General Adversarial Networks
Report On
Unpaired Image-To-Image Translation

Team Members:

Utkarsh Khurana (20UCS215)

Shubhang Jain (20UCS192)

July 2023

Course Instructor

Dr. Indra deep Mastan

Department of Computer Science Engineering

The LNM Institute of Information Technology, Jaipur

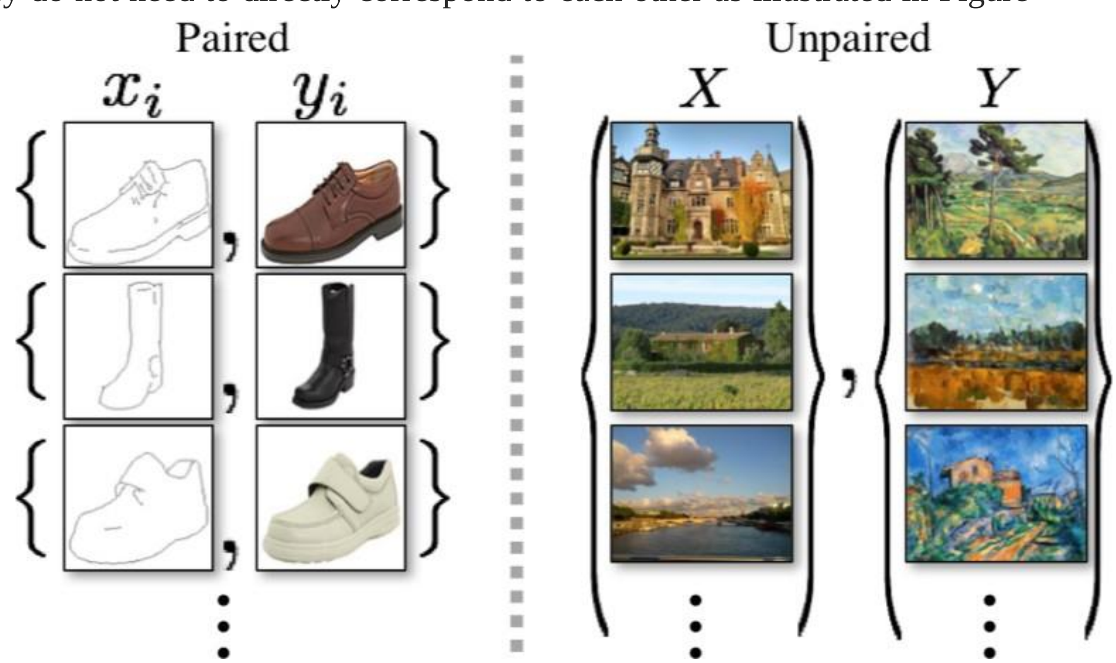
ABSTRACT

Unpaired image-to-image translation is a class of vision problems whose goal is to find the mapping between different image domains using unpaired training data. Cycle-consistency loss is a widely used constraint for such problems**.** It has gained a great deal of attention for applications in which paired data are unavailable or difficult to collect. A key problem of unpaired image-to-image translation is determining which properties of the source domain to preserve in the translated domain. This project aims to study & develop unpaired image translation with the goal of converting unpaired paintings into photorealistic images using CycleGAN & its corresponding architecture.

INTRODUCTION

Unpaired image-to-image translation learns mapping between two piles of images and examines common elements of the two piles (content) and unique elements of each pile (style). CycleGAN, a type of GAN which performs cross-domain transfer tasks, such as turning painting into photos, and vice versa, photo enhancement, and many more. It allows us to do unsupervised image-to-image translation, from two domains.
We will be focusing on one of its applications of unpaired image translation using CycleGAN in this report, specifically using them to convert paintings into photos. It deciphers an image from one domain X to another domain Y without paired images

To turn paintings into photos, normal GANs need paired images. This limitation is overcome by CycleGAN, that is, it does not need paired images to train the network. It requires unpaired images in the sense that while an *x* and *y* set of images are still required, they do not need to directly correspond to each other as illustrated in Figure



In other words, if we wanted to translate between paintings and photos, we still need to train on a set of paintings and a set of photos, but the paintings would not need to be of the exact photos in your dataset. This unsupervised learning capability of CycleGAN is very useful. It does so by training two Generator networks (*G & F*) and two Discriminator networks *(D_X & D_Y)*, unlike normal GANs, which train one Generator and one Discriminator network.

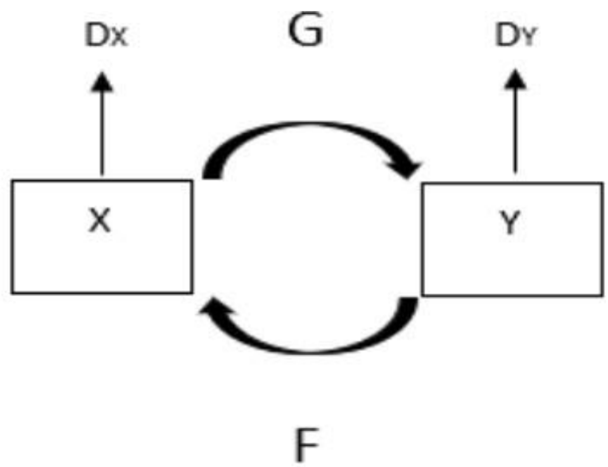


Figure 2: Illustration of CycleGAN with two generator & adversarial discriminator networks

We find this project of interest to us because history is always fascinating to learn about and it helps us develop an understanding of how the world was at the time. It is exciting to have an insight into the exact scenario the painter had at the time of painting. Also, paintings are perishable and have a higher maintenance cost.
CycleGAN lets us translate paintings into photos, which in turn lets us have an insight into our history and how the world appeared to the painter, thus our reasoning for opting for this project.

BACKGROUND

Generative Adversarial Networks (GAN) have achieved impressive imaging, image processing, and presentation learning. Newer methods use the similar idea for conditional imaging applications like Text2Image, Image Painting, future prediction, and other areas such as video and data in 3D.
The success of GAN is due to the loss of the adversary, which forces generated images to be essentially indistinct from the real photos. This loss is significant for imaging tasks, as it is the goal that many computer graphics aim to optimize.

Image-to-Image Translation

The concept of image-to-image translation uses a non-parametric texture model for a single pair of input-output training images. Newer methods use a series of input and output examples to learn a parametric translation function using CNN.

Unpaired-Image-to-Image Translation

Several other methods also address the unpaired configuration, where the goal is to relate two data domains: X and Y.There is also the use of adversarial nets with additional conditions to force the exit near the entrance. in a predefined metric space, such as B. B. Class namespace, image pixel space, and image feature space.

Our formulation is not based on a predefined, task-specific similarity function between output and input. We also do not assume that both input and output domains exist in the same low-dimensional insertion space. Thus our method is a universal solution for many visual and graphical tasks.

Using simple forward and backward coherence for visual tracking has been a traditional trick. In the field of languages, revising and enhancing translations using reverse translation and matching is a method used by human translators (including Mark Twain) and machines. Higher-order cycle consistency has been used in motion structure, 3D shape matching, segmentation, dense semantic alignment, and depth estimation.

Neural style transfer is another way of performing an image-to-image translation. Here, a new image is synthesized by mixing the content of one image with the style of another image (usually a painting) based on the match of the statistics of the image. matrix. Grams of previously trained depth characteristics.

On the other hand, our main goal is to learn a function between the two image groups, rather than between two specific images, which tries to capture high-level correspondences between appearance structures. Therefore, our method is applicable to other tasks as well, such as B. Painting → photography, object transformation, etc. where single sample transfer methods don't work well.

The other simple approach to the solution would be to train the model for a mapping between the given input image and the output image

However, the above approach requires a training set with paired input-output images.

- Such a large dataset is not available and would be expensive to collect.
- Moreover, the output would be complex in such artistic styles to define.

Hence, we'll need a network that doesn't require pre-existing paired images, but rather, that can generate a mapping on two different datasets, named as the source domain and target domain, such that they are indistinguishable.

RELATED WORKS

GANs have attained pronounced success in video and image editing and generation and representation learning. The one significant contributing factor to GANs' positive result is the concept of adversarial loss that makes the generated images very similar to real images.

Image-to-Image Translation

In this, we have a dataset that has input-output images, and then from there, we learn a function with the help of CNN(Convolution Neural Network). Our approach uses a conditional GAN to learn how to map the source to target without being provided with paired training examples like this.

Unpaired-Image-to-Image Translation

Here we have to relate the two domains X and Y. For this purpose, many models and frameworks have been proposed which includes

- a Bayesian model which incorporates a prior based on a Markov random field and a likelihood term.
- Cross-modal scene networks along with some Coupled GANs using a weight-sharing strategy
- There is also a framework prompted by dual learning in machine translation.
- Another remarkable work makes the input, output have some common "content" features even when they are differing in "manner". Here too adversarial networks are used to make the output very similar to the input in an already defined metric space, like image feature space or class label space.

In all these above approaches an already defined function of similarity among the input and output is stressed upon while in our formulation there is no such constraint. That is what makes ours a generalized and better solution.

Cycle Consistency

This is very much the same as our work because in this we are using a cycle consistency loss to use transitivity to supervise CNN training. In modern times, enforcing simple forward-backward consistency is simpler but not that consistent.
Nowadays, higher-order cycle consistency is being used in structure from motion and similar works. Ours is just more consistent but somewhat similar to these.

Neural Style Transfer

This combines the content of one image with the style of another image by toning the Gram matrix statistics of pre-trained deep features whereas we learn a function in the twodomains X and Y and not in between some particular two images. This generalized nature of our model is what makes it easily applicable in scenarios such as converting paintings to photos and transfiguring objects etc.

There are several baseline models with whom we can compare and contrast our method.Some of them are:

CoGAN

It learns one generator for each domain. Translation from X to Y is further done by latent representation and then using this representation into style Y.

Pixel loss + GAN

This applies adversarial loss to train a translation from X to Y with some regularization term at a certain level.

Feature loss + GAN

This is in essence a variant of Pixel loss+GAN where the Least Absolute Deviation is calculated over deep image features.

pix2pix

This is the simplest implementation as is trained on paired data. The main aim of performing this was to know the "upper - bound" i.e to create a level for further comparison.

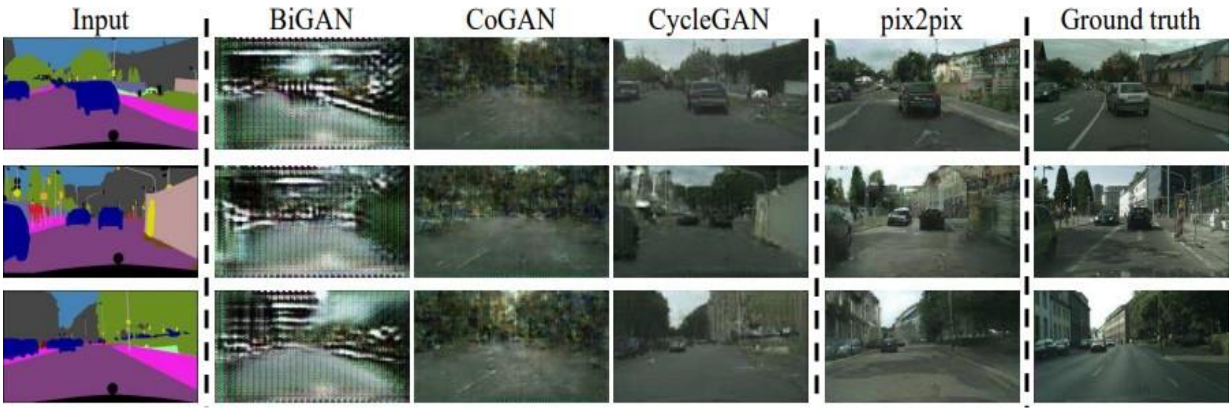


Figure 3: Results obtained from different baseline methods

The comparison with these models results in the following conclusions :

- Pixel loss + GAN method fails to generate results anywhere near to the desired.
- Quite similar to the Pixel loss+GAN, Feature loss + GAN also fails to achievedesired levels of performance.
- Our method (Cycle GAN) achieves results almost similar to the pix2pix method.
- None of the other methods as illustrated in Figure3 prove to be as good

as our method.
As Cycle GAN can produce translations similar to that of a fully supervised model,we can say that our approach is the best when compared to these baseline methods.

METHODOLGY

Overview

This project utilizes deep learning techniques, specifically cycle-consistent adversarial networks (CycleGANs). The network structure consists of generators and discriminators, with adversarial loss, cycle-consistency loss, and identity loss forming the key components of the loss function. Regularization techniques such as perceptual loss and total variation regularization are employed to improve image quality. A comprehensive quality metric is used to evaluate the performance of the architecture. Overall, it ensures the generation of high-quality and realistic translations between different image domains. Various applications of GANs are used in style transfer, object transfiguration, season transfer etc.

Network Structure of CycleGAN

CycleGAN trains two generator networks and two discriminator networks. Since it deals with the cross-domain transfer, let X be the source domain and Y be the target domain.

The generator networks are as follows:

Generator A: The aim is to learn a function G: X → Y such that G(X) is similar to Y. It takes image *A* from X and transforms it into image *B* that is comparable to an image fromY.
Generator B: The aim is to learn a mapping F: Y → X such that F(G(X)) is similar to X. It takes image *B* from Y and transforms it into image *A* that is comparable to an image from X.

The architecture of both generator networks is the same, but we train them separately.

The discriminator networks are as follows:

Discriminator A: It works to distinguish between the image generated by the G mapping,represented as G(X), and the real image *y* from the domain Y.

Discriminator B: Its work is to distinguish between the image generated by the F mapping, represented as F(Y), and the real image *x* from the domain X.
The architecture of both discriminator networks is the same, but we train them separately.

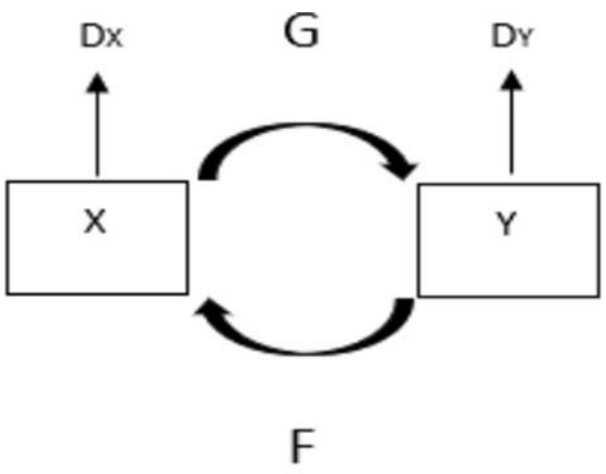


Figure 2: Illustration of CycleGAN with two generator & adversarial discriminator networks

Generator Architecture

It contains the following blocks:

- *Convolution block:* It contains 2D convolution layers, followed by instance batchnormalization and 'relu' as activation function.
- *Residual block:* It contains two 2D convolution layers which are followed by a batch normalization layer each with a momentum equal to 0.9. The generator network contains six residual blocks.

  - *Upsampling block:* It contains a transpose 2D convolution layer with relu as the activation function. The generator network has two upsampling blocks.
  - *Final convolution layer:* The last layer is a 2D convolution layer and uses tanh as the activation function.

Discriminator Architecture

It is a deep convolutional neural network and contains several convolution blocks. It inputs an image and predicts whether the image is fake or real.

LOSS FUNCTION

The loss function of the model is a weighted sum of the adversarial loss and cycle-consistency loss.

$$\begin{aligned}\mathcal{L}(G, F, D_X, D_Y) =& \mathcal{L}_{\mathrm{GAN}}(G, D_Y, X, Y) \\ &+ \mathcal{L}_{\mathrm{GAN}}(F, D_X, Y, X) \\ &+ \lambda\mathcal{L}_{\mathrm{cyc}}(G, F),\end{aligned}$$

Adversarial Loss

It is the loss between the images generated by the generator networks and the image from the real distributions. Since we have two generator mappings, we apply the adversarial loss to both.

$L_{GAN}(G, D_Y, X, Y)$ is the first adversarial loss, and $L_{GAN}(F, D_X, Y, X)$ is the second adversarial loss.

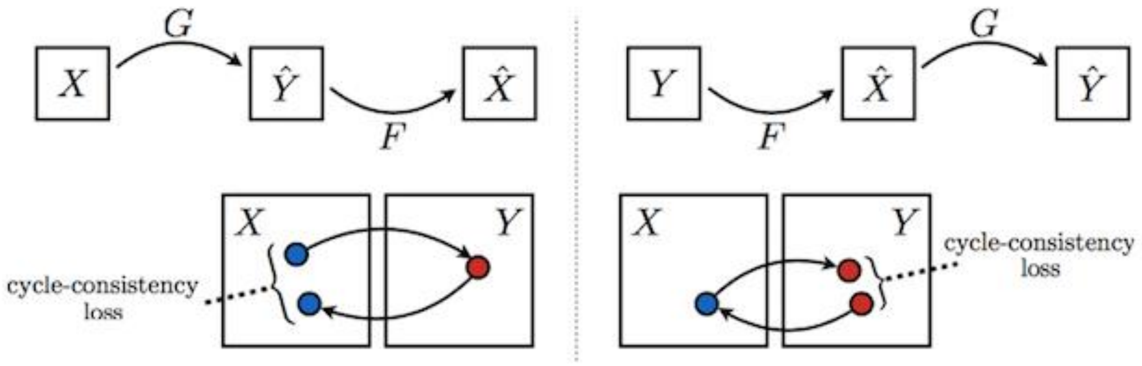
Cycle Consistency Loss

The problem with only adversarial loss is that the network can map the same set of input images to any permutation of images in target domain Y. There are many possible mapping functions between xi and yes. Cycle consistency loss overcomes this problem by reducing possible mappings.

A cycle consistent mapping is such that a function can translate an image from X to another image in Y, and generate back the original image.

$x \rightarrow G(x) \rightarrow F(G(x)) \sim x$ Forward cycle consistent mapping

$y \rightarrow F(y) \rightarrow G(F(y)) \sim y$ Backward cycle consistent mapping



<u>REGULARIZATION</u>

To apply regularization we use Batch Normalization. This acts as a regularizer to overcome the overfitting problems in the Deep Learning model to some extent. Apart from that it also speeds up the learning process of the model as it standardizes input. We have applied the same in both generator and discriminator after the last convolution layer.

DATASET

In this project, we work with the Monet2Photo dataset. This consists of 1193 Monet paintings and 7038 natural photos with each split into train and test subsets. Many CycleGAN datasets including this are easily available in the TensorFlow Datasets collection. After loading the dataset the first generator maps train_x to train_y and the second generator maps in reverse. Further, we define a few hyperparameters like lambda, initializer, and optimizers.

QUALITY METRICS

In GAN generator models there is no objective loss function and so often to judge we use manual and several other qualitative/quantitative measures. We have compared our method to various baseline methods both in terms of quality of the result produced and quantitatively. Apart from this we did study ablation for the full loss function.
While comparing to baseline methods we initially did record the results on paired datasets, to use them as a metric for comparison. Then we recorded the per-class and per-pixel accuracies of models and frameworks and compared them with our method.

EXPERIMENTAL SETUP

The experiment utilizes two datasets: one containing Monet-style paintings and the other with real photos.
Monet Dataset: A collection of Monet-style paintings. [Dataset is partitioned as Test_Monet and Train_Monet]
Photo Dataset: A set of real-world photos. [Dataset is partitioned as Test_Photo and Train_ Photo]

This project has been conducted on GoogleColab using the notebook: "CycleGAN_Monet_to_Photo.ipynb". As GoogleColab provides a free GPU environment, which is we used for training the deep learning model efficiently.

Library used:
TensorFlow: Deep learning framework for implementing and training the CycleGAN model.
Keras: This library offers a collection of pre-built layers, loss functions, optimizers, and utilities that simplify the process of constructing complex neural networks.
NumPy: For numerical operations and data manipulation.
Matplotlib: For visualizing images and training progress.
PIL (Pillow): For image processing tasks.

Model Architecture:
The CycleGAN consists of two sets of Generators and Discriminators:
Generators (G): There are two generators, one for transforming images from Monet to Photo domain (G_monet2photo) and the other for Photo to Monet domain

(G_photo2monet). We designed the generator with 15 convolutional , 17 instance normalizations and relu activation function .

conv2d_transpose (Conv2DTranspose)	(None, 128, 128, 128)	295040	['add_5[0][0]']
instance_normalization_15 (InstanceNormalization)	(None, 128, 128, 128)	256	['conv2d_transpose[0][0]']
activation_9 (Activation)	(None, 128, 128, 128)	0	['instance_normalization_15[0][0]']
conv2d_transpose_1 (Conv2DTranspose)	(None, 256, 256, 64)	73792	['activation_9[0][0]']
instance_normalization_16 (InstanceNormalization)	(None, 256, 256, 64)	128	['conv2d_transpose_1[0][0]']
activation_10 (Activation)	(None, 256, 256, 64)	0	['instance_normalization_16[0][0]']
reflection_padding2d_13 (ReflectionPadding2D)	(None, 262, 262, 64)	0	['activation_10[0][0]']
conv2d_15 (Conv2D)	(None, 256, 256, 3)	9411	['reflection_padding2d_13[0][0]']
instance_normalization_17 (InstanceNormalization)	(None, 256, 256, 3)	6	['conv2d_15[0][0]']
activation_11 (Activation)	(None, 256, 256, 3)	0	['instance_normalization_17[0][0]']

```
Total params: 7,845,065
Trainable params: 7,845,065
Non-trainable params: 0
None
```

Generator Model

Discriminators (D): There are two discriminators, one for distinguishing real Monet images from fake ones (D_monet) and the other for real Photo images from fake ones (D_photo).

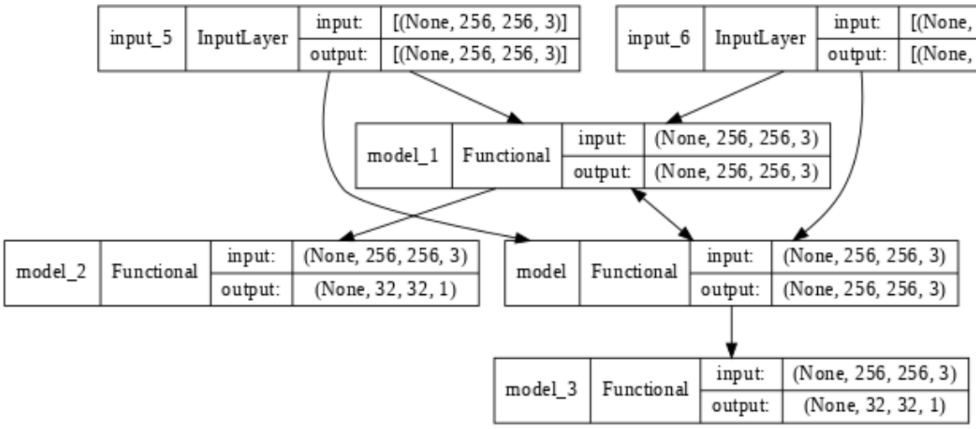
Layer (type)	Output Shape	Param #
input_3 (InputLayer)	[(None, 256, 256, 3)]	0
conv2d_32 (Conv2D)	(None, 128, 128, 64)	3136
leaky_re_lu (LeakyReLU)	(None, 128, 128, 64)	0
conv2d_33 (Conv2D)	(None, 64, 64, 128)	131200
leaky_re_lu_1 (LeakyReLU)	(None, 64, 64, 128)	0
instance_normalization_36 (InstanceNormalization)	(None, 64, 64, 128)	256
conv2d_34 (Conv2D)	(None, 32, 32, 256)	524544
leaky_re_lu_2 (LeakyReLU)	(None, 32, 32, 256)	0
instance_normalization_37 (InstanceNormalization)	(None, 32, 32, 256)	512
conv2d_35 (Conv2D)	(None, 32, 32, 512)	2097664
leaky_re_lu_3 (LeakyReLU)	(None, 32, 32, 512)	0
instance_normalization_38 (InstanceNormalization)	(None, 32, 32, 512)	1024
conv2d_36 (Conv2D)	(None, 32, 32, 1)	8193

```
Total params: 2,766,529
Trainable params: 2,766,529
Non-trainable params: 0
```

Discriminator Model

Next we did CycleGAN as shown in the snapshot.

```
Total params: 21,223,188
Trainable params: 15,690,130
Non-trainable params: 5,533,058
```



CycleGAN Architecture

We have compiled our model using Adam optimizer and MAE(Mean Absolute Error) and MSE(Mean Squared Error) loss functions.

ABLATION STUDY

Ablation study, in the context of Machine Learning, is a systematic approach that involves the deliberate removal or exclusion of specific features or components from a network or model to examine their impact on performance. This technique provides valuable insights into the underlying mechanisms and functionalities of the network.
In our model of CycleGANs, we conducted an ablation study to gain a deeper understanding of its behavior and identify crucial features that contribute significantly to its robustness. Several critical aspects of our CycleGAN model were subjected to ablation, and their effects were meticulously analyzed.

a) **Loss function:** The loss function in our CycleGAN model is formulated as a weighted combination of adversarial loss and cycle-consistency loss. To explore the significance of the loss function, we performed ablations by completely removing it from the training process. The results were striking, as the absence of the loss function led to a substantial degradation in the quality of the generated outputs. This observation emphasizes the loss function's critical role in guiding the model during training and ensuring the production of meaningful results.

Furthermore, we examined the impact of removing the cycle-consistency loss alone. This experiment also yielded noteworthy findings, as the elimination of cycle consistency caused a significant decline in the quality of the generated photos. It highlights the importance of the cycle-consistency term in enforcing the consistency between the original and reconstructed images, which is instrumental in enhancing the overall performance of the CycleGAN model.

b) **Cycle Consistency:** To delve deeper into the significance of cycle consistency, we performed additional ablations by selectively removing the cycle-consistency term in one direction at a time. We established two models for this analysis:

i) **GAN + Only Forward cycle loss**: In this configuration, we retained the GAN loss and only the forward cycle-consistency loss. Despite this partial preservation of cycle consistency, the results were unsatisfactory. The model exhibited instability and mode collapse, implying that the forward cycle loss alone was insufficient to ensure the desired level of performance and coherence in the generated images.

$$\mathbb{E}_{x\sim p_{\text{data}}(x)}[\|F(G(x))-x\|_1]$$

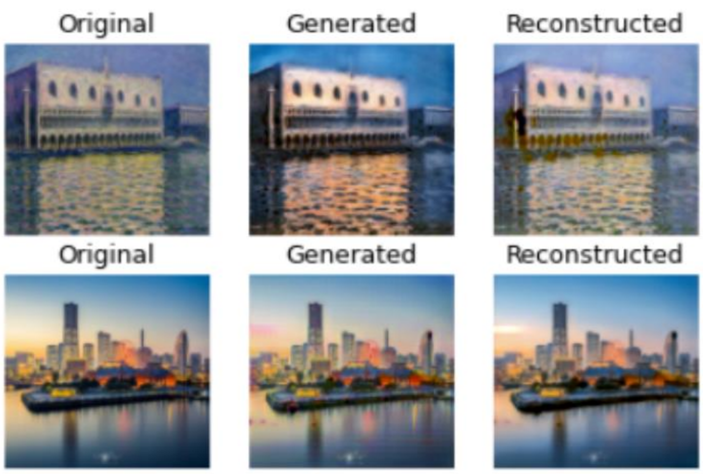
ii) **GAN + Backward cycle loss:** Similarly, we retained the GAN loss and only the backward cycle-consistency loss in this setup. Once again, the outcomes were suboptimal, indicating that relying solely on the backward cycle loss led to instability and mode collapse, like the previous ablation.

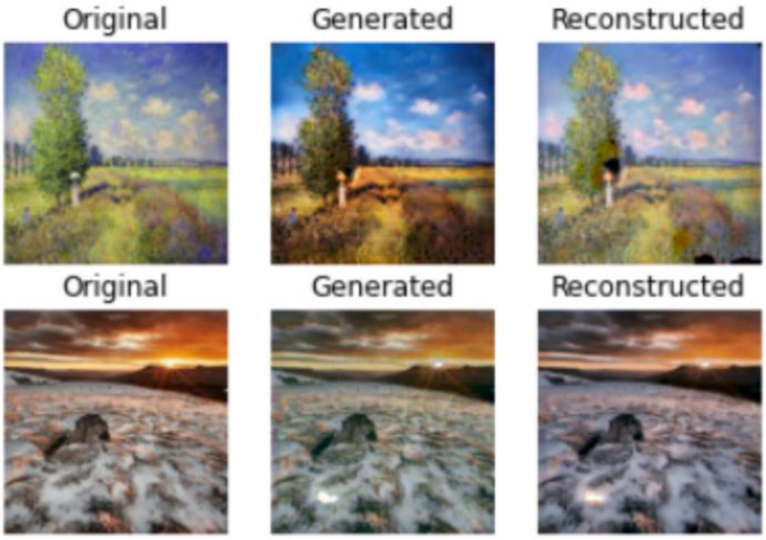
$$\mathbb{E}_{y\sim p_{\text{data}}(y)}[\|G(F(y))-y\|_1]$$

Overall, these ablation experiments shed light on the vital role of cycle consistency in the CycleGAN model. The bidirectional enforcement of cycle-consistency is crucial for maintaining stability and preventing mode collapse during the image-to-image translation process.

Our ablation study provided comprehensive insights into the significance of different components in our CycleGAN model. It demonstrated the indispensability of the loss function and the crucial role of cycle consistency in achieving high-quality image generation. This understanding empowers us to make informed decisions in designing and optimizing our CycleGAN model and contributes to the advancement of image-to-image translation techniques in the realm of Machine Learning.

RESULTS



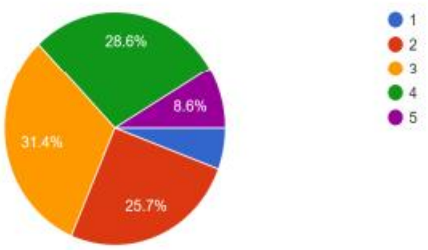


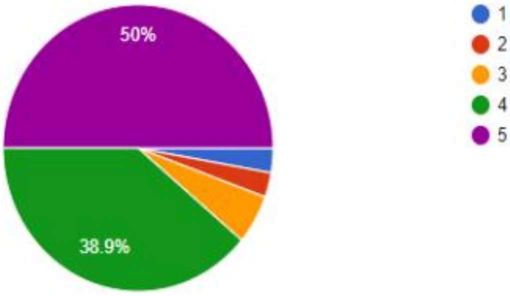
Apart from this from our ablation studies , we concluded some major insights:

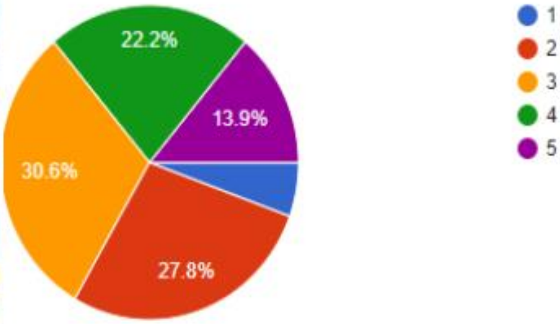
- Both the cycle GAN as well as cycle consistency loss functions when removed decrease the quality of results . Therefore , we conclude that they are important.
- Secondly , we tried evaluating the model with loss in one direction only but that resulted in mode collapse .

EVALUATION

We conducted a survey in which 13 colleagues participated through google form. They had to rate the photos generated from our model on the scale from 1 to 5, where 1 is the worst(looks like the painting) & 5 corresponds to the best(looks most realistic).
Below attached are results corresponding to the photos generated.







DISCUSSION

The future scope of the unpaired image-to-image translation using CycleGAN project is quite promising, as it falls within the broader domain of generative models and computer vision. Some of them are:

Improved performance: As the technology continues to develop, we can expect to see improved performance from CycleGAN models. This means that models will be able to generate more realistic and high-quality images.

New applications: As the technology becomes more widely available, we can expect to see new applications for CycleGAN. For example, CycleGAN could be used to create new virtual worlds or to generate realistic avatars for online games.

Impact on society: CycleGAN has the potential to have a significant impact on society. For example, CycleGAN could be used to improve the quality of medical images, which could lead to earlier diagnosis and treatment of diseases.

As advancements in this field continues, there are several applications and potential research directions that can be explored using this project:

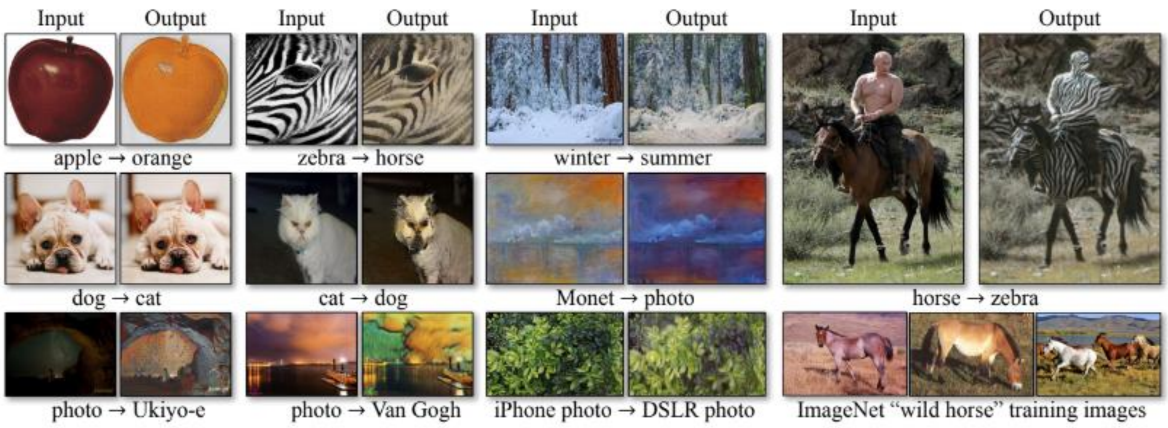
- **Artistic Style Transfer:** The ability to translate images between different artistic styles has numerous applications in the creative industry.
- **Super-resolution:** CycleGAN can be used to super-resolve images. This means that it can take a low-resolution image and generate a high-resolution image that is similar to the original image.
- **Season Transfer**: CycleGAN can be used to change the season of an image, transforming images taken in one season (e.g., summer) to look as if they were

captured in a different season (e.g., winter).

- **Data Augmentation:** By generating additional samples in a different domain, it can help improve the generalization and performance of models.
- **Object transfiguration:** The modal can be used to translate one object class from imagenet to another object class.
- **Image restoration:** CycleGAN can be used to restore images that have been corrupted or damaged. For example, you could use CycleGAN to repair a blurry photo or to remove noise from an image.
- **Colorizing legacy photographs:** This method can be used to turn black & white photos into colored images.

LIMITATIONS

There are numerous usual failure instances. Tasks that require geometric modifications have little success. This failure is probably due to generator architectures which might be tailor-made for exact overall performance on look modifications. Handling numerous and intense transformations, especially geometric modifications, is essential for our work.
Some failure instances are probably due to the distribution traits of the schooling datasets. There may additionally be a lingering hole among the outcomes, potential with paired schooling information and people completed through unpaired methods. Sometimes, this gap can be very challenging or impossible to eliminate.
To resolve this uncertainty, some weak semantic tracking is required. Integrating invalid or semi-monitored data can result in significantly more powerful translators, even at a fraction of the cost of annotation of fully monitored systems. However, in most cases, completely unpaired data abound and should be used. We will try to push the limits of what is possible in this "unattended" environment.



Typical Failure cases of our method

The biggest risk that can lead to the failure of this project is inadequate performance or poor results in the image translation process. Some of those major risks are:

Poor Quality Output: The generated images may suffer from artifacts, distortion, or unrealistic appearances, leading to unsatisfactory results. This could render the project ineffective and limit its usability in practical applications.

Lack Of Training: If the training data is not diverse enough, the model will not be able to learn to generate a variety of images.

Mode Collapse: The model may struggle to capture the full diversity of the target domain, resulting in mode collapse where it produces only a limited set of outputs, irrespective of the input. This severely restricts the model's ability to perform diverse translations.

Overfitting: The model might be overfit to the training data, meaning it becomes excessively tuned to the specific training examples, making it less capable of handling unseen data during inference.

To address these risks, effort should be made to gather a large and diverse dataset from both domains to ensure the model can learn robust mappings. Data augmentation techniques can be applied to increase the effective size of the dataset. We should experiment with different architectures and hyperparameter configurations to find the best-suited ones for the specific task. Define appropriate evaluation metrics that align with the project's objectives, such as Fréchet Inception Distance (FID). Regularization techniques such as dropout, weight decay, or adversarial training should be implemented to prevent overfitting during training.

CONCLUSION

The main takeaway would be that CycleGAN represents a significant advancement in the field of image-to-image translation, offering a more efficient and versatile approach to handle unpaired data while generating high-quality and realistic results. However, it would also emphasize the need for further research to address its limitations and enhance its performance for even broader applications.

In this project, we have learned how to turn paintings into photos, a application of unpaired image to image translation using CycleGAN. We started with an introduction to CyleGANs and explored the architectures of networks involved in CycleGANs. We also explored the different loss functions required to train CycleGANs. This was followed by an implementation of CycleGAN in the Keras framework. We trained the CycleGAN on the monet2photo dataset and visualized the generated images, the losses, and the graphs for different networks. We also explored the real-world use cases of CycleGANs.

In conclusion, the project's contribution to the field of Deep Learning lies in the development of CycleGAN, a powerful and innovative model capable of unpaired image-to-image translation. By addressing the challenges of unpaired data and introducing the cycle consistency concept, CycleGAN expands the applications of generative models and offers a more efficient and flexible approach to image translation tasks. Its impact can be seen in various domains, from computer vision research to practical applications in creative fields.

REFERENCES

- Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros; Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017
- Generative Adversarial Networks Projects Kailash Ahirwar, Chapter 7
- Official PyTorch Implementations of Pix2Pix and CycleGAN
- An Introduction to GANs in Tensorflow[Online]- http://blog.aylien.com/introduction-generative-adversarial-networks-code-tensorflow/
- Generative Adversarial Nets (Goodfellow et al., 2014)
- CycleGAN tutorial by Tensorflow[Online] - https://www.tensorflow.org/tutorials/generative/cyclegan
- Taihong Xiao[Online] A Mathematical View towards CycleGAN
- Hardik Bansal Archit Rathore. [Online]understanding and implementing cyclegan in tensorflow.
- Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew P. Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 105–114, 2017.
- Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired imageto-image translation using cycle-consistent adversarial networks. In Computer Vision (ICCV), 2017 IEEE International Conference on, 2017.

APPENDIX

The project is implemented on the google colab. Link for the same is given below:

https://colab.research.google.com/drive/1pXUbkf4m3mcpKkttIOOEe9t9kxQvyc6#scrollTo=uG-ZPoKSqDzf

CONTRIBUTIONS OF TEAM MEMBERS

	Name	ID	Percentage Contribution
1	Utkarsh Khurana	20ucs215	*50%*
2	Shubhang Jain	20ucs192	*50%*