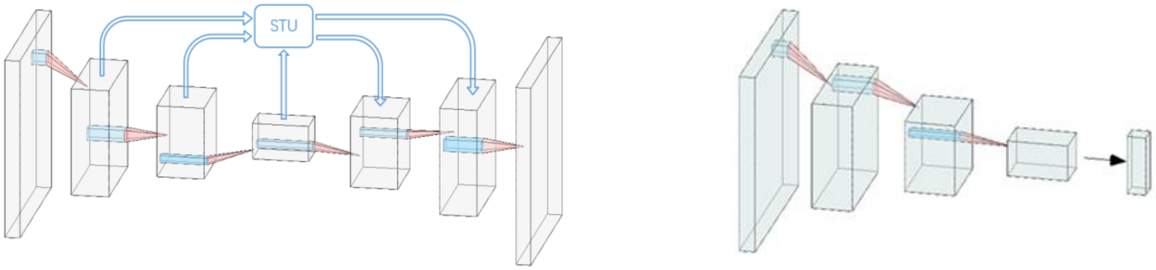

Figure 2: Left: The generator structure. Gated convolution layers and STUs are highlighted. Right: The attribute classifier structure.

Wasserstein GAN-Gradient Penalty and DRAGAN have been proposed aiming at facilitating a more stable training process so that a faster network convergence can be achieved.

In particular, Wasserstein Loss[7], proposed by Gulrajani et al., replaces Jensen-Shannon divergence by Wasserstein metric since the fact that the raw data distribution and the sample distribution do not intersect (to be precise, the intersection has measure zero) may result in a constant JS-divergence.

However, on anime data, the success of WGAN-GP cannot be easily assumed. Since we have carefully prepared our dataset, all images have a pure background and a centered head position. Further, as we are doing attribute manipulation, the presence of a source image as input prevents the network from diverging from authentic distribution too much and thus the advantage of WGAN is faded. Consequently, we anticipate ANIMEGAN to generate images that resemble raw data with less difficulty. Experiments, which are presented in the next section, agree with our expectation. Based on these analyses, we simply use the vanilla GAN loss, i.e.

$$E_{x\sim P_r}[\log D(x)] - E_{x\sim P_g}[\log(1 - D(x))]$$

And

$$E_{x\sim P_g}[-\log D(x)]$$

indicating an availability for future study and enhancement.

4.2.2. Attribute Editing Loss

Attribute editing loss measures whether the network has indeed generated desired attributes without introducing unnecessary ones. This can be done by classifying the generated image and comparing it with the input attribute vector. Note that the classifier referred to here is not a vanilla classifier that chooses a label for each image, but for each label (attribute) and each image, decides whether the image possesses that attribute. Thus the classification

result is a confidence vector with dimension being the number of total attributes we studied.

There are two possible ways to obtain an attribute classifier. We may train a CNN classifier together with the generator and discriminator, or we may append an existing one, which we have already used for dataset labeling. Realizing that the existing classifier is well encapsulated and that we cannot retrieve its analytical gradient, we will save it for benchmark analysis and train a new one.

Finally, the attribute editing loss is as follows:

$$E_{x\sim P_g}[\log |C(x) - A(x)|\ ]$$

Where $C(x)$ is the classifier and $A(x)$ is the attributes desired.

4.3. Overall Architecture

The left side of Figure 2 presents the topology of the generator of ANIMEGAN. It uses gated convolutions for the encoder and vanilla convolutions for others. STU is inserted to skip-connect corresponding layers of the same dimension.

The right side of Figure 2 presents the topology of the attribute classifier. It takes an attribute-edited image as input and produces a vector where each element indicates the confidence of possession of that attribute.

5. Experiment

In this section, we first compare three possible loss models, then analyze the overall performance of ANIMEGAN from the perspective of reconstruction and attribute editing capability.

Comparison with similar networks is also detailed.

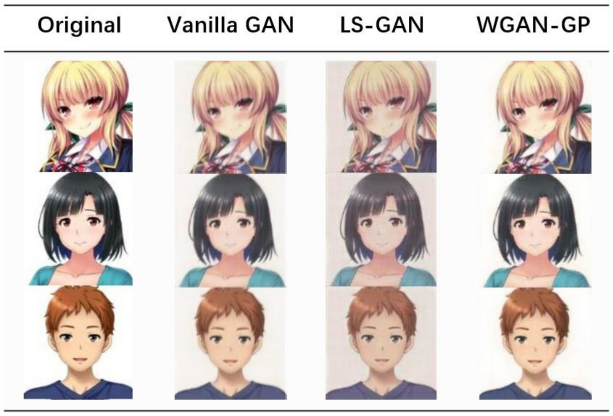

Figure 3: Comparison of WGAN-GP with two inferior GAN results.

Vanilla GAN	LS-GAN	WGAN-GP
26.72	22.49	27.88
26.23	23.41	27.35
25.94	22.86	27.10

Table 1: PSNR evaluation for three GAN networks.

5.1. Loss model

As we have noted, although WGAN-GP has achieved appealing results on general purpose datasets, e.g. LSUN bedrooms, its performance on anime attribute editing cannot be simply assumed due to several natures in such tasks. To demonstrate this, we train three networks with vanilla GAN loss, LS-GAN loss and WGAN-GP for 2,000 iterations respectively and evaluate their reconstruction capability based on visual quality as well as PSNR scores.

Figure 3 presents a few samples generated by the three networks. Observe that while images from LS-GAN are heavily polluted, vanilla GAN yields a slightly darkened background and WGAN-GP produces visually-indifferentiable results. PSNR evaluation (Table 1) agrees with this intuition.

5.2. Attribute Editing

5.2.2. Qualitative Results

As shown in Figure 5, ANIMEGAN is able to capture large facial attributes, e.g. hair, and successfully apply a change of color to it. Several interesting properties of the network are revealed from the first row. Firstly, even when its hair style differs from other samples significantly, the network is still able to accurately capture its region, suggesting that the network has high capability of spatial cognition. Further, although having a single eye may seem peculiar to a naïve classifier, the image goes through ANIMEGAN properly without being attempted to add the missing eye on. This shows that neither the discriminator overfits nor the generator simply memorizes samples seen, even when our dataset is relatively small. Having this said, we conclude that the network in general is highly extendable and allows further scaling. However, as is exhibited in the fourth row, when Purple Hair is applied, surprisingly, a blue hair is resulted. The reason is unclear but one certain thing is that the feature of hair color is not learned simply via RGB values, which, in this case, suffices and digging into more intrinsic property even does harm to it.

	PSNR	SSIM
IcGAN	15.28	0.430
FaderNet	30.62	0.908
AttGAN	24.07	0.841
StarGAN	22.80	0.819
STGAN	31.67	0.948
ANIMEGAN	28.18	0.88

Table 2: Reconstruction evaluation of GAN variants.

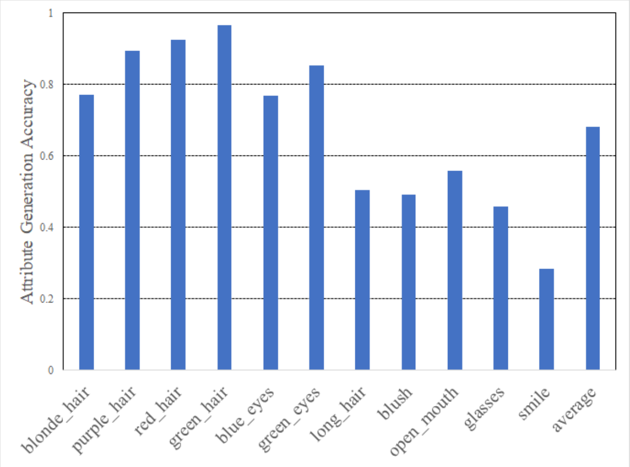

Figure 4: Re-classification accuracy over destination labels.

5.2.3. Quantitative Evaluation

The performance of attribute editing can be evaluated from two aspects, i.e., image quality and attribute generation accuracy.

As being a widely adopted metric to evaluate the visual quality of generated images, we list PSNR/SSIM results in Table 2 and compare it with IcGan, FaderNet, AttGan, StarGan and the original STGan. Benefited from the idea of difference-attribute-vector, ANIMEGAN exhibits higher reconstruction capability than IcGan, AttGan and StarGan and is comparable to STGan. This is consistent with visual intuition.

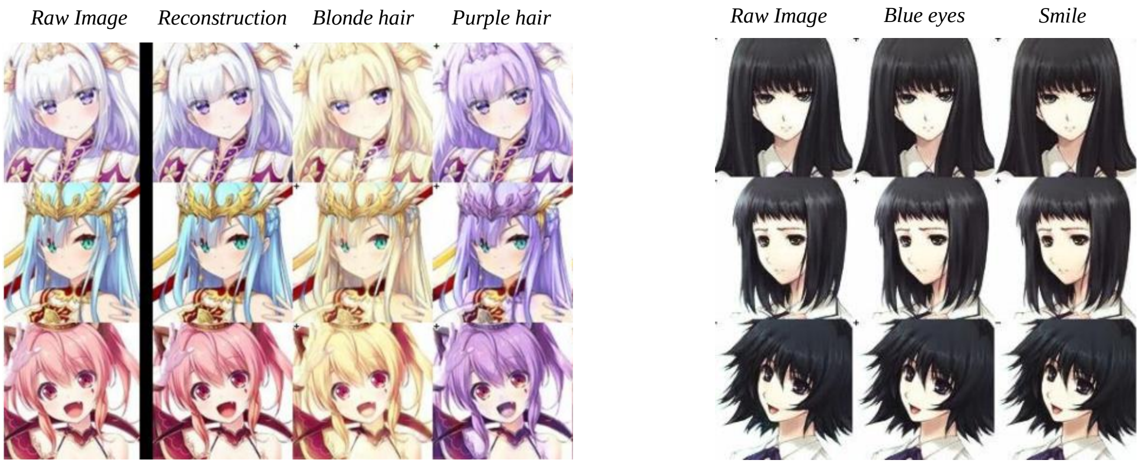

Figure 5: ANIMEGAN results for facial attribute editing.

As for attribute generation accuracy, we use Illustration2vec library, which is also used in the filtering and labeling process when building our own dataset, as the illustration attributes classifier. Then Figure 4 shows the attribute generation accuracy, i.e., accuracy of classification on the modified facial attributes. We check whether the corresponding tag of the modified facial attributes has a confidence exceeding pre-set threshold (in Figure 4, the threshold is set to be 0.5), to determine whether the classification for this attribute is correct. We can see that our ANIMEGAN performs well in hair color and some eyes color modification. For the attribute *purple hair, red hair, and green hair*, ANIMEGAN achieves over 90% accuracy. It also has reasonable accuracy in modification of more subtle and complicated features such as blush, open mouth and glasses. Although it does not perform well on some ambiguous and abstract attributes such as smile, the average accuracy obtained still exceed 70%.

6. Future Extension

As shown in Figure 5, unfortunately, some subtle attributes fail to be extracted by ANIMEGAN. They either possess a small region in image (e.g. eye color) or are relatively abstract (e.g. smile). SC-FEGAN suggests a possible solution to this problem: random masks that resemble sizes and positions of eyes and mouths are applied to raw data before feeding them into the generator. This provides an incentive to the network to focus on these subtle regions. We anticipate that ANIMEGAN can be augmented in a similar way and leave this for future work.

7. Conclusion

In this paper, we study attribute editing tasks in the anime domain. We build a new ANIME dataset with unprecedented clearness and uniformity. Then, we explore the design space of possible network models and propose ANIMEGAN which takes advantage of gated convolutions and Selective Transfer Units. We also note that WGAN-GP makes no significant performance gain. We analyze the reconstruction capability and attribute editing accuracy of our model and show that our ANIMEGAN outperforms other methods on anime face manipulation tasks.

References

[1] Ming Liu, Yukang Ding, Min Xia, Xiao Liu, Errui Ding, Wangmeng Zuo and Shilei Wen. STGAN: A Unified Selective Transfer Network for Arbitrary Image Attribute Editing. arXiv preprint *arXiv:1904.09709*, 2019.

[2] Guim Perarnau, Joost van de Weijer, Bogdan Raducanu, and Jose M Alvarez. Invertible conditional gans for image editing. arXiv preprint *arXiv:1611.06355*, 2016.

[3] Guillaume Lample, Neil Zeghidour, Nicolas Usunier, Antoine Bordes, Ludovic Denoyer, et al. Fader networks: Manipulating images by sliding attributes. In *Advances in Neural Information Processing Systems*, pages 5967–5976, 2017.

[4] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Arbitrary facial attribute editing: Only change what you want. arXiv preprint *arXiv:1711.10678*, 2017.

[5] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8789–8797, 2018.

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[7] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777, 2017.

[8] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5967–5976, 2017.

[9] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[10] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[11] Ying-Cong Chen, Huaijia Lin, Michelle Shu, Ruiyu Li, Xin Tao, Xiaoyong Shen, Yangang Ye, and Jiaya Jia. Faceletbank for fast portrait manipulation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3541–3549, 2018.

[12] Mu Li, Wangmeng Zuo, and David Zhang. Deep identity-aware transfer of facial attributes. *arXiv preprint arXiv:1610.05586*, 2016.

[13] Free-Form Image Inpainting with Gated Convolution Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, Thomas Huang. *ICCV 2019* Oral.