Multimedia Tools and Applications
https://doi.org/10.1007/s11042-023-14853-9

Abstraction-perception preserving cartoon face synthesis

Sy-Tuyen Ho[1] · Manh-Khanh Ngo Huu[1] · Thanh-Danh Nguyen[1] · Nguyen Phan[1] · Vinh-Tiep Nguyen[1] · Thanh Duc Ngo[1] · Duy-Dinh Le[1] · Tam V. Nguyen[2]

Received: 25 October 2021 / Revised: 12 August 2022 / Accepted: 6 February 2023

Abstract
Portrait cartoonization aims at translating a portrait image to its cartoon version, which guarantees two conditions, namely, reducing textural details and synthesizing cartoon facial features (e.g., big eyes or line-drawing nose). To address this problem, we propose a two-stage training scheme based on GAN, which is powerful for stylization problems. The abstraction stage with a novel abstractive loss is used to reduce textural details. Meanwhile, the perception stage is adopted to synthesize cartoon facial features. To comprehensively evaluate the proposed method and other state-of-the-art methods for portrait cartoonization, we contribute a new challenging large-scale dataset named *CartoonFace10K*. In addition, we find that the popular metric FID focuses on the target style yet ignores the preservation of the input image content. We thus introduce a novel metric FISI, which compromises FID and SSIM to focus on both target features and retaining input content. Quantitative and qualitative results demonstrate that our proposed method outperforms other state-of-the-art methods.

Keywords Cartoon face synthesis · Generative adversarial network · Neural style transfer

1 Introduction

Cartoon is a popular form of artistic entertainment productions, especially to young adults. Hence, the cartoonizing portrait image problem attracts not only artists but also researchers in the computer science community. Compared with the manual work in the past, this task is automatically handled with the help of computer vision techniques. There is a wide range of real-world applications using cartoonized portrait images, for example, on-line game characters customization, face filters for social networks, or cartoon and comic production support.

✉ Tam V. Nguyen
tamnguyen@udayton.edu

[1] University of Information Technology, VNU-HCM, Ho Chi Minh City, Vietnam

[2] University of Dayton, Dayton, OH, USA

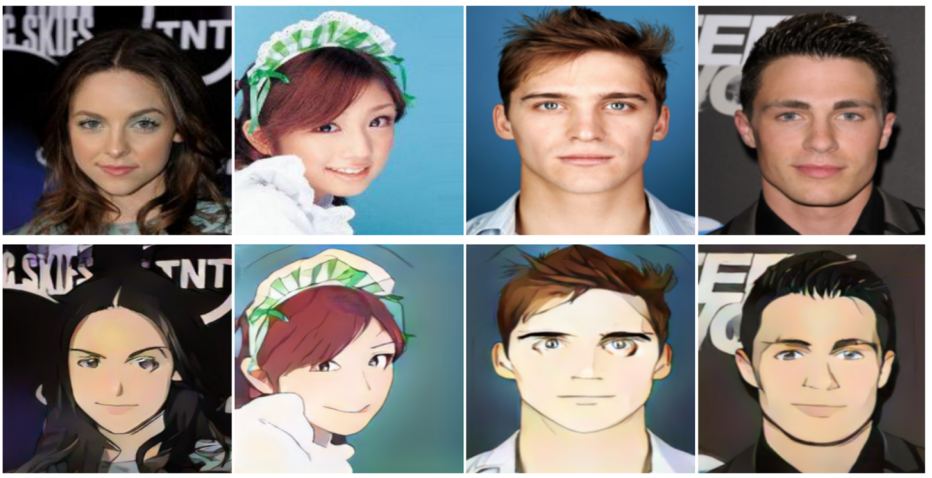

Fig. 1 Given input portrait images (top), our approach can automatically cartoonize them into their cartoon version (bottom) with cartoon facial features while preserving the identity

From the observation and analyzing cartoon portrait images, we propose a novel framework for portrait cartoonization satisfying both two criteria (Fig. 1), namely, (1) simplifying texture by color quantization and (2) synthesizing cartoon facial features (e.g., big eyes or line-drawing nose). To achieve these criteria, an artist usually spends many hours [26] with professional skills and support softwares. On the contrary, our portrait cartoonization method is beneficial for reducing time and allowing everyone to create their own artwork automatically.

Image stylization has been studied in Non-Photorealistic Rendering (NPR) [10] and in Image Analogies (IA) [16], which are not involved with deep learning. To apply such methods, we have to sophisticatedly design a pipeline for each style. Nonetheless, both NPR and IA only use low-level features, which is difficult for synthesizing cartoon facial features.

Recently, Convolutional Neural Networks [13] (CNN) has become an emergent solution to domain transfer problem [6, 20, 21, 30, 45]. To be specific, Neural Style Transfer (NST) [9] is the first example-based deep learning method that allows users to automatically transfer image style. NST and other upgraded versions [8, 18, 21] are designed for general cases and demand a manually defined style loss, which is unable to cartoonize face style. Based on Generative Adversarial Network [12], the authors in [20, 41, 46] proposed various methods to learn a mapping function between distributions of the source domain and target domain without a defined fixed style loss. However, these methods require paired datasets, which are expensive or unavailable in many domain translation tasks. To deal with un-paired dataset, several methods have been proposed in [2, 22, 23, 30, 32, 42, 45]. Especially, CartoonGAN [5] and [38] are two specific systems for cartoonizing real-world image. However, the result does not satisfy the requirement of synthesizing cartoon facial features because the previous methods mainly focused on edges, texture, or surface. Besides, the popular metrics such as Inception Score (IS) [36], Frechet Inception Distance (FID) [17], and Kernel Inception Distance (KID) [3] for evaluating an unsupervised image-to-image translation system only take the similarity of actual targets and produced features into account while ignoring the preservation of the global structure.

Before going deeper into our work, we provide Table 1 for an easier reference for important abbreviations.

To address these above-mentioned problems, we first design a two-stage training process to cartoonize portrait images. Based on the two aforementioned criteria, we separate our training into two stages: (1) an abstraction stage to reduce detailed textures and (2) a perception stage to not only simplify image but also synthesize the facial features to cartoon domain. Then, we collect a new challenging large-scale dataset to comprehensively evaluate the proposed method and the other state-of-the-art methods for portrait cartoonization. Furthermore, we propose a new performance metric, which compromises both FID and SSIM [39]. In summary, the main contributions of this paper are three-fold:

- We introduce a two-stage training scheme, which is adopted by two remarkable changes in a cartoonized portrait image compared to the real-world one.
- We propose a new evaluation metric dubbed FISI by integrating FID and SSIM metrics for the cartoonizing portrait images problem in particular and the unsupervised image-to-image problems in general.
- We collect the *CartoonFace10K* dataset, which includes 10,000 images per domain. To the best of our knowledge, in term of gender balance, this is the largest dataset for cartoon face synthesis.

2 Related work

2.1 NPR-based methods

Non-photorealistic rendering [10] (NPR) was the first approach to solve the style transfer problem in the computer graphics community. Specifically, NPR algorithms were adopted for 2D art-work or image-based-artistic rendering (IB-AR) [25]. IB-AR consists of stroke-based rendering (SBR) [15] and region-based rendering (RBR). SBR is a process to replace brush-strokes till obtaining desired style while RBR first segments images to different parts and then adjusts content in each part to complete the style transfer task. Kolliopoulos [24] and Gooch et al. [11] combined SRB and RBR to generate images with suitable brush-strokes in semantic parts. NPR-based methods do not require many resources like deep learning approaches and give remarkable results, especially with color-paint or oil-paint. Some NPR algorithms [28, 35] had developed for cartoonizing portrait images. However, they require a sophisticated system design and only focus on texture transferring.

Table 1 List of important abbreviations within our paper

Abbreviation	Description
GAN	Generative Adversarial Networks, a deep learning based generative model
FID	Frechet Inception Distance, a metrics that measures the similar between real and generated images by feature vectors
SSIM	Structural Similarity Index, a metrics that measures the similarity between two given images based on three key features of an image including Luminance, Contrast, and Structure
FISI	F-score of Inception and Similarity, our proposed metrics for portrait cartoonization problem. FISI takes into account the advantages of both SSIM and FID

2.2 Style transfer using CNN

In recent years, Convolution Neural Network (CNN) has produced state-of-the-art results in many computer vision problems, including the automatic image stylization. Neural Style Transfer [9] (NST) was the first CNN-based model for the style transfer problem. Given content and style images as input, NST optimizes the high-level features extracted from the VGG [37] network of the resulting image with content and style images through content loss and style loss, respectively. Later, FastNST model [21] improved the speed of NST by updating a feed-forward network instead of updating the resulting image directly. However, FastNST [21] only transforms a single style. Therefore, [18] introduced adaptive instance normalization (AdaIN) to optimize the style of output image through the mean and variance of them. However, the demand of manually defining the style loss is a big concern when applying NST-based methods to the cartoonizing portrait images problem. It is impossible to define a style loss that satisfies criteria (1) and (2), which are mentioned in the first part of the Introduction.

2.3 GAN-based methods

GAN currently is an emerging topic in the group of generative models. Basically, a GAN framework consists of a generator and a discriminator. At first, both of them are bad at generating and distinguishing samples. During the training, they automatically learn to improve their ability by solving a min-max game. GAN has a wide range of applications in many computer vision problems, especially in image-to-image translation.

The group of the image-to-image problems initialized by the study of [20], which is based on a conditional GAN. The extended versions [40, 41] solve the problem of video resolution and conversion, respectively. Zhu et al. [46] deals with the multi-domain image-to-image translation problem. However, due to the requirement of a paired dataset, this group of models is limited to many practical image-to-image translation problems.

To overcome the lack of paired dataset, CycleGAN [45], DualGAN [42], and DiscoGAN [22] use a cycle-consistency loss to constrain the similarity of semantic content between input and output. Besides, UNIT model [30] assumes a latent space, where a translated image and the original one should map to the same latent code. Other works [6, 7, 19, 27] expand the previous methods to handle multi-domain translation. Based on CycleGAN [45], UGATIT [23] addresses the transformation between two domains with a large difference (i.e. cat-to-dog). The authors integrate Class Activation Mapping [44] into both generator and discriminator to increase the weight of important features. However, a bijection relationship is required between two domains when using a cycle-consistency hypothesis. CUT [32] allows one-side translation without cycle-consistency loss by maximizing the mutual information between input and corresponding patches of the output. More recently, [43] formulates the image-to-image translation problem under the contrastive learning. Concretely, [43] introduces a versatile metric called MoNCE, which facilitates the contrastive learning from informative negative samples. More specifically, CartoonGAN and [38] focused on the cartoonizing real-world photo problem. The main difference between the two models with generic image-to-image models is the design of distinctive loss functions for cartoon domain. However, CartoonGAN and [38] struggle in transforming facial features to cartoon domain when being applied to the problem of cartoonizing portrait images. This is due to the difficulty in training a GAN network.

In this paper, we propose a GAN-based method with a two-stage training scheme, which simulates the two biggest changes in cartoonized portrait image: reducing textural details and synthesizing cartoon facial features (e.g., big eyes or line-drawing nose).

3 Proposed method

3.1 Data collection

In this work, our main task is to transfer images between the cartoon human portrait domain and the real human portrait domain. Thus, we are in need of the two image sets corresponding to the two aforementioned domains. To the best of our knowledge, *selfie2anime* [23] with 3,400 samples and *Danbooru2019* [4] with 302,652 samples are two public datasets for the task of cartoonizing human portraits images. However, these datasets only contain female cartoon portrait, meanwhile there is no sample for male. This is considered as an imbalanced factor, which significantly affects the performance of the learning models.

Cartoon portrait To overcome the problem, we create a new dataset named *CartoonFace10K*. Firstly, we access the website of *anime-planet* [34] to collect 50,245 images of cartoon characters. We use gender filter for separately collecting male and female characters. Secondly, a cartoon facial detector[1] is leveraged to remove non-human images, e.g. the character of Doraemon or Pikachu. Following the removal stage, there are 14,021 cartoon human face images. To enhance the confidence, we do a manual check across all images to ensure the purity of our proposed dataset. Finally, our *CartoonFace10K* contains 10,000 cartoon human face images. To date, this is the largest and the most qualified dataset of cartoon human face synthesis research. Furthermore, our dataset maintains a ratio of 1:1 for male and female cartoon human faces to avoid gender imbalance in data.

Human portrait Unlike cartoon portraits, human portraits are much easier to collect and there are also available datasets. Hence, we make use of the CelebA dataset [29] to select 10,000 images. The number of selected images is equal to those in cartoon human face part to ensure the balance in our experiments.

In our experiments, we proceed to train all configurations with un-paired datasets, which includes 9,000 images each domain for training phase. Then in the testing phase, 1,000 human face images are used for cartoon face synthesis. Meanwhile, the remaining cartoon images serve in the evaluation process with the FID metric.

3.2 Abstraction-perception preserving cartoon face synthesis

In this section, we describe our two-stage training scheme, which orients to cartoonizing portrait images problem. By observing the cartoon facial features, we find that a cartoonized portrait has two major changes compared to original input: simplifying detailed textures while preserving edge information (Fig. 2-2,3), and synthesizing cartoon facial features (Fig. 2-1). To the aspect of training GAN, it is difficult to map two image domains with matching complex criteria. In our case, the output cartoon portraits must satisfy three factors of color and texture, identity and cartoon facial features. Thus, we first aim at training a

[1] https://github.com/nagadomi/lbpcascade_animeface

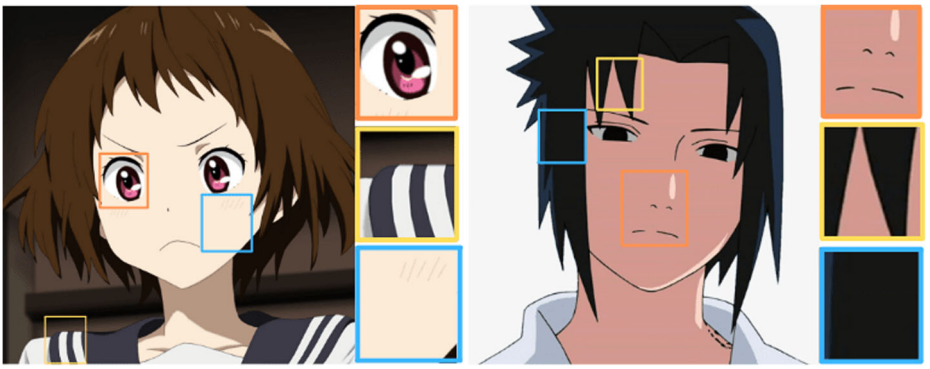

Fig. 2 Common characteristics of cartoon face images 1. Distinctive facial features (orange boxes); 2. Sharpened edges (yellow boxes); 3. Smoothed color blocks (blue boxes)

network to smooth images in order to gain a fine weight for the network. Then in the next step, we leverage the trained network to maintain the aforementioned criteria, which is simpler than doing all at the same time. Hence, we propose a two-stage training process including (1) the abstraction stage for reducing texture and smoothing color, (2) the perception stage for not only simplifying images but also translating the facial features to cartoon domain.

In the remainder of this paper, we formulate our training as the process of learning to transfer images from human face domain H to cartoon face domain C. The training dataset comprises of human face dataset (CelebA) $Data(H) = \{\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n\}$ and cartoon face dataset (our *CartoonFace10K*) $Data(C) = \{\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_m\}$, where n and m are number of human face images and cartoon face images, respectively. Our model consists of two subnetworks, namely Generator G and Discriminator D.

Abstraction Stage The objective of this stage is to smooth the color and reduce the details of texture in the input image. This initialization emulates the demand of having smooth and sparse color blocks with clear edges. Due to the lack of ground truth, we conduct to look for a method to generate the smooth version of human face images. The edge-preserving smoothing problem is consistent with the goal of this phase. In this work, we use the ResNet-based architecture [47] as *Abstractor A* to generate ground truth $\mathbf{y}_i$ for the abstraction stage. Once we have the ground truth, the output of the abstraction stage $G(\mathbf{h}_i)^a$ is expected to be exact to the abstraction of input. Hence, in this stage, we use an L_2-based loss called abstractive loss as bellow:

$$\mathcal{L}_{abstractive}(G) = \mathbb{E}_{\mathbf{h}_i \sim Data(H)}[\|\mathbf{y}_i - G(\mathbf{h}_i)^a\|_2]. \tag{1}$$

A more straightforward design for our proposed method is that we skip the Abstraction Stage and directly use the Abstractor as the Generator for the Perception Stage. However, the architecture of Abstrator includes a series of convolution layers with 16 residual blocks in the middle. This architecture is designed for the Edge-Preserving Image Smoothing problem, where we want to reduce the details of the input image and do not have to synthesize something new. However, in the Perception Stage, we also want the generator to synthesize cartoon facial features therefore the architecture of Abstrator is no longer relevant. As a result, we decide to use the two-stage training scheme.

Perception Stage We assume that the generator overcomes the local minimum and is closer to become a good cartoonization system after the abstraction stage. Hence, we continue to train the generator according to the GAN's mechanism.

Due to the lack of paired dataset, we use the perceptual loss [21] to ensure that input and output have the same visual semantic meaning. Furthermore, the generator should not only have the ability of the abstractor but also synthesize the cartoon facial features. Therefore, the perceptual loss is more suitable than L_2-based loss, which is too strict for the objective of this stage. Note that the perceptual loss shares the same objective with abstractive loss, but at a more flexible level. In other words, the perception stage still keeps the constrain of the abstraction stage. To balance both the performance and inference time, we choose ResNet-18 [14] ϕ as an auxiliary classifier to compute perceptual loss. Equation (2) formulates the perceptual loss between the high-level features of the abstraction of input $\mathbf{y}_i$ and the output of generator in perception stage $G(\mathbf{h}_i))^p$ in our proposed framework:

$$\mathcal{L}_{perceptual}(G) = \mathbb{E}_{\mathbf{h}_i \sim Data(H)}[\|\phi^l(\mathbf{y}_i) - \phi^l(G(\mathbf{h}_i))^p\|_1]. \tag{2}$$

In the perceptual stage, we utilize the adversarial training to avoid the requirement of manually defining a style loss, which is very difficult in cartoon face synthesis. The adversarial loss as in (3) is used to match the distribution of human face image features to cartoon face image features:

$$\begin{aligned} &\mathcal{L}_{adversarial}(G, D) = \mathbb{E}_{\mathbf{c}_i \sim Data(C)}[\log(D(\mathbf{c}_i)))] \\ &+\mathbb{E}_{\mathbf{h}_i \sim Data(H)}[\log(1 - D(G(\mathbf{h}_i)^p))]. \end{aligned} \tag{3}$$

Then, we jointly train the discriminator and the generator to optimize the final objective:

$$\min_G \max_D \mathcal{L}_{total} = \mathcal{L}_{adversarial} + \lambda \mathcal{L}_{perceptual}\ , \tag{4}$$

where the $\mathcal{L}_{perceptual}$ drives model to preserve semantic content of input image, while $\mathcal{L}_{adversarial}$ contributes to synthesizing target domain. In our work, we set $\lambda = 90$, which harmonizes style translation and content preservation. The overview of our framework for cartoon face image synthesis is illustrated in Fig. 3.

3.3 FISI: F-score of inception and similarity

In literature, **Inception Score (IS)** is a well-known metric to evaluate GAN models. Technically, IS uses the architecture of an Inception-v3 model pre-trained on ImageNet as its backbone network. The IS is calculated via the probability of whether an image belongs to a specific class. Therefore, this metric works well only if the considered images contain an object which has been defined in ImageNet. Moreover, IS is known to be suitable for evaluating image generation problem whereas our work aims at transferring image style.

Fretchet Inception Distance (FID) is another evaluation metric to evaluate the quality of synthesized images. Similar to IS, the backbone of FID is an ImageNet pre-trained Inception-v3. FID score is computed in the last coding layer of the network to leverage semantic features. Respectively, both the real and the generated images are fed into the network of Inception-v3 to extract information. Let $(\mathbf{m}, \mathbf{C})$ be the mean and covariance of image features in source domain T and $(\mathbf{m}', \mathbf{C}')$ denotes the generated image features in

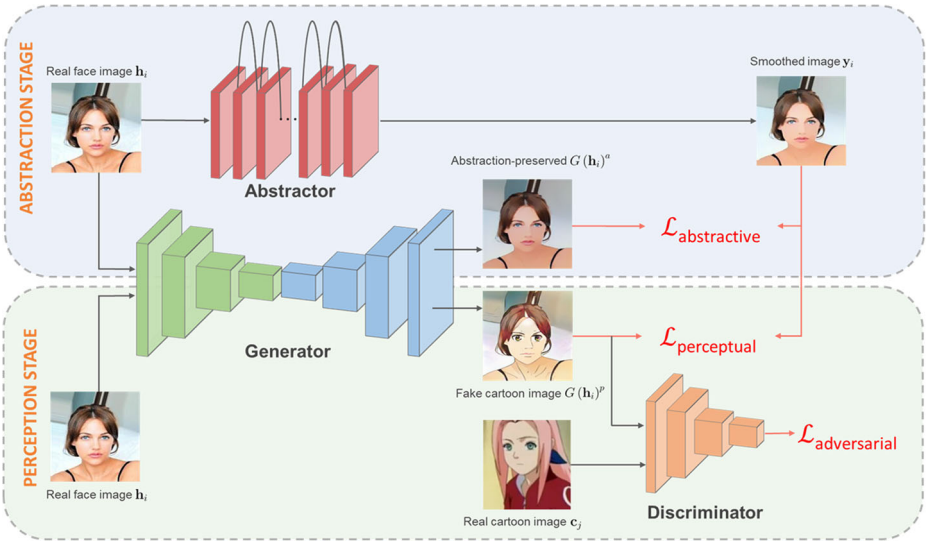

Fig. 3 The proposed framework. There are two fine-tuning stages for cartoon face synthesis, namely, abstraction and perception. In the abstraction stage, input image $\mathbf{h}_i$ and its smoothed version of $\mathbf{y}_i$ are considered as a pair of images for training the generator G to produce the abstraction-preserved $G(\mathbf{h}_i)^a$. In the perception stage, G is retrained to generate perception-preserved $G(\mathbf{h}_i)^p$ from $\mathbf{h}_i$. $G(\mathbf{h}_i)^p$ and $\mathbf{c}_j$ are the fake and real cartoon faces, respectively

generated domain K. We have the difference between the two distributions measured by Frechet distance shown below.

$$\begin{aligned} FID(T, K) &= d((\mathbf{m}, \mathbf{C}), (\mathbf{m}', \mathbf{C}')) \\ &= \mathbf{m} - \mathbf{m}'^2_2 + \text{Tr}(\mathbf{C} + \mathbf{C}' - 2(\mathbf{C}\mathbf{C}')^{\frac{1}{2}}), \end{aligned} \tag{5}$$

where Tr refers to the trace of linear algebra operation. With the computed FID results, the lower the score is, the better quality of generated images is achieved. Note that FID score compares the differences among images based on semantic features rather than their labels, thus it is suitable to be an evaluation metric in our work. Despite paying attention to the similarity of the real and the generated images, FID does not focus on spatial information. The reason is that FID relies on high-level feature maps (i.e. the feature maps after the last average pooling in Inception-v3), where the global structure information is vanished. Let us take Fig. 4 as an example of this statement. It is obvious that the generated cartoonized results yield a poor visual effect meanwhile FID achieved the value of 35.41 (the lowest score in our experiments). To overcome this issue, we propose to combine FID with Structure Similarity [39].

Structural Similarity (SSIM) is an evaluation metric based on the assumption that human visual perception system well extracts the structural information of the captured scenes. Therefore, SSIM is designed to compare pairs of images based on the degradation of structural information rather than the quantifying errors. To be more specific, there exist three factors that affect the SSIM score, namely luminance, contrast and structure. Let $\mathbf{x}$ and $\mathbf{y}$ be the two exemplary images that need to compare. Each factor is computed as below.

Fig. 4 Exemplary results (the bottom row) produced by CUT [32] from portrait images (the top row). In this case, FID equals to 35.41, which is the lowest FID score in our experiment. In other words, the synthesis results well capture cartoon features yet ignore the global structure of the inputs

- **Luminance** is computed as $l(\mathbf{x}, \mathbf{y}) = \frac{2\mu_x\mu_y+C_1}{\mu_x^2+\mu_y^2+C_1}$, where μ_x and μ_y are respectively the means of intensity of $\mathbf{x}$ and $\mathbf{y}$.
- **Contrast** is calculated as $c(\mathbf{x}, \mathbf{y}) = \frac{2\sigma_x\sigma_y+C_2}{\sigma_x^2+\sigma_y^2+C_2}$, where σ_x and σ_y are the standard deviations of each image $\mathbf{x}$ and $\mathbf{y}$, respectively.
- **Structure** is computed as $s(\mathbf{x}, \mathbf{y}) = \frac{\sigma_{xy}+C_3}{\sigma_x\sigma_y+C_3}$. Here, the structural information is evaluated via σ_x and σ_y.

Finally, the combination of luminance, contrast, and structure is used to measure the structural similarity as below:

$$SSIM(\mathbf{x}, \mathbf{y}) = [l(\mathbf{x}, \mathbf{y})]^{\alpha}[c(\mathbf{x}, \mathbf{y})]^{\beta}[s(\mathbf{x}, \mathbf{y})]^{\gamma}. \tag{6}$$

By default, $C_3 = \frac{C_2}{2}$, $\alpha = \beta = \gamma = 1$, and C_1 is added to avoid a zero denominator. Study further, we realize that SSIM complements the lack of spatial structural information in FID score. We define the SSIM between two domains $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_v\}$ and $Y = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_v\}$ is measured by (7). Note that two datasets is paired (i.e. y_i indicates the Y domain version of x_i)

$$SSIM(X, Y) = \frac{\sum_{i=1}^{v} SSIM(\mathbf{x_i}, \mathbf{y_i})}{v} \tag{7}$$

The proposed FISI In our study, we observe that the FID metric is good at measuring the differences among the features of generated images and images in the target domain. Meanwhile, SSIM can handle the structural information when generating images, which is out of reach of FID. Therefore, we propose a novel metric named **FISI**, the short form of **F**-score of **I**nception and **SI**milarity, which is the harmonic mean of FID and SSIM.

When forming the FISI metric, there exists a problem related to the value ranges of FID and SSIM. In particular, FID score ranges from 0 to *positive infinite* while the range of SSIM value is adjusted to be in [0, 1]. Should we simply sum these two metrics together,

the difference in value range heavily affects the proposed metric. Thus, we normalize both FID and SSIM scores to [0, 1] in order to have the similar range of value. Note that our main task is to transfer images from domain A to B, which is the ideal case. To investigate the upper bound of FID, we first visualize the distribution of 1000 randomly selected images per domain. We use ResNet-18 [14] pre-trained on ImageNet dataset to obtain the features map. Then, we apply t-SNE to reduce the dimension of the feature from 1000 to 1. Finally, the distribution is estimated by *Kernel Density Estimation*. The Fig. 5 illustrates the process of shifting produced images distribution B' to B. However, there always exists a gap between the generated domain B' and the target domain B. As a result, the upper bound value of FID is equal to the distance between source domain A and generated domain B. Based on this hypothesis, we can normalize the FID score as in (10).

For fair evaluation, in (8), we formulate a novel FISI metric by the F_β score of FID and SSIM. Note that SSIM is more biased forward to the human visual perception than FID. Therefore, follow previous works [1, 31, 33], we weight the precision (as $S_{B'A}$ in (8)) greater than the recall (as $1 - F_{norm}$ in (8)). This weighting helps the FISI better mimic the perception of human beings than when conveys the balance between the precision and the recall (i.e. $\beta = 1.0$)

$$\begin{aligned} FISI(B, B', A) &= F_\beta(S_{B'A}, 1 - F_{norm}) \\ &= (1 + \beta^2)\frac{(1-F_{norm})S_{B'A}}{\beta^2(1-F_{norm})+S_{B'A}}, \end{aligned} \tag{8}$$

where $\beta = 0.5$ while $S_{B'B}$ and F_{norm} are defined as follows:

$$S_{B'A} = SSIM(B', A), \tag{9}$$

$$F_{norm} = \frac{FID(B', B)}{FID(A, B)}. \tag{10}$$

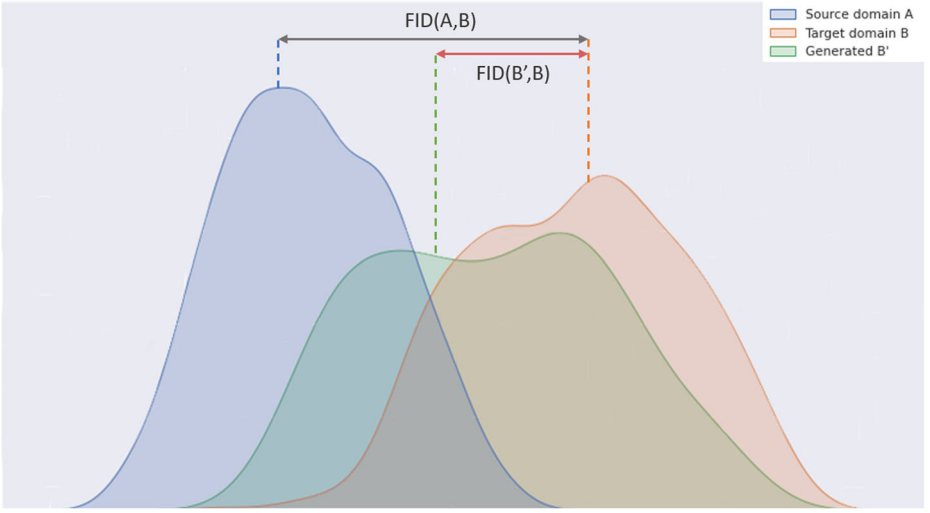

Fig. 5 Let A, B', and B be the domains of source, generated, and target, respectively. The image-to-image problem aims to shift the distribution of B' from A to be closed to B. Therefore, we consider the FID_{AB} as the upper bound value of $\text{FID}_{B'B}$

4 Experiments

4.1 Experimental setup

All experiments were performed on a NVIDIA Geforce RTX 2080TI. We use the Adam optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.999$ to optimize both generator and discriminator. Batch size and learning rate are 16 and 0.0002, respectively. During training, all images are resized to 256×256. The architectures of Discriminator and Generator are adopted from CartoonGAN [5] with reducing half of the number of filters in all convolution layers for simplifying the calculation. As a result, on a NVIDIA Geforce T4, our model can process an image within only 4.6ms, which enable real-time inference.

In our experiments, we train 10 epochs for Abstraction Stage and 90 epochs for Perception Stage. For a fair comparison, we train others baseline with the same epochs. We compare our method with other baselines including: CUT [32], CycleGAN [45], CartoonGAN [5], Wang et al. [38], UGATIT [23], and MoNCE [43].

4.2 Quantitative comparison

In order to compare the performance of our proposed model and others, we use three evaluation metrics, namely FID, SSIM and our proposed FISI. We conduct a quantitative comparison on *selfie2anime* and our dataset, which is introduced in Section 3.1. There are 100 and 1,000 images per domain for testing purpose, respectively. We denote images of human face as source domain A and images of cartoon face as target domain B. Based on the theory of our proposed FISI metric, we first compute the upper bound of the FID score. To be specific, the FID upper bounds of *selfie2anime* and our dataset are 289.29 and 236.78, respectively. Then, we normalize the computed FID score to the specific range of 0 and 1.

The quantitative results on our collected data are presented in Table 2. The images generated by CartoonGAN well keep global structure of inputs. And they are so similar to the real images that they get the highest FID score. Wang and Yu [38] improves the cartoon features of CartoonGAN, which leads to a significant decrease in FID score. However, the global structure of [38] is not as good as CartoonGAN (decrease in SSIM score). Although CycleGAN, CUT, UGATIT, and MoNCE give impressive FID scores, these methods disrupt the spatial structure of inputs. Their SSIM scores are low and their FISI scores are decreased

Table 2 Quantitative results of the comparison of different methods in terms of FID, SSIM and FISI. The top-3 results are highlighted in red, green, and blue

Methods	FID to cartoon ⇓		SSIM to real ⇑		FISI ⇑	
	selfie2 anime	Cartoon Face10K	selfie2 anime	Cartoon Face10K	selfie2 anime	Cartoon Face10K
CUT	**81.37**	**35.41**	0.49	0.43	0.52	0.48
CycleGAN	**88.14**	**42.12**	0.56	0.49	0.58	0.53
CartoonGAN	138.58	165.87	**0.71**	**0.73**	**0.66**	**0.57**
Wang et al. [38]	133.27	115.64	0.57	0.52	0.56	0.52
UGATIT	121.32	83.35	**0.59**	**0.60**	**0.59**	**0.61**
MoNCE	92.01	48.74	0.53	0.46	0.55	0.50
Ours	**101.13**	**64.77**	**0.67**	**0.65**	**0.67**	**0.66**

accordingly. In other words, previous methods only focus on one out of the two criteria of whether capturing cartoon human face features or preserving global structure of portraits. In comparison with previous works, our proposed method handles both of the two criteria demonstrated by the highest FISI score among the five methods.

4.3 Qualitative comparison

Qualitative comparison among previous methods and ours is visualized in Fig. 6. Overall, CartoonGAN [5] and the model of [38] only pay attention to texture and color translation. Moreover, CartoonGAN uses an edge-promotion loss function to capture more information of edges in the images. Despite an ineffective learning of cartoon features, CartoonGAN maintains identifiable features of a character.

Meanwhile, the results of CycleGAN and CUT maintain the local cartoon facial features. However, they fail to keep the global structures of input images. Both cycle-consistency-based (i.e. CycleGAN and UGATIT) and contrastive-learning-based (i.e. CUT and MoNCE) methods produce good cartoonized portrait images. Nonetheless, they seem to over-change the input and the results turn out to be difficult to recognize its identity. In another work,

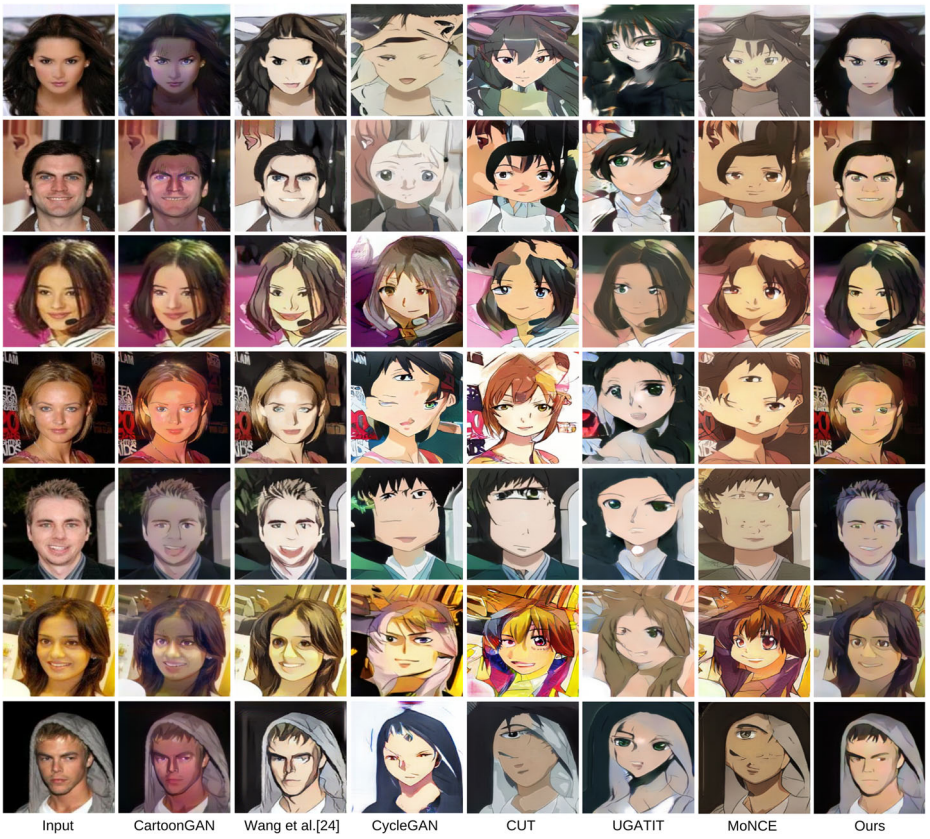

Fig. 6 Qualitative comparison. The first column is human face images. The successive columns are respectively the cartoonized results using CartoonGAN [5], Wang et al. [38], CycleGAN [45], CUT [32], UGATIT [23], MoNCE [43], and ours, respectively

Wang and Yu [38] separates the objectives of GAN into three factors: texture, surface, and structure. This change makes the training process of GAN more efficient as the model tries to optimize each kind of feature. As a result, the proposed method of [38] gives better results than CartoonGAN at capturing cartoon features. In detail, portraits generated by [38] have sharp edges, smooth lines or curves and clearly separated blocks of colors. Note that the output of [38] still keeps identity features of the character, which is a strong point derived from CartoonGAN.

The last column in Fig. 6 presents the results of our proposed method. The proposed framework generates images with fine color, texture and even cartoon facial features. Among methods, our approach yields the visually better results.

4.4 User study

Image aesthetics are highly subjective and depend on individual opinion. Hence, we conduct a user study to compare our method with previous works. There are in total 125 participants whose ages ranged from 18 to 40 years old ($\mu = 21.36$) participating in the study. All participants are students and staff from the university and the research institute. Among them, 45 participants are female. Prior to the experiment, participants signed the consent form and only after that the instructions about the experiment could be given to the participants. They are requested to try all image synthesis methods in a random order. In particular, there are ten random portrait images shown in each survey. For each portrait image, we generate five cartoonized versions, namely, Wang et al. [38], CartoonGAN, CUT, CycleGAN, and our proposed framework. We gave the participants a brief about different methods. The participants are asked to rate from 1 to 5 for each method. Finally, we compute the mean and the standard deviation scores for each method. The higher mean score, the better the average quality of cartoonized images. The lower standard deviation score shows that the method is stable in its mean. As shown in Table 3, our method achieves the highest mean and the lowest standard deviation of rating. This demonstrates that the participants prefer ours method over baselines. Meanwhile, CycleGAN obtains the lowest mean rating since it produces unstructured results. CartoonGAN receives the mixed results. In particular, it reaches the second-highest mean rating. However, some participants notice the unnatural lighting in the CartoonGAN generated results. This is the reason why CartoonGAN receives the highest std rating.

Table 3 The result of user study

Methods	Mean rating ⇑	STD rating ⇓
CUT	2.2227	1.1103
CycleGAN	1.7984	0.9922
CartoonGAN	3.7555	1.3201
Wang et al. [38]	3.1960	0.9752
Ours	**4.0269**	**0.8875**

The higher mean rating indicates the higher favor of cartoonized images whereas the lower STD rating implies the better stability of the rating. The best performance is marked in boldface

Table 4 Quantitative analysis of each component in our proposed method based on the three evaluation metrics

Configuration	FID to cartoon ⇓	SSIM to real ⇑	FISI ⇑
w/o perceptual loss	204.89	0.24	0.21
w/o adversarial loss	188.68	0.77	0.49
w/o abstraction stage	70.81	0.54	0.57
Full model	**63.91**	**0.65**	**0.66**

The best performance of each criterion is marked in boldface

4.5 Ablation study

In this subsection, we investigate the impact of each component in the proposed framework. The quantitative and qualitative results of our ablation study are shown in Table 4 and Fig. 7, respectively.

Firstly, following the two-stage training configuration as in Section 4.1, but at the perceptual stage, we train with adversarial loss alone and remove the perceptual loss. The results express no information of the real portraits but only the general shapes. This problem occurs due to the lack of semantic constraints between the input and the output. Accordingly, the low value of SSIM score means there exists little structural similarity in the resulting images. At the same time, the FID score of this configuration is higher than that of the full model. This score demonstrates that our model cannot capture cartoon facial features well without perceptual loss, as visualized in Fig. 7-(b).

Secondly, we also conduct a two-stage training scheme and removing the adversarial loss in the perception stage. The resulting images are similar to the input portraits in the realistic style and lack of cartoon features. This indicates that the adversarial loss plays a vital role in producing cartoon features. The results shown in Fig. 7-(c) are smoothed versions of the input portraits without adding cartoon features. The quantitative result of this configuration shows a high value of the FID to demonstrate that it is not similar to cartoon images. Due

Fig. 7 Visualization when removing each component individually in our proposed approach

to the removal of the adversarial loss, the perceptual factor itself reduces details of the input image.

Last but not least, we demonstrate the efficiency of the proposed two-stage training scheme by ablating the abstraction stage. For a fair comparison, we train perception stage with 100 epochs and both perceptual and adversarial losses are used as in Fig. 3. The results in Fig. 7-(d) illustrate that ablating abstraction stage causes messy textural details. In the third line of Table 4, both the FID to cartoon face domain and the SSIM to human face domain are not as good as the full framework. Ablation studies show that all components contribute to the final performance of the full model.

4.6 Failure cases

Figure 8 shows failure cases of our proposed method. First, even though the abstraction stage reduces messy texture details, our proposed method sometimes produces noise in resulting images (especially in the region of the forehead). Second, some translations produce unexpected wrinkles on characters' faces. In these cases, the cartoonizer seems not to be aware of important details (needed to be kept) and unimportant details (needed to be removed) during the translations. Therefore, a facial-aware attention should be taken into account in order to avoid unexpected wrinkles.

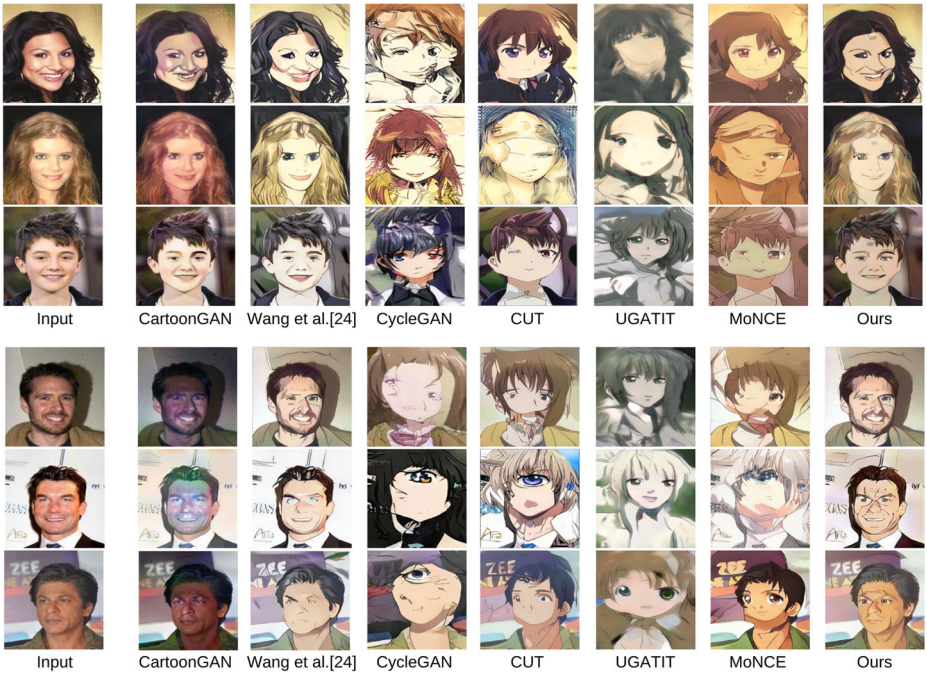

Fig. 8 Failure cases by our proposed method are typically caused by: (top 3 rows) noise on forehead and (bottom 3 rows) unexpected wrinkles on the face

5 Conclusion and future work

In this paper, we propose a cartoon face synthesis method with three main contributions. Firstly, we propose a novel framework which utilizes a two-stage training strategy: an abstraction stage to reduce detailed textures and a perception stage for not only simplifying the image but also translating the facial features to cartoon domain. Secondly, we introduce a new evaluation metric named FISI as the harmonic mean of the two FID and SSIM metrics. Thirdly, we collect and introduce a new dataset named *CartoonFace10K* for cartoonizing portrait images problem. Via the extensive experiments, our approach establishes the new state-of-the-art on *selfie2anime* and *CartoonFace10K* datasets.

Though the proposed method achieves significant results for portrait image cartoonization, we still face challenging problem when applying our framework to videos for dynamic facial expression synthesis. Therefore, we aim to address the problem of automatically cartoonizing portrait video in the future.

Acknowledgements We gratefully acknowledge the support of NVIDIA Corporation with the GPU donation.

Funding This research is funded by Vietnam National University Ho Chi Minh City (VNUHCM) under grant number C2022-26-01. This work is also supported by the National Science Foundation (NSF) under Grant 2025234. Thanh-Danh Nguyen is funded by the Master, PhD Scholarship Programme of Vingroup Innovation Foundation (VINIF), code VINIF.2022.ThS.104.

Data Availability Data available on request from the authors

Declarations

Conflict of Interests The authors have no conflicts of interest to declare that are relevant to the content of this article.

References

1. Achanta R, Hemami SS, Estrada FJ, Süsstrunk S (2009) Frequency-tuned salient region detection. In: CVPR 2009
2. Benaim S, Wolf L (2017) One-sided unsupervised domain mapping. In: Advances in neural information processing system (2017)
3. Bińkowski M, Sutherland DJ, Arbel M, Gretton A (2018) Demystifying MMD GANs. In: International conference on learning representations (2018)
4. Branwen G, Anonymous, Community D (2019) Danbooru2019 portraits: a large-scale anime head illustration dataset. https://www.gwern.net/Crops#danbooru2019-portraits. Accessed: DATE
5. Chen Y, Lai Y, Liu Y (2018) Cartoongan: generative adversarial networks for photo cartoonization. In: 2018 IEEE/CVF conference on computer vision and pattern recognition, pp 9465–9474
6. Choi Y, Choi M-J, Kim M, Ha J-W, Kim S, Choo J (2018) Stargan: unified generative adversarial networks for multi-domain image-to-image translation. In: 2018 IEEE/CVF Conference on computer vision and pattern recognition, pp 8789–8797
7. Choi Y, Uh Y, Yoo J, Ha J-W (2020) Stargan v2: diverse image synthesis for multiple domains. In: 2020 IEEE/CVF Conference on computer vision and pattern recognition, pp 8185–8194
8. Dumoulin V, Shlens J, Kudlur M (2017) A learned representation for artistic style. In: International conference on learning representations (2017)
9. Gatys LA, Ecker AS, Bethge M (2016) Image style transfer using convolutional neural networks. In: 2016 IEEE Conference on computer vision and pattern recognition, pp 2414–2423
10. Gooch A (2001) Non-photorealistic rendering
11. Gooch B, Coombe G, Shirley P (2002) Artistic vision: painterly rendering using computer vision techniques. In: Proceedings of the 2nd international symposium on non-photorealistic animation and rendering, pp 83–90

12. Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville AC, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing system (2014)
13. Goodfellow I, Bengio Y, Courville AC (2015) Deep learning. Nature 521:436–444
14. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: 2016 IEEE Conference on computer vision and pattern recognition, pp 770–778
15. Hertzmann A (1998) Painterly rendering with curved brush strokes of multiple sizes. In: SIGGRAPH '98
16. Hertzmann A, Jacobs C, Oliver N, Curless B, Salesin D (2001) Image analogies. In: SIGGRAPH '01
17. Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S (2017) Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Advanced in conference on neural information processing systems (2017), pp 6629–6640
18. Huang X, Belongie SJ (2017) Arbitrary style transfer in real-time with adaptive instance normalization. In: 2017 IEEE International conference on computer vision (2017), pp 1510–1519
19. Huang X, Liu M-Y, Belongie S, Kautz J (2018) Multimodal unsupervised image-to-image translation. In: Eupopean conference on computer vision (2018)
20. Isola P, Zhu J-Y, Zhou T, Efros AA (2017) Image-to-image translation with conditional adversarial networks. In: 2017 IEEE Conference on computer vision and pattern recognition, pp 5967–5976
21. Johnson J, Alahi A, Fei-Fei L (2016) Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision (2016)
22. Kim T, Cha M, Kim H, Lee J, Kim J (2017) Learning to discover cross-domain relations with generative adversarial networks. In: International conference on machine learning (2017)
23. Kim J, Kim M, Kang H, Lee KH (2020) U-gat-it: unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. In: International conference on learning representations (2020)
24. Kolliopoulos A (2005) Image segmentation for stylized non-photorealistic rendering and animation
25. Kyprianidis JE, Collomosse J, Wang T, Isenberg T (2013) State of the 'art': a taxonomy of artistic stylization techniques for images and video. IEEE Trans Visual Comput Graphics, 866–885
26. Laovaan How to draw yourself as an anime character. Youtube. https://youtu.be/9YSpzmWwBkI. Accessed 24 Oct 2021
27. Lee H-Y, Tseng H-Y, Huang J-B, Singh MK, Yang M-H (2018) Diverse image-to-image translation via disentangled representations. In: European conference on computer vision (2018)
28. Li H, Liu G, Ngan KN (2011) Guided face cartoon synthesis. IEEE Trans Multimedia. 1230–1239
29. Liu Z, Luo P, Wang X, Tang X (2015) Deep learning face attributes in the wild. In: Proceedings of international conference on computer vision (2015)
30. Liu M-Y, Breuel T, Kautz J (2017) Unsupervised image-to-image translation networks. In: Advances in neural information processing system
31. Nguyen TV, Liu L (2017) Salient object detection with semantic priors. In: 2017 International joint conference on artificial intelligence. arXiv:abs/1705.08207
32. Park T, Efros AA, Zhang R, Zhu J-Y (2020) Contrastive learning for unpaired image-to-image translation. In: European conference on computer vision (2020)
33. Perazzi F, Krähenbühl P, Pritch Y, Sorkine-Hornung A (2012) Saliency filters: contrast based filtering for salient region detection. In: 2012 IEEE conference on computer vision and pattern recognition, pp 733–740
34. Planet A Anime planet website. Anime Planet. https://www.anime-planet.com. Accessed 24 Oct 2021
35. Rosin PL, Lai Y (2015) Non-photorealistic rendering of portraits. In: CAE '15
36. Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X (2016) Improved techniques for training gans. In: Advanced in conference on neural information processing systems (2016)
37. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. arXiv:abs/1409.1556
38. Wang X, Yu J (2020) Learning to cartoonize using white-box cartoon representations. In: 2020 IEEE/CVF Conference on computer vision and pattern recognition, pp 8087–8096
39. Wang Z, Bovik A, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 13:600–612
40. Wang T, Liu M-Y, Zhu J-Y, Liu G, Tao A, Kautz J, Catanzaro B (2018) Video-to-video synthesis. In: Advances in neural information processing system (2018)
41. Wang T, Liu M-Y, Zhu J-Y, Tao A, Kautz J, Catanzaro B (2018) High-resolution image synthesis and semantic manipulation with conditional gans. In: 2018 IEEE/CVF conference on computer vision and pattern recognition, pp 8798–8807
42. Yi Z, Zhang H, Tan P, Gong M (2017) Dualgan: unsupervised dual learning for image-to-image translation. In: 2017 IEEE international conference on computer vision (2017), pp 2868–2876

43. Zhan F, Zhang J, Yu Y, Wu R, Lu S (2022) Modulated contrast for versatile image synthesis. arXiv:abs/2203.09333
44. Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A (2016) Learning deep features for discriminative localization. In: 2016 IEEE Conference on computer vision and pattern recognition, pp 2921–2929
45. Zhu J-Y, Park T, Isola P, Efros AA (2017) Unpaired image-to-image translation using cycle-consistent adversarial networks. In: 2017 IEEE International conference on computer vision (2017), pp 2242–2251
46. Zhu J-Y, Zhang R, Pathak D, Darrell T, Efros AA, Wang O, Shechtman E (2017) Toward multimodal image-to-image translation. In: Advances in neural information processing system (2017)
47. Zhu F, Liang Z, Jia X, Zhang L, Yu Y (2019) A benchmark for edge-preserving image smoothing. IEEE Trans Image Process 28:3556–3570