Final Project Report

CIS-5810 –Computer Vision & Computational Photography *December 15, 2023*

Project Team Members: Tamjid Imtiaz, Rafiz Sadique
Project Title: DeepToon: Cartoonization of Reality using Deep Learning

Abstract:

This work explores the difficult problem of creating visually appealing cartoon-style graphics from real-world scene shots, which presents a number of difficulties in the fields of computer vision and creative style transition. We note that current techniques have not been able to produce animated results that are sufficiently good. Typical problems with these techniques are that the resulting graphics lack unique animation textures, the original material is lost, and the network parameters are too demanding, taxing the memory capacity. In this work, we investigate two different architectures: styleGAN and animeGAN, the latter being particularly noteworthy as a new and lightweight generative adversarial network designed for fast animation style transfers. Our experiments show that animeGAN outperforms styleGAN in producing high-quality images. Test findings demonstrate that it can quickly convert real-world photos into high-quality anime representations, both in a new dataset and in comparison to baseline datasets.

Introduction:

Within digital media, the conversion of real-world images into cartoon representations has a distinct appeal and usefulness. Beyond just being for fun, cartoons have developed into a potent medium with a variety of uses in virtual character creation, animation, and advertising. Their capacity to visually engage and simplify complicated themes and emotions makes them appealing to people of all ages and cultural backgrounds. Seeing this potential, our research sets out to automate the cartoonization process—converting common photos and movies into visually appealing cartoons—by utilizing advances in computer vision and deep learning.

Cartoon universe generation has always been a very time-consuming process that requires a great deal of artistic talent and manual labor to either simplify or amplify features obtained from real-life settings and characters. Even while this manual intervention yields amazing outcomes, it is frequently laborious and unscalable for real-time or large-scale datasets. In order to overcome this difficulty, we used cutting-edge deep learning models, particularly AnimeGAN and StyleGAN, to automate and expedite this transformation process.

The project's creative application of deep neural networks forms its basis. A model called AnimeGAN was created expressly for the purpose of photo animation, and it excels at turning ordinary photos into anime-style graphics. Its streamlined architecture and customized loss algorithms allow it to preserve the integrity of the input while striking a balance between

creative stylization and content preservation. However, StyleGAN—which is renowned for its excellent generating powers—offers an alternative strategy. It can produce detailed and varied cartoon-like visuals while preserving the essential elements of the original inputs by adjusting latent spaces. Not only do we use these models in our work, but we also modify and refine them to meet our unique needs, like changing cartoon styles and video cartoonization. Also, we used a specific dataset which is not previously used for the cartoonization purpose.

By combining traditional image processing techniques with modern deep learning algorithms, we are not only automating an artistic process but also opening new avenues for creative expression and digital content creation. Our system is designed to cater to various digital platforms, offering tools for creators and enthusiasts to transform ordinary visuals into extraordinary cartoon-style content with ease and efficiency.

Related Works:

Image-to-image (I2I) translation in computer vision involves converting images from one domain to another, such as turning semantic maps into real images, grayscale images into color, or low-resolution images into high-resolution ones. Generative adversarial networks (GANs) have gained attention in artificial intelligence as they use a two-player zero-sum game approach. GANs consist of a generator and discriminator trained simultaneously through a min-max game. Recently, GAN-based I2I translation methods have shown promise. Isola et al. introduced "pix2pix," utilizing conditional GANs (cGANs) and U-Net neural networks for various tasks like synthesizing photos from label maps and colorizing images [4]. Wang et al. extended pix2pix to "pix2pixHD" for high-resolution photo-realistic image synthesis from semantic label maps using conditional GANs [10]. "CycleGAN" is another approach that can translate images from one domain to another without paired examples [14]. Almahairi et al. proposed "Augmented CycleGAN" to learn many-to-many mappings between domains in an unsupervised manner, producing diverse outputs for each input across different domains [1].

Style transfer methods based on GANs and convolutional neural networks (CNNs) have been widely explored and achieved impressive results. In recent years, there has been a growing interest in animation style transfer. Chen et al. introduced a GAN-based style transfer method that effectively cartoonizes photos [3]. Maciej et al. presented a solution for transforming videos into comics, involving two stages and a keyframes extraction algorithm to capture comprehensive video context [7]. Their network structure aligns with the method proposed by Chen et al. [3], and they incorporate additional training strategies.

Generative Adversarial Networks (GANs) have been extensively employed to tackle the task recently. This cartoonization task can be divided into two distinct categories based on previous research efforts. The first category pertains to scene cartoonization, while the second one relates to portrait cartoonization. In the realm of scene cartoonization, much of the prior work has revolved around the development of specialized loss functions aimed at enhancing image clarity, ultimately producing an abstract rendition of the original scene

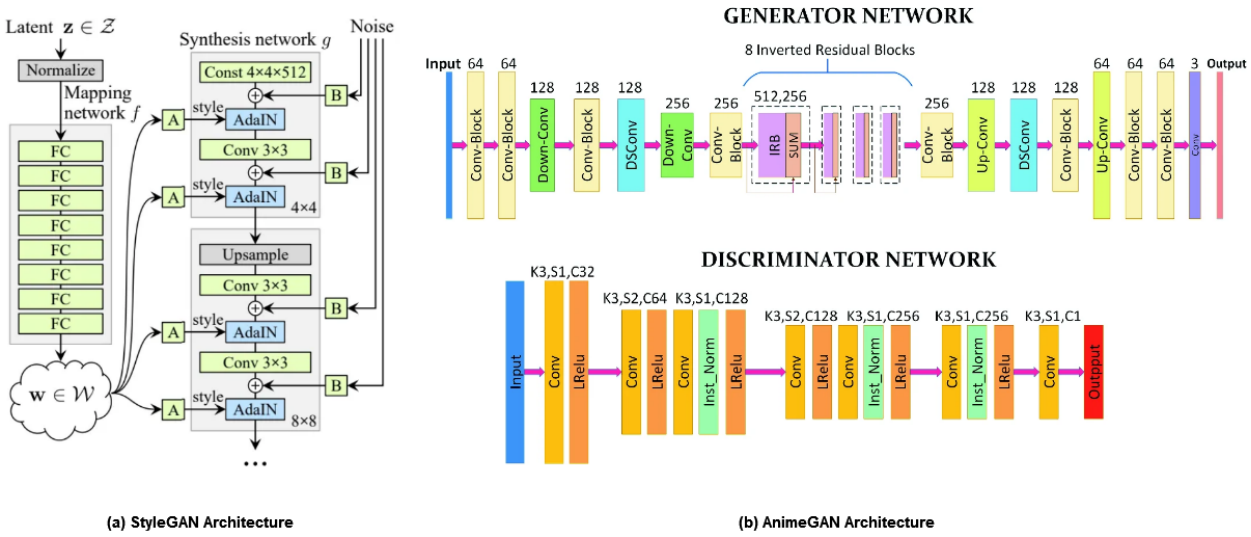

Figure 1: Network architecture of StyleGAN and AnimeGAN. Images collected from [5] and [2].

[2, 11]. Conversely, portrait cartoonization is primarily concerned with generating images reminiscent of manga or anime styles with deliberate geometric deviations from the original facial features [12, 13]. A novel multi-style generative adversarial network (GAN) is proposed in [8] where a hierarchical semantic loss with sparse regularization, edge-promoting adversarial loss, and a style loss are proposed to transform the realistic images into their cartoon form. A recently propsed novel lightweight GAN, called AnimeGAN [2], utilizes three unique loss functions to train the model. In order to make the generated images have the content of the original photos, they introduces the pre-trained VGG19 [9] as the perceptual network to obtain the L1 loss of the deep perceptual features of the generated images and original photos.

Methodology:
Our project, DeepToon, utilizes two sophisticated deep learning models, AnimeGAN and StyleGAN, to automate the cartoonization of real-world images and videos. The methodology encompasses data preparation, network architecture design, training processes, and performance evaluation, tailored to achieve high-quality cartoon-like visuals.

StyleGAN (Style Generative Adversarial Network) is a generative model architecture used for generating high-quality, realistic images. It was developed as an extension of the original GAN (Generative Adversarial Network) framework.

Style GAN employs the baseline progressive GAN architecture with specific modifications in the generator component, while the discriminator architecture closely resembles that of the baseline progressive GAN. Let's examine these architectural changes in detail which is visualized in Fig.4(a).

Style GAN utilizes the baseline progressive GAN architecture, where the size of the generated image progressively increases from a low resolution (4×4) to high resolution

(1024×1024). This is achieved by adding a new block to both the generator and discriminator models to support larger resolutions after initially training the model on smaller resolutions, ensuring stability.

In contrast to previous Baseline Progressive GAN architectures, the authors employ bi-linear sampling rather than nearest neighbor up/down sampling in both the generator and discriminator. Bi-linear sampling is implemented by applying a separable 2nd order binomial filter for low pass filtering after each upsampling layer and before each downsampling layer.

The mapping network's objective is to transform the input latent vector into an intermediate vector, where different elements control various visual features. Instead of directly providing the latent vector to the input layer, a mapping is employed. In this paper, a latent vector (z) of size 512 is mapped to another 512-dimensional vector (w) using an 8-layer MLP (8 fully connected layers). The output of the mapping network (w) undergoes a learned affine transformation (A) before passing into the synthesis network, which employs the AdaIN (Adaptive Instance Normalization) module to convert the encoded mapping into the generated image.

The input to AdaIN is represented as $y = (y_s, y_b)$, generated by applying A to w. The AdaIN operation is defined by the following equation:

$$AdaIN(x_i, y) = y_{s,i} \left(\frac{(x_i - \mu_i)}{\sigma_i} \right) + y_{b,i} \tag{1}$$

Here, each feature map x is normalized separately, scaled, and biased using the corresponding scalar components from style y. Consequently, the dimensionality of y is twice the number of feature maps (x) on that layer. The synthesis network consists of 18 convolutional layers, with 2 for each of the resolutions (4×4 to 1024×1024).

Unlike previous style transfer models that use random input to create the initial latent code of the generator, Style GAN replaces the initial input with a constant matrix of size 4×4×512. This change contributes to improved network performance.

Gaussian noise (represented as B) is added to each activation map before the AdaIN operations. A distinct noise sample is generated for each block and scaled based on the layer's scaling factors.

Style GAN employs an intermediate vector at each level of the synthesis network, potentially causing the network to learn correlations between different levels. To mitigate this, the model randomly selects two input vectors (z_1 and z_2) and generates intermediate vectors (w_1 and w_2) for them. It then trains some levels with the first vector and switches (at a random split point) to the other vector to train the remaining levels. This randomized split point ensures that the network does not overly learn correlations.

On the other hand, AnimeGAN comprises two convolutional neural networks: one is

the generator G, responsible for converting real-world scene photos into anime-style images, while the other is the discriminator D, tasked with distinguishing between real target domain images and those generated by the generator. You can see the architecture of AnimeGAN in Fig.4(b).

Fig.4(b) depicts AnimeGAN's generator as a symmetrical encoder-decoder network with standard and depthwise separable convolutions, inverted residual blocks, and upsampling/downsampling modules. Its final convolutional layer lacks normalization but includes tanh activation. The inverted residual block also integrates depthwise and pointwise convolutions.

Eight consecutive and identical IRBs are employed in the generator's midsection to effectively reduce parameter count. These IRBs, with pointwise convolutions of 512 and 256 kernels and a depthwise convolution of 512 kernels, significantly lower parameter numbers and computational demands compared to standard residual blocks. Notably, the generator's last convolution layer omits an activation function. The discriminator network, mirroring the generator, adopts the architecture referenced in literature [3], as shown in Fig.4(b). It utilizes standard convolutions in all layers, with spectral normalization [6] applied to each layer's weight for enhanced training stability.

To enhance AnimeGAN's training stability for photo animation, it employs a least squares loss function from LSGAN. The generator's total loss function $L(G, D)$ is defined as:

$$L(G, D) = \omega_{adv} L_{adv}(G, D) + \omega_{con} L_{con}(G, D) + \omega_{gra} L_{gra}(G, D) + \omega_{col} L_{col}(G, D) \quad (2)$$

where:

- $L_{adv}(G, D)$ is the adversarial loss for animation transformation.
- $L_{con}(G, D)$ is the content loss for preserving the input photo content.
- $L_{gra}(G, D)$ is the grayscale style loss for imparting clear anime style textures.
- $L_{col}(G, D)$ is the color reconstruction loss for maintaining original colors.

The weights ω_{adv}, ω_{con}, ω_{gra}, and ω_{col} balance the style and content.

The content loss $L_{con}(G, D)$ uses high-level features extracted by VGG19:

$$L_{con}(G, D) = E_{p_i \sim S_{data}(p)} \left[\| VGG_l(p_i) - VGG_l(G(p_i)) \|_1 \right] \quad (3)$$

The grayscale style loss $L_{gra}(G, D)$ compares the Gram matrix of the features:

$$L_{gra}(G, D) = E_{p_i \sim S_{data}(p), E_{j \sim S_{data}(j)}} [\|Gram(VGG_l(G(p_i))) - Gram(VGG_l(x_i))\|_1] \quad (4)$$

Color reconstruction loss $L_{col}(G, D)$ in YUV color space is defined as:

$$L_{col}(G, D) = E_{p_i \sim S_{data}(p)} [\|Y(G(p_i)) - Y(p_i)\|_1 + \|U(G(p_i)) - U(p_i)\|_H + \|V(G(p_i)) - V(p_i)\|_H] \quad (5)$$

This structured approach with specific loss functions and equations enables the generation of high-quality anime-style images while preserving the content of the original photos.

Experiments and results:

Dataset: In this project, we utilized a dataset sourced from the paper by Chen et al. (2020) [2]. The dataset comprises two main components: real-world photos serving as content images and anime images serving as style images for training. The resolution of all training images was standardized to 256 x 256 pixels. To train our model, we employed 6656 real-world photos as content images, which were previously used in training the CycleGAN model [14]. For the style images, we adopted a unique approach. To capture distinct animation styles from various artists, we selected key frames from animated films created and directed by specific artists. Specifically, we used 1792 animation frames from Miyazaki Hayao's film "The Wind Rises" for training the Miyazaki Hayao style model, 1650 animation frames from Makoto Shinkai's "Your Name" for the Makoto Shinkai style model, and 1553 animation frames from Kon Satoshi's "Paprika" for the Kon Satoshi style model. Additionally, we used two independent publicly available dataset selfie2anime [1]and Dragon Ball z dataset[2] which is available to Kaggle.

Experimental Setup: AnimeGAN is designed for simple end-to-end training using unpaired datasets. Due to the highly nonlinear nature of GAN models and the tendency for random initializations to lead to suboptimal local minima, we pre-train the generator for improved convergence speed. This pre-training is conducted using only the content loss for a single epoch, with an initial learning rate of 0.0001. For styleGAN, an adam optimizer with initial learning rate 0.0001 is selected. For both the network, the batch size is limited to 2.

Result: In order to compare the To evaluate the performance of the StyleGAN and AnimeGAN architectures, researchers employed the selfie2anime dataset, which is specifically designed for translating selfie photographs into anime-style images. The qualitative results of this experiment are illustrated in Fig.2, where a visual comparison of the outputs from both the StyleGAN and AnimeGAN generators is presented.

In Fig.2, it is observable that the StyleGAN architecture, while capable of generating images with a cartoonish essence, significantly alters the original content. This distortion

[1]https://www.kaggle.com/datasets/arnaud58/selfie2anime/data
[2]https://www.kaggle.com/datasets/insaiyancvk/dragon-ball-z-dataset

often results in the loss of the defining characteristics of the input images, suggesting a compromise of content fidelity in the pursuit of a new style.

Conversely, the AnimeGAN architecture demonstrates a remarkable ability to transform facial images while adhering closely to the distinct cartoon styles exemplified by three renowned animation styles: Hayao, Paprika, and Shinkai. The AnimeGAN not only retains the structural integrity and recognizable features of the original images but also skillfully applies the stylistic elements associated with each of the three animation styles. This suggests a more nuanced understanding and application of style features by AnimeGAN, leading to visually pleasing and content-accurate stylized portraits. Similar superior performance is seen on the landscape images shown in Fig.3.

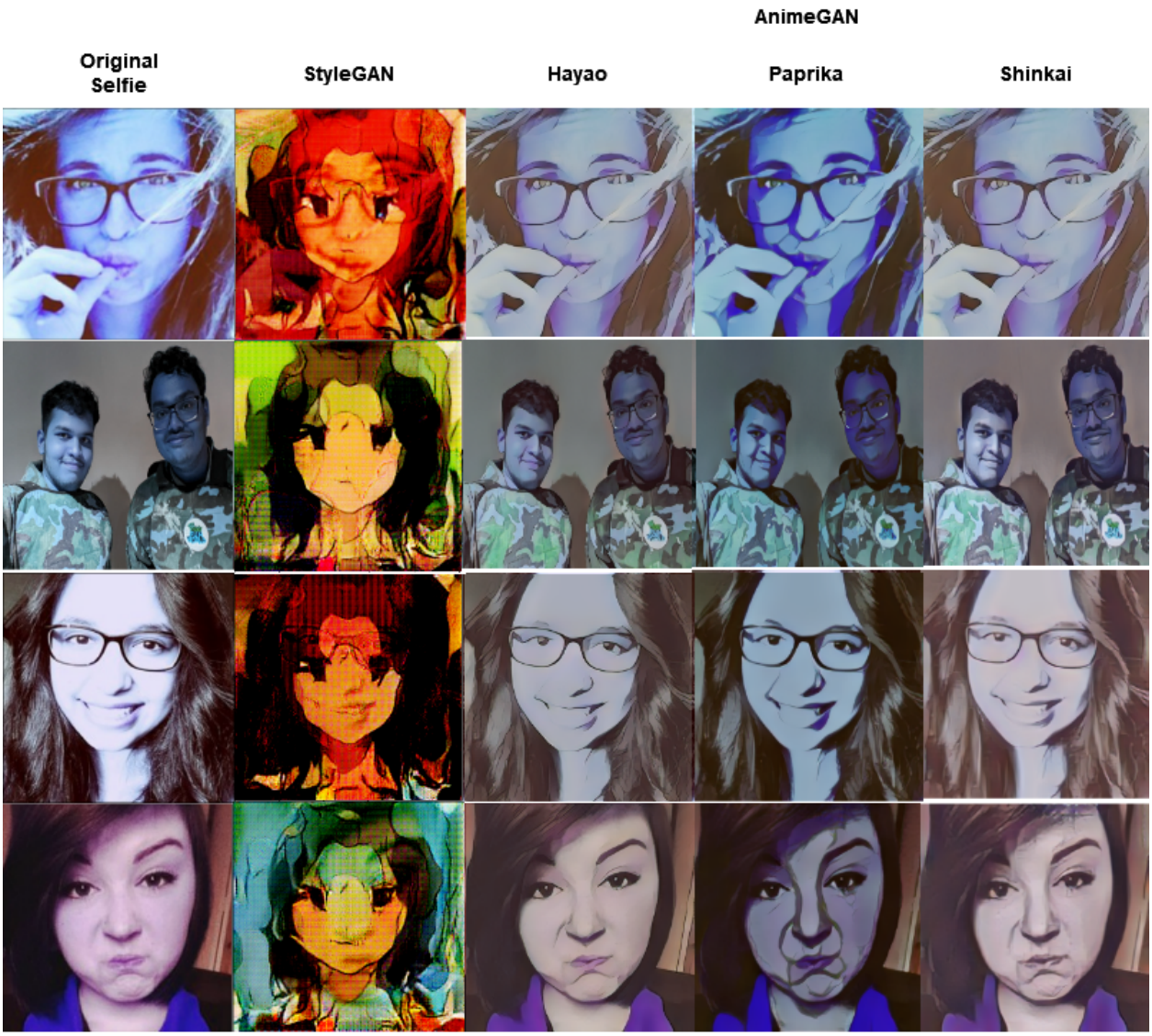

Figure 2: Performance of styleGAN and animeGAN in selfie2anime dataset

Qualitative analysis:

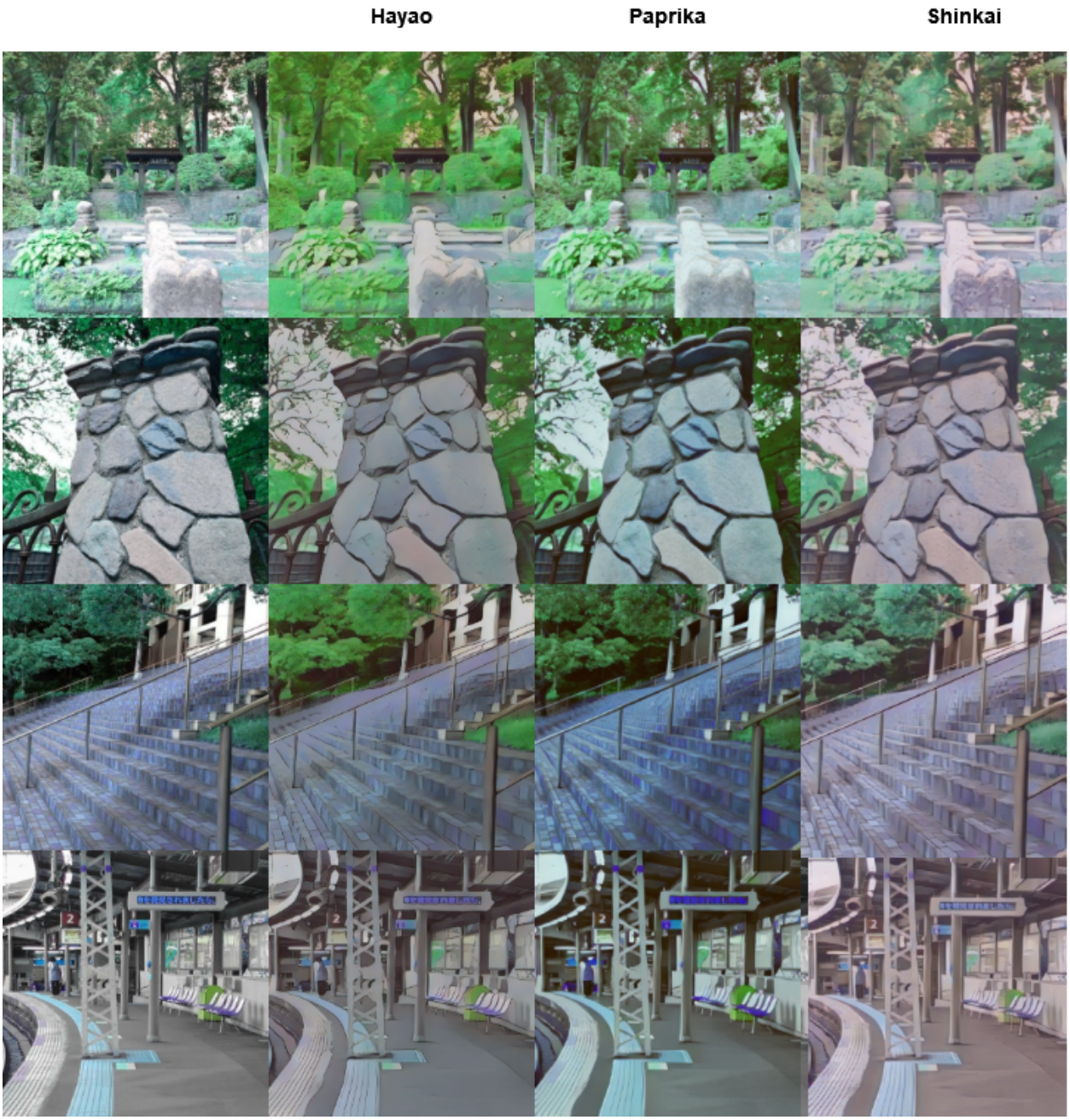

Figure 3: Network architecture of StyleGAN and AnimeGAN. Images collected from [5] and [2].

To evaluate the network's ability to qualitatively transform real photographs into cartoons, we utilized the Dragon Ball Z dataset. Visual inspection of the real photos adapted to Dragon Ball Z style revealed that the StyleGAN architecture tends to distort images similarly to previous observations. Conversely, the AnimeGAN architecture yielded superior results, effectively retaining the distinctive style of the Dragon Ball Z cartoons.

This project offers several advantages. Transforming real photos into cartoon images

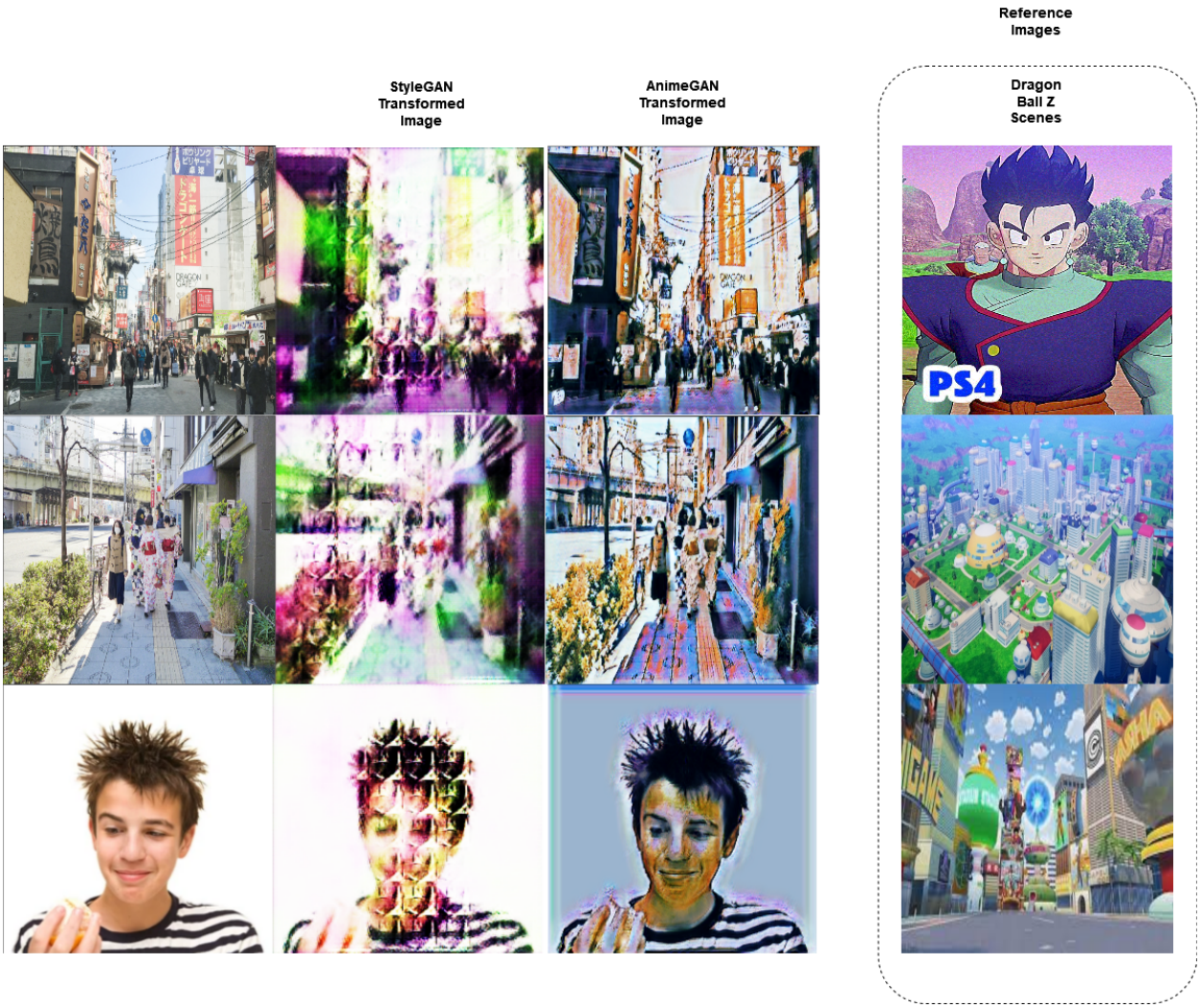

Figure 4: Network architecture of StyleGAN and AnimeGAN. Images collected from [5] and [2].

provides a new avenue for artistic expression, enabling individuals to envision themselves and their surroundings in a unique, stylized manner. In the realms of animation and gaming, this technology can expedite character creation and afford a more personalized experience. It also has the potential to capture and present cultural dress and events in a stylized manner, which may appeal to future generations. However, alongside these benefits come privacy concerns. The storage of real photographs for conversion poses a risk of data breaches, possibly leading to the exposure of private images. Moreover, if the cartoon renderings closely resemble the originals, there's a possibility of reverse engineering to uncover the subject's identity. Unconsented cartoonization of individuals raises ethical issues, especially if the resulting images are disseminated or commercialized without permission.

Conclusions:

This project investigates the complex process of transforming real-life scene photos into

enthralling cartoon-style imagery, which presents substantial challenges in computer vision and artistic stylization. Our method innovatively combines neural style transfer with generative adversarial networks (GANs) to address these issues. We've noted that previous techniques often don't produce satisfactory cartoon-like results, struggling with generating distinctive animated textures, preserving original content, and requiring extensive network resources. Our analysis focuses on two main architectures: styleGAN and animeGAN. The latter emerges as an efficient, innovative GAN specifically tailored for quick and effective animation style conversions. Through our experiments, we've found that animeGAN is superior in crafting high-definition images that convincingly turn real-world photos into premium anime renditions, showcasing its prowess both on a new dataset and against established benchmarks.

References

[1] A. Almahairi, S. Rajeshwar, A. Sordoni, P. Bachman, and A. Courville. Augmented cyclegan: Learning many-to-many mappings from unpaired data. In *International conference on machine learning*, pages 195–204. PMLR, 2018.

[2] J. Chen, G. Liu, and X. Chen. AnimeGAN: a novel lightweight gan for photo animation. In *International symposium on intelligence computation and applications*, pages 242–256. Springer, 2020.

[3] Y. Chen, Y.-K. Lai, and Y.-J. Liu. Cartoongan: Generative adversarial networks for photo cartoonization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9465–9474, 2018.

[4] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[5] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[6] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

[7] M. Pęśko, A. Svystun, P. Andruszkiewicz, P. Rokita, and T. Trzciński. Comixify: Transform video into comics. *Fundamenta Informaticae*, 168(2-4):311–333, 2019.

[8] Y. Shu, R. Yi, M. Xia, Z. Ye, W. Zhao, Y. Chen, Y.-K. Lai, and Y.-J. Liu. GAN-based multi-style photo cartoonization. *IEEE Transactions on Visualization and Computer Graphics*, 28(10):3376–3390, 2022.

[9] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[10] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[11] X. Wang and J. Yu. Learning to cartoonize using white-box cartoon representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8090–8099, 2020.

[12] R. Yi, Y.-J. Liu, Y.-K. Lai, and P. L. Rosin. ApdrawingGAN: Generating artistic portrait drawings from face photos with hierarchical GANs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10743–10752, 2019.

[13] R. Yi, Y.-J. Liu, Y.-K. Lai, and P. L. Rosin. Unpaired portrait drawing generation via asymmetric cycle mapping. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8217–8225, 2020.

[14] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.