Image Translation for Time of Day Lighting Correction Using StarGAN

Derek Bowdle
Department of Computer Science
Johns Hopkins University
Baltimore, MD, United States
dbowdle1@jh.edu

***Abstract*— This paper focuses on application of deep learning techniques in image processing with special concentration on the immense transformative capabilities of Generative Adversarial Networks (GANs), including Convolutional Neural Networks (CNNs), Conditional GANs (cGANs), Generative Adversarial Networks for Cyclic Transformation (CycleGANs), and Unified Generative Adversarial Networks (StarGANs). It delves into the subtle issues emerging in image-to-image translation, particularly with the task of translating images taken at various times of day to a corresponding clear daytime one. The task has proven to be quite a challenge, given that lighting, color dynamics, and visible details vary sharply between the different domains. The StarGAN model is used to handle unpaired image datasets that require translations between many domains, thus one is used on our novel dataset. We have gathered a new dataset containing surf camera images, but at different times of the day, to be able to use it for training our model. The processed dataset is analyzed, and the performance of the model is estimated, which is used to translate visual characteristics of night scenes to corresponding day scenes in a realistic way. We consider that this work shows promising potential and widescale use cases of StarGAN with a focus on practical scenarios with unpaired data and some important outcomes in the field of environmental monitoring, security, or digital art, among other fields. The paper concludes with a discussion on the existing limitations in methodologies and future scope of research for the enhancement of image translation technology in terms of its fidelity and applicability.**

Keywords—Convolutional Neural Network, Autoencoder, Generative Adversarial Network, Conditional Generative Adversarial Network, Cycle Generative Adversarial Network, Unified Generative Adversarial Networks, Image Translation, Image Style Transfer

I. Introduction

Developments in deep learning have allowed computer scientists to make huge advances in all areas of image processing. Specifically, many of these advancements have enabled the building and training of networks that generate and modify images in various and particular ways. These techniques have a broad range of applications, but they also contribute to the advancement of the field of image analysis in a very specific and valuable way. They allow researchers to generate datasets that can be used to train other image analysis networks by either creating an entirely novel dataset or supplementing other non-generated images, allowing for the network to arrive at a more robust state or to train on data that would not have been possible to generate.

This leads to another complex problem that exists within this space: the translation of poorly lit images into their clear daytime versions. Drastic changes exist between these two domains, with respect to illumination, color dynamics, and visible details make this task very challenging. The poorly lit scenes usually comprise low levels of light, with artificial sources and much more contrast, whereas the day scenes usually comprise only the broad spectrum natural light, which fundamentally changes in most conditions. This presents a serious tradeoff between realism and faithfulness for the resulting translation, something that must be balanced for any effective or useful network.

A. Definitions

A Convolutional Neural Network (CNN) is a kind of Deep Learning algorithm used purely for processing data with grid-like matrix input such as image inputs. This is the reason why convolutional neural networks are such a strong choice in computer vision. The network naturally learns spatial hierarchies of features adaptively through backpropagation. This capability is granted by the convolutional layers, which are one of the fundamental building blocks of a CNN. The convolutional layers produce feature maps that summarize the presence of detected features, such as edges or textures, over the entire input image through the use of filters run across the input. Additionally, within convolutional layers, CNNs contain pooling layers that down sample specific layers and effectively reduce the number of parameters and computations of the network, thus limiting overfitting. This allows the architecture to obtain great efficiency in tasks like, for example, image classification or object detection, capturing important characteristics with much less pre-processing than other algorithms for image classification.

An autoencoder is a special type of neural network model used within the realm of unsupervised learning, aiming towards efficient data encodings. The objective of an autoencoder is to compress input data into the lowest possible dimensionality representation and then reconvert the original data from that compressed form with the least amount of information lost. The compressed representation (latent space) of the data holds all the most essential information about the original input, which makes the additional lost dimensionality just non-essential noise. The general structure of an auto-encoder is an encoder, latent space, and a decoder. There are various uses of autoencoders, like in unsupervised learning for dimensionality reduction, where they help to bring out important features in data for other tasks such as image recognition. They are also used in data de-noising to

effectively eliminate the noise and in the detection of anomalies to identify the outliers by learning the normal patterns in data. This, in turn, makes autoencoders a very good tool in the handling of big amounts of unlabeled datasets in wide domains.

The Generative Adversarial Network (GAN) is a very advanced deep learning architecture and has been particularly influential in the area of computer vision. This model has two deep neural networks: the Generator and the Discriminator. It goes on to an adversarial process, where the Generator tries to generate examples (e.g., images) that seem to come from real data distribution, and at the same time, the Discriminator tries to distinguish between these real data instances and the ones created by the Generator. This dynamic competition drives both networks to improve iteratively, with the generator learning to produce increasingly realistic outputs. All these have revolutionized the idea through their ability to synthesize images and videos to be highly realistic. Application areas include photo-realistic image generation to super-resolution, style transfer, and many more, all showcasing the power of one of the most important technologies of artificial intelligence and computer vision.

A Conditional Generative Adversarial Network (cGAN) is a further extension of the basic framework of the Generative Adversarial Network (GAN) model, to use conditional variables within the model and extend its capability to the much wider horizon of deep learning in general and computer vision in particular. In cGANs, both the generator and the discriminator are conditioned on extra information, which is typically class labels, tags, or some other image, to guide the process of generating data. This conditioning allows the generator to produce specific types of images rather than generating from a random noise vector alone. For example, the cGAN may be told to generate images of particular objects, styles, or scenes from the conditional data provided. Particularly in these tasks that require controlled output, such as photo editing or content-specific image generation, or even the creation of complex scenario simulations in computer vision, which make it a game-changer for practical applications.

A Cycle Generative Adversarial Network (CycleGAN) is type of Generative Adversarial Networks, designed specifically for the task of image-to-image translation without paired examples. This is particularly handy for use in computer vision, where exact before and after images are usually not available for training. CycleGAN consists of two sets of generators and discriminators, each corresponding to a domain of interest. The main innovation in CycleGAN adds the cycle-consistent loss, which tries to ensure that an input image from one domain, when translated to another domain and then back, should look as close to the original image as possible. This proposed method serves as a powerful technique for translating images across domains, such as from horses to zebras, translations of the seasons in landscape photos, and translations between paintings and photographs. This ability of CycleGAN to learn the transformations in an unsupervised manner, again without the paired data, makes it very powerful for all sorts of artistic and photorealistic applications in computer vision, as well as providing a way to preprocess data for other techniques that require more consistent datasets for training.

Image-to-Image Translation: it may be defined to be a process of converting the image from one style or domain to another while keeping the content of the image. This is mostly done with advanced architectures of neural networks. For example, Generative Adversarial Networks (GANs) or one of their kinds like CycleGANs. The general goal of image translation is to alter some of the visuals in the overall image, which might refer to style, texture, appearance of any kind, and not necessarily change the inner structure and/or content. A few examples of applications are artistic style transfer (e.g., from painting to photo, or vice versa), changing seasonal attributes in landscape images, and translations such as conversion of daytime photographs to nighttime. The technology has further enhanced the capability of creative image manipulation, not limiting it to software but also enhancing its practical applications; for example, augmenting the data during the training of machine learning models when it requires different imagery.

"Image Style Transfer" is a really interesting application of computer vision using deep learning. The aesthetic style from one image is transferred and applied to the other one's content, while both are merged into the single output preserving the structure of original content but acquiring artistic hints from the referenced style. This has majorly used Convolutional Neural Network (CNN) in its process and has largely been implemented through models such as the Neural Style Transfer algorithm, which separates and recombines content and style features from different images. It uses the networks layers to code content and style independently, therefore making it a synthesis of images in a way that reflects a creative combination of the essence of both inputs. As a result image style transfer is pretty popular use of deep learning such as in applications that make the user's photo look like the styles of famous artworks. Thus, this is used lots in broader applications in graphic design, entertainment, and advertising, adding a new level of creativity and customization in visual content.

A Unified Generative Adversarial Network, which is more simply referred to as a StarGAN, is a type of GAN for multi-domain translation in image-to-image generation that uses a single GAN model. It is particularly useful for tasks like changing facial attributes; e.g., hair color, age, or gender. More traditional methods for dealing with multiple domains rely on many models. Specifically in the case of a multi CycleGAN for the task of translation a separate model is required for each translation between each domain. StarGAN packs them all into one unified model. To be more precise, in this model, the conditional generative approach takes an input image along with a domain label describing which domain should the input be transformed into, and the output is a modified image according to the input domain. This is what makes StarGAN efficient: the one model can be used to address a large variety of transformations. This conditional approach is structurally similar to cGAN but unlike a method such as pix2pix data is not required to be paired and many features can be used for the same image. A StarGAN makes use of a cycle consistency loss term, enforcing the input identity to remain the same, except for the intended modifications. Furthermore, the discriminator of StarGAN does not only discriminate real and fake images but also scores based on the domain to which the real images belong.

With this dual role, the model learns more solid features pertaining to the domain. All these make StarGAN highly applicable and efficient, thereby an impactful tool in this field.

II. Literature review

Previous works have used different techniques to perform image translation, mostly centering around the use of CNNs or some form of GAN. For the purposes of this paper, image to image translation encapsulates image style transfer. If viewed through a particular lens image style transfer is just a form of image translation where you don't have a dataset for the result of the translation. In regards to the time of day image translation, it is impossible to capture the exact same scenery in both daytime and nighttime conditions. While many things in the image might be constant many details will change in the time between day and night, which motivates a desire to find techniques that do not rely on paired images for training, like the ones used in style transfer. The inherent nature of the style transfer task is that it is impossible to say definitively what the resulting image would look like, the accuracy of the output is subjective, yet through techniques like a cycleGAN we can still preserve faithfulness to the original image through translating back to the original domain. This is something that we will try to leverage.

A. Key Works

In the paper, "Image-to-Image Translation with Conditional Adversarial Networks" the authors make an attempt to come up with an image-to-image translation framework to translate an input image in one style or domain context-sensitively to another one following conditional Generative Adversarial Networks (cGANs). These include learning to make the network understand and manipulate the complex image data, while preserving the core structure and content rigidly but allowing the appearance of the same to change as that of the desired output domain. Such a method can be applied for different types of applications, including turning aerial photos to maps, turning sketches into photo-realistic images, and most importantly, changing the seasonal attributes in landscape photos. The aim is generally to provide a powerful, flexible, and automatic way of image translation that can be applied not only to many specific problems of the image processing and artistic creation domain but also, if possible, to some practical cases such as medical imaging or video enhancement.

This paper demonstrates the flexibility of the model across a number of diverse applications, each of which shows an important capability of the technology. One such application is photographic image synthesis wherein segmented images of models such as a city block image with objects like roads, buildings, trees labeled on the pixels are translated into detailed and, one can say, photo-realistic urban scenes. This is very useful for urban planning and in the generation of virtual environments.

The second-largest application that was tested in the paper was semantic segmentation. In other words, the model would act vice versa in relation to the synthesis task: from real-world images to segmented maps. This is important; e.g., in the case of autonomous driving, the precise perception of the environment is essential. Besides, it also did the task of colorization in the sense of turning all the black-and-white photos into color, while realizing the realistic color in its knowledge context from the grayscale images.

Further demonstrating more of its flexibility, the model is doing style transfer, meaning that the visual style of images is transformed in such a way that it looks like a particular artistic style or era, hence helping in creative work with digital art. Finally, the current paper moves on to the super-resolution procedure of upscaling image resolution. This is essential in improving the quality of digital media and, particularly, video production and photography details of the improved nature, which can greatly heighten visual realism. All this brings to light the wide utilities and impacts of conditional GANs across industries and creative fields.

The method presented by the authors in this article is able to produce realistic lighting and color tone changes in images from daytime to represent lighting conditions corresponding to nighttime. This includes dimming the overall lighting, introducing shadows, and changing the sky's appearance. This really does quite a convincing change of scenes into a night setting and would hence be used for a whole range of purposes, like increased night aesthetics of the media or even training systems of computer vision with varied lighting conditions.

However, translating night to day is inherently more challenging for several reasons. Night images typically contain less visible information and more noise due to low lighting conditions. These key details necessary for realistic day time images, such as clear sky colors, textures on surfaces, and shadows indicating sun position, either do not exist in night images or are poorly defined. This causes a lack of information, which allows the network to give inferences and reconstructions of a possible daytime scene correctly.

Night images also generally have a compressed dynamic range, thus they offer less variation in the brightness levels to work on. This further adds to the challenge of adequately brightening the image without making it appear unnatural. While the day-to-night translation mostly consists of adding shadows and changing light within existing details, night-to-day translation has to make inferences and add details that were not actually seen in the input image. This makes it quite a more complicated and challenging job for the conditional adversarial network.

The work presented in "Image-to-Image Translation with Conditional Adversarial Networks" and the paper "A Neural Algorithm of Artistic Style" are both based on a similar neural network-based architecture but have a very different purpose and method. One looks at the translation of images from one form to another while preserving the structural integrity but the styles or even the elements of the same, such as texture, are changed. This would then allow conditional Generative Adversarial Networks (cGANs) to learn the mapping from an input image to an output image given some condition; for example, in this case, it is a semantic label or an edge map. Maintaining direct correspondence of the input between the output ensures that the transformation is applicable to those applications demanding high fidelity and precise context adjustments, for example, converting the daytime scene to nighttime.

In contrast, "A Neural Algorithm of Artistic Style" is a paper that looks at the idea of style transfer: the styles of features of one image, typically an artwork, are transferred to the content of another image, like a photograph, preserving primarily the structure of the content. It uses Convolutional Neural Networks (CNNs) with the aim to reduce the difference at each layer between a content image and it in terms of their content features and between a style reference with respect to style features. Through the development of this technique, it has mainly found use in artistic enhancements, allowing one to create visually unique images that connect famous aesthetics with everyday photography.

Basically, both papers argue that these are artistic handling and transformation of the images under the realms of advanced machine learning techniques, but "Image-to-Image Translation" has more emphasis on the accuracy level of the context-based, context-aware transformations, ranging from practical to creative outputs. On the contrary, "Artistic Style Transfer" works with an aesthetic mixture of style and content to give the user, in turn, an even wider field of their usage, by providing him with creative and decorative results.

The "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks" or simply CycleGAN with respect to image-to-image translation, is very different from previous methods such as those described in 'Image-to-Image Translation with Conditional Adversarial Networks' or 'A Neural Algorithm of Artistic Style.' Perhaps what distinguishes CycleGAN as a model is to be an unpaired image translation model whose functioning seems quite distant from the norms of conditional adversarial networks requiring paired images for training. This is of particular advantage when the source and target images need to perfectly correspond but this goal cannot be achieved, as is the case with many real-world problems.

Furthermore, cycle-cons, loss makes sure that an image translated from one domain to another and again translated back to its original domain suffered only a little loss of information. This forms a cyclical process that assists in learning robust translations of images that maintain the necessary qualities of the image across the domains without the need for direct paired examples. This is directly against both the direct mapping and paired training approach of conditional GANs and more in lines with real-world scenarios where most probably this perfect pairing of image domains is far from happening.

Compare this with "A Neural Algorithm of Artistic Style" (vs. "A Neural Algorithm of Artistic Style"): compared to "A Neural Algorithm of Artistic Style," CycleGAN flips from the problem of how to think about mixing the style of one picture with the content of another one into translating generic features between two different sets of images that have no relation to each other (e.g., translating horses into zebras or summer scenes into winter). CycleGAN takes a different approach from traditional style transfer, which optimizes for a composite image with respect to the content and style at runtime. Instead, CycleGAN trains a generative model to efficiently generate new images given sufficiently many input-output pairs for training. This brings even more flexibility and practical application for continuous or multiple image translations, using CycleGAN as a powerful tool that will span very many applications needing dynamic and diverse image translation free from the shackles of paired training data.

The paper "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation" introduces the model StarGAN: A Universal Approach to Domain-Guided Image-to-Image Translation. This is much less complex than previous approaches, which needed a model for each source-target domain pair and hence much easier to scale up. What makes StarGAN unique is that it uses domain labels to condition the generative process, so translation across multiple domains, like different attributes or styles, can be performed under one framework.

Notably, this includes a domain classification loss that plays nicely with the adversarial loss and cycle-consistency loss, all the three usual suspects that have become a natural part of any GAN-based image-to-image translation task. Further, due to loss, it ensures that the model produces images that are classified into the correct domain, so it improves the effectiveness of the model. Moreover, StarGAN also allows scaling and inviting new domains without the need to rebuild the model afresh, something other multi-domain approaches missed.

On the other hand, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks" suggested an effective method of translating images between domains without paired examples, though only between two domains. CycleGAN also uses a dual-GAN architecture, in which each GAN is tasked with image-to-image translation of two image domains, using a cycle consistency loss to ensure that an image can be translated back and forth between domains to end up close to its original self. This is an important loss that requires use when learning effective translation models in the absence of paired training data.

The notable difference between the two is in scalability and efficiency, instead of giving the direct comparison of ideas presented in CycleGAN, with regard to adversarial training and cycle consistency, StarGAN generalizes the foundational ideas. While CycleGAN needs to have a model for each pair of the domains, meaning its scalability to train multiple domains is quite limited, StarGAN simply adopts a single-model approach for many domains, making training efficient and scalable.

Additionally, new domains can be extended much more easily with StarGAN. When compared with CycleGAN, where we had to build and train some new model from scratch every time a new pair of domains appeared, it won't be that much of a hassle. StarGAN overall improved the scope of flexibility and scalability by deriving the principles of CycleGAN yet utilizing a conditional input like a cGAN. StarGAN best suits multi-domain translations by a single model which is fitting for applications like style transfer and facial attribute modification, where a number of attribute adjustments need to be done at the same time.

B. Other Relevant Works

The paper "A Style-Based Generator Architecture for Generative Adversarial Networks," uses a radically new generator architecture that is very different from what is usually proposed for GANs. This architecture increases the quality of the generated images through a style-based method, where there

exists a mapping network for taking the latent code to an intermediate latent space. This space enables users to manipulate the style of the output at different scales through adaptive instance normalization (AdaIN) at each convolutional layer, providing fine-grained control in the synthesis process. The method encompasses progressively growing the network to enhance stability and quality in producing high-resolution and high-fidelity realistic images. This, therefore, allowed controlling several attributes at the same time, in this case, for example, pose, face shape, complexion details, without affecting the details of the other attributes.

The authors apply AdaIN (Adaptive Instance Normalization) in each of the convolution layers of the generator, such that every convolution layer of the generator changes the mean and variance of the content inputs to those of the style inputs. This, therefore, turns out to be one of the core constituents of mixing styles into the content-aware outputs. And the mapping network maps input latent code to this intermediate latent space for a more disentangled and controlled disentanglement process of generation.

This allowed for improved realism and diversity in the generated images by introducing stochastic details, such as hair texture and skin imperfections, at every layer through noise inputs. Building on the concept of progressively growing of GANs, StyleGAN adapts this approach with the use of style-based controls for improved synthesis at different resolutions of the image. The impact of architecture into the field has been wide and huge, through inspiration for more investigations and wide applications: in computer graphics, fashion, and design.

Comparing StyleGAN and StarGan highlights distinct focus areas and applications. Where StyleGAN slightly can render more details and lifelike images that are suitable for generating human face images with relative fine details, StarGAN is designed for more general switching between domains. StyleGAN allows you to show a high level of control to produce very detailed images, while StarGAN offers a general solution across different domains. Both papers are representing huge steps in their respective fields and are pushing the frontier of what GANs can do in terms of image quality, versatility, and controllability.

The paper "Style Transfer of Black and White Silhouette Images using CycleGAN and a Randomly Generated Dataset" is definitely a recommendation for an impressively new method of the application of CycleGANs for style transfer dealing with black and white silhouette images. Therefore, the main objectives of this research project are to establish the fact that CycleGANs are very good at transferring style into silhouette images, which traditionally have been characterized by emphasis on shape rather than the stylistic of color and texture. This presents a unique challenge for style transfer technologies, which typically focus on color images.

The authors opted to try CycleGAN due to its ability to learn translations between two unpaired image datasets. Generating a random dataset of silhouette images to train the model is a novel step they have brought in. The dataset they have is such that it contains a large variety of shapes and outlines, so it helps the model generalize to a larger extent over different styles of silhouettes. This paper shows the dataset preparation, CycleGAN architecture, and training procedures used, always emphasizing the flexibility and efficiency of the model for the task at hand.

The results from the paper are very strong, showing how a sound model can be tested with a quality and quantity assessment. The authors compare images before and after style transfer using some quantifiable metrics, such as style and content loss, showing how in both cases their model can do some impressive style transfer, while keeping the main structural elements of the silhouettes. This balance is critical for the success of style transfer in silhouette images.

Finally, the paper concludes with a discussion of a brief exploration of some implications these findings might have in fields like graphic design and animation, where such technology could go a long way in enhancing creative processes. They even further suggest possible improvements in research, such as looking for other GAN architectures to achieve better efficiency and quality, increasing the size of their dataset to include more complex silhouettes, or even adding color. These are future directions that signal potential for the proposed CycleGAN to revolutionize style transfer applications in a very big variety of artistic domains. The present research, a step-change contribution in the niche area of silhouette image style transfer using deep neural networks with wholly new methodologies and insights, may apply to many other fields of machine learning and digital art creation.

III. Description of Dataset

For this project a dataset was assembled by capturing still screenshots surf cameras provided by website Surfline. Surfline is a website, headquartered in Huntington Beach, California, that provides comprehensive surf forecasts, live surf reports, and surf news service to surfers. It gives the requisite data for surfers such as wave conditions, directions of swells, and wind data. In addition the service is well known due to its network of more than 600 live HD cameras. To gather the images an automated script was used to capture images once every 10 minutes from a smaller selection of these cameras, mainly in and around Southern California. The images are usually taken from a structure near by the beach and feature either only the ocean, or sometimes the beach or any other objects that might be around.

Fig. 1. Daytime Example image: Lower Trestles, San Clemente, California

For this project the images were all collected from the surf camera placed at Lower Trestles, San Clemente, California. They were gathered over the last 8 months along with the day

and time they were taken at then combined into a single directory.

The images were then binned by dawn, day, dusk, and night. Specifically the time of sunset and sunrise was on the day the image was taken was used to determine how close the time was to either event, so that day and night images were far away and dawn and dusk images occurred close to them. This binning rule is described in Table 1.

Fig. 2. Dawn Example image: Lower Trestles, San Clemente, California

Fig. 3. Dusk Example image: Lower Trestles, San Clemente, California

Fig. 4. Nightime Example image: Lower Trestles, San Clemente, California

Once the images were binned, they were then preprocessed and put into a Tensorflow dataset. First each image was resized to 128 by 128 pixels to decrease the number of nodes needed in the initial convolutions, and thus ensure a reasonable size for training.

TABLE I. DATASET SUMMARY

Time	Properties	
	Total images	***Binning rule***
Dawn	1,590	Sunrise to Sunrise + 1hr
Day	12,341	Sunrise + 2hr to Sunset – 2hr
Dusk	1,604	Sunset – 1hr to Sunset
Night	13,448	Sunset + 2hr to Sunrise – 2hr

[a.] Dataset was gathered from images on surfline.com

IV. METHODS

A. Network

For this project a StarGAN was used which is structurally the same as the one in "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation", though different hyperparameters were used. The StarGAN architecture is based on the architecture of a CycleGAN but has some key differences: It only has a single generator and discriminator, It has an additional conditional input, and has some different components in its loss functions namely a domain classification loss component.

The architecture of a CycleGAN is comprised of two sets of GANs, each containing a generator and a discriminator. Each generator is just an autoencoder and each discriminator is a CNN. The first generator G translates images in the domain X to domain Y and, similarly, the second generator F does so in the opposite direction. The discriminators (D_Y and D_X) then try, in an analogous way, to classify whether samples are real and from their domain or are fake translations from the generator belonging to the other domain. Discriminator D_Y distinguishes between real Y images and those fraudulently created by G, and D_X distinguishes between real X images and those generated by F.

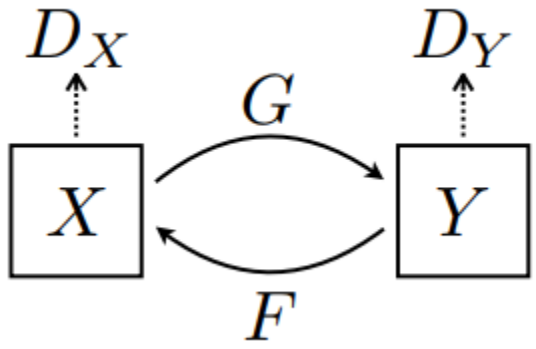

Fig. 5. CycleGAN Structure

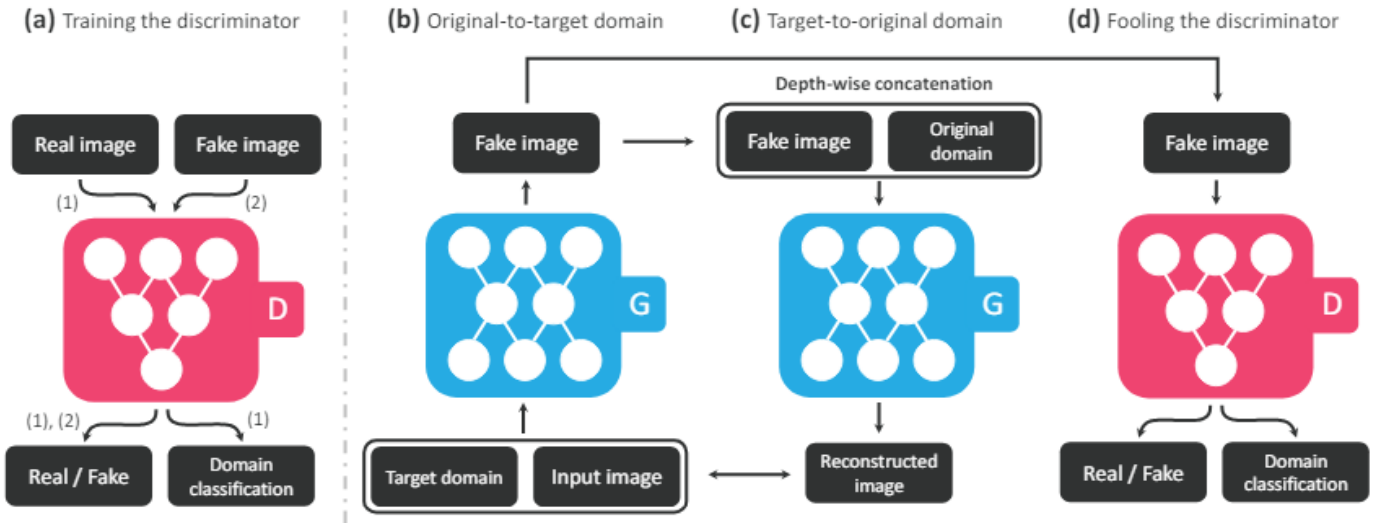

Fig. 6. StarGAN Structure - Source: *StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation*

The crux of the CycleGAN model architecture is the cycle consistency loss. The model uses a typical adversarial loss component as well but the cycle consistency loss component ensures that the image translation process preserves the important features in each domain. The network is penalized for this loss in cases where the translated image to another domain and then back again differs significantly from the original one. It is given as the sum of the L1 norm of the difference between an image and its cycles through both domains, further attesting that the model is capable of coherent translation, contextually appropriate for the task. This cycle consistency can be mathematically expressed as:

$$||F(G(x))-x||+||G(F(y))-y||$$

The principal change that was made to the CycleGAN to make a StarGAN is the addition of a domain input expressed as an additional channel inputted into the generator. This tells the single network what domain to translate to, and as a result we can now use the network to translate to multiple domains. If we then think about how we could modify the structure of the CycleGAN while utilizing this new ability we get the following changes. First generators G and F become a single generator G that takes in an image from either X or Y and the input of the desires domain to transfer to so Yor X. The generated fake image then still needs to be checked, but because we only have a single generator we only need a single discriminator D, yet we still need a way to keep track of the domain so we have the discriminator output a vector, in addition to its real/fake estimate, that provides and estimate of which domain it believes the input belongs to.

These changes to the generator input and discriminator output then necessitate a change to our loss function. First we keep the adversarial loss component and remake the cycle consistency loss as a reconstruction loss which does the same task as the it does in the CycleGAN yet because we only have one generator now we apply the generator twice and appropriate domain inputs to arrive back at the same image. In the same notation as the equation above with c as the domain that looks like:

$$||G(G(x,c), c')-x||$$

The new piece that we must add is a component that accounts for the domain classification. This is done through two loss terms one for the discriminator and one for the generator. In the original paper this is described as a "probability distribution over domain labels computed by D" but essentially it is pushing the classifications to be accurate when the discriminator is run by itself on a real image and when run on the output of the generator. The formula for the domain classification loss of the discriminator is given by:

$$L_{cls}^{r} = -\log D_{cls}(c'|x)$$

This differs slightly from the formula for the domain classification loss of the generator which is given by:

$$L_{cls}^{f} = -\log D_{cls}(c|G(x,c))$$

B. Training Procedures

As with all GANs the training alternates between training the generator and discriminator. I used a batch size of 16 for each training attempt I made and learning rate of 0.002 with the adam optimizer. I used the default Loss functions as described in the StarGAN Paper. The full functions for the generator and discriminator from the paper are:

$$L_D = -L_{adv} + \lambda_{cls} L_{cls}^{r}$$

$$L_G = L_{adv} + \lambda_{cls} L_{cls}^{f} + \lambda_{rec} L_{rec}$$

Where L_{adv} is the adversarial loss, L_{cls} is the domain classification loss, L_{rec} is the reconstruction loss and the λ terms are hyperparameters that can balance the various terms. I experimented with different values of λ but found that the values of $\lambda_{cls} = 1$ and $\lambda_{rec} = 10$. I also experimented with the number of residual blocks used in the generator. The original paper used 6, when less were used (3,4 or 5) there were problems with convergence. Lastly I experimented with training with different numbers of epochs, but it quickly became clear that if the

network can remain stable and the generator and discriminator stay balanced then more is better. Due to time and resource constraints the most epochs I could do was 30, which ultimately gave the best results.

C. *Learning Curves*

The most difficult part of properly training any GAN is balancing the generator and discriminator loss, StarGAN is no exception. In order to ensure that the model has been properly trained we must check that neither the discriminator nor the generator overpowers one another. If this balance is not preserved then the model ceases to train effectively.

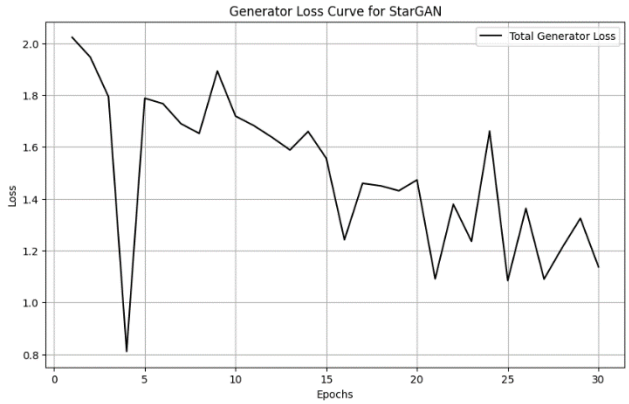

Fig. 7. Generator loss curve for final training attempt

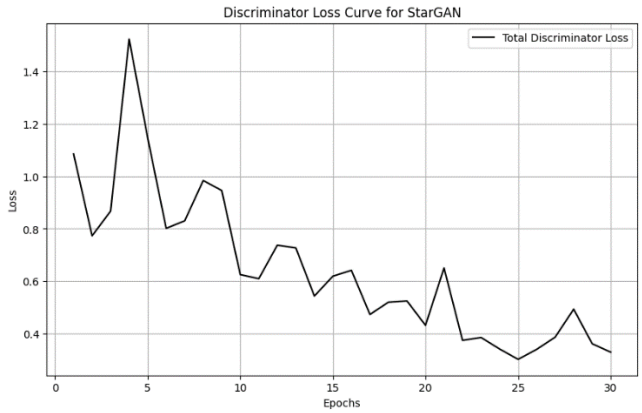

Fig. 8. Discriminator loss curve for final training attempt

In the final training attempt both the generator and discriminator loss remined stable. Neither outcompeted one another so the model would be considered effectively trained.

V. Results

To evaluate the model each time I used a set of 100 images from the training dataset and 100 images from the dataset that were never scene during training. Each set of images were evenly distributed across the four classes, 25 from each of dawn, day, dusk, and night. The images were then fed into the discriminator and class with the highest probability from the class vector output was counted for each.

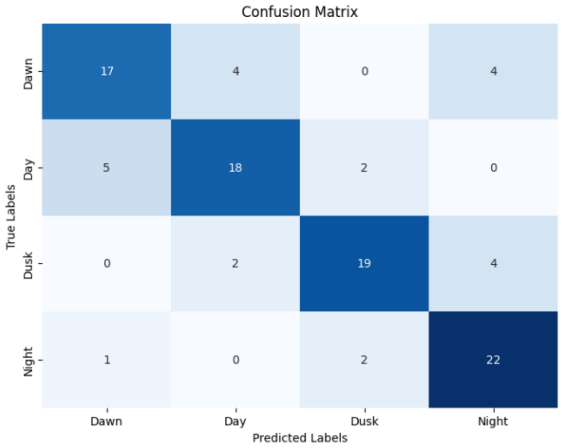

Fig. 9. Training Data Confusion Matrix

The classifier in the discriminator does a fairly good job differentiating the classes for images from the training data. It seems to have the largest issues with the dawn class.

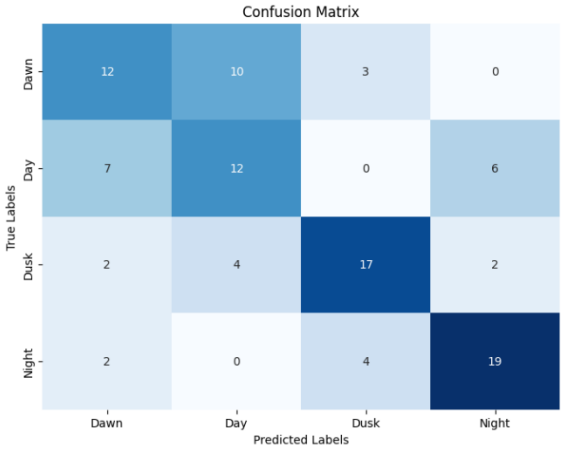

Fig. 10. Validation Data Confusion Matrix

The classifier in the discriminator does still differentiate the classes for images from the validation data, but does so pretty poorly. Some concerning results are that it appears to with some frequency confuse night with day, which seems the easiest to differentiate. Due to this issue not being present in the training data confusion matrix it is likely an overfitting issue.

VI. Conclusion

The model was somewhat successful though improvement was greatly needed. Two things would possibly improve it: more diverse data, and more training time. The website that the images were collected from offers hundreds of surf cameras, for this project only one was utilized. If the dataset was comprised of images from a large diversity of location it would possibly preform better on data outside of its training set. Secondly, as mentioned previously more training time and resources would have possibly improved performance of the model. The original paper trained for 100 epochs though some StarGANs will train upwards of 200 epochs. This of course comes with challenges in keeping the generator and discriminator balanced and avoiding overfitting, so changes to the network may be needed as a result.

References

[1] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," *Berkeley AI Research (BAIR) Laboratory, UC Berkeley*, Aug. 2020.

[2] L. Gatys, A. Ecker, and M. Bethge, "A Neural Algorithm of Artistic Style," *Journal of Vision*, vol. 16, no. 12, p. 326, Sep. 2016, doi: https://doi.org/10.1167/16.12.326.

[3] L. Gatys, A. S. Ecker, and M. Bethge, "Image Style Transfer Using Convolutional Neural Networks," *IEEE Computer Society*, pp. 2414–2423, 2016.

[4] S. Inoue and T. Gonsalves, "Style-Restricted GAN: Multi-Modal Translation with Style Restriction Using Generative Adversarial Networks," *arXiv (Cornell University)*, Jan. 2021, doi: https://doi.org/10.48550/arxiv.2105.07621.

[5] S. Kavitha, B. Dhanapriya, G. Naveen Vignesh, and K. R. Baskaran, "Neural Style Transfer Using VGG19 and Alexnet," *2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA)*, Oct. 2021, doi: https://doi.org/10.1109/icaeca52838.2021.9675723.

[6] T. Karras, S. Laine, and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks," Mar. 2019.

[7] W. Suwannik, "Style Transfer of Black and White Silhouette Images using CycleGAN and a Randomly Generated Dataset," *arXiv (Cornell University)*, Aug. 2022, doi: https://doi.org/10.48550/arxiv.2208.04140.

[8] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation," *Clova AI Research, NAVER Corp.*, pp. 1–15, Sep. 2018.

[9] Y. Liao and Y. Huang, "Deep Learning-Based Application of Image Style Transfer," *Mathematical Problems in Engineering*, vol. 2022, pp. 1–10, Aug. 2022, doi: https://doi.org/10.1155/2022/1693892.