Selfie-Cartoonization-GAN: Generative Adversarial Networks for Hand-Painted Selfie

Jie Liu
Beihang University
China
ljie@buaa.edu.cn

Yuke Jia
Beihang University
China
16231161@buaa.edu.cn

Chenyu Yan
Beihang University
China
ycycs@buaa.edu.cn

Abstract

In this paper, we propose a solution to transforming the real-world selfies into hand-painted ones, which is able to be used as social network profile to help our friends recognize us conveniently. Our solution belongs to learning based methods, which have succeed in mapping real-world photos into cartoon style images. However, existing methods produce unsatisfactory results for hand-painting ones due to the fact that hand-paintings have clear edges, smooth color shading and relatively simple textures and the foreground and background shoule be distinguished as well as possible. In the paper, we propose a generative adversarial network framework for hand-paintings generation. Our method takes unpaired selfies and hand-paintings for training. In order to get the result not influented by background color, we add a simple but efficient U-Net to the discriminator network to distinguish foreground and background. Experimental results show that our method is able to generate satisfactory hand-paintings from real-world selfies and outperforms state-of-the-art methods.

1. Introduction

Hand-painted selfies are widely popular in our daily life. Compared to the real-world selfies, hand-painted selfies have some adventages when used to be our social network profile photos. Using hand-painted selfies as our profile photos can make it easy for our friends to recognize us by profile photos and protect our privacy from being leaked, for it is difficult for strangers to know what we look like in real world. However, manually recreating a real-world selfie in hand-painted style is very laborious and takes a lot of skill. To obtain a look-liking hand-painted selfie, artists have to draw every single line and shade to complete every detail, which takes too much time. Meanwile, existing image editing software/algorithms with standard features cannot produce satisfactory results for hand-paintings. Therefore, a specific technique that can automatically transform realworld photos to detailed hand-paintings is very useful for artists so that they may have time to do other work. Although stylizing images in an artistic manner has been widely developed, state of the art methods focused on overall style fail to produce a detailed hand-painting. Hand-paintings are simplied and abstracted from the read-world photos instead of adding some patterns and changing some color. And hand-paintings have noticeable features, such as clear edges, smooth tones and relatively simple textures, which is different from other artworks. In this paper, we propose SelfieGAN, a novel CartoonGAN-based [1] approach to transfer real-world photos to hand-paintings. Our method takes a set of real-world selfies and a set of hand-paintings for training. In order to get enough data to produce high quality results, we do not require pairing or correspondence between two sets of images. Our goal is to map the real-world selfies into the hand-painted ones and keep the appearance unchanged so that our firends can recognize us by the hand-paintings. To achieve this goal, we propose to use a dedicated CartoonGAN-based architecture together with a simple yet effective U-Net [5]. The main contributions of this paper are:
(1)We propose a dedicated GAN-based approach that effectively learns the mapping from real-world selfies to hand-painted ones using unpaired image sets for training. Our model is able to produce high quality hand-paintings, which is more detailed than state-of-the-art methods.
(2)We add a simple yet effective U-Net to the CartoonGAN, which is used for model to distinguish foregrounds and backgrounds. Before the discriminator network, we add an U-Net to get the foreground of the selfie to make the discriminator network ignore the background.

2. Related work

2.1. Nerual Style Tansfer

Convolutional Neural Networks (CNNs) [3, 4] has received considerable attention in style transformation. But for our task, hand-paintings have some noticeable features, such as clear edges, smooth tones and relatively simple textures. The network may incorrectly view red background as textures and generate the red face which we are not willing to produce.

2.2. Stylization with GANs

There are many conditional GANs used for stylization. Pix2pix [2] model is used for image-to-image translation, but needs paired data. It is difficult to train on our data sets for our work. In order to solve unpaired training sets, there is CycleGAN [8] can be used for mapping a image to another. However, CycleGAN [1] is not able to produce images with clear edges, which we think is the most important difference between real-world selfies and the hand-paintings. And CartoonGAN solves the problem to get the clear edges and get the result that saves the content of the image, but foreground and background cannot be distinguished well in the result so that the hand-paintings look like strange and awful. So our method is aimed to get this problem solved. For there have been many prior and concurrent works, our setup is considerably simpler than most others.

3. Our model

Formally, the goal of selfie cartoonization problem is to estimate a map $F_{S\to T}$ from domain A formed by real selfies to domain B formed by hand-writing cartoon portraits. The mapping function is learned using independently sampled data instances χ_S and χ_T such that the distribution of the mapped instances $F_{S\to T}(\chi_S)$ matches the target distribution P_T. Like other GAN frameworks, a generative function G produces vivid images to confuse the discriminative function D, while the optimization procedure of D aims to distinguish the real cartoon portraits in the domain T from the fake ones generated by G. Let $\mathcal{L}$ be the loss function, G^* and D^* be the weights of the networks. Our objective is to solve the min-max problem:

$$(G^*, D^*) = \arg\min_G \max_D \mathcal{L}(G, D)$$

Our starting point is Chen et al.'s CartoonGAN approach[1] which proposes a solution to transforming photos of real-world scenes into cartoon style images. We present the detail of our network architecture in Section 3.1 and propose three loss functions for G, D and S in Section 3.2. We also propose a training order which is summarized in Section 3.3.

3.1. Selective CartoonGAN architecture

Refer to Figure 1. In Selective CartoonGAN, the generator G firstly translates input selfie x_i into a cartoon portrait $G(x_i)$, then $G(x_i)$ and a true cartoon portrait from target manifold y are sent into the Segmentation network S and become $S(G(x_i))$ and $S(y)$. we apply a Unet [5] based architecture as the S network. Complementary to the generator network, the discriminator network D is used to judge whether the input image is a real cartoon portrait. We use the same G and D architecture as G and D in Chen et al.'s cartoonGAN.

3.2. Loss function

The loss function $\mathcal{L}(G, D)$ in Eq.(1) consists of three parts: (1) the adversarial loss $\mathcal{L}(G, D)$ (Section 3.2.1), which drives the generator to learn the target style;(2) the perception loss $\mathcal{L}(G, D)$ (Section 3.2.2), which ensures consistency of content; and (3) the binary cross entropy loss $\mathcal{L}_{\}}(S)$ (Section 3.2.3), which drives the segmental network to learn mask and the total network to ignore background when doing image translation. We use a simple additive form for the loss function:

$$\mathcal{L}(G, D) = \mathcal{L}(G, D) + w_1\mathcal{L}(G, D) + w_2\mathcal{L}_{\}}(S),$$

where w_1 and w_2 are the weight to balance the three given losses. Larger w_1 leads to more content information from the input images to be retained. Larger w_2 leads to more background information from the input images and background style from the target images to be ignored. In all our experiments, we set $w_1 = 0.01$ and $w_2 = 10$ which achieves a good balance of style, content and background preservation.

3.2.1 Adversarial loss $\mathcal{L}(G, D)$

The adversarial loss is applied to both networks G and D and its value indicates to what extent the output image of the generator G looks like a cartoon image. As Chen et al.[1] did in cartoonGAN, we also use the same edge-promoting adversarial loss as:

$$\begin{aligned}\mathcal{L}_{adv}(G, D) = {} & \mathbb{E}_{c_i\sim S_{\text{data}}(c)}\left[\log D\left(c_i\right)\right] \\ & + \mathbb{E}_{e_j\sim S_{\text{data}}(e)}\left[\log\left(1 - D\left(e_j\right)\right)\right] \\ & + \mathbb{E}_{p_k\sim S_{\text{data}}(p)}\left[\log\left(1 - D\left(G\left(p_k\right)\right)\right)\right]\end{aligned}$$

where $S_{\text{data}}(c)$ represents the hand-writing cartoon portraits, $S_{\text{data}}(e)$ represents images generated from $S_{\text{data}}(c)$ by removing clear edges and $S_{\text{data}}(p)$ represents the input images.

3.2.2 Perception loss $\mathcal{L}(G, D)$

In image translation area, cycleGAN is a widely used method to handle unpair training data. However, as CartoonGAN shows, the perceptions loss is also a nice way to do image translation with unpair training data. We use the high-level feature maps in the VGG network [7] pre-trained

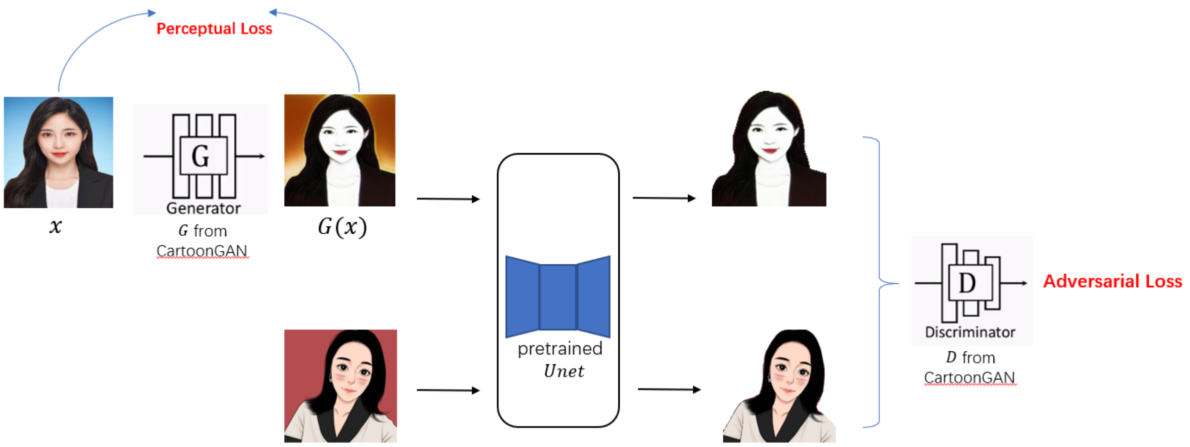


Figure 1. Our model architecture.

by [6], which has been demonstrated to have good object preservation ability. Accordingly, we define the content loss as:

$$\mathcal{L}_{con}(G, D) = \mathbb{E}_{p_i \sim S_{data}(p)} \left[\| VGG_l \left(G \left(p_i \right) \right) - VGG_l \left(p_i \right) \|_1 \right]$$

where l represents the feature maps of a specific VGG layer.

3.2.3 Binary cross entropy loss $\mathcal{L}_\}(S)$

To make the S network learn to distinguish between persons and background, we adopt the binary cross entropy loss to measure the distance from input mask to learned mask:

$$\mathcal{L}_{seg}(S) = \mathbb{E}_{p_i \sim S_{data}(p)} \left[m_i \log S(p_i) + (1 - m_i) \log(1 - S(p_i)) \right]$$

where m_i refers to the ground truth mask of p_i and $S(p_i)$ refers to the learned mask.

3.3. Training order

Firstly, we train G with perception loss for 1 epoch to only reconstructs the content of input images. Secondly, we train S with binary cross entropy loss for 20 epochs to learn distinguish between persons and background and filter out the background style information. Thirdly, we train D and G with adversarial loss to learn style.

4. Experiments

We implemented our model in TensorFlow and Python language. And some additional material is available at github[1]. All experiments were performed on an NVIDIA GTX 1080 Ti GPU.

Selfie-Cartoonization-GAN can convert a selfie to a hand-painted cartoon image, at the same time it can ignore the influence of background. To selfie transformation, the background doesn't belong to style. However, the existing methods can not avoid it.

To compare selfie-Cartoonization-GAN with the existing methods, we collected the training and test data as presented in Section 4.1. In Section 4.2, we present a comparison between the proposed method and representative stylization methods.

4.1. Data

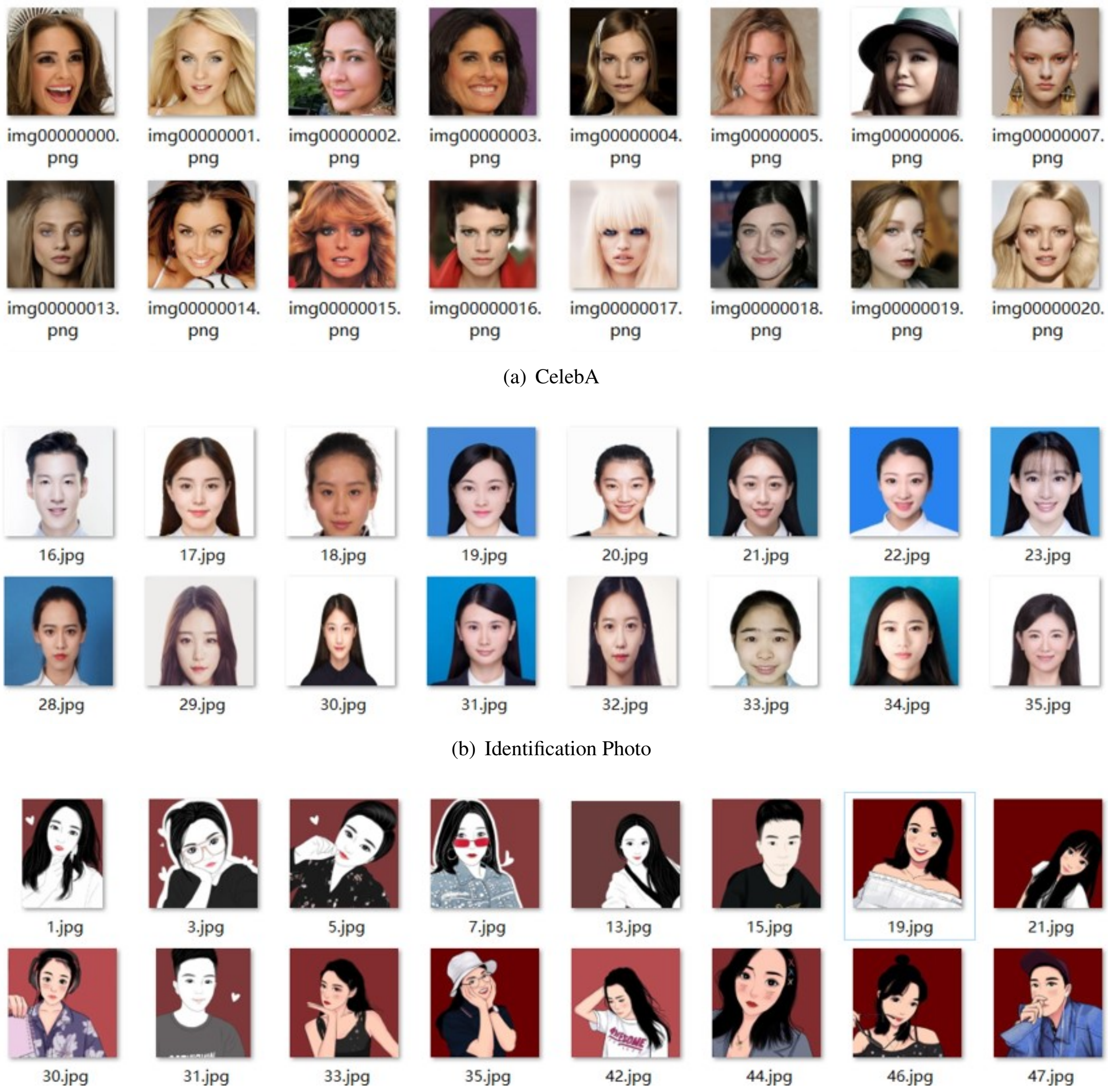
The training data contains unpaired real-world photos and hand-painted images. Because there is no open source dataset for the hand-painted selfie, our dataset is some small. And our dataset is collected by Baidu image, Google image, and Taobao comments.

Dataset	Size
CelebA	350
Identification Photo	340
Hand-painted	311

Table 1. Dataset Information

For real-world photos, we collected 2 datasets. One is CelebA, which has 350 photos of celebrities. And the other one has 340 identification photos. It's apparent that the identification photo is easier to train than CelebA. Because the hand-painted images have different styles. We select two similar style to make up the hand-painted dataset, which has 311 images. Some images of hand-painted dataset are shown in Figure 2-(c).

[1] https://github.com/yifan123/selfie-Cartoonization-GAN



(a) CelebA

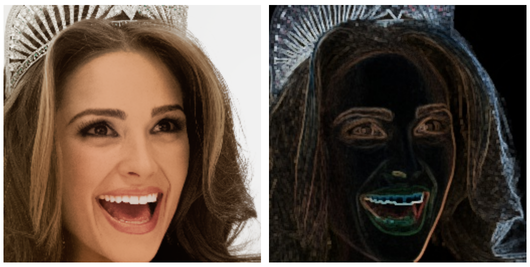
(b) Identification Photo

(c) Hand-painted

Figure 2. Some images of our datasets.

4.2. Comparison with existing methods

We first try the classic style transform method, Neural Style Transfer. NST takes one style image I_s and one content image I_c as input then extracts the style of I_s and transforms the style to I_c. Because the NST can implement the arbitrary style transform, we use a pretrained demo application of NSF to experiment with the effect of selfie transformation. The results are shown in Figure 4-(b). From the results, we find the NST mainly extracted the style of the red color of the background.

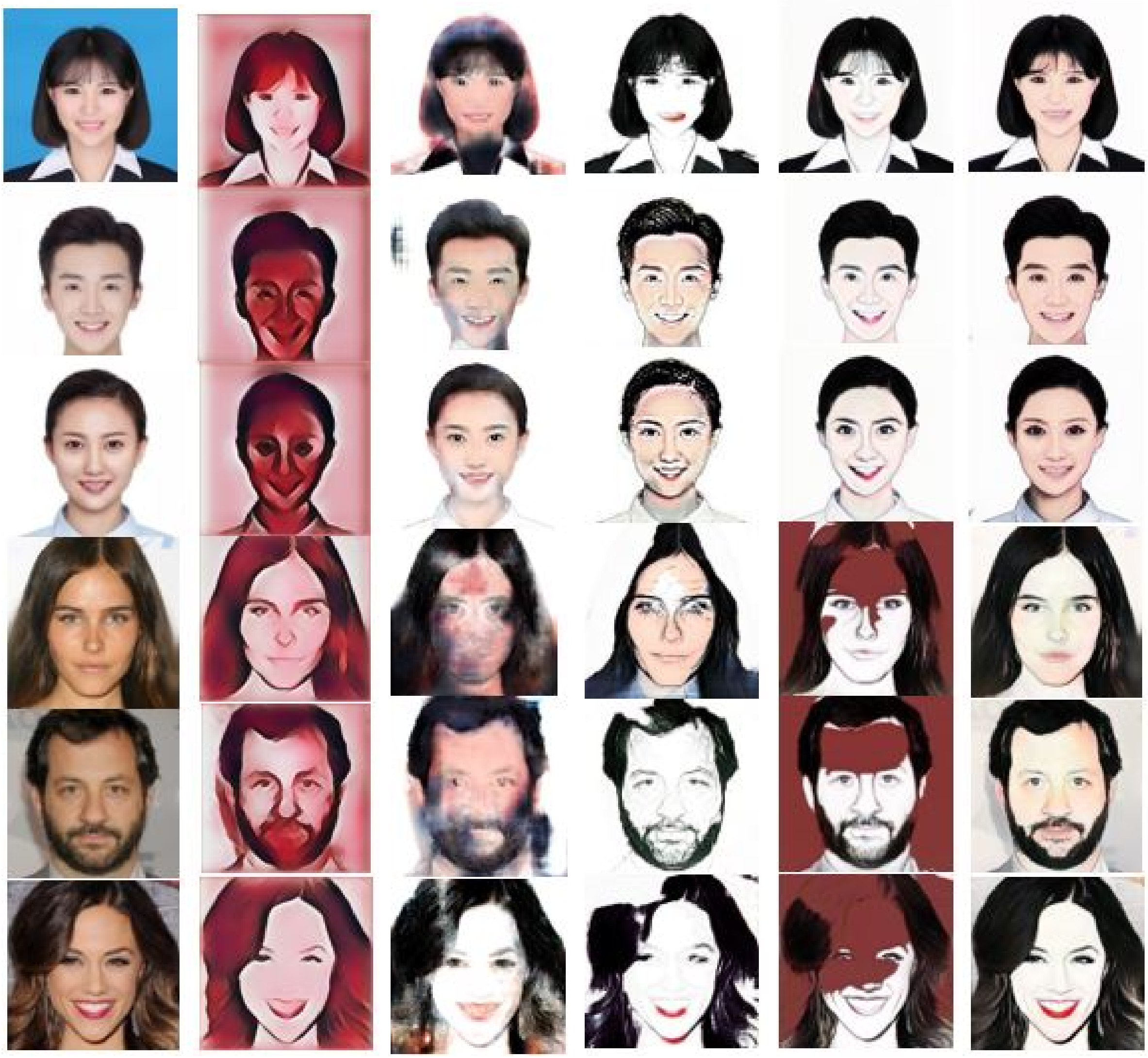


(a) Original Image

(b) Outline of the Image

Figure 3. Example of outline extraction.

Next, we compare two main current GAN, pix2pix and



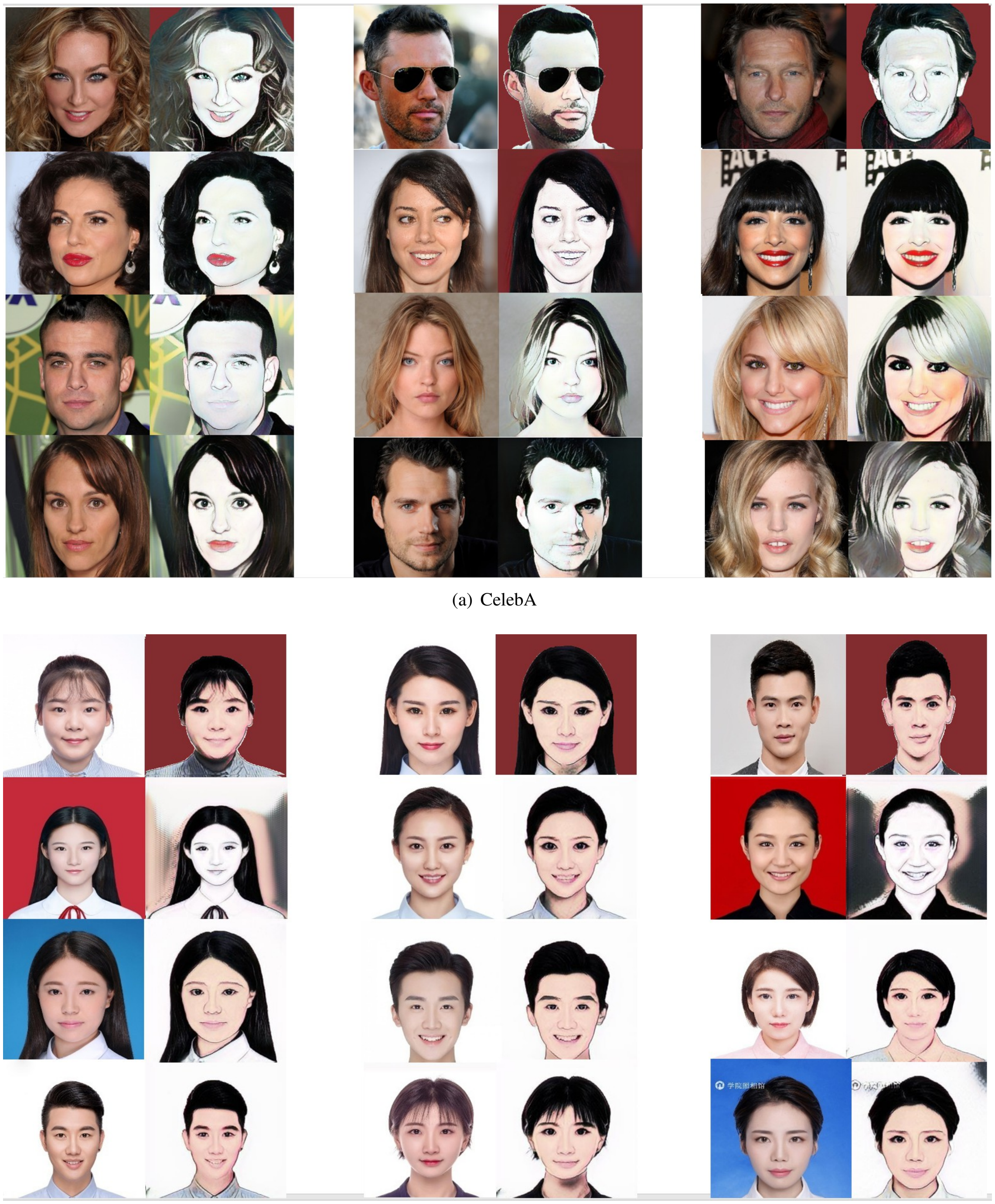
(a) Input Photo (b) Nerual Style Transfer (c) pix2pix (d) CycleGAN + $L_{identity}$ (e) CartoonGAN (f) Our Results

Figure 4. Comparison of selfie-Cartoonization-GAN and some existing methods.

CycleGAN. There is a problem that the training of pix2pix need the paired dataset[2], so we use the absolute difference between dilated image and eroded image to extract the outline. By this way, we can use the outline of hand-painted images and original images (the paired data) to train pix2pix, and in the test phase, we use the outline of test image as input. In other words, pix2pix learn to color the outline. For CycleGAN, we first train it without the identity loss[8], but we find it is unstable and hard to train cycleGAN, especially the content preservation. After adding identity loss, cycleGAN performed better. Although the main content of the output of pix2pix and cycleGAN may look not too bad, the details are terrible.

We also attempted to reduce the influence of background by replacing the background to white, the images with a different color(to neutralize the influence of color) and blurring. And some of the results became better indeed. However, the influence of the background was not disappeared completely. And it's a little tedious to replace the background.

The last one is CartoonGAN. For hand-painted selfie, presentation of clear edges is an important characteristic. Because of the loss of discriminate the edge of input[1], CartoonGAN has the best edge details among the existing methods. However, CartoonGAN still can not avoid the influence of the background. After adding a segmentation net-



(a) CelebA

(b) Identification Photo

Figure 5. Results of our model(To get the contrast effect, we change the background of the 1st row to red color by photoshop. And others are indeed directly generated by our model.)

work(Unet) between Generator and Discriminator to segment the portrait, our model can solve this problem. More results of our model are shown in Figure 5.

5. Conclusion and future work

In this paper, we proposed selfie-Cartoonization-GAN, a Generative Adversarial Network to transform selfie photos to hand-painted images. This work is motivated by

CartoonGAN. To selfie transformation, the background does not belong to style. By adding a segmentation network(Unet), selfie-Cartoonization-GAN can ignore the influence of background and produce high-quality hand-painted selfie images. And we also compare the results of selfie-Cartoonization-GAN and the existing methods. The experiments show that selfie-Cartoonization-GAN is able to learn a model that transforms the style of selfie photos. At the same time, it ignores the influence of background and keeps clear edge information.

In future work, we prepare to collect more hand-painted data and modify the facial angles of the dataset. Because of our humans' high sensibility of facial features, we aim to add attention to some important facial features, like eyes and mouth.

References

[1] Yang Chen, Yu-Kun Lai, and Yong-Jin Liu. Cartoongan: Generative adversarial networks for photo cartoonization. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[2] Phillip Isola, Jun Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. 2016.

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *International Conference on Neural Information Processing Systems*, 2012.

[4] . Lawrence, S., C L Giles, A C Tsoi, and A D Back. Face recognition: a convolutional neural-network approach. *IEEE Transactions on Neural Networks*, 8(1):98–113, 1997.

[5] Olaf Ronneberger. Invited talk: U-net convolutional networks for biomedical image segmentation. 2015.

[6] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, and Michael Bernstein. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[7] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *Computer Science*, 2014.

[8] Jun Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision*, 2017.