Towards Scalable and Diverse Unpaired Image-to-Image Translation: Following the Road Map Generative Adversarial Networks

Yue Kuang* Ziqiao Ma* Zhuowen Shen* Ruobing Wang*
Univerisity of Michigan
{kuangyue,marstin,mickshen,rubywang}@umich.edu

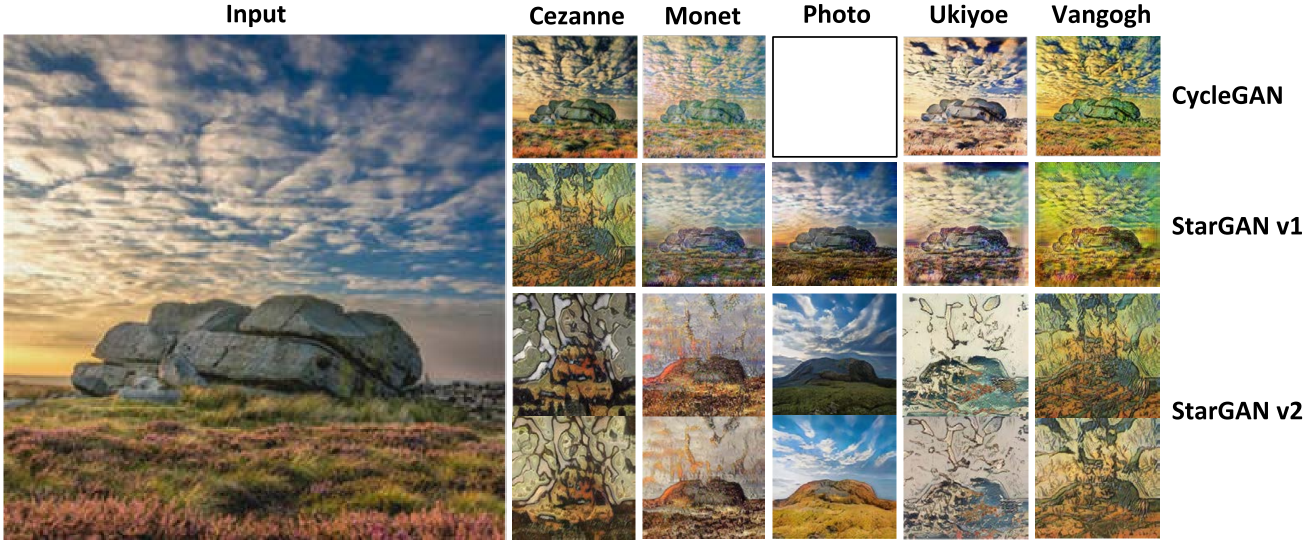


Figure 1: The unpaired image-to-image translation experiment results for CycleGAN, StarGAN v1 and StarGAN v2 on each domain of the collection style transfer dataset.

Abstract

Image-to-image translation aims at learning a mapping between an input image and an output image using a training set of aligned image pairs. However, for many tasks, paired training data are not available, making it challenging to map images from one domain to another. In this project, we investigate state-of-the-art methods in unpaired image-to-image translation and present our work in reproducing three successful generative adversarial network models for translation. The baseline is CycleGAN [22], a successful approach in image-to-image translation for two domains. To enable scalability over multiple domains, we reproduced StarGAN v1 [2], a model that can perform image-to-image translations for multiple domains. To generate diverse images across multiple domains, we further reproduced StarGAN v2 [3]. Qualitative comparisons and quantitative evaluations will be presented on neural style transfer task where paired training data does not exist. We confirm the scalability superiority of both versions of StarGAN over CycleGAN, and the improvement in the diversity of generated images from StarGAN v2.

1. Introduction

Traditional Image translation models require dataset comprised of paired examples, which is expensive to prepare. In concern of this problem, unpaired image-to-image translation aims to learn a mapping between different visual domains and change a particular aspect of a given image to another [10]. Specifically, the term `domain` refers to a set of images that can be grouped as a visually distinctive labels, while within each `domain`, images can take on a unique appearance, defined as their `style` [3].

Neural style transfer is a typical unpaired image-to-image translation task, which synthesizes a collection of images by combining the content of one image with the style of another image, typically a painting [6]. The intuition be-

*indicates equal contribution

hind the task is that, with the knowledge of the collection of an artist's paintings and the collection of landscape photographs, we can reason about the stylistic differences between the two collections and imagine what a scene might look like if we were to "translate" it from one collection to the other.

The quality of unpaired image-to-image translation has experienced significant development following the introduction of generative adversarial networks (GANs). This quality improvement is particularly remarkable in terms of domain scalability and style diversity. CycleGAN [22] is an early successful approach in unpaired image-to-image translation task. However, it cannot be applied directly to more than two domains. In solution to such concerns, StarGAN [2], a model that can perform translations for multiple domains was proposed. StarGAN v2 [3] make a step further on its path to improve both the diversity and scalability of the generated images.

In this project, we reproduce these successful unpaired image-to-image translation models. Using collections of landscapes and paintings of four famous artists, we train the model to learn how to render natural photographs into the respective styles, and vice versa. We present the qualitative comparisons and quantitative evaluations and confirm the scalability superiority of both versions of StarGAN over CycleGAN, and the improvement in the diversity of generated images from StarGAN v2.

2. Related Work

2.1. Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) [7] were proved remarkable powerful in various computer vision tasks, such as image generation [5], image translation [22, 2, 3], super-resolution imaging [13], and face image synthesis [20, 2, 3]. The networks consist of two models: a generator and a discriminator. The generator tries to create as realistic examples as possible to deceive the discriminator. The discriminator tries to distinguish fake examples from true examples. Both the generator and discriminator improve through adversarial learning. This adversarial process gives GANs notable advantages over other generative algorithms. [8].

2.2. Image-to-Image Translation

Recent work have achieved impressive results in image-to-image translation. For instance, pix2pix [10] learns this task in a supervised manner using cGANs [15]. It combines an adversarial loss with a L1 loss, which requires paired data samples. To alleviate the problem of obtaining data pairs, unpaired image-to-image translation frameworks have been proposed. CycleGAN [22], DiscoGAN [12], and DualGAN [18] have achieved remarkable success in image-to-image translation with unpaired training data. However, all these frameworks are only capable of learning the relations between two different domains at a time. StarGAN [2, 3] and CoGAN [14] well solve this problem which can perform image-to-image translations for multiple domains using only a single model.

2.3. Neural Style Transfer (NST)

Neural Style Transfer (NST) has recently become an active research area, motivated by both scientific challenges and industrial demands that re-drawing an image in a particular style requires a well-trained artist and lots of time. [11] summarized current NST methods into two categories: *image-optimisation-based online neural methods* which iteratively optimises an image, or *model-optimisation-based offline neural methods* which optimises a generative model offline and produces the stylised image with a single forward pass. Many GAN models, falling into the second type, were proved to be promising by the NST community and has become a trending direction [19, 21]. The CycleGAN [22] model is a noticeable milestone as it is the first work that formulate NST tasks as an application of unpaired translation. With the rapid development of unpaired image-to-image translation models, we are eager to see how state-of-the-art GANs can address NST problems.

3. Method

3.1. Problem Formulation

We first adapt and develop a unified sets of notations to formulate the task and describe the computational models.

Let X be the original domain of images with training samples $\{x_i\}_{i=1}^N$, and and $\mathcal{Y} = \{Y_1, Y_2, \cdots, Y_c\}$ be the domains of target images with label space $C = \{1, 2, \cdots, c\}$. The unpaired image-to-image translation can be abstracted as a task that builds a mapping $G : (X, \cdot) \to Y \in \mathcal{Y}$ such that given a source image $x \in X$, the output $\hat{y} = G(x)$ is indistinguishable from images $y \in Y$.

3.2. Computational Models

CycleGAN CycleGAN [22] performs unsupervised training using a collection of images from the source and target domains that do not need to be associated in any way. It is a cross-domain model, and addresses one original domain X and one target domain $Y \in \{Y_1, Y_2, \cdots, Y_c\}$ at a time.

As is illustrated in Figure 2, the model includes two generators $G : X \to Y$ and $F : Y \to X$ and two adversarial discriminators D_X and D_Y. D_X learns to distinguish between images $\{x\}$ and translated images $\{F(y)\}$; in the same way, D_Y learns to distinguish between images $\{y\}$ and translated images $\{G(x)\}$. The objective contains two types of terms: adversarial losses for matching the distribution of generated images to the data distribution in the tar-

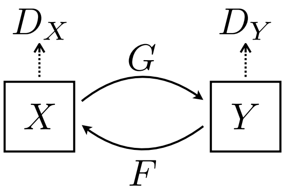


Figure 2: CycleGAN [22] model structure: two generators G and F, and associated adversarial discriminators D_Y and D_X.

get domain; and cycle consistency losses, $||x - F(G(x))||_1$ and $||y - G(F(y))||_1$, for capturing the intuition that if we translate from one domain to the other and back again we should arrive at where we started. The complete model can be found in the Appendix A in Figure 5.

However, since different approaches should be built independently for every pair of domains, it cannot be used directly for more than two domains. With k domains, CycleGAN will have to train $k^2 - k$ generators to provide a complete translation between all domains, which is computationally inefficient.

StarGAN Version 1 (StarGAN v1) CycleGAN is ineffective in multi-domain image translation tasks because it takes one target domain $Y \in \mathcal{Y}$ each time. In solution to such concerns, StarGAN v1 [2], one of the most influential GANs in the community, was proposed.

To achieve this, the generator G now translate an input image $x \in X$ into an output image $\hat{y}$ conditioned on the target domain label $c \in C$, *i.e.* $G : (X, C) \to Y \in \mathcal{Y}$. The target domain label c is randomly generated so that G learns to flexibly translate the input image. With a conditional label input, the problem can now be framed as a Conditioned Image Synthesis task, so an auxiliary classifier [16] can be introduced to enable a single discriminator to control multiple domains in $\mathcal{Y}$. The discriminator D produces probability distributions over both sources and domain labels, $D : x \to \{D_X(x), D_\mathcal{Y}(x)\}$. The complete model can be found in the Appendix A in Figure 6

Given the huge success of StarGAN v1 in solving the efficiency problem by training on multiple domains at once by indicating each domain with a predetermined label, the in-domain diversity have been the concern. This version of StarGAN learns a deterministic mapping per domain, contradicting the multi-modal nature of the data distribution.

StarGAN Version 2 (StarGAN v2) Compared to the StarGAN, StarGan v2 [3] makes a step further on its path to addressing the diversity of the generated images.

StarGAN v2 embeds its domain label c into a domain-specific style code s that can represent diverse styles of a specific domain. The generator G instead translate an input image $x \in X$ into an output image $\hat{y} = G(x, s)$ given the style code s. Specifically, given the domain $Y \in \mathcal{Y}$, the cross-domain style code is generated by the mapping network F from a latent code z, and the in-domain style code is generated by the style encoder E from an reference image $x \in X$.

$$s_1 = F_Y(z), \qquad s_2 = E_Y(x)$$

The complete model can be found in the Appendix A in Figure 6

4. Experiments

4.1. Dataset

One major benefits of StarGANs is their ability to map one input image to multiple outputs simultaneously. In order to compare the performance of CycleGAN and StarGANs, we want our dataset having multiple output domains corresponding to one input domain. We chose the collection of style transfer dataset used in CycleGAN [22], in which input images are transferred into the artistic styles of Monet, Van Gogh, Cezanne, and Ukiyo-e.

4.2. Evaluation Metrics

To quantitatively evaluate and compare the performances of CycleGAN [22] and two versions of StarGAN [2, 3], we use two metrics: the Fréchet Inception Distance (FID) Score [9] and accuracy measured by a style classifier.

Fréchet Inception Distance (FID) The FID score [9] is designed to evaluate the performance of GANs by the evaluating the statistical similarity of a set of generated images against a set of real images from the target domain. The score is developed upon a suitable feature function φ, most commonly, the Inception network's convolutional feature. Let p_r and p_g be the probability distribution of the original data and generated images. The FID models $\varphi(p_r)$ and $\varphi(p_g)$ are considered as Gaussian random variables with empirical means μ_r and μ_g and empirical covariance Σ_r and Σ_g.

The FID score is calculated by the following formula:

$$\text{FID}(p_r, p_g) = ||\mu_r - \mu_g|| + tr\Big(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}\Big)$$

From the formula above, a lower FID score indicate better quality images. For identical sets of images, the FID score vanishes to zero.

Style Classifier The FID metric cannot well handle the overfitting problem [1]. In solving this concern, we train a extra style classifier. If the GAN overfits on the dataset, we should expect to observe a small FID value but a poor classification accuracy.

The classifier is pre-trained on the ImageNet dataset [4] first and then fine-tuned on a dataset containing four artists' paintings (3401 images in total, 80%/10%/10%/ splitting for training, validation and testing) using a ResNeXt architecture [17]. It obtained a near-perfect test accuracy 98.5%.

4.3. Results[1]

We trained the three GAN models on the dataset with hyperparameters chosen as the advised ones from the official implementations. A example batch of the synthesized images is summarized in Figure 1 on the front page. The two rows on StarGAN v2 represent synthesized images from the same domain with multi-modal styles.

The FID scores were measured on the fake images against the real works of the artists. The results for the three models are shown in Table 1. According to the table, StarGAN v2 generated fake images that has a closest statistical distribution to the real images, therefore, performs best among the models.

	Original	CycleG	StarG v1	StarG v2
Cezanne	487.42	392.86	397.39	**363.15**
Monet	424.61	376.58	363.47	**241.03**
Ukiyo-e	509.92	443.21	**438.93**	443.67
Vangogh	420.86	393.99	335.68	**312.55**

Table 1: FID scores for models evaluation. The ”Origin” column measures the similarity between source photos and different artists' paintings.

To check if the low FID score is a result of overfitting, we then classify fake images generated by CycleGAN, StarGAN v1 and StarGAN v2 using the above-mentioned classifier individually and report the performance in the Table 2. We noticed that CycleGAN and StarGAN v2 perform similarly and slightly better than StarGAN v1. Given that StarGAN v2 had a significantly lower FID score but a almost equal classification accuracy compared to CycleGAN, we suspect that the StarGAN v2 model might have been slightly overfitted.

Model	CycleG	StarG v1	StarG v2
Accuracy	67.11%	64.52%	**67.83%**

Table 2: Accuracy for model evaluation.

4.4. Error Analysis

Some synthesized images of CycleGAN and StarGAN v1 are very close to the original ones (Figure 3), and this is particularly the case for source images with huge difference from target style. For such images, it may take longer training for the model to translate it into desirable styles.

The synthesized images of StarGAN v2 are highly dependent on the reference image chosen for in-domain style diversity. As is shown in Figure 4, when the reference image has a similar structure to the source image, the synthesized image would look promising, and vice versa.

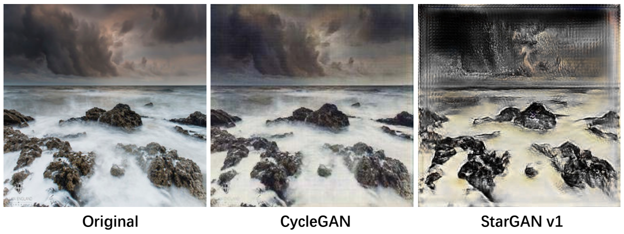


Figure 3: The synthesized images for Ukiyo-e's style produced by CycleGAN and StarGAN v1. They look similar to the original image, which is far from the style of Ukiyo-e.

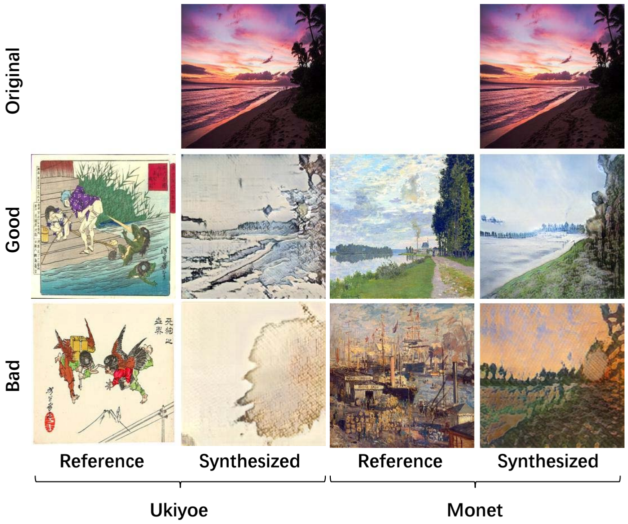


Figure 4: The synthesized images of StarGAN v2 are highly dependent on the reference image chosen for in-domain style diversity.

5. Conclusions

In this project, we compare and evaluate the performances of CycleGAN, StarGAN v1 and StarGAN v2 on unpaired image-to-image translation tasks. While all of them perform well, CycleGAN has the limitation of being only capable of learning the relations between two different domains at a time. Both StarGANs are more efficient by being able to train and test over multiple domains simultaneously. Compared with the other two models, StarGAN v2 generate images that are closer to real works. What's more, it also brings diversity in styles of the images.

[1] The source code for this project can be access at GitHub.

References

[1] Mikołaj Bińkowski, Dougal J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018.

[2] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation, 2018.

[3] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains, 2020.

[4] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[5] Emily L Denton, Soumith Chintala, arthur szlam, and Rob Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28, pages 1486–1494. Curran Associates, Inc., 2015.

[6] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2414–2423, 2016.

[7] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[8] Jie Gui, Zhenan Sun, Yonggang Wen, Dacheng Tao, and Jieping Ye. A review on generative adversarial networks: Algorithms, theory, and applications, 2020.

[9] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018.

[10] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks, 2018.

[11] Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, Yizhou Yu, and Mingli Song. Neural style transfer: A review. *IEEE transactions on visualization and computer graphics*, 2019.

[12] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. volume 70 of *Proceedings of Machine Learning Research*, pages 1857–1865, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

[13] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network, 2017.

[14] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. *CoRR*, abs/1606.07536, 2016.

[15] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

[16] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans, 2017.

[17] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks, 2017.

[18] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. *CoRR*, abs/1704.02510, 2017.

[19] Lvmin Zhang, Yi Ji, and Xin Lin. Style transfer for anime sketches with enhanced residual u-net and auxiliary classifier gan, 2017.

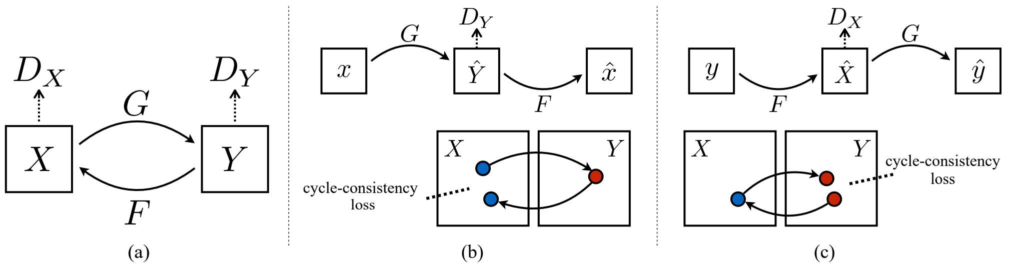
[20] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder, 2017.

[21] A. Zhu, X. Lu, X. Bai, S. Uchida, B. K. Iwana, and S. Xiong. Few-shot text style transfer via deep feature similarity. *IEEE Transactions on Image Processing*, 29:6932–6946, 2020.

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

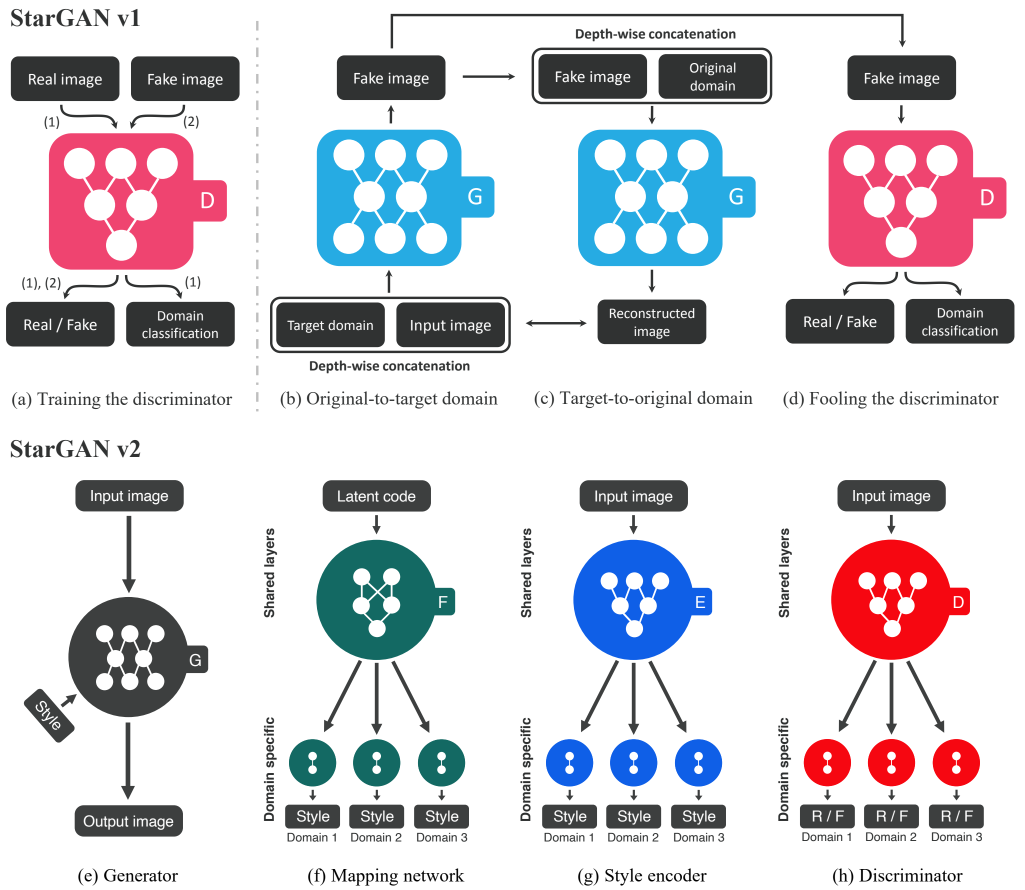
A. Appendix

The complete GAN architectures of CycleGAN, StarGAN v1 and StarGAN v2 are placed in this section.



(a) (b) (c)

Figure 5: (a) CycleGAN [22] contains two mapping functions $G : X \rightarrow Y$ and $F : Y \rightarrow X$, and associated adversarial discriminators D_Y and D_X. D_Y encourages G to translate X into outputs indistinguishable from domain Y , and vice versa for D_X and F. To further regularize the mappings, two cycle consistency losses captures the intuition that translating from one domain to the other and back again should arrive at where it is started: (b) forward cycle-consistency loss: $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$, and (c) backward cycle-consistency loss: $y \rightarrow F(y) \rightarrow G(F(y)) \approx y$



(a) Training the discriminator (b) Original-to-target domain (c) Target-to-original domain (d) Fooling the discriminator

(e) Generator (f) Mapping network (g) Style encoder (h) Discriminator

Figure 6: Overview of StarGAN v1 [2] and v2 [3]. StarGAN v1 consists of two modules, a discriminator D and a generator G. StarGAN v2 consists of two extra modules, a Mapping Network F and a Style Encoder E. (a) D learns to distinguish between real and fake images and classify the real images to its corresponding domain. (b) G takes in as input both the image and target domain label and generates an fake image. The target domain label is spatially replicated and concatenated with the input image. (c) G tries to reconstruct the original image from the fake image given the original domain label. (d) G tries to generate images indistinguishable from real images and classifiable as target domain by D. StarGAN v2 consists of four modules as well. (e) G translates an input image into an output image reflecting the domain-specific style code. (f) The F transforms a latent code into style codes for multiple domains, one of which is randomly selected during training. (g) The E extracts the style code of an image, allowing the generator to perform reference guided image synthesis. (h) D distinguishes between real and fake images from multiple domains. All modules except the G contain multiple output branches, one of which is selected when training the corresponding domain.