AI

MDPI

Article

Multi-CartoonGAN with Conditional Adaptive Instance-Layer Normalization for Conditional Artistic Face Translation

Rina Komatsu and Tad Gonsalves *

Department of Information & Communication Sciences, Faculty of Science and Technology, Sophia University, 7-1 Kioi-cho, Chiyoda-ku, Tokyo 102-8554, Japan; r_komatsu@eagle.sophia.ac.jp
* Correspondence: t-gonsal@sophia.ac.jp

Abstract: In CycleGAN, an image-to-image translation architecture was established without the use of paired datasets by employing both adversarial and cycle consistency loss. The success of CycleGAN was followed by numerous studies that proposed new translation models. For example, StarGAN works as a multi-domain translation model based on a single generator–discriminator pair, while U-GAT-IT aims to close the large face-to-anime translation gap by adapting its original normalization to the process. However, constructing robust and conditional translation models requires tradeoffs when the computational costs of training on graphic processing units (GPUs) are considered. This is because, if designers attempt to implement conditional models with complex convolutional neural network (CNN) layers and normalization functions, the GPUs will need to secure large amounts of memory when the model begins training. This study aims to resolve this tradeoff issue via the development of Multi-CartoonGAN, which is an improved CartoonGAN architecture that can output conditional translated images and adapt to large feature gap translations between the source and target domains. To accomplish this, Multi-CartoonGAN reduces the computational cost by using a pretrained VGGNet to calculate the consistency loss instead of reusing the generator. Additionally, we report on the development of the conditional adaptive layer-instance normalization (CAdaLIN) process for use with our model to make it robust to unique feature translations. We performed extensive experiments using Multi-CartoonGAN to translate real-world face images into three different artistic styles: portrait, anime, and caricature. An analysis of the visualized translated images and GPU computation comparison shows that our model is capable of performing translations with unique style features that follow the conditional inputs and at a reduced GPU computational cost during training.

Keywords: artificial intelligence; image processing; image generation; generative adversarial network

Citation: Komatsu, R.; Gonsalves, T. Multi-CartoonGAN with Conditional Adaptive Instance-Layer Normalization for Conditional Artistic Face Translation. *AI* **2022**, *3*, 37–52. https://doi.org/10.3390/ai3010003

Received: 26 September 2021
Accepted: 14 January 2022
Published: 24 January 2022

1. Introduction

Studies exploring deep learning modeling have expanded to the field of image processing. For example, in the field of image recognition, Shutanov et al. [1] explored the possibility of using convolutional neural networks (CNNs) to recognize traffic signs. Li et al. [2] trained a humanoid robot to recognize emotions from facial images by combining a CNN and long short-term memory (LSTM). In an effort to improve image task quality levels, Tian et al. [3] proposed a guided denoising method that works by extracting latent noise information through an attention-guided CNN.

Most image recognition and improvement tasks require the preparation of both input and target paired data, such as classified labels for recognition and cleaned images to improve noisy input images. However, preparing target images is often a cumbersome task depending on the image processing method. This is particularly true in the case of image-to-image translation tasks, such as translating real-world photos into segmented images under supervised learning conditions, because of the need to search for and generate paired images.

Despite this difficulty, the CycleGAN translation model overcame the problem of preparing paired datasets and made it possible to perform image-to-image (one-to-one domain) translation learning with unpaired image datasets. Later, Zhu et al. [4] proposed a method of employing CycleGAN adversarial loss to mimic target domain features and cycle consistency loss to retain the content image features. After the introduction of CycleGAN, Choi et al. [5] proposed a multi-domain translation architecture, named StarGAN, that can translate an image among N domains by utilizing a conditional vector. This multi-domain translation model eliminated the need to prepare and train N generator-discriminator pairs when using one-to-one domain translation to output N kinds of translated images.

However, the translation performance of models based on CycleGAN is strongly dependent on the feature differences between the source and target domain datasets. For example, as pointed out in [6], previous models (such as CycleGAN) were unable to complete image translation tasks that involved large shape changes. Accordingly, implementing a translation model for N domains that is robust to extreme appearance changes can be seen as a new frontier for image-to-image translation. Unfortunately, implementing such a model is difficult because of the need to consider the GPU computational cost at the start of conditional translation training.

Another issue is that existing translation models capable of dealing with extreme appearance changes consume significant amounts of mobile-integrated blockchain (MiB) on GPUs because of the need to repeatedly use the generator to obtain cycle consistency loss. This means that, in situations where a generator consists of multiple CNN layers and complex normalization layers, additional computational resources are required when the generator is reused.

In response to these issues, this study aims to construct an N domain translation model that deals with extreme appearance translations by saving computational costs at the start of the training. In our experiments, we attempted to train a single model to translate a real-world face into three different artistic face styles: portrait, anime, and caricature, each of which has unique features such as painting brushstrokes and coloring. Here, it should be noted that translating to anime and caricature styles is particularly demanding because the appearance gaps between these styles and real-world faces are extremely large.

Hence, when considering the most appropriate model to address these issues, inspired by the low computational cost of CartoonGAN [7], we decided to develop a model that could perform conditional N domain translations and be adaptable to extreme translations. The result was our novel adaptation of CartoonGAN, named Multi-CartoonGAN.

Our proposed Multi-CartoonGAN model not only employs the VGG19 pretraining method of CartoonGAN to prevent calculating content loss and save parameter training computational costs, but also implements an original normalization function called conditional adaptive layer-instance normalization (CAdaLIN) for use in the generator. This normalization process was inspired by AdaLIN in U-GAT-IT [6], which is a normalization process that summarizes the instance-normalized and layer-normalized results. The affine parameters of AdaLIN were obtained from pooled feature maps using adaptive average and adaptive max pooling.

Our study utilizes a conditional vector as the affine parameter for AdaLIN, which is calculated via linear layers by the input conditional vector. This method not only outputs affine parameters, but also reports the content of the conditional input to the generator instead of inserting a conditional vector directly.

Taken together, the contributions of our proposed model can be summarized as follows:

- In addition to performing suitable artistic translations of low-appearance translations, such as translating real-world faces to portrait style, our proposed Multi-CartoonGAN can also handle large appearance gap changes such as face-to-anime and face-to-caricature translations;

- Our proposed CAdaLIN normalization method contributes to conditional large appearance translations better than other processes, such as inserting a conditional vector, similar to in StarGAN;
- The computational cost of training on GPUs and the time required to complete one training epoch using our model can be reduced to less than that possible with U-GAT-IT;
- By using t-SNE clustering to classify results, our model produces more significant variations between the portrait and anime styles than are possible with CycleGAN, Single GAN, StarGAN, and U-GAT-IT;
- Our N-to-N domains translation model can solve the trade-off problem between extreme appearance translation that necessarily requires a great deal of technical layers, and the GPU computational cost along with memory consumption required for robust training.

2. Related Studies

2.1. Neural Style Transfer

Gaty et al. [8,9] proposed an image translation method that uses a pretrained VGG16 to mix spatial content features and style design features [10]. This neural style transfer method aims to find the best image synthesis by repeated backpropagation. However, the optimization process for this neural style transfer method tends to take a long time. Johnson et al. [11] addressed the optimization time problem by training feed-forward translation networks through perceptual loss functions.

In the latest research arena, image-to-image translation via neural style transfer is not only useful for developing networks, but can also be used to propose normalization functions for image synthesis. For example, Ulynov and Vedaldi [12] proposed a process called instance normalization, which calculates the mean and variance from each channel of a feature map and showed that the performance could be improved by replacing batch normalization with instance normalization. Dumoulin [13] expanded the diversity of available painting styles by utilizing a process called conditional instance normalization, which selects units in the layers for use as affine parameters (scaling and shifting) during instance normalization. Huang et al. [14] also attempted to expand the diversity of style synthesis availability by a process known as adaptive instance normalization, which computes affine parameters for instance normalization based on the input style.

The flexibility of the neural style transfer method makes it possible to mix any image content and style simply by completing the optimization process or training feed-forward networks. However, the neural style transfer process translates the entire area of the content image and lacks the ability to specify the information that determines which area should be translated. This means that, if users want to attempt a particular area translation task, such as a real face to portrait style, the model needs to use other criteria to ensure that some features of the translated image in particular areas, such as the entire face area of the subject, are maintained.

2.2. One-to-One Domain Translation

For translation learning of a particular area, it is necessary to strike a balance between the limited spatial content features to be retained and the likely target domain features to be reflected. The architectures discussed below attempt to use adversarial training to perform learning translation for a particular area (or the whole area) of a content image.

The first architecture we will discuss is pix2pix [15], which is a pioneering image-to-image translation method that implements the L1 norm and adversarial loss. To accomplish this, the architecture first prepares a paired dataset containing real-world photos for input and target images that correspond to each of those photos. Next, during translation training, the pix2pix generator learns how to generate a translated image from an input photo. Then, by comparing the L1 norm and deceiving the discriminator via adversarial loss, it becomes possible to generate an image that closely approaches the target image. However, this

one-to-one domain translation model requires manually preparing a paired dataset for the comparison of outputs and targets.

Other architectures, such as CycleGAN [4], DualGAN [16], and DiscoGAN [17], attempt to resolve this preparation problem by training both mapping models (both generators), thus combining adversarial loss and cycle consistency loss. As a result, the translated images can be reconstructed by inputting the images into another mapping model.

Generators learn output images by maintaining source input content features and deceiving the discriminators so that the generated images are judged as real and belonging to the target domain.

Kim et al. [6] proposed the AdaLIN process, in which parameters can be trained by adaptively selecting the ratio between the instance and layer normalization processes, thus making it possible to construct robust translation models capable of handling tasks that require large shape gap changes, such as selfie-to-anime translations. New architectures have also been developed [7,18] that utilize a pretrained VGG19 [10] to calculate the content loss.

However, even though these novel one-to-one domain translation models can operate using unpaired datasets, if the users desire to obtain N-style translated images with these models, they must still train $N \times (N - 1)$ models. As a result, it takes a long time to complete the translation training.

2.3. Multi Domains Translation

As a potential way to reduce the required number of translation models for N domains, ComboGAN [19] combines encoder and decoder methods. In this architecture, if users want to translate x in domain X to y indomain Y, the generator combines the indexed x encoder and the indexed y decoder and outputs the translated images. However, this method also requires users to train N ComboGANs consisting of an N generator and discriminator pairs to output the N-style-translated images.

A pertinent research question is whether it is possible to control the output form of an image with a single architecture. One potential solution is the use of the conditional GAN architecture (cGAN) [20], in which a conditional vector that relates to the targeted category is provided as an additional input data to the generator and discriminator.

The StarGAN [5] translation process, for example, utilizes a conditional vector in which the input image and the resized conditional vector are concatenated and input into the generator. In this architecture, the generator trains the translated image output by following the conditional input, and the discriminator attempts to determine whether the output is real or fake through the output feature map patches and to classify the domains to which the input image belongs. AttGAN [21] also employs an encoder–decoder-based generator that is extended to consider the relationship between latent representation (conditional input) and the attribution that is output from the discriminator. IcGAN [22] is a version of cGAN that has been expanded by including an encoder that compresses the input content image to achieve latent representation, after which the cGAN outputs the conditional translated image through inputs.

RelGAN [23] employs a conditional vector that represents the differences between the original and target domain attributions. This vector is then input into the generator along with the input image. To upload the conditional input into the generator, StarGAN and RelGAN inject concatenated data (the content image and the conditional vector that has been resized to the channel size) directly into the generator. However, other related studies have proposed methods that do not involve data injection. For example, Chen et al. [24] proposed Gated-GAN, which uses different branches between the encoder and decoder at the generator to perform N-style translations. Yu et al. [25] pointed out that injection methods can lead to mode collapses when used with existing normalization methods, such as batch or instance normalization, and proposed an alternative central biasing normalization (CBN) process as the latent code injection method.

In (CBN), the conditional input is input as a bias and added to the normalized feature map. Multi-domain translation models have attempted conditional translations by changing the conditional input insertion method. For example, SingleGAN [26] employed CBN for conditional translations. Constructing a multi-domain translation model with technical normalization layers can solve extreme appearance gap translation tasks. However, if we stack an ever-increasing number of layers in a translation model, the computational cost of training on GPUs will increase. This makes it difficult to set a small batch size at the start of training and drastically increase the time required to complete a single training epoch.

3. Our Proposed Model

As stated above, our study focused on the CartoonGAN translation architecture [7], which utilizes a pretrained VGGNet and has been shown to require much less training time than CycleGAN. Although CartoonGAN is a one-to-one domain translation model, we further developed this model into a multi-domain translation model by employing CAdaLIN. This section introduces an overview of our model and describes the full loss objectives used for learning conditional translation.

3.1. Overview of Our Multi-Cartoongan

Figures 1 and 2 show the construction of the generator (G) and discriminator (D) in Multi-CartoonGAN. Here, G aims to output translated artistic face images $G(x, c)$ from an input real-world photo x by following c, which is the conditional input that relates to the selected artistic style. The construction of our D is inspired by NICE-GAN, a one-to-one domain architecture whose discriminator work as encoder for its generator [27]. NICE-GAN constructs multi-scale discriminators that output different scales of the receptive field. Here, D attempts to determine whether the image is real or fake using two different scales of output feature maps and the logit of class activation maps (CAM). We employed the CAM logit to discriminate the feature units extracted by the pooling layers. Additionally, D classifies the input images into appropriate artistic styles.

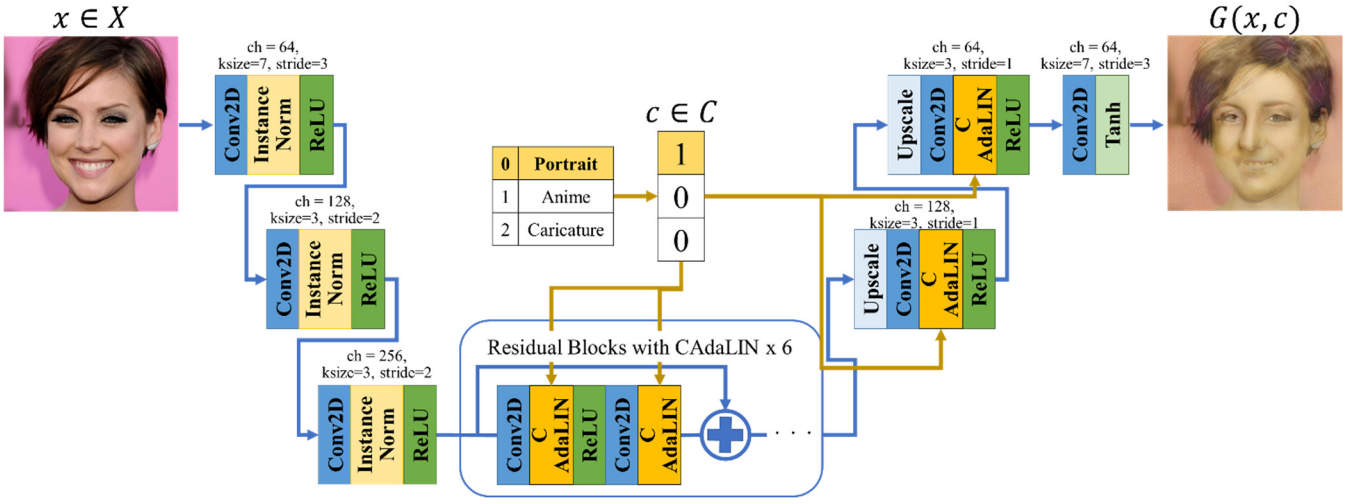


Figure 1. Overview of the generator in the Multi-CartoonGAN model.

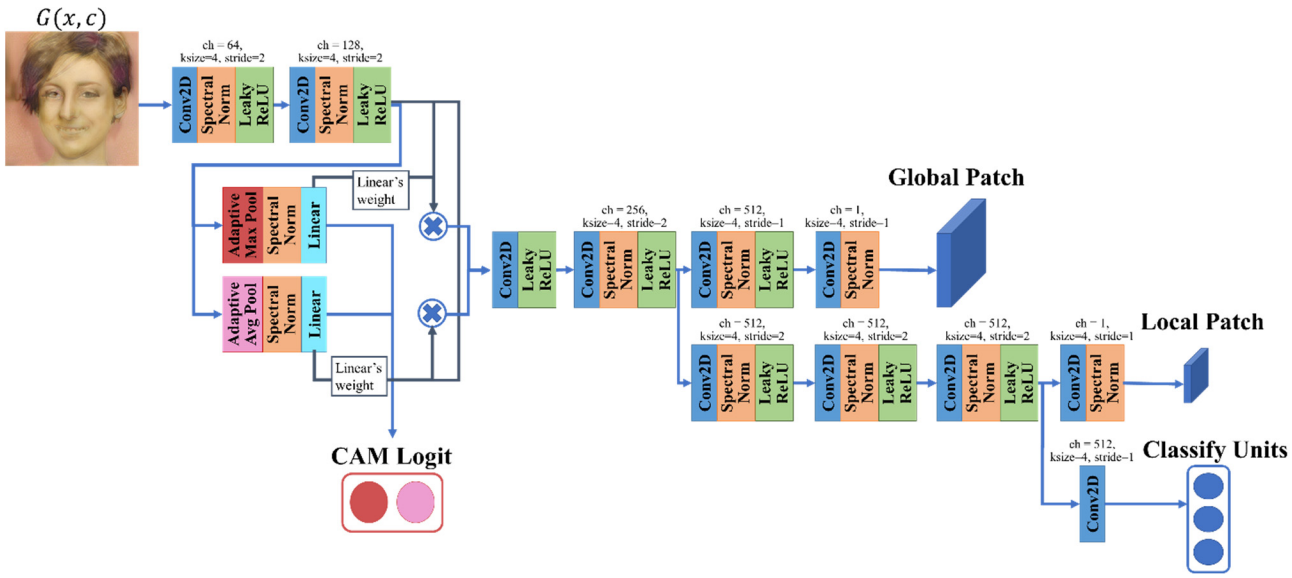


Figure 2. Overview of the discriminator in the Multi-CartoonGAN.

In the training phase, G outputs the translated image $G(x, c)$ by following the content of conditional vector c. Meanwhile, D outputs the feature map, the CAM logit, and the classified units. Discrimination is performed on the feature map patches to determine whether the input image is from the real artistic domain images or a fake translated image from G. Discrimination is also performed on the CAM logit from the attention feature maps via the adaptive pooling layers. The classified units provide the criteria that determine the artistic style that the input image belongs to.

3.2. Conditional Adaptive Layer-Instance Normalization in the Generator

As pointed out in [25], conditional input injections directly into the generator can lead to unstable conditional training. To address this problem, a conditional vector is employed as the layer hyperparameter in our generator. This idea was inspired by AdaLIN [6], which controls the affine parameters so that scaling and shifting through the extracted feature maps is performed by the adaptive max-pooling of the adaptive average pooling layers. Our results suggest that, assuming that a different input for the fully connected layers provides a different output, the affine parameters from the conditional vector can also be used to control flexible scaling and shifting.

Figure 3 presents an overview of the proposed CAdaLIN. In the normalization process, the conditional input c $\in$ C is input to the linear layers, and each layer outputs the affine parameter for scaling and shifting to the normalized feature maps.

3.3. Full Objectives

Equations (1)–(8) show the G and D loss objectives of our Multi-CartoonGAN. Equation (1) is the learning loss objective function for G, which includes content loss, adversarial loss, classification loss, and adversarial loss with the CAM logit. Equation (2) is the objective function for D and includes adversarial loss, classification loss, adversarial loss with CAM logit, and gradient penalty loss.

Based on Equation (1), our model obtains a content loss that compares the differences between x and $G(x, c)$ through the output from conv4-4 in the pretrained VGG19 following the CartoonGAN [7] method. The adversarial loss in Equation (4), the classification loss in Equation (5), and the adversarial loss with the CAM logit in Equation (7) are calculated for D. The adversarial loss in Equation (4) and the CAM logit adversarial loss in Equation (5) are discriminated as real or fake by the least-squares generative adversarial loss (LSGAN) method [28], which adopts the least squares loss when discriminating and performs stable training.

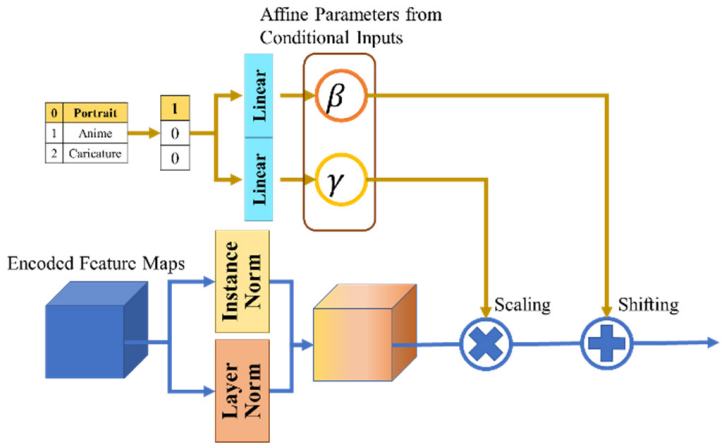


Figure 3. Overview of conditional adaptive instance-layer normalization.

CAM logits are output through adaptive average-pooling and adaptive max-pooling layers. *G* seeks to deceive *D* so that both the feature patches and the CAM logit are treated as real data. In contrast, the classification loss for *G* seeks to be recognized by the classifying network in *D* such that $G(x, c)$ belongs to domain c.

Using Equation (2), the adversarial losses in Equations (4) and (7) are discriminated as real or fake based on the judgments from two outputs: the feature map and the CAM logit. The classification loss in Equation (6) means that *D* is trained to recognize the artistic style from real artistic input images in the collection. To stabilize the adversarial training, we placed spectral normalization [29] in each layer of *D* and the gradient penalty [30].

$$L_G = \lambda_{con} L_{VGG} + \lambda_{GAN} L_{GAN}(G,\ D) + L_{cls}^{G} + L_{cam}^{D}(G,\ D) \tag{1}$$

$$L_D = \lambda_{GAN} L_{GAN}(G,\ D) + L_{cls}^{D} + L_{cam}^{D}(G,\ D) + \lambda_{gp} L_{gp} \tag{2}$$

$$L_{VGG} = \mathbb{E}_{x \sim p_{data}(x),\ c} \| F_{conv4\_4}(G(x,\ c)) - F_{conv4\_4}(x) \|_1 \tag{3}$$

$$\begin{aligned} L_{GAN}(G,\ D) &= \mathbb{E}_{y \sim p_{data}(y)} \left[(D(y))^2 \right] \\ &+ \mathbb{E}_{y_{blur} \sim p_{data}(y_{blur})} \left[(1 - D(y_{blur}))^2 \right] \\ &+ \mathbb{E}_{x \sim p_{data}(x), c} [(1 - D(G(x, c)))^2] \end{aligned} \tag{4}$$

$$L_{cls}^{G} = \mathbb{E}_{x \sim p_{data}(x), c} [-\log(D_{cls}(c|G(x, c)))] \tag{5}$$

$$L_{cls}^{D} = \mathbb{E}_{y \sim p_{data}(y), c} [-\log(D_{cls}(c|y))] \tag{6}$$

$$\begin{aligned} L_{cam}^{D}(G,\ D) &= \mathbb{E}_{y \sim p_{data}(y)} \left[(D_{cam}(y))^2 \right] \\ &+ \mathbb{E}_{y_{blur} \sim p_{data}(y_{blur})} \left[(1 - D_{cam}(y_{blur}))^2 \right] \\ &+ \mathbb{E}_{x \sim p_{data}(x), c} \left[(1 - D_{cam}(G(x, c)))^2 \right] \end{aligned} \tag{7}$$

$$L_{gp} = \mathbb{E}[(\| \nabla_{\hat{x}} D(\hat{x}) \|_2 - 1)^2] \tag{8}$$

4. Experimental Setup

4.1. Real-World Face Image Dataset

As the source domain of this study, we employed the CelebA-HQ dataset [31], which consists of 28,000 cropped face images for training and 2000 similar images for testing. Our study used a conditional translation training dataset based on a deep learning model and a testing dataset that performs evaluations by visualizing a translated image from a testing image and then evaluating the computational performance.

4.2. Artistic Face Images Dataset

As explained above, we evaluated our proposed model by translating real-world face images into three artistic styles: portrait, anime, and caricature. For each artistic style, we employed the MetFaces portrait dataset [32], which consists of facial artworks obtained from the New York Metropolitan Museum of Art. We also used an additional anime face dataset [6], which consists of a collection of animation character images. For caricatures, we selected an additional dataset WebCaricature [33,34]. Before conducting experiments, the face images from the original data were cropped by referring to an annotation file that indicates the axis of the facial points. Regarding the number of images in each art category, we could collect 1336 portrait images, 3400 anime girl images, and 6042 caricature images.

4.3. Training Setup

When loading images from the source and target domains, they were resized to 286 × 286 pixels, after which some of those images, selected at random, were cropped to 256 × 256 pixels. For the training optimizer, we employed the Adam optimization algorithm [35] with the learning rate set to 0.0001 and (β_1, β_2) = (0.5, 0.999). In addition, the optimizer weight decay was set to 0.0001. During the training phase, we trained the model for 50 epochs with a batch size of 15.

4.4. Enviroment about Software and Hardware

We implemented Multi-CartoonGAN and performed the training phase using the open-source PyTorch deep learning library. To speed up training, we employed an NVIDIA GeForce RTX 3090 GPU.

5. Results

This section introduces three aspects of the results obtained: visualizing translated images using the trained Multi-CartoonGAN, comparing the time needed to complete one training epoch with the GPU MiB training cost, and using k-means clustering [36] to determine whether the model could output each translated image style by differentiating appropriately among the target artistic domains.

After conditional translation training, we tested whether our model could output appropriately translated results that followed the conditional inputs and reflected the correct artistic style domain. Figure 4 shows the results of inputting real-world face images for translation.

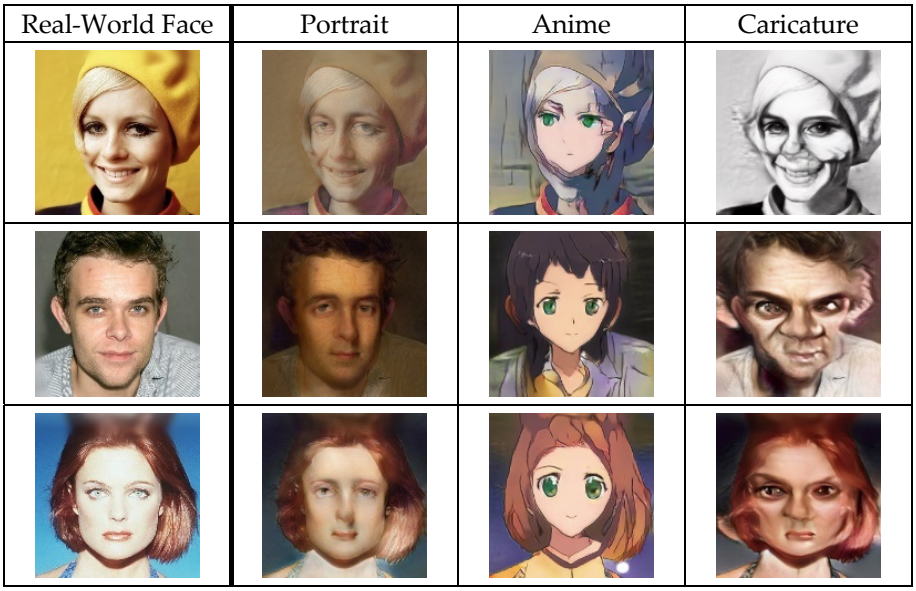


Figure 4. *Cont.*

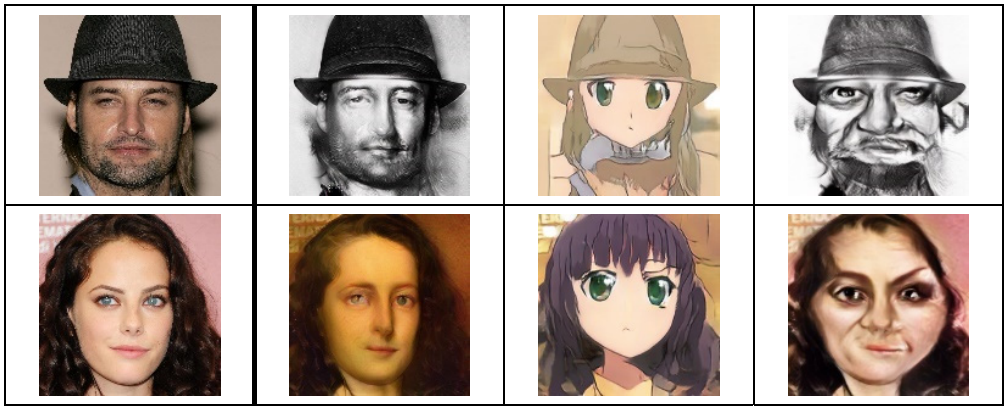


Figure 4. Visualizing conditional artistic face translation using the trained Multi-CartoonGAN.

Viewing the entire field of translated images, it can be seen that each image has unique features that maintain the structure of the content image, such as the face location. By evaluating the features of each translated image, we can see that each image has unique features that reflect the intended artistic style. For example, translating to the *portrait*-style results in images that show paintbrush strokes and subdued colors, while *anime*-style results have unique eye designs that are much bigger than real-world faces. Interestingly, when translating image into anime style, the male face images are forcibly translated into girly faces because we employed an anime face image dataset which consists of almost animation girl characters. Finally, the *caricature* style results show design exaggerations that reflect particular facial characteristics.

These results show that the trained Multi-CartoonGAN can perform conditional translations that reflect the unique features of the three ordered artistic style domains. As for the translation quality of image visualization among various translation models, Figure 5 shows the results of comparing the three styles using the same input image for Multi-CartoonGAN, CycleGAN, U-GAT-IT, NICE-GAN, and StarGAN.

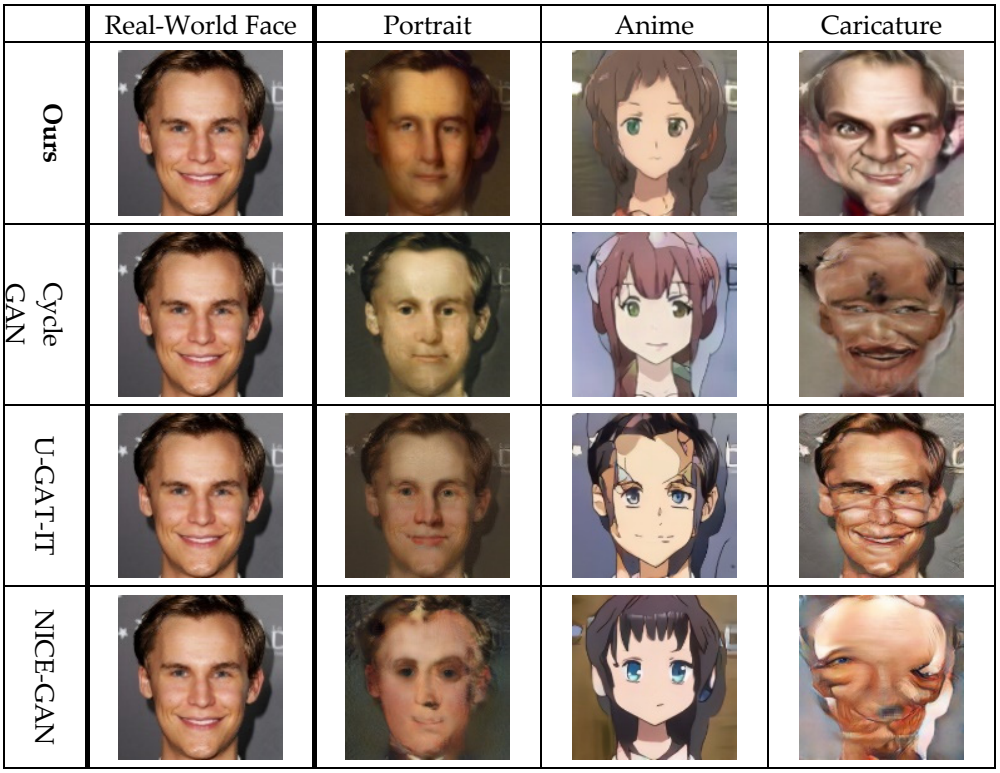


Figure 5. *Cont.*

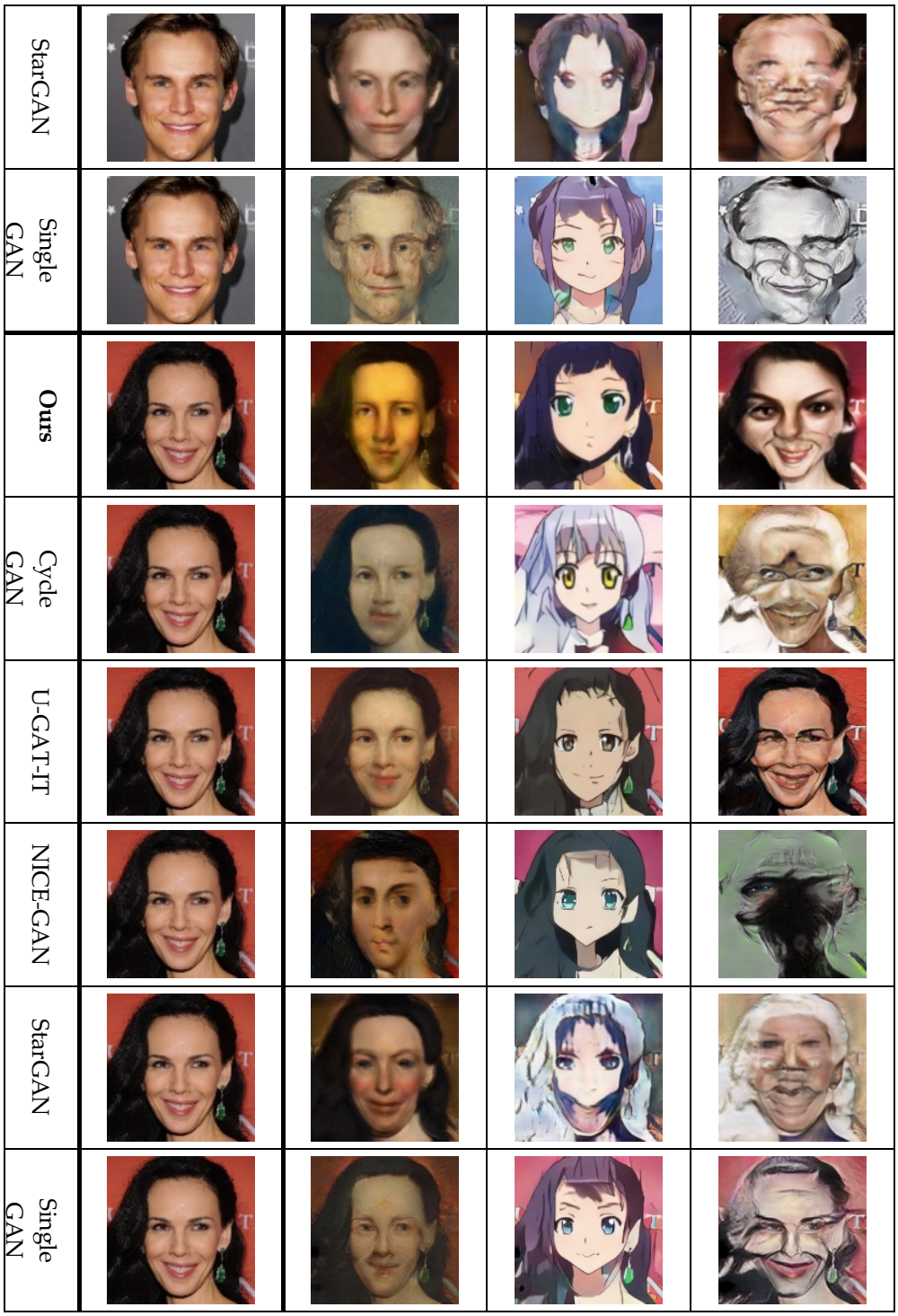


Figure 5. Comparing conditional artistic face translation results among translation models.

In the translation results using trained CycleGAN and NICE-GAN, the portrait style results show appropriate painting styles that maintain the subject's features, while the anime style results show typical anime coloring and facial characteristics. However, when translating to caricature style, the results did not change the shape of the subject's face line, such as Multi-CartoonGAN. Instead, only the coloring was changed, much like a film negative, thus producing coloring results that were much different from the original image. In the translation results using U-GAT-IT, the images translated into portrait style and anime style show the likely coloring and shade painting. However, the images translated

to the caricature style did not drastically change considering the face shape feature. This phenomenon is also visible in SingleGAN's results.

As for StarGAN, the image translation results show coloring changes and feature distortions that are radically different from the intended artistic style. Therefore, we can conclude that our proposed model performs better than StarGAN.

5.1. Comparing Computational Cost and Time to Complete One Poch

This section introduces the performance results related to the computational cost of training on a GPU and the time required to complete one training epoch using our prepared datasets (the iteration number is up to 28,000 based on the real-world face image number).

Figure 6 shows a graph comparing the computational cost of training on a GPU among the translation models. Because Multi-CartoonGAN employs CartoonGAN, which calculates content loss using a pretrained VGGNet, our model could reduce the calculation costs to less than those for U-GAT-IT and NICE-GAN, and Nice-Gan light, even when adopting CAdaLIN for technical normalization.

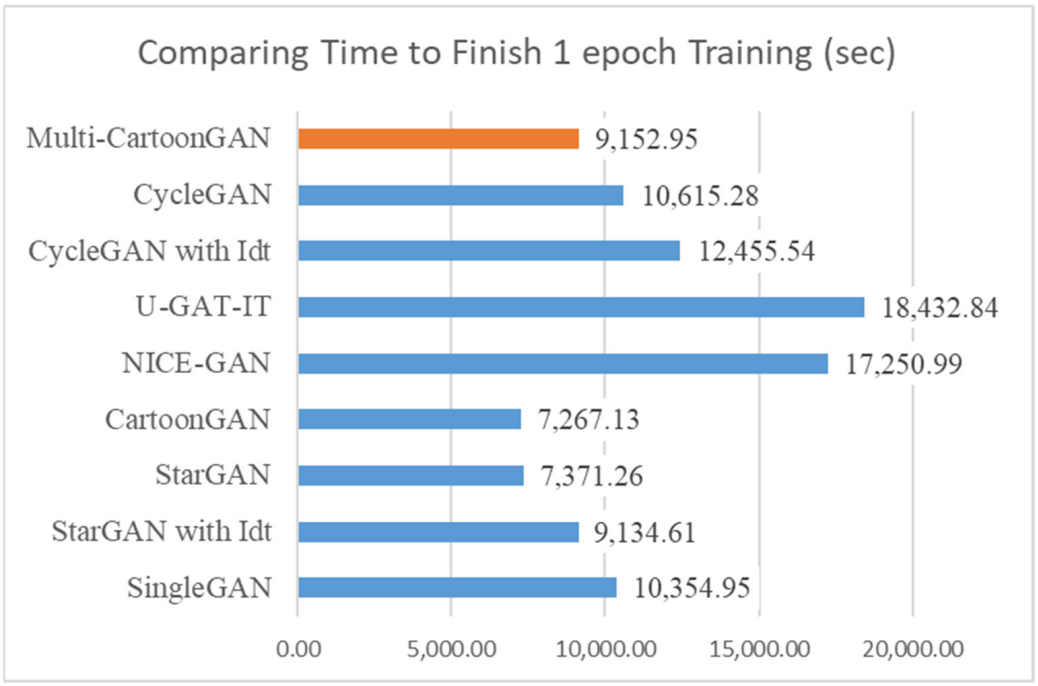


Figure 6. Comparing the GPU computational cost (Idt refers to the model learned identity mapping for preventing extreme color tinting).

As for the time required to complete one training epoch, Figure 7 shows a graph with the comparison results for the counting time when the batch size was set to 1. Here, it can be seen that the identity mapping loss (Idt) for our model was similar to that for StarGAN but faster than that for CycleGAN, U-GAT-IT, SingleGAN, and NICE-GAN, because it was necessary for those models to reuse the generator when calculating content losses.

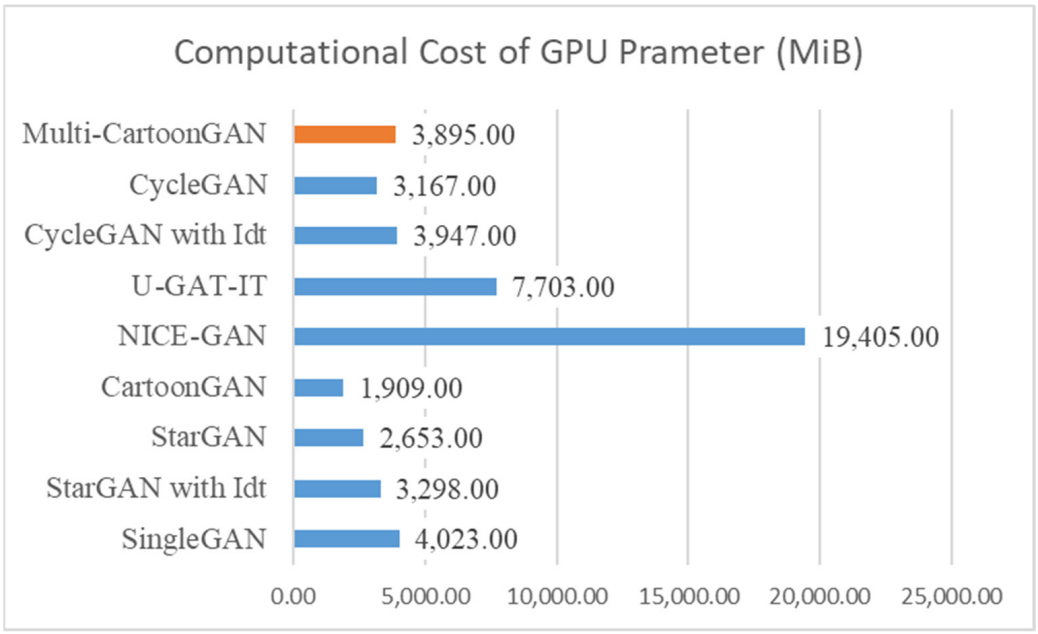
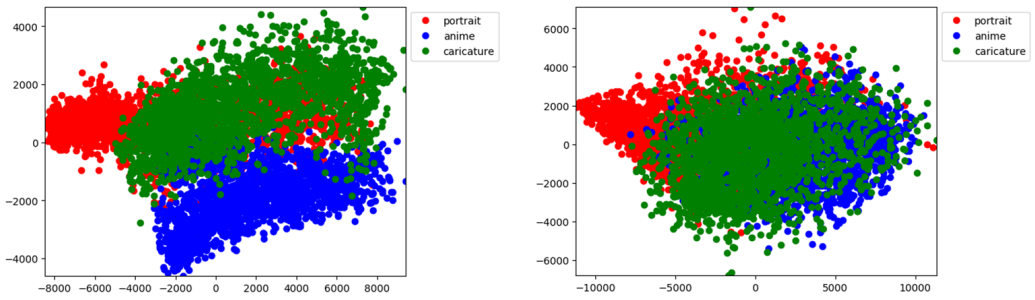


Figure 7. Comparing the time to complete one training epoch (Idt refers to the model learned identity mapping for preventing extreme color tinting).

5.2. Clustering by t-SNE

Using trained versions of image-to-image translation models, we experimented to determine if the models could output images with appropriate differences among the requested target domains. To evaluate these differences, we used t-SNE clustering, which visualizes the similarities over the entire dataset by converting a high-dimensional dataset into a matrix [36]. An advantage of using t-SNE clustering is that it projects high-dimensional data onto a low-dimensional space by preserving the clustering in a high-dimensional space, which helps to visualize the clusters.

Figure 8 shows the clustering results for each image-to-image translation model. In the CycleGAN, Single GAN, StarGAN and U-GAT-IT results, it can be seen that the three clusters were nearly mixed, thus indicating that the translated images for each style had the same features and no major differences.



(**a**) (**b**)

Figure 8. *Cont.*

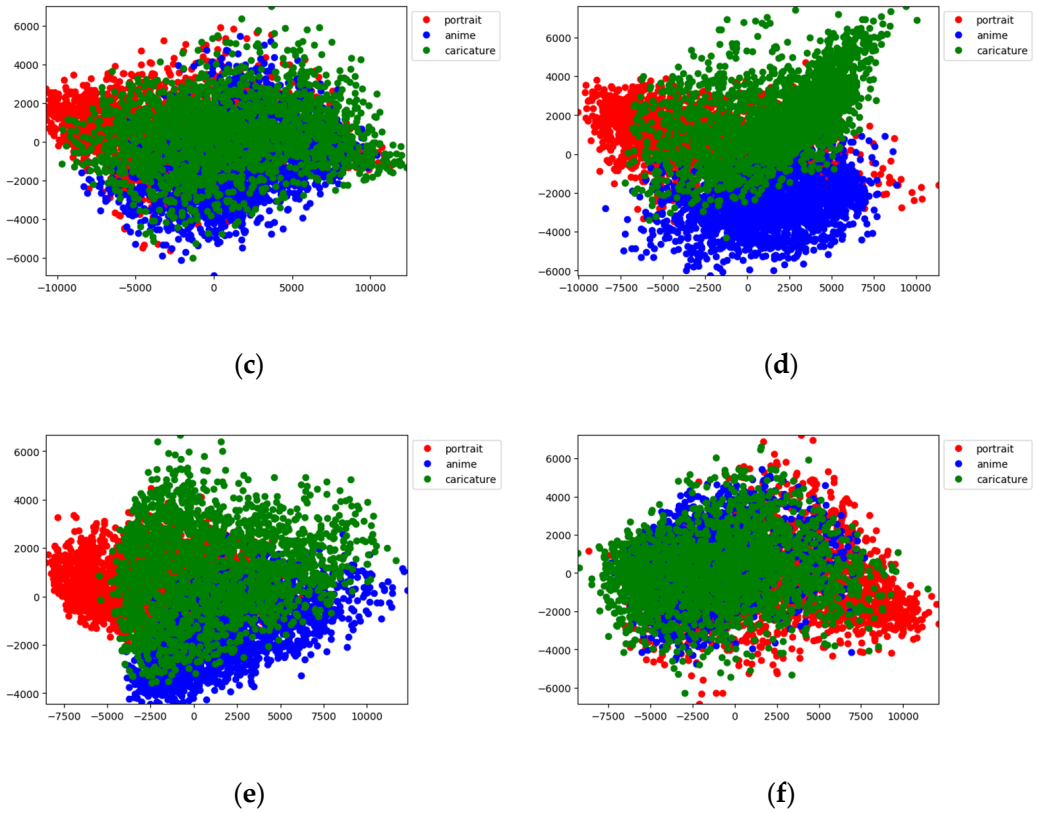


Figure 8. Results of K-means clustering using three translation models. (**a**) t-SNE with our Multi-CartoonGAN, (**b**) t-SNE clustering with CycleGAN, (**c**) t-SNE clustering with U-GAT-IT, (**d**) t-SNE clustering with NICE-GAN, (**e**) t-SNE clustering with StarGAN, (**f**) t-SNE clustering with SingleGAN.

In contrast, the NICE-GAN and Multi-CartoonGAN translation results show clear demarcations between the anime and other styles. The portrait and caricature clusters are somewhat mixed because the art style forms are very similar in terms of describing paintings based on image content. The fact that most anime cluster areas did not mix with the others indicates that the translated anime-style images had the intended characteristic features.

6. Conclusions

In this study, we proposed a conditional translation model capable of handling large appearance changes when the forms of the content and target domains are radically different. Additionally, the model was designed with the aim of keeping the GPU computational costs as low as possible. In our proposed model, which we named Multi-CartoonGAN, we implemented a new conditional input insertion method, named CAdaLIN, and found that our normalization function could perform conditional translations that produced appropriate differences between the intended styles.

Based on the results of our experiments, the following observations can be made:

- The t-SNE cluster results for our model show that more improvements could be seen in the differences between the portrait and caricature styles;
- If the pose and shade features in the real-world face domain are different from those in the target domain, our model tends to output corrupted images;
- Because our model outputs a single translated image from a single content image and a single conditional vector signal, there is little translation diversity from a single input image.

Regarding the first point above, we note that our model often outputs translated caricature style images with blurred colors similar to brush paintings. This causes the translated features results of the portrait and caricature styles to appear nearly mixed.

The second point is the problem between the form of content domains that include a variety of face patterns and directions and the artistic image style domains, which collect face images that are oriented primarily in the front direction. This corruption problem often appeared when attempting translations to anime and caricature styles.

Finally, from our visualization results, we found that some translated images had quite similar designs. This occurs when using a fixed conditional input because there is no additional auxiliary information to tell the generator that it should output different forms of the translated images.

To overcome these issues, we intend to further develop our model based on the following ideas:

- Adding an adversarial loss function that compares the extracted edge features of the target domain images with those of the translated images using a Canny edge filter;
- Concatenating the face mask images in the generator so that the face information can be described in detail;
- Developing our model into a guided multimodal translation architecture, such as StarGAN v2 [37] and MUNIT [38], using appropriate content and style images, and then training the model to output translated images that retain the spatial features of the content image and the painting features of the style image.

N-N domains translations dealing with three different artistic styles ended up limiting the current study to small datasets. In the future, the authors plan to further test the applicability of their model to larger and more diversified datasets. Moreover, the translation evaluation was primarily of a qualitative nature by subjects that are well versed with artistic styles. In the future, the authors plan to extend the qualitative evaluation using the collective intelligence available on Amazon Mechanical Turk to further validate the results, apart from using the Fréchet Inception Distance (FID) and Kernel Inception Distance (KID) quantitative metrics.

Author Contributions: Conceptualization, R.K. and T.G.; methodology, R.K.; software, R.K.; validation, R.K. and T.G.; formal analysis, R.K. and T.G.; resources, R.K. and T.G.; data curation, R.K.; writing—original draft preparation, R.K.; writing—review and editing, T.G.; visualization, R.K. and T.G. supervision, T.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Shustanov, A.; Yakimov, P. CNN Design for Real-Time Traffic Sign Recognition. *Procedia Eng.* **2017**, *201*, 718–725. [CrossRef]
2. Li, T.-H.S.; Kuo, P.-H.; Tsai, T.-N.; Luan, P.-C. CNN and LSTM Based Facial Expression Analysis Model for a Humanoid Robot. *IEEE Access* **2019**, *7*, 93998–94011. [CrossRef]
3. Tian, C.; Xu, Y.; Li, Z.; Zuo, W.; Fei, L.; Liu, H. Attention-guided CNN for image denoising. *Neural Netw.* **2020**, *124*, 117–129. [CrossRef] [PubMed]
4. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv* **2017**, arXiv:1703.10593.
5. Choi, Y.; Choi, M.; Kim, M.; Ha, J.W.; Kim, S.; Choo, J. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation. In Proceedings of the IEEE conference on computer vision and pattern recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8789–8797.
6. Kim, J.; Kim, M.; Kang, H.; Lee, K. U-GAT-IT: Unsupervised Generative Attentional Networks with Adaptive Layer-Instance Normalization for Image-to-Image Translation. In Proceedings of the International Conference on Learning Representations 2020, Addis Ababa, Ethiopia, 26–30 April 2020.
7. Chen, Y.; Lai, Y.-K.; Liu, Y.-J. CartoonGAN: Generative Adversarial Networks for Photo Cartoonization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 9465–9474.
8. Gatys, L.; Ecker, A.; Bethge, M. A Neural Algorithm of Artistic Style. *J. Vis.* **2016**, *16*, 326. [CrossRef]

9. Gatys, L.A.; Ecker, A.S.; Bethge, M. Image Style Transfer Using Convolutional Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2414–2423.
10. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the International Conference on Learning Representations 2015, San Diego, CA, USA, 7–9 May 2015.
11. Johnson, J.; Alahi, A.; Fei-Fei, L. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Berlin, Germany, 2016; Volume 9906, pp. 694–711. [CrossRef]
12. Ulyanov, D.; Vedaldi, A. Instance Normalization: The Missing Ingredient for Fast Stylization. *arXiv* **2016**, arXiv:1607.08022.
13. Dumoulin, V.; Shlens, J.; Kudlur, M. A Learned Representation for Artistic Style. *arXiv* **2016**, arXiv:1610.07629.
14. Huang, X.; Belongie, S. Arbitrary Style Transfer in Real-Time with Adaptive Instance Normalization. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 1510–1519.
15. Isola, P.; Zhu, J.-Y.; Zhou, T.; Efros, A.A. Image-to-Image Translation with Conditional Adversarial Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1125–1134.
16. Yi, Z.; Zhang, H.; Tan, P.; Gong, M. DualGAN: Unsupervised Dual Learning for Image-to-Image Translation. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2849–2857.
17. Kim, T.; Cha, M.; Kim, H.; Lee, J.K.; Kim, J. Learning to Discover Cross-Domain Reactions with Generative Adversarial Networks. In Proceedings of the International Conference on Machine Learning, PMLR, Lanzarote, Canary Islands, 9–11 April 2018; pp. 1857–1865.
18. Chen, J.; Liu, G.; Chen, X. AnimeGAN: A Novel Lightweight GAN for Photo Animation. In *International Symposium on Intelligence Computation and Applications*; Springer: Singapore, 2019; pp. 242–256.
19. Anoosheh, A.; Agustsson, E.; Timofte, R.; Gool, L.V. ComboGAN: Unrestrained Scalability for Image Domain Translation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018; pp. 783–790.
20. Mizra, M.; Osindero, S. Conditional Generative Adversarial Nets. *arXiv* **2014**, arXiv:1411.1784.
21. He, Z.; Zuo, W.; Kan, M.; Shan, S.; Chen, X. Arbitrary Facial Attribute Editing: Only Change What You Want. *arXiv* **2017**, arXiv:1711.10678.
22. Perarnau, G.; Weijer, J.-V.-D.; Rauducanu, B.; Álvarez, J.M. Invertible Conditional GANs for image editing. *arXiv* **2016**, arXiv:1611.06355.
23. Wu, P.-W.; Lin, Y.-J.; Chang, C.-H.; Chang, E.Y.; Liao, S.-W. RelGAN: Multi-Domain Image-to-Image Translation via Relative Attributes. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 5914–5922.
24. Chen, X.; Xu, C.; Yang, X.; Song, L.; Tao, D. Gated-GAN: Adversarial Gated Networks for Multi-Collection Style Transfer. *IEEE Trans. Image Process.* **2018**, *28*, 546–560. [CrossRef] [PubMed]
25. Yu, X.; Ying, Z.; Li, T.; Liu, S.; Li, G. Multi-Mapping Image-to-Image Translation with Central Biasing Normalization. *arXiv* **2018**, arXiv:1806.10050.
26. Yu, X.; Cai, X.; Ying, Z.; Li, T.; Li, G. SingleGAN: Image-to-Image Translation by a Single-Generator Network Using Multiple Generative Adversarial Learning. In Proceedings of the Asian Conference on Computer Vision, Perth, Australia, 2 December 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 341–356. [CrossRef]
27. Chen, R.; Huang, W.; Huang, B.; Sun, F.; Fang, B. Reusing Discriminators for Encoding: Towards Unsupervised Image-to-Image Translation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 8168–8177.
28. Mao, X.; Li, Q.; Xie, H.; Lau, R.Y.K.; Wang, Z.; Smolley, S.P. Least Squares Generative Adversarial Networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2794–2802.
29. Miyato, T.; Kataoka, T.; Koyama, M.; Yoshida, Y. Spectral Normalization for Generative Adversarial Networks. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
30. Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A. Improved Training of Wasserstein GANs. *arXiv* **2017**, arXiv:1704.00028.
31. Karras, T.; Aila, T.; Laine, S.; Lehtinen, J. Progressive Growing of GANs for Improved Quality, Stability, and Variation. *arXiv* **2017**, arXiv:1710.10196.
32. Karras, T.; Aittala, M.; Hellsten, J.; Laine, S.; Lehtinen, J.; Aila, T. Training Generative Adversarial Networks with Limited Data. *arXiv* **2020**, arXiv:2006.06676.
33. Huo, J.; Li, W.; Shi, Y.; Gao, Y.; Yin, H. WebCaricature: A Benchmark for Caricature Recognition. In Proceedings of the British Machine Vision Conference, Newcastle, UK, 3–6 September 2018.
34. Huo, J.; Gao, Y.; Shi, Y.; Yin, H. Variation Robust Cross-Modal Metric Learning for Caricature Recognition. In *ACM Multimedia Thematic Workshops*; Association for Computing Machinery: New York, NY, USA; Mountain View, CA, USA, 2017; pp. 340–348. [CrossRef]
35. Kingma, D.P.; Ba, J.L. Adam: A Method for Stochastic Optimization. In Proceedings of the International Conference on Learning Representations 2015, San Diego, CA, USA, 7–9 May 2015.

36. van der Maaten, L.; Hinton, G. Visualizing Data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
37. Choi, Y.; Uh, Y.; Yoo, J.; Ha, J.-W. StarGAN v2: Diverse Image Synthesis for Multiple Domains. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 8188–8197.
38. Huang, X.; Liu, M.-Y.; Belongie, S.; Kautz, J. Multimodal Unsupervised Image-to-Image Translation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 172–189.