Toon2real: A GAN Based Approach for Translating Cartoons to Photorealistic Images

Labiba Kanij Rupty[1], K M Arefeen Sultan[2], Mohammad Imrul Jubair[3], Sayed Hossain Khan[4], and MD. Nahidul Islam[5]

[1] Ahsanullah University Of Science And Technology, Dhaka, Bangladesh
labknr98@gmail.com
[2] krsultan069@gmail.com
[3] mohammadimrul.jubair@ucalgary.ca
[4] sayedhossainkhan36@gmail.com
[5] nahidul19967@gmail.com

Abstract. In terms of Image-to-image translation, Generative Adversarial Networks (GANs) has achieved great success even when it is used in the unsupervised dataset. In this work, we aim to translate cartoon images to photorealistic images using GAN. We apply several state-of-the-art models to perform this task; however, they fail to perform good quality translations. We observe that shallow difference between these two domains causes this issue. Based on this idea, we propose a method based on CycleGAN model for image translation from cartoon domain to photorealistic domain. To make our model efficient, we implemented Spectral Normalization which added stability in our model. We demonstrate our experimental results and show that our proposed model has achieved the *lowest Fréchet Inception Distance score* and better results compared to other state-of-the-art techniques, such as UNIT and Single-GAN.

Keywords: Cartoon-to-real, Image-to-image-translation, GANs

1 Introduction

Cartoons occupy a huge part in our entertainment sector. Film industries, in recent days, are remaking movies from the popular past cartoons and presenting them for current generation. Such an example is – the upcoming *The Lion King (2019)* from *The Lion king (1994)*. Therefore, we realize the necessity of recreating realistic images from the cartoons which can contribute to photorealistic rendering in computer graphics as well as in film industries. In this paper, we propose an approach which converts images from cartoons into their corresponding photorealistic images. From Figure 1, we can see an outcome of our work where the cartoon scene is translated into a photorealistic one.

Image-to-image translation using Generative Adversarial Network (GAN)[3] has been one of the most desiring fields of deep learning research lately. In a GAN architecure, a discriminator network tries to measure the probability

of whether an image has come from an authentic data source or from a fake generated source of the generator. On the other hand, the generator tries to maximize the probability of the discriminator's making mistake and while doing that, it learns to generate more accurate data as much as close to real data. Tremendous success of GANs [3], led other researchers to work on unsupervised settings of datasets, such as [25], [10]. Although these models have succeeded to translate one domain of image to another in general, there hasn't been any specific research for generating images of the photorealistic domain from cartoon domain. This is an extremely hard task; the reason is the domain gap between these two distributions is too shallow.

(a) Cartoon scene (b) Our result

Fig. 1. An example of cartoon to real world translation. (a) *Input image*: which is from the animated film "My Neighbour Totoro". (b) *Our result*: transforming the cartoon image (a) to real world image.

As a result, the discriminator can be easily erroneous to determine the generated data as real ones. This is the reason why most state-of-the-art models tend to fail in case of generating cartoon to real images. We illustrate in our result section that several models intent to keep the original content of cartoon domain while generating photo-realistic images.

To satisfy our objective on this task, we have taken an approach built on CycleGAN[25]. We implemented spectral normalization technique[17] which helps our model to converge faster. Our approach also keeps the content of photo-realistic domain. In addition to these, we also created our own dataset for our model. We show that our method has the lowest FID score than the other baseline models and also, it tends to show more stabilization in training than the others.

2 Related Works

In this section, we review on different relevant variations of GAN and several past works on image-to-image translations.

GANs have achieved great results in various image generation tasks, which are image super-resolution[8], image-to-image translation[25], [23], [13], text-to-image synthesis[24], [19] etc. For stabilizing and improving the training of GAN, several works were proposed such as adding weight normalization and regularization techniques [4], [17], designing new generative architectures[18], [9] to improve visual results and modifying learning objectives [2], [16]. *Miyato et al.*[17] first proposed spectral normalization technique which constrains the Lipschitz constant of the discriminator network by limiting the spectral norm of each layer.

Recently, GAN[3] based approach has given tremendous results in image-to-image translation tasks. *Zhu et al.*[25] proposed a cycle consistency loss to reduce the infinite mappings of input images to any distribution in the target domain. *Adversarial loss* alone can't solve the random permutation mappings of target distribution, rather it helps the input image to be translated into target domain. Similar to CycleGAN, *Kim et al.* [10] proposed a method for preserving the key attributes between the input and the transformed image, while maintaining a cycle consistency criterion. Similarly, *Yi et al.* [22] proposed dual-GAN mechanism based on dual learning from natural language translation[5]. In UNIT[13] framework, *Liu et al.* proposed a shared-latent space assumption, which denotes that the pair of corresponding images in different domains can be mapped to a same latent representation in a shared-latent space. *Liu et al.* used the combination of generative adversarial network(GAN), based on CoGAN[14] and variational autoencoders(VAEs)[11], [12], [20]. Similarly to CycleGAN[25], in SingleGAN[23] framework, the authors used *cycle consistency loss*[25], where one generator is used instead of using two generators[25]. In our previous work, CycleGAN based model was implemented to transform cartoon images into photo-realistic domain[21]. While all these methods achieve compelling results, they take too much time for training. The reason behind it is that these models are not stabilized during training. Even after the training is finished, the models produce blurry or less vibrant results. In the next section we discuss on the approach we took to solve these fundamental issues and to obtain better outcomes.

3 Formulation

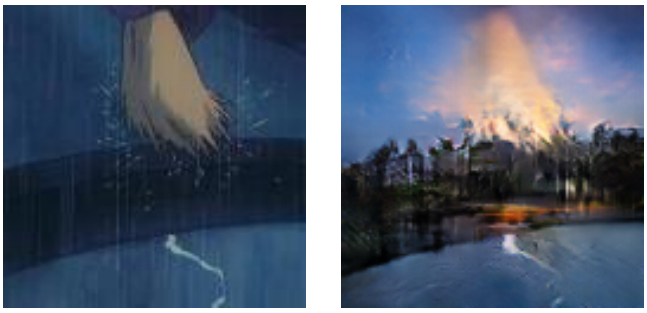
Our main objective is to transform *cartoon* images to *photo-realistic* images by learning the mapping of a *cartoon* domain C to the *photo-realistic* domain R. To satisfy this objective, we adopted *generative adversarial network*[3] to train our models where, as mentioned earlier, *two* networks will simultaneously learn the probability distribution of the domains, C and R to defeat each other respectively. In our case, the first network is a generator, $G_r : C \rightarrow R$ which learns the mapping between the distribution of domain C and R and will generate fake images r_{fake}, matched to domain R using the mapping, whereas the second network, Discriminator D_r will learn the probability distribution of domain R and try to differentiate between the generated images, r_{fake} and the images r from domain R. Our overall process is discussed below including our dataset preparation.

Dataset Collection: Due to the lack of paired data between cartoon domain and photo-realistic domain, we took an approach to collect unpaired dataset for both domains. We collected around $3.5K$ images from various *cartoon* movies and scraped $6.2K$ images from *Flickr* which were tagged as several categories like—*scenery, sunrise, sunset, sea, sky, beach* etc.

Adversarial Loss: Although in [3], a binary cross-entropy based Adversarial Loss function was proposed, we use a *Least Squares Loss(LSGAN)* function for our training. According to *Mao et. al* [15], we have explored that *LSGAN* performs better in the case of vanishing gradient problem and thus shows more stability during training and produces much more higher quality images in the case of *Image-to-image Translation*. So, our adversarial loss stands as follows -

$$For\, Generator\, G_r,\, \mathcal{L}_{G_r} = \frac{1}{m}\sum_{i=1}^{m}(1 - D_r(G_r(c)))^2 \tag{1}$$

However, due to the deep similarities between cartoon and photo-realistic images, we observed that, using only a single generator fails to map the differences between these *two* domains. To resolve this issue, we use *two* additional networks in our model, where a generator, G_c tries to generate images of *Cartoon* domain and a discriminator, D_c tries to discriminate the generated image from *cartoon* domain. The additional networks also perform according to the previously mentioned loss function.



(a) Input from Cartoon Domain

(b) Output without Reconstruction Loss

Fig. 2. (a) A scene taken from "Kiki's Delivery Service". (b) Translated output of image (a), without reconstruction loss, where the model generates an incomprehensible structure.

Reconstruction Loss: Adding an additional generator solves the issue of mapping differences; however it still lacks in content preservation. We noticed from our training that, while training with only adversarial loss, the model tends to produce images which fail to match with the input data. It happens because

of multiple mappings of the target domain. For example, in Figure 2 the generator(without reconstruction loss) generated an image, with some unstructured pixels in the middle which consists of multiple mappings of the target domain.

However, using an additional loss function, by using the technique of forward and backward loss[10], [25], we've solved this content issue and have kept the similarities in the domain translation. The motive of this function is that, an image generated from an input can be reconstructed back to the input again such that $x = F(G(x))$, where F and G are generators and thus, it is able to map an image of target domain which is as close as possible to the image of input domain. In our paper, we call it *Reconstruction Loss*. The equation is as follows -

$$Forward\,Consistency\,Loss, \mathcal{L}_{f\_cyc} = \frac{1}{m}\sum_{i=1}^{m}(F_r(G_r(c)) - c) \tag{2}$$

$$Backward\,Consistency\,Loss, \mathcal{L}_{b\_cyc} = \frac{1}{m}\sum_{i=1}^{m}(G_r(G_c(r)) - r) \tag{3}$$

Training Stabilization: Training GAN with efficiency is a hard nut to crack. Prior to previous works, it is known that, discriminator tends to make the training slower and show more inconsistency during training. We used *Spectral Normalization* technique, which was first proposed by *Miyato et al.*[17], to stabilize our training. Benefit of spectral normalization is that it doesn't need extra hyper-parameter tuning. Also, the computational cost is relatively small compared to other weight normalization techniques. *Miyato et al.*[17] found better or same results with image generation tasks by utilizing this normalization technique. We can see from Figure 3 that, using this technique stabilized the training, where Figure 3(b) is ours, which shows a much smoother curve of FID scores than Figure 3(a), which is the FID score-graph of CycleGAN[25]. Also, we can see that ours achieved the least FID score within 145 epoch, whereas CycleGAN[25] takes more epochs for that.

PatchGAN: As discriminators, we used PatchGAN which was first proposed in *Isola et al.* [1]. The intuition of using this discriminator is that it works best for extracting the high-frequency details of the distribution. Another beneficial feature is, due to working on $N \times N$ patches, it takes fewer parameters and thus decreases the computation cost.

4 Implementations & Analysis

In this section, we discuss the implementation of our approach followed by illustrating its results.

Network structure: For generative networks we implemented the architecture from *Johnson et al.* [8] who achieved amazing results for neural style transfer and super-resolution. The network includes two stride-2 convolutions, 6 residual blocks[6], and two fractional strided convolutions with stride 1/2. We

used instance normalization technique as it performs well on style transfer tasks. For the discriminator network, we used 70×70 PatchGANs[1]. For ease of representation, we showed the architectures of generator and discriminator in Table 1 and 2, where the notation CONV(N,K,S) stands for $K \times K$ Convolutional layer with N filters and S stride size. Residual blocks[6] is denoted as residual(N,K) where N is the number of filters and K is the size of filter.

Table 1. Generator Architecture

Layers	Generator
1	CONV-(N64,K7,S1), InstanceNorm, Relu
2	CONV-(N128,K3,S2), InstanceNorm, Relu
3	CONV-(N256,K3,S2), InstanceNorm, Relu
4	residual-(N256,K3)
5	residual-(N256,K3)
6	residual-(N256,K3)
7	residual-(N256,K3)
8	residual-(N256,K3)
9	residual-(N256,K3)
10	CONV-(N128,K3,S1/2), fractional strided, InstanceNorm, Relu
11	CONV-(N64,K3,S1/2), fractional strided, InstanceNorm, Relu
12	CONV-(N3,K7,S1), InstanceNorm, Relu

Table 2. Discriminator Architecture

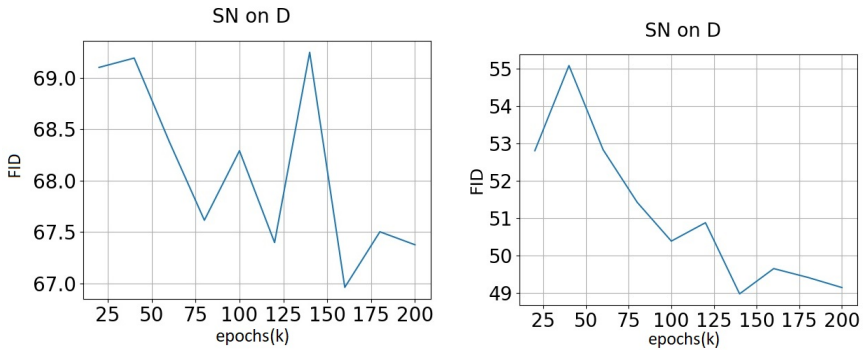
Layers	Discriminator
1	CONV-(N64,K4,S2), LeakyRelu
2	CONV-(N128,K4,S2), InstanceNorm, LeakyRelu
3	CONV-(N256,K4,S2), InstanceNorm, LeakyRelu
4	CONV-(N512,K4,S2), InstanceNorm, LeakyRelu

Evaluation metric: We chose the Fréchet Inception Distance (FID)[7] for quantitative evaluation. As FID score measures the difference between the generated dataset and the target dataset, it has shown more consistency with human evaluation. it calculates the Wasserstein-2 distance between the translated image and the real world images from an intermediate layer of an Inception-v3 network. Lower the FID score, the closer the distance between translated image and real domain images. As our task is image-to-image translation where we want our output to have the content of input cartoon images and the style of real-world images, we calculated a weighted average between them, where we used 80% weight for target data and 20% weight for input data. From Table 3 we can see that our work has shown the least FID score compared to other state of the art models.

Table 3. Fid scores of ours, UNIT and SingleGAN models.

Models	Our Work	UNIT	SingleGAN
FID	**48.4225**	55.9214	54.0130

Evaluation of stabilization technique: By utilizing spectral normalization technique on discriminator network shown in Figure 3(b), we started to gain lower FID score from the very initial of training compared to baseline model, which is implemented based on CycleGAN[25] model. Spectral normalization is used on discriminator network on baseline model which is shown in 3(b). From 3(b), the quality of transforming images doesn't improve monotonically during training. For example, the FID score of our work starts to drop at the 37th epoch. On the contrary, baseline model's FID score starts to rise after 125th epoch and it crosses the initial FID scores, whereas in our work, the scores didn't rise like the baseline model did. From this, we can clarify that we achieved a more stabilized model and better scores. We can also clarify from Figure 3 that the stabilization technique also takes fewer training epochs to achieve better scores.



(a) CycleGAN model (b) Ours

Fig. 3. Here, FID scores for CycleGAN (a) and for our method (b) are shown from 20 epochs up to 200 epochs.

Comparison with state of the art models: We compared our work with state of the art techniques, i.e UNIT [13] and SingleGAN's base model[23]. In Figure 4, we show a close-up view of an example, explaining that our work preserves much better vibrance and content preservation, where UNIT's[13] output becomes blurred and SingleGAN[23] makes unrealistic color on the content of output image. In Figure 5, we showed that our work achieved more color consistency and content preservation in cartoon to real world domain translation task. The UNIT[13] framework fails to keep content preservation of input images where it makes the content blurry. SingleGAN[23] results are overall less vibrant,

it couldn't preserve the real world images colorization and sometimes the model fails to preserve the content of input image.

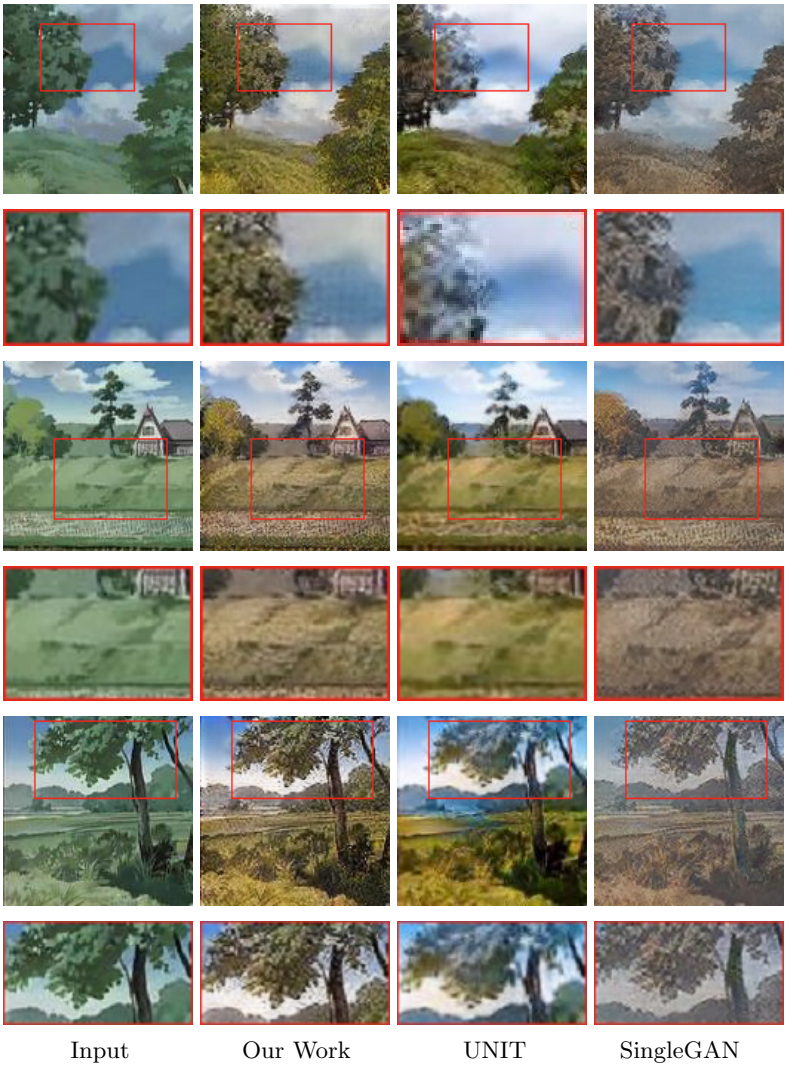


Fig. 4. Detailed comparisons in terms of contrast and content preservation. (a) Input images of cartoon scenes (a portion is amplified inside red bounding box for better observation). (b) *Result of our work*: shows more contrast on content, compared to other works. (c) *Result of UNIT*[13] which shows blurry content than (b). (d) *Result of SingleGAN*: which doesn't provide real world colorization on tree leaves and grasses; the image became faded and less vibrant.

Fig. 5. More qualitative results of our work compared to others. (a) Input images of cartoon scenes. (b) *Results of our work*: the results are much more vibrant and can preserve the content of input image compared to other models. (c) *Results of UNIT*[13]: which sometimes fail to preserve the content of input image and makes image blurry or too much smooth. (d) *Results of SingleGAN*[23]: which makes the overall image less vibrant and sometimes fail to keep content preservation.

Limitations: Despite achieving better FID score of all, it is still too high to be a perfect image translation score. In fact, we can see from Figure 6 that, the output fails to achieve the meaningful (semantically and geometrically) structure of real-world objects—in this example a *cat*. This problem is also common in UNIT[13] and other image-to-image translation models.

5 Conclusion

In this paper, we showed a GAN based approach to translate images from cartoon domain to photo-realistic domain. We implemented our model based on CycleGAN[25], where we used Reconstruction Loss for content preservation of input image and the PatchGAN for better texture extraction. By implementing spectral normalization technique on discriminator network, we showed that our model achieves better training stability and the lowest FID score of all the other models. Our future plan is to lessen our current limitations by investigating more geometry and content aware model to improve the texture so that the gap with the photorealistic domain decreases. In addition to FID score, we have

(a) Cartoon scene (b) Our result

Fig. 6. A failure case of our method. In the output image (b), the *cat* remains cartoonish as in input (a) and is not translated into a realistic one.

plans to arrange human-involved and perceptual evaluation processes to asses the correctness of your outcomes.

References

1. 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. IEEE Computer Society (2017), http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=8097368
2. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein gan. CoRR **abs/1701.07875** (2017)
3. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. pp. 2672–2680 (2014), http://papers.nips.cc/paper/5423-generative-adversarial-nets
4. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. CoRR **abs/1704.00028** (2017), http://arxiv.org/abs/1704.00028
5. He, D., Xia, Y., Qin, T., Wang, L., Yu, N., Liu, T.Y., Ma, W.Y.: Dual learning for machine translation. In: Advances in Neural Information Processing Systems. pp. 820–828 (2016)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. CoRR **abs/1512.03385** (2015), http://arxiv.org/abs/1512.03385
7. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Klambauer, G., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a nash equilibrium. CoRR **abs/1706.08500** (2017), http://arxiv.org/abs/1706.08500
8. Johnson, J., Alahi, A., Li, F.: Perceptual losses for real-time style transfer and super-resolution. CoRR **abs/1603.08155** (2016), http://arxiv.org/abs/1603.08155
9. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196 (2017)

10. Kim, T., Cha, M., Kim, H., Lee, J.K., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017. pp. 1857–1865 (2017), http://proceedings.mlr.press/v70/kim17a.html
11. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
12. Larsen, A.B.L., Sønderby, S.K., Larochelle, H., Winther, O.: Autoencoding beyond pixels using a learned similarity metric. arXiv preprint arXiv:1512.09300 (2015)
13. Liu, M., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. CoRR **abs/1703.00848** (2017), http://arxiv.org/abs/1703.00848
14. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: Advances in neural information processing systems. pp. 469–477 (2016)
15. Mao, X., Li, Q., Xie, H., Lau, R.Y.K., Wang, Z., Smolley, S.P.: Least squares generative adversarial networks. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. pp. 2813–2821 (2017). https://doi.org/10.1109/ICCV.2017.304, https://doi.org/10.1109/ICCV.2017.304
16. Metz, L., Poole, B., Pfau, D., Sohl-Dickstein, J.: Unrolled generative adversarial networks. arXiv preprint arXiv:1611.02163 (2016)
17. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. CoRR **abs/1802.05957** (2018), http://arxiv.org/abs/1802.05957
18. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. CoRR **abs/1511.06434** (2015), http://arxiv.org/abs/1511.06434
19. Reed, S.E., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., Lee, H.: Generative adversarial text to image synthesis. CoRR **abs/1605.05396** (2016), http://arxiv.org/abs/1605.05396
20. Rezende, D.J., Mohamed, S., Wierstra, D.: Stochastic backpropagation and variational inference in deep latent gaussian models. In: International Conference on Machine Learning. vol. 2 (2014)
21. Sultan, K.M.A., Rupty, L.K., Pranto, N.I., Shuvo, S.K., Jubair, M.I.: Cartoon-to-real: An approach to translate cartoon to realistic images using GAN. CoRR **abs/1811.11796** (2018), http://arxiv.org/abs/1811.11796
22. Yi, Z., Zhang, H.R., Tan, P., Gong, M.: Dualgan: Unsupervised dual learning for image-to-image translation. In: ICCV. pp. 2868–2876 (2017)
23. Yu, X., Cai, X., Ying, Z., Li, T.H., Li, G.: Singlegan: Image-to-image translation by a single-generator network using multiple generative adversarial learning. CoRR **abs/1810.04991** (2018), http://arxiv.org/abs/1810.04991
24. Zhang, H., Xu, T., Li, H., Zhang, S., Huang, X., Wang, X., Metaxas, D.N.: Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. CoRR **abs/1612.03242** (2016), http://arxiv.org/abs/1612.03242
25. Zhu, J., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. pp. 2242–2251 (2017). https://doi.org/10.1109/ICCV.2017.244, https://doi.org/10.1109/ICCV.2017.244