WEATHER TRANSLATION FOR SCENERY IMAGES USING CYCLEGAN

Nian-Yun Wu*, Ai-Jung Yang, Yi-Chung Chen

Department of Computer Science and Engineering, National Chung Hsing University, Taichung, Taiwan.
(itshelenwu@gmail.com; s110066038@smail.nchu.edu.tw; chenyich@nchu.edu.tw)

Session 6: New Technology

ABSTRACT:

This study is primarily based on the CycleGAN (Cycle-Consistent Generative Adversarial Networks) model, aiming to achieve the transformation of images from one weather condition to another while preserving the structure of the images. In addition to using the TensorFlow and Keras libraries to build the model, various auxiliary functions and layers are employed to define the generator, discriminator, and loss functions. During the training process, adversarial loss, cycle consistency loss, and identity loss for both the generator and discriminator are utilized. Additionally, data augmentation techniques are incorporated to enhance the robustness and performance of the model. Specifically, the technique of tf.concat is employed to randomly overlay images from an additional dataset onto the input photos, thereby augmenting the training data. This augmentation not only accelerates the training speed of the model but also facilitates easier recognition of differences in features among different images. Through multiple rounds of training, the model learns the mapping relationship between images from two domains, enabling the mutual transformation of images while preserving their structural integrity.

Key Words: CycleGAN, Image Transformation, Tf.concat, Data Augmentation

1. INTRODUCTION

Recently, with the rise of the concept of deep travel, many people have started to travel to various countries through self-guided tours. Providing information about tourist attractions that vary according to different times will greatly help people understand the current situation of these attractions. For example, describing a theme of cherry blossoms in a tourist attraction in spring, but the same tourist attraction in summer would not include cherry blossoms. With this additional information, it can assist people in deciding whether to visit these attractions during their travels.

In this project, we explore an intriguing issue: the attractiveness of many world-renowned tourist attractions is often influenced by weather conditions. For example, Mount Fuji is frequently shrouded in thick clouds, preventing visitors from enjoying its true magnificent scenery. However, tourists typically need to plan their trips weeks or months in advance, making it difficult for them to accurately predict the weather conditions during their visit. Additionally, the tourism industry faces pressure regarding environmental conservation and sustainable development. Therefore, we aim to advance the prediction of tourist attraction appearances in advance, enabling visitors to anticipate the possible scenery at specific dates in the future. This allows for better trip planning, effective resource management, and reduced environmental impact. Furthermore, we also consider the issue of photo copyrights. Traditionally, tourists may rely on photos from online sources or travel guides to understand the true appearance of attractions. However, these photos may be subject to copyright restrictions, limiting their free use. Hence, through this research, we seek to provide a method for generating predictive images of tourist attractions using existing resources. This helps travelers gain better insights into destination scenery while avoiding copyright issues.

2. RELATED WORK

2.1 General Adversarial Networks

Generative adversarial networks (GANs) have undergone significant advancements since their introduction by Goodfellow et al. in 2014 (Goodfellow et al., 2014). These networks, comprising a generator and a discriminator engaged in a competitive training process, have proven effective in generating realistic data samples across various domains (Arjovsky et al., 2017; Liu et al., 2017). The classic GAN architecture, initially an unconditional generation model, has been extended to address specific tasks such as image editing and conditional image generation (Mirza & Osindero, 2014; Isola et al., 2017). One notable extension is the conditional GAN (cGAN) proposed by Mirza & Osindero (2014), which introduces conditional information to synthesize images with specific categories. Building upon cGAN, Phillip et al. introduced Pix2Pix (Isola et al., 2017), incorporating the U-Net structure to enhance generation quality and utilizing adversarial and L1 loss during training. However, the reliance on paired training data in Pix2Pix can be a limitation due to its scarcity. To tackle this challenge, CycleGAN (Zhu et al., 2017) was

introduced, combining two GANs and employing a cycle consistency loss training strategy for unpaired image translation tasks. This framework has since become a cornerstone in the field, with ongoing efforts to refine its performance through adjustments in constraint conditions and network structures (Zhu et al., 2017; Yi et al., 2017). Additionally, advancements such as Wasserstein GANs (Arjovsky et al., 2017) have contributed to stabilizing training processes, further expanding the capabilities of GANs in generating diverse and realistic samples.

2.2 Image-to-Image Translation

Image-to-image translation is a pivotal task aiming to establish mappings between input and output images across diverse domains, encompassing tasks like semantic segmentation, style transfer, and super-resolution. The seminal work of Isola et al. introduced the pix2pix framework in 2016, employing conditional GANs to achieve supervised image translation, yielding impressive results in generating photorealistic images by training on paired input-output examples (Isola et al., 2017). Subsequent advancements in the field have extended the capabilities of image translation methodologies. For instance, CycleGAN introduced by Zhu et al. enables unpaired image translation through cycle consistency loss, while UNIT by Liu et al. learns a shared latent space for domain translation without necessitating paired data (Zhu et al., 2017; Liu et al., 2017). Moreover, other tasks within the realm of image translation, such as image coloring, domain adaptation, and data augmentation, have also witnessed significant progress. Approaches such as those by Gatys et al., Mo et al., and Ma et al. have contributed to refining image translation methodologies by addressing challenges like maintaining realistic results in complex backgrounds and instance-level translation (Gatys et al., 2016; Mo et al., 2019; Ma et al., 2018).

2.3 CycleGAN

The basis of this study majorly originates from the groundbreaking work of Zhu et al., which introduced CycleGAN (Zhu et al., 2017), a revolutionary contribution in the realm of unsupervised image-to-image translation tasks. CycleGan presents a solution for the challenge of translating images from one domain to another without the need for paired training data. The architecture of CycleGAN consists of two generator networks and two discriminator networks engaged in a sophisticated adversarial training process. By leveraging a cycle consistency loss function, it enables the translation process to be consistent when mapping images back and forth between two domains, thereby facilitating diverse applications such as style transfer, artistic rendering, and domain adaptation. Furthermore, its flexibility and versatility make it an indispensable tool for researchers and practitioners alike, enabling them to explore new avenues in image transformation and synthesis. The impact of CycleGAN extends beyond computer vision, influencing advancements in other areas such as natural language processing, where similar adversarial training techniques have been applied to tasks like text generation and machine translation.

3. OUR APPROACH

3.1 Methodology

Our work takes great inspiration from CycleGAN, building upon its fundamentals whilst incorporating our own touches to achieve a defining network. The significance of CycleGAN lies in its ability to seamlessly bridge the gap between disparate image domains, offering a versatile and powerful tool for image transformation tasks, which fits directly with our objective of translating images to different domains. Unlike conventional methods that require paired training data, CycleGAN utilizes adversarial training and cycle consistency to facilitate smooth translation between different image domains. This is achieved through the implementation of a cycle consistency loss function, ensuring that the translated images maintain the essential characteristics of the original input, resulting in realistic outcomes.

3.2 Problem Formulation

The goal of this study is to enable a seamless translation of clear-weather images to other time, weather or seasonal domains. We establish the specific domains to which we aim to translate the source images, namely: night, gloomy, fog, autumn, and winter. These target domains represent the desired outcomes for our initial sunny and summery input imagery. Since we formulate our problem to be a one-to-one correspondence between source and target domains, each target domain necessitates an individual dataset to enable precise translation. Thus we curate five distinct datasets and implement data augmentation techniques if necessary. We mostly adopt CycleGAN's architectural framework for our network. Given that CycleGAN's model architecture comprises two generators G and F, each designed to process inputs from their respective domains X and Y, we define domain X to always represent clear weather images, while domain Y represents one of our predefined target domains.

3.3 Datasets

For the capturing of diverse scenic imagery across various weather conditions and time intervals, we turn to live camera feeds on YouTube. These feeds offer continuous coverage of tourist destinations, and thus will be a perfect way to showcase how the same location looks during different times of the day and hopefully also during different weather conditions. Selections for capture include scenic spots like Mount Fuji, Hangang, Taitung Duoliang Station, etc. Employing the youtube-dl

and ffmpeg command line utilities in conjunction, we systematically gather our dataset. While youtube-dl facilitates the download of live stream content from YouTube, ffmpeg is utilized for programmatically extracting and storing still images from the feed at ten-minute intervals.

Documenting seasonal changes at a single location poses a trickier challenge. Given the impracticality of continuously recording a live camera feed over several months, an alternative approach is warranted. Thus, we turn to the Flickr API. Flickr offers an extensive repository of images, and by leveraging its API, we procure a substantial volume of scenic imagery tagged with specific seasonal descriptors–which in our case are autumn and winter.

Following the acquisition of our images, we standardize the datasets by resizing all images to 256x256 pixels. Subsequent data augmentation, such as random cropping and horizontal flipping, is exclusively applied to weather-related datasets.

3.4 Concatenation

Despite having conducted data augmentation on our datasets, initial experiments have shown lackluster performances. Consequently, we have devised a method to expedite the model's acquisition of nuanced features and differentiations between input images and target domain outputs: concatenation. Initially, we establish an additional dataset filled with images from both the source domain—comprising clear weather images—and the target domain, whichever weather or seasonal conditioned images that we aim to translate to. Instead of inputting a single clear-weather image into generator G as in the CycleGAN model, we randomly select two images from the additional dataset, concatenate them onto our clear-weather input image along the channel dimension using tf.concat, then use this concatenated tripartite image as the input for generator G. A key to be noted is that even though our input is now an image of size (256, 256, 9) as opposed to the original (256, 256, 3), the generated output image of the target domain maintains the original dimensions of (256, 256, 3). The images used for concatenation from the additional dataset serves the sole purpose of better highlighting the features within the primary input source image, thereby enhancing the accuracy of domain mappings and lifting training efficiency.

3.5 Losses

We adopt the loss functions from the original CycleGAN. The CycleGAN network has three loss functions: adversarial loss, cycle-consistency loss, and identity loss. These three losses work together to train the CycleGAN model effectively, allowing it to learn meaningful mappings between different image domains while preserving image content and structure.

3.5.1 Adversarial Loss:

The adversarial loss, also known as the GAN loss, is based on the concept of adversarial training, where the generator network competes against the discriminator network in a min-max game. It is used for matching the distribution of generated images to the data distribution in the target domain. The goal of the generator is to produce realistic images similar to those of domain Y to deceive the discriminator, while the discriminator aims to distinguish between real and translated images. Mathematically, the adversarial loss is formulated as:

$$\mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) = \mathbb{E}_{y \sim p_{\text{data}}(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log(1 - D_Y(G(x))] \quad (1)$$

where G = generator network
D_Y = discriminator network for domain Y
X = input domain
Y = target domain

The objective is for the generator G to minimize this loss, while the discriminator D_Y aims to maximize it. This forces the generated images to be indistinguishable from real photos.

3.5.2 Cycle-Consistency Loss:

The cycle-consistency loss is a regularization term introduced in CycleGAN to ensure that the mapping between the source and target domains is consistent in both directions. It is computed by feeding an image from one domain through the generator for that domain, and then passing the generated image through the generator for the opposite domain. The resulting image should ideally be similar to the original input image. Mathematically, the cycle-consistency loss is defined as:

$$\mathcal{L}_{\text{cyc}}(G, F) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1]. \quad (2)$$

where G = generator network for forward mapping (domain X to domain Y)
F = generator network for backward mapping (domain Y to domain X)
$||\cdot||_1$ = $L1$ norm

The objective is to minimize this loss, ensuring that if we translate from one domain to the other and back again we should arrive at where we started, and prevent the learned mappings G and F from contradicting each other.

3.5.3 Identity Loss: The identity loss is an additional regularization term used in CycleGAN to preserve the identity of images that already belong to the target domain. It ensures that if an image from the target domain is passed through the generator, it remains unchanged. Mathematically, the identity loss is computed as:

$$\mathcal{L}_{\text{identity}}(G, F) = \mathbb{E}_{y \sim p_{\text{data}}(y)}[||G(y) - y||_1] + \mathbb{E}_{x \sim p_{\text{data}}(x)}[||F(x) - x||_1] \quad (3)$$

where G = generator network for forward mapping (domain X to domain Y)
F = generator network for backward mapping (domain Y to domain X)
$||\cdot||_1$ = $L1$ norm

The objective is to minimize this loss, ensuring that the generators do not modify images that already belong to their respective domains.

4. IMPLEMENTATION

4.1 Network Architecture

Generators G and F share identical architectures and exhibit symmetry in their design. Both generators employ a ResNet architecture, comprising 9 residual blocks, with each block housing 2 convolutional layers. The generator's structure encompasses several key components: Firstly, there are 2 convolutional layers utilized for the initial processing of images, employing 7x7 convolutions. Following this, there are 2 downsampling modules, each comprising 1 convolutional layer, which serve to decrease the size of the output feature map relative to the input. This downsampling aids in extracting more global features from the image while concurrently reducing computational overheads and memory usage. For upsampling operations, fractionally-strided convolutions are employed. Additionally, the generator incorporates 2 upsampling modules, each featuring 1 transposed convolutional layer. These modules work to increase the size of the input feature map, facilitating the restoration of high-resolution details within the image. Throughout these operations, Instance Normalization is employed to ensure proper normalization of the data. The discriminators X and Y also share identical architectures. They employ PatchGANs, a technique in which the discriminator assesses patches of the image rather than the image as a whole. The discriminator is structured as follows: it begins with 4 convolutional layers to process the images initially, with each layer consisting of a 4x4 convolution. Following this, there are 3 downsampling modules, each comprising 1 convolutional layer. These modules capture local information within the image while minimizing sensitivity to specific local details, thereby facilitating binary classification (real or fake).

4.2 Training Details

The generator loss encompasses three components: adversarial loss, cycle consistency loss, and identity loss, which are combined linearly. On the other hand, the discriminator loss involves the adversarial loss computed between real and generated images. Adversarial loss is calculated using the Mean Squared Error (MSE) loss function. To tackle mode collapse and training instability issues associated with the original binary cross-entropy loss, the Least Squares GAN (LSGAN) approach is employed. LSGAN utilizes MSE loss instead of binary cross-entropy and introduces target values within a specific range for generated and real samples, thereby enhancing training stability and accelerating generator convergence speed. During the training of the CycleGAN model, the adv_loss_fn function is utilized to compute the adversarial loss for the generator, with real labels set to 1 and generated labels set to 0. Similarly, when computing the adversarial loss for the discriminator, the adv_loss_fn function is used with real labels set to 1 and generated labels set to 0. The Adam optimizer is employed to train both the generator and discriminator, with a learning rate of 2e-4 and beta_1 of 0.5. A batch size of 1 is set to stabilize dynamics between the generator and discriminator, thereby enhancing training stability. Additionally, the weights λ_cycle and λ_identity are defined to control the contribution of cycle consistency loss and identity consistency loss, respectively. λ_cycle determines how much the generator transforms generated images back to the original images, with a value of 10.0 in the code. On the other hand, λ_identity regulates how much the generator preserves the identity of input images while performing transformations, with a value of 0.5 in the code. Furthermore, the code includes functionality to augment training data by appending additional images to the primary dataset, thereby enhancing model generalization and performance. Reflection Padding2D layers are employed to implement reflection padding, which reduces boundary effects and improves the performance of the generator.

5. EVALUATION

During the training process of CycleGAN, there exists a competitive relationship between the generator and the discriminator. The generator is responsible for producing realistic generated images, making it difficult for the discriminator to distinguish between real and generated images. Meanwhile, the discriminator continually improves its ability to recognize images, enabling the model to correctly identify the authenticity of the images.

After undergoing two rounds of scene transformations through the GAN network model, the input image will obtain generated images with the same scene as the

original input. At this point, CycleGAN calculates the feature discrepancy between the input image and the generated image using the cycle consistency loss function and provides feedback to the network for weight modification. A larger value of the cycle consistency loss function indicates a greater feature discrepancy and lesser similarity between the two images, while a smaller value indicates greater similarity. During the training process, CycleGAN aims to reduce the value of the cycle consistency loss function, gradually increasing the similarity between the two images.

Figure 8.1 shows images generated by CycleGAN. Column A represents the original input daytime images, column B represents nighttime generated images produced by the first GAN, and column C represents images of the original daytime scene obtained through the second GAN from column B. By comparing images in columns A, B, and C, it can be observed that the generation network effectively performs scene transformation, converting daytime elements into nighttime scenes while retaining the structure of the original input images. Upon closer observation of objects such as Mount Fuji, Disneyland, Tokyo Tower, and Tokyo Rainbow Bridge in the original input images, it is evident that these important elements are successfully preserved after transformation. The experimental results demonstrate that CycleGAN network model not only generates corresponding nighttime scene images but also achieves a high degree of similarity to the original input images when further transforming the generated images. However, careful observation reveals that the CycleGAN network model tends to mistakenly convert parts with yellow-brown hues into nighttime lights, even if the original images actually depict vegetation.

Figure 8.2 demonstrates the use of the concatenate technique to improve the generation of nighttime scene images by the CycleGAN network model. Column A represents the scenario before applying the concatenate technique, while column B represents the scenario after applying the concatenate technique. By comparing columns A and B, it can be observed that the nighttime sky's color becomes darker and the scenery becomes clearer after utilizing the concatenate technique.

Figure 8.3 showcases the image translation results of scenic imagery from summer to autumn and winter, while Figure 8.4 showcases the translation results of clear-weather images to gloomy and foggy weather conditions.

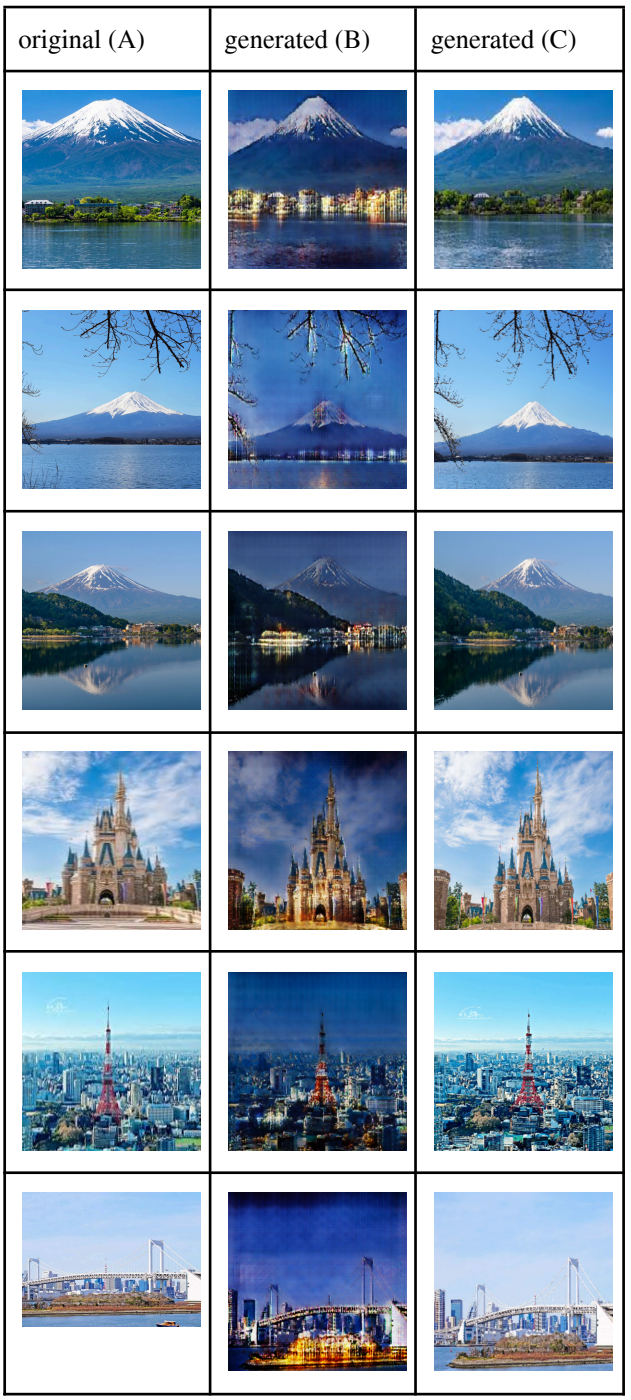

Figure 8.1. Images generated by CycleGAN

Figure 8.2. Using the concatenate technique to improve the generation images

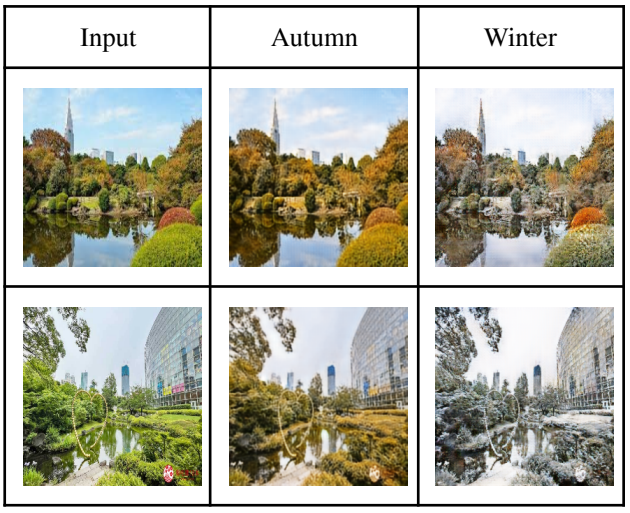

Figure 8.3. Image translation results for different seasonal conditions

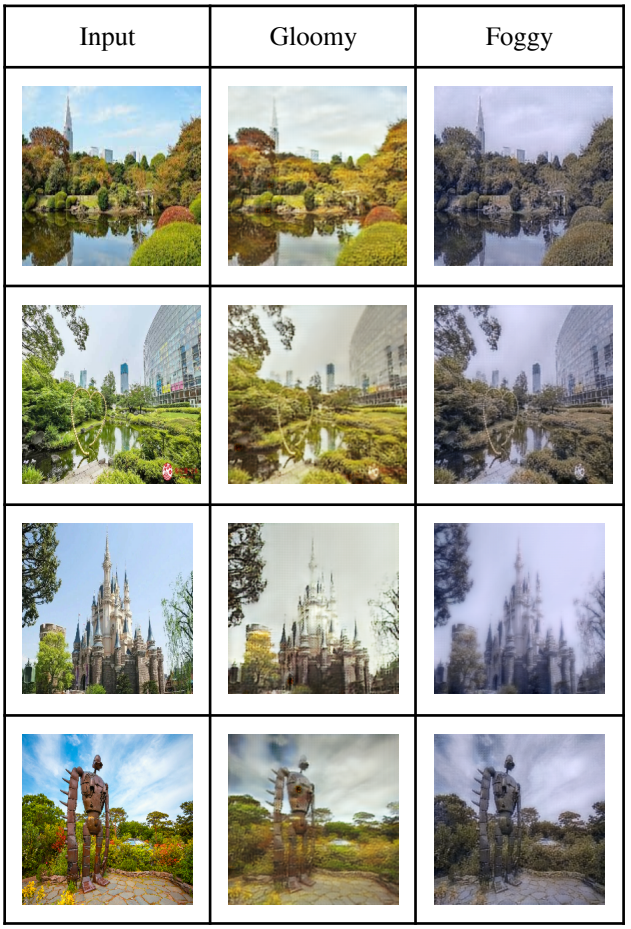

Figure 8.4. Image translation results for different weather conditions

6. CONCLUSIONS

In this work, we develop a framework that transforms the weather or seasonal conditions in regular scenic images. Our model enables the transition from clear weather to these target domains: night-time, gloomy, fog, autumn and winter. By capturing real-life scenery from

24-hr live cameras using youtube-dl command-line program and ffmpeg command-line tool in tandem, as well as using the Flickr API to fetch specific seasonal image data, we create unique abundant datasets specifically tailored for each of our target domains. We propose a unique method of utilizing concatenation for data augmentation purposes and speeding up the training process. By concatenating multiple images along the channel dimension and feeding them as one singular input into our framework, the model is able to quickly learn the features of our main input image and thus produce a highly accurate mapping to its target transition domain. We combine this concatenation technique with the fundamentals of CycleGAN, thus allowing our model to provide visually realistic weather and seasonal translation results. In future works, we aim to expand the coverage of our domains to enable even more possible transition end-results, and even to possibly allow not just 1-to-n but m-to-n weather translations.

7. REFERENCES

References from Journals:

Arjovsky, M., Chintala, S., & Bottou, L. (2017). *Wasserstein GAN*. arXiv preprint arXiv:1701.07875.

Gatys, L. A., Ecker, A. S., & Bethge, M. (2016). Image style transfer using convolutional neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). *Generative adversarial nets.* In Advances in neural information processing systems (pp. 2672-2680).

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

Liu, M. Y., Breuel, T., & Kautz, J. (2017). *Unsupervised image-to-image translation networks.* In Advances in neural information processing systems (pp. 700-708).

Yi, Z., Zhang, H., Tan, P., & Gong, M. (2017). DualGAN: *Unsupervised dual learning for image-to-image translation.* In Proceedings of the IEEE international conference on computer vision (pp. 2849-2857).

References from Other Literature:

Mirza, M., & Osindero, S. (2014). *Conditional generative adversarial nets.* arXiv preprint arXiv:1411.1784.

Ma, J., Wu, Y., Zhang, L., & Gong, D. (2018). *Exemplar-guided generative adversarial networks for image-to-image translation.* In Proceedings of the European Conference on Computer Vision.

Mo, J., Zhu, S., & Yi, D. (2019). *InstaGAN: Instance-aware Image-to-Image Translation.* In Proceedings of the IEEE International Conference on Computer Vision.

Zhu, J. Y., Park, T., Isola, P., & Efros, A. A. (2017). *Unpaired image-to-image translation using cycle-consistent adversarial networks.* In Proceedings of the IEEE international conference on computer vision (pp. 2223-2232).