Artistic Style Transfer with Unpaired Training Data

Ruize Xu, Jianping Ye, Lichuan Zhang
Department of Electrical and Computer Engineering
Western University
London, Ontario, Canada, N6A 5B9
Email: rxu293@uwo.ca, jye64@uwo.ca, lzhan888@uwo.ca

***Abstract*– Computer generated image is one of the many challenges to be tackled in the field of computer vision. Generative adversarial networks are gaining increasing popularity as they are proven to be notably effective. In this article, we identify the current challenges and potentials of image-to-image feature learning and translation in the absence of paired training data. We also explore three cutting-edge models on style mapping and translation. The purposed models were trained, tuned, and evaluated. The findings of this study demonstrate that Cycle-Consistent Adversarial Network outperforms both DiscoGAN and neural style transfer model, achieving an Inception Score of 3.82 and Fréchet Inception Distance of 175.83. Moreover, the study provides evidence that CycleGAN is more suitable for collective style translation, and neural style transfer is more favored in case-specific scenarios.**

Keywords- unpaired Image-to-Image translation, style transfer, generative adversarial network, cycle-consistent adversarial network, neural style transfer, discover cross-domain adversarial network, convolutional neural network, machine learning.

I. INTRODUCTION

Major breakthroughs of deep learning models in recent year have dramatically draw people's attention to computer vision. Although the top one accuracy for object recognition algorithms had gone up to 90% as of today [1], we do not see similar improvement in computer image generation algorithms. The first reason is the lack of effective objective scorings mechanism to directly replace human subjective scorings. A second reason is that the data that is used for image generation or style transfer are mostly unpaired, meaning that there's no correct label for input images. A third reason is that the process for generating image is longer than object recognition. Object recognition algorithms summarized features in a picture and use them for classification. In contrast, image generation algorithms not only have to learn the features, but it must also be able to generate it and having another model (discriminator) to evaluate the output. Therefore, we have two models that compete against each other that make the overall performance unstable and harder to train.

Nevertheless, we do see big potentials in image generation algorithms. Many applications can be developed based on image generation algorithms. For example, they can be used for facial view generation. People can take a picture of themselves, and they can modify the angle between the camera and their face [2]. Another example can be cloth translation. Customer can upload their picture, and algorithms can generate the picture of them wearing the cloth they choose [2]. The image generation algorithm can also help designer to expand their imagination and reduce their work. Common tasks like translating photo to painting and vice versa can expand designer's imagination. Also, the designer can specify the desired feature to be generated into the output. This can reduce their time spend in drawing and be able to evaluate their ideas faster than before.

II. RELATED WORK

Image-to-image translation is a class of task whose objective is to learn a mapping between an input-output image pair. Initially the field suffers from the difficulty of obtaining paired training data as they are generally either too costly to prepare or simply not available. However, it was only until the groundbreaking paper by Zhu *el al.* [3], the paired data limitation was finally lifted. Our research discovers three promising approaches: cycle-consistent adversarial network (CycleGAN) by Zhu *el al.* [3], discover cross-domain adversarial network (DiscoGAN) by Kim *el al.* [4], and neural style transfer by Gatys *el al.* [5].

CycleGAN future develops the idea of transitivity and propose cycle consistency loss. The idea behind cycle consistency loss is that if there is a mapping X->Y, then its inverse mapping Y->X should return to its starting point, that is X->Y->X' ≈ X. Therefore, the model involves two generators-discriminator pairs to enforce this forward-backward translation.

DiscoGAN aims to discover relations between different domains also in the absence of paired data. Cross-domain relations could be similar colors, or objects. DiscoGAN adopts the cycle-consistency idea but it has two reconstruction loss, one for each domain, while CycleGAN involves just one.

Neural Style transfer leverages the power of Convolutional Neural Network optimized for object recognition to extract high-level semantic information that allow us to separate image content from style [5]. This approach separates, then incorporates the style of a style reference image into a content image.

In the current paper, all three approaches were investigated as they represent generally the state-of-the-art techniques in the field of unpaired image-to-image translation. Modifications on network architectures were attempted to optimize performance on our dataset.

III. METHODOLOGY

This section elaborates on the proposed methods and detailed procedures followed by an analysis and approach to the problem. It is divided into the following sections: data set details, data preparation, data preprocessing, exploratory data analysis, modeling, hyperparameter tuning, and evaluation process.

A. Data Set Details

The datasets used in the project comes from the Kaggle competition “I’m something of a painter myself” [6]. There are two separate sets of images, that are Monet’s painting and photographed pictures. The original Monet’s painting dataset contains 300 samples. We expanded Monet’s collection by including another Monet’s dataset found on Kaggle, which provides 1193 Monet’s paintings [7]. In order to avoid potential overlaps in these two Monet’s datasets, we decided to use the later Monet's dataset since it provides more training instances. In addition to Monet’s paintings, the competition provides 7038 photos to be translated to Monet’s style. During model development, 1193 photos were used in accordance with the size of Monet’s dataset. Both Monet’s paintings and photos have dimension of 256 by 256.

B. Data Preparation

1) Data Splitting: The data set was divided into training and test sets with ratios of 90% and 10%, respectively. The data set is batched with batch size of 1. The final training set contains 1073 instances, while the test set have 120 samples.

C. Data Preprocessing

Data augmentations were implemented since they were proven to improve accuracy and reduce overfitting [3, 8, 12]. Noticeably, data augmentation was done only on the train set, while the test set need only to be normalized.

1) Random horizontal flipping: The image is randomly flipped horizontally.

2) Random Resizing: The image is resized to a slightly larger size with bicubic interpolation method.

3) Random Cropping: The resized image is randomly cropped back to retain its original dimensions.

4) Data Scaling: The image pixel values were scaled to between –1 and 1.

D. Exploratory Data Analysis

The following images are randomly sampled from the training set of Monet and photos. The left column displays two Monet paintings, and the right column shows photos. From Fig. 1, it is visually detectable that objects in photos have sharper boundaries than those in Monet’s. The color segments are relatively more continuous and clustered. In contrast, colors in Monet’s paintings are more mixed into each together and show large variations regionally. Moreover, color channel distribution plots are displayed in Fig. 2. In the two distribution plots of photos on the right, we can see each color channel occupies nearly the entire dynamic range, but color channels in Monet take relatively narrower dynamic range. The effects of data augmentation are depicted in Fig. 3. The augmented image could be slightly off or up compared to their originals. Also, they may have opposite orientations.

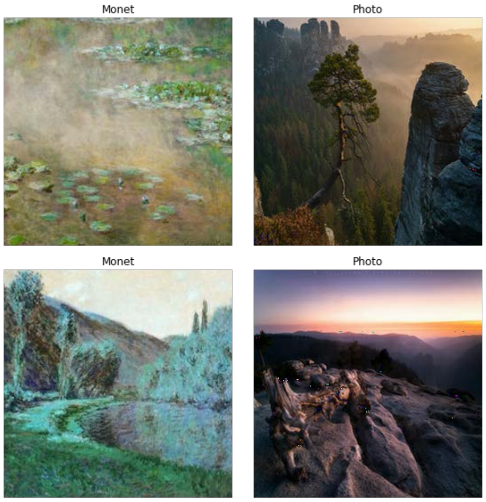


Fig. 1: Visualization of Monet versus Photo

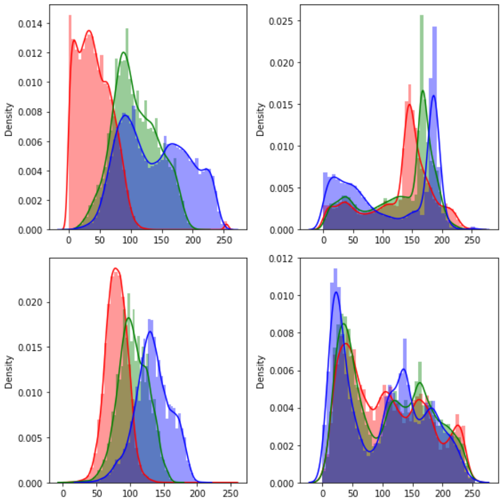


Fig. 2: Color Channels distribution plots of Monet (left) versus Photo (right)

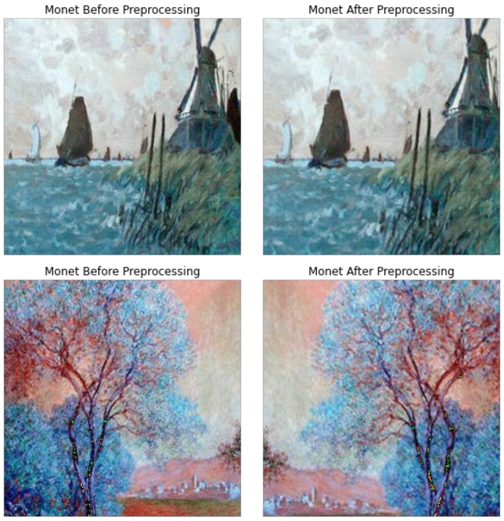

Monet Before Preprocessing
Monet After Preprocessing
Monet Before Preprocessing
Monet After Preprocessing

Fig. 3: Data Augmentation Effects

E. Modeling

The models were implemented using Keras machine learning library with a TensorFlow GPU backend [9, 10, 11, 17]. As mentioned earlier, CycleGAN, DiscoGAN, and neural style transfer models were developed to compare effectiveness. The GAN models were trained for 30 epochs. Early stopping was implemented to interrupt training when the monitored loss makes little progress for a specified number of epochs. Finally, mean square loss function was selected as loss function during training as it allows more efficient computation on gradients.

1) Cycle-Consistent Adversarial Network: The model consists of two generator-discriminator pairs. The two generators are responsible for generating images from domain X to domain Y, and Y to X, respectively. Accordingly, the two discriminators try to distinguish real X from fake X, and real Y from fake Y, respectively. During training, images generated by one generator is feed into the other generator to produce cycled images. The cycled image is expected to be as consistent as possible to its original. Cycle-consistent loss is computed as the mean absolute difference between original input and its cycled output. The generator model contains down-sampling blocks, which are convolutional layers, residual blocks, and finally up-sampling blocks, which are transposed convolutional layers to restore the original image size. The discriminator model is a PatchGAN structure that classify each N*N patch in an image as real or fake [16]. Noticeably, ReLU activation function is used in the generator, whereas the discriminator uses leaky ReLU. Instance Normalization is also utilized since the batch size was determined to be 1. Adam optimizers with learning rate of 0.0002 and first moment of 0.5 were used to stabilize training as suggested by Zhu *el al.* [3].

2) Discover Cross-Domain Adversarial Network: CycleGAN and DiscoGAN were basically published at the same time, but somehow CycleGAN are much more widely known and used than DiscoGAN. However, DiscoGAN has a lot of similarities to CycleGAN. It also seeks to have two GANs that "can map each domain to its counterpart domain", and "distinguish one domain from the other" [4]. Meanwhile, the differences to CycleGAN are also obvious. Firstly, it uses the encoder-decoder structure [4] as what normal Deep Convolutional GAN uses. Secondly, DiscoGAN does not use residual connections. Finally, it calculates two separate reconstruction loss and apply individually on each generator. In some scenarios, the training may encounter the mode collapse problem. Mode collapse refers to situation that different input images may be mapped to the same output image by the discriminator.

3) Neural Style Transfer: The neural style transfer approach takes a content image and a style reference image and learn to incorporate the style statistics of the style refences image into the content image through extracting intermediate layers activation of a pertained image classification network, for instance, VGG19. The model defines total loss function as the summation of style loss, content loss, and total variation loss [5]. Style loss is the L2 distance between Gram matrices of feature maps from the style reference image and the output image. The gram matric is a measure of means and correlations across different feature maps. Content loss is the L2 distance between the content image and the output image. Total variation loss is calculated from the high frequency components of the image [11]. It is used to reduce the artifacts that happened commonly on high frequency components. This generally makes the picture looks smoother. After defining three losses, gradient descent is applied to find an output image that minimize these losses and generate final output [18].

F. Hyperparameters Tuning

A randomized search is attempted for each model to optimize performances. Due to substantial amount of training time is required, we limit the number of hyperparameter combinations for each model to be maximum of four. For the two GAN models, the combination that yields the lowest photo-to-Monet generator loss was selected. For the neural style transfer model, the optimal setting was selected based on the highest Inception Score that the produced images obtained. Then, the corresponding models was used to evaluate the test set. Hyperparameters tested for each algorithm are summarized in Table I.

TABLE I
HYPERPARAMETERS AND PARAMETER DISTRIBUTIONS

Model	Hyperparameter	Parameters Tested
CycleGAN	Number of down-sampling & up-sampling blocks in generator	1, 2, 3
DiscoGAN	Number of iterations	1000,2000,3000
DiscoGAN	Learning rate	0.0002,0.0004
Neural Style Transfer	Number of iterations	1000, 1500, 2000

G. Reverse Scaling and Evaluation Process

Before evaluation, preprocessing normalization is inverted to retain original scale of pixel values, which is 0 to 255. After the models are trained and tuned, the test set is used to evaluate model performance.

IV. RESULTS AND DISCUSSION

Table II provides the optimal hyperparameters settings of each model.

TABLE II
OPTIMAL HYPERPARAMETER SETTINGS

Model	Optimal Hyperparameters
CycleGAN	Down-sampling & up-sampling blocks in generator = 2
DiscoGAN	Number of iterations = 3000
DiscoGAN	Learning rate = 0.0002
Neural Style Transfer	Number of iterations = 1000

A. Performance Measures

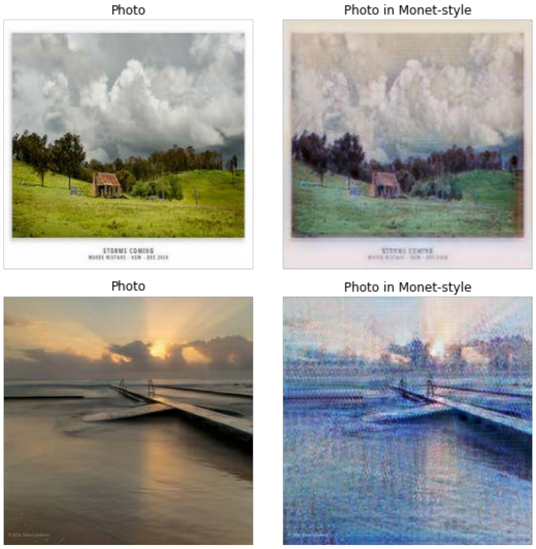
1) Fréchet Inception Distance (FID): FID is a metric to measure the distances between feature vectors calculated for real and generated images [6, 13]. By comparing the statistics between two collection of images, it captures similarity between synthetic images and real images from the target domain. A pre-trained deep neural network for image classification, inception V3, is used to extract computer vision features from two sets of images. The quality of generate images is inversely related to the FID result, that is smaller FID indicates better image quality.

2) Inception Score (IS): Inception score is another popular metric used in studies to measure GAN performance. It uses a pre-trained deep neural network for image classification, inception V3, to classify the generated images. Specifically, the probability of the image belonging to each class is predicted. This metric aims to capture two aspects of generated images: quality and diversity. Since the metric is found to be highly correlated with human subjective evaluation [14, 15], it is the primary metric for quantitative evaluation in this project. A higher IS score is desired.

3) Human Subjective Evaluation: The produced images were presented to evaluators and they were asked to identify which set of results is aesthetically and visually similar to Monet's style.

B. Cycle-Consistent Adversarial Network

Fig. 4 shows the translated photos produced by CycleGAN. It is observed that the translated images are notably differentiable from there originals. The translation of the first and fourth image are the most successful. Color segments and object boundaries are correctly identified and maintained while modifying the style statistics. The second image shows considerable noise and color mismatch.



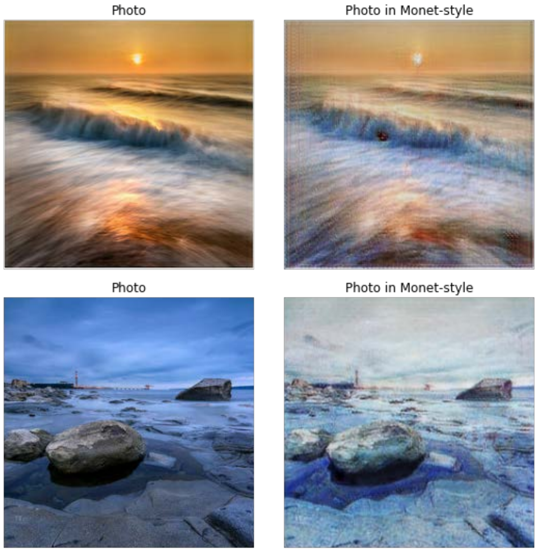
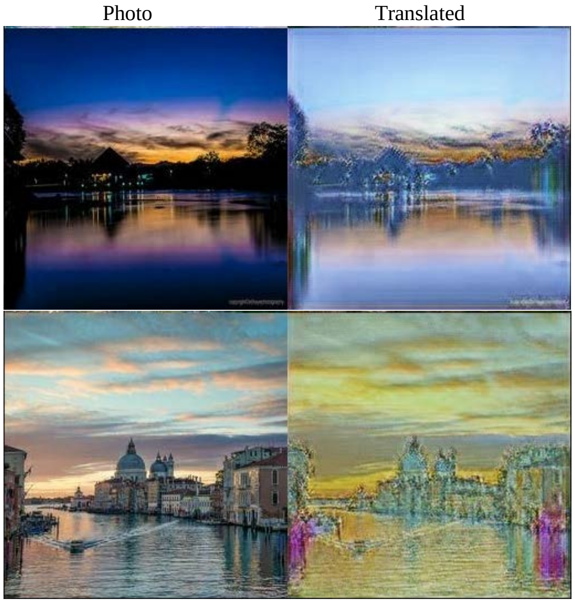


Fig. 4: CycleGAN Photos versus Monet-Style Photos

C. Discover Cross-Domain Adversarial Network

As we can see in Fig. 5, DiscoGAN can learn some of the characteristics from Monet's paintings. The translated images are visually different from their originals. In the translated photo set, the first one has a lot of blurry boundaries which might due to lack of details (almost entirely black) in the original photo. The second and the fourth experienced great color and style changes but a bit noisy, while the third one has the best visual effect.



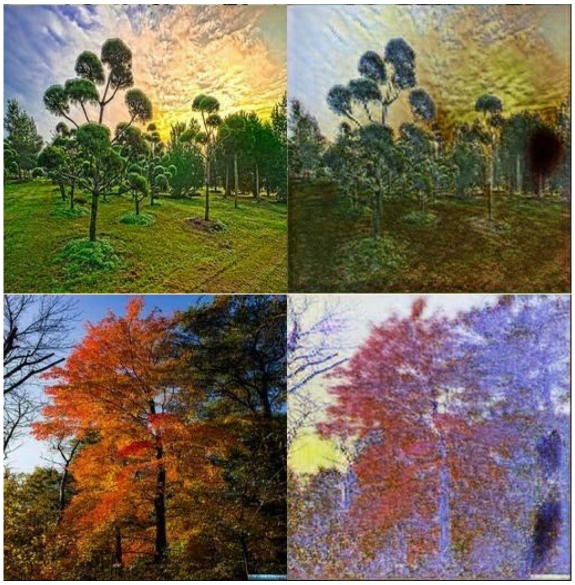


Fig. 5: DiscoGAN Photos versus Monet-Style Photos

D. *Neural Style Transfer*

In the result of the neural style transfer seen in Fig. 6, the produced images are clearly different than that of CycleGAN. The model successfully blends the style reference image into the content image. However, color segments and object boundaries can be distorted which result in slightly worse visual effects than CycleGAN.

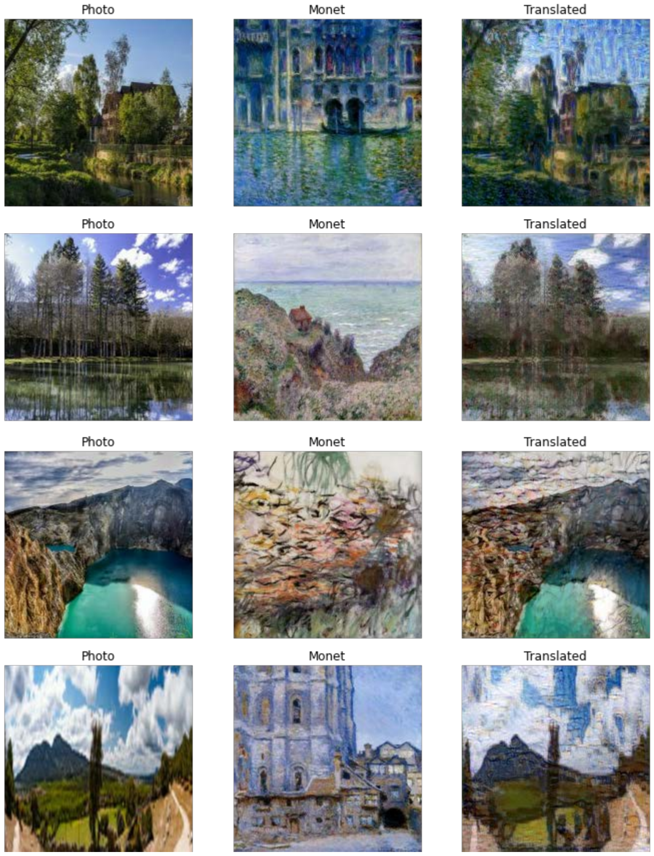


Fig. 6: Neural Style Transfer Photos versus Monet-Style Photos

E. *Comparison of Models and Discussion*

The comparison plots below (Figs. 7 & 8) show quantitative metric performance of the three models. It is shown that CycleGAN and neural style transfer algorithm achieve very similar FID results, and CycleGAN has the highest inception score.

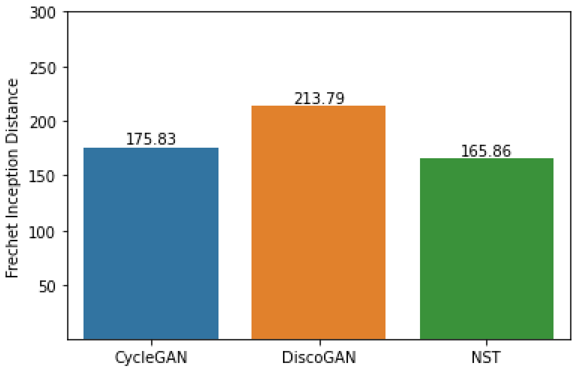


Fig. 7: Comparison Plot of Fréchet Inception Distance (FID)

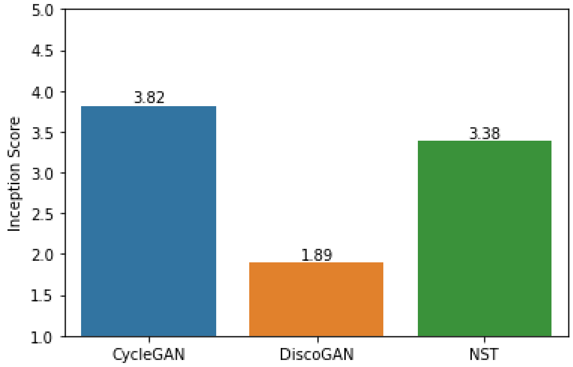


Fig. 8: Comparison Plot of Inception Score (IS)

In subjective evaluation, it turns out that images generated by CycleGAN are the most accepted by evaluators, followed by neural style transfer, and finally DiscoGAN.

In general, CycleGAN produces the most satisfactory results despite with some occasional fail cases, which can be attributable to the inherent instability of GAN. However, it is worth noticing that training CycleGAN involves significant time cost and the model is very sensitive to hyperparameters, which may require even more time to fine-tune. The outstanding advantage of CycleGAN is that it can collectively learn the style representation of a group of images without the need to find a proper one-to-one match between training data. CycleGAN is hugely benefit by adopting the ResNet structure. The residual connections allow gradients to bypass nonlinear activation functions and flow directly into certain layers, alleviating the vanishing gradient problem in a deep neural network. The PatchGAN discriminator is another component that set CycleGAN apart from other GAN models. With PatchGAN, more gradients signal can flow back to the generator which eventually help modifying local style statistics easier.

DiscoGAN follows the encoder-decoder structure in the generator, which mainly consist of convolutional-deconvolutional blocks. No residual connections are implemented. In the discriminator, the layout is comparable to usual CNN with sigmoid activation to output the probability that the entire image is real or fake. Due to these limitations, the results are less convincing, and the training process is more susceptible to mode collapse and unstable gradients.

In contrary to CycleGAN and DiscoGAN, neural style transfer takes an alternative path. Its shinning points include: (1) no training is required, thus lead to much quicker translation time; and (2) one-to-one image translation brings infinite possibilities. As this approach leverages on pretrained network, all we need to do is apply gradient descent algorithm to optimize the loss functions such that the balance between transferred style and content distortion is struck. The time needed to translate an image can be as fast as only 10 seconds on a normal laptop. Secondly, a photo can be translated to have different visual effects depending on which Monet's painting is selected as the reference image. This could potentially be an issue because it implies that style image may require manual pruning and careful selection to make the output image looks reasonable. This issue can be slightly mitigated by the quick translation speed. However, this algorithm has another weakness: the translated image inevitably carries some characteristics of the style image. For instance, some objects from the style image might appear on inappropriate positions in the content image, leading to a blurry and rugged appearance.

In summary, CycleGAN is the most suitable and best performing model in terms of collection style transfer. Nevertheless, neural style transfer is more favorable when one would like to translate the style of a specific painting onto an image.

V. Conclusion

This study investigated the unpaired image-to-image translation challenge of a Kaggle competition and examined the effectiveness and applicability of three deep learning models on tackling the task. Specifically, CycleGAN, DiscoGAN, and neural style transfer model were constructed to translate Claude Monet's painting style onto photos. The project started with data preprocessing in which several data augmentation techniques were adopted as a form of regularization. Then, exploratory analysis was conducted to visualize the artistic difference between paintings and photos, also their discrepancies in terms of color channel distributions. Next, three models were built, trained, tuned, and evaluated with FID and IS metric quantitatively. The produced images are further compared by human subjective evaluation.

The metrics results suggest that CycleGAN achieves an Inception Score of 3.82 and Fréchet Inception Distance of 175.83, both representing the top performance among the three models. The results of neural style transfer are slightly worse than that of CycleGAN.

The findings of this study demonstrate the outstanding effectiveness of CycleGAN on collection style transfer, whereas the neural style transfer model is more favorable in scenarios where case-specific translation is required.

This study can be further extended to include other painters' work. CycleGAN and DiscoGAN can be further optimized by using some modern training techniques or novel architectures to stabilize training and shorten training time while maintaining the quality of produced images.

References

[1] "ImageNet Benchmark (Image Classification)," *The latest in machine learning*. [Online]. Available: https://paperswithcode.com/sota/image-classification-on-imagenet. [Accessed: 20-Mar-2021].

[2] J. Brownlee, "18 Impressive Applications of Generative Adversarial Networks (GANs)," *Machine Learning Mastery*, 12-Jul-2019. [Online]. Available: https://machinelearningmastery.com/impressive-applications-of-generative-adversarial-networks/. [Accessed: 20-Mar-2021].

[3] Jun-Yan Zhu, Taesung Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2242–2251, doi: 10.1109/ICCV.2017.244.

[4] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, "Learning to Discover Cross-Domain Relations with Generative Adversarial Networks," 2017.

[5] L. Gatys, A. Ecker, and M. Bethge, "A Neural Algorithm of Artistic Style," *Journal of vision (Charlottesville, Va.)*, vol. 16, no. 12, p. 326–, 2016, doi: 10.1167/16.12.326.

[6] "I'm Something of a Painter Myself," *Kaggle*. [Online]. Available: https://www.kaggle.com/c/gan-getting-started. [Accessed: 20-Mar-2021].

[7] D. Oliveira, "TFRecords Monet paintings 256x256," *Kaggle*, 01-Sep-2020. [Online]. Available: https://www.kaggle.com/dimitreoliveira/tfrecords-monet-paintings-256x256. [Accessed: 21-Mar-2021].

[8] J. Brownlee, "How to Implement GAN Hacks in Keras to Train Stable Models," *Machine Learning Mastery*, 12-Jul-2019. [Online]. Available: https://machinelearningmastery.com/how-to-code-generative-adversarial-network-hacks/. [Accessed: 21-Mar-2021].

[9] A. K. Nain, "Keras documentation: CycleGAN," *Keras*. [Online]. Available: https://keras.io/examples/generative/cyclegan/. [Accessed: 20-Mar-2021].

[10] T. Kim and T. Han, "SKTBrain/DiscoGAN," *GitHub*. [Online]. Available: https://github.com/SKTBrain/DiscoGAN. [Accessed: 20-Mar-2021].

[11] F. Chollet, "Keras documentation: Neural style transfer," *Keras*. [Online]. Available: https://keras.io/examples/generative/neural_style_transfer/. [Accessed: 20-Mar-2021].

[12] A. Géron, "Generative Adversarial Networks," in *Hands-on Machine Learning with Scitkit-Learn, Keras, and Tensorflow*, 2nd ed., Sebastopol, CA, USA: O'Reilly Media, 2019, pp. 752.

[13] J. Brownlee, "How to Implement the Frechet Inception Distance (FID) for Evaluating GANs," *Machine Learning Mastery*, 10-Oct-2019. [Online]. Available: https://machinelearningmastery.com/how-to-implement-the-frechet-inception-distance-fid-from-scratch/. [Accessed: 21-Mar-2021].

[14] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved Techniques for Training GANs," 2016.

[15] J. Brownlee, "How to Implement the Inception Score (IS) for Evaluating GANs," *Machine Learning Mastery*, 10-Oct-2019. [Online]. Available: https://machinelearningmastery.com/how-to-implement-the-inception-score-from-scratch-for-evaluating-generated-images/. [Accessed: 05-Apr-2021].

[16] P. Isola, Jun-Yan Zhu, Tinghui Zhou, and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5967–5976, doi: 10.1109/CVPR.2017.632.

[17] T. Du, "DiscoGAN," *GitHub*, 07-Jun-2020. [Online]. Available: https://ustccoder.github.io/2020/06/12/generative_adversarial%20DiscoGAN/. [Accessed: 14-Apr-2021].

[18] Tensorflow, "Neural Style Transfer: Creating Art with Deep Learning using tf.keras and eager execution", 3-Aug-2018. [Online]. Available:

https://medium.com/tensorflow/neural-style-transfer-creating-art-with-deep-learning-using-tf-keras-and-eager-execution-7d541ac31398. [Accessed: 10-Apr-2021].

APPENDIX

A. Research

Preliminary research was conducted by all members and at that time, the assigned model of each member was identified.

B. Model Development

Each member is responsible entirely for the development of their research and assigned model, including data loading, preprocessing, modeling, tuning, and evaluation.

C. Presentation

Each member is responsible for a particular part of the presentation as well as describing the model they experimented and the corresponding results.

D. Report Writing

FD: First Draft
MR: Major Revision

	Ruize Xu	Jianping Ye	Lichuan Zhang
Abstract			FD & MR
Introduction			FD & MR
Related Work	FD	MR	FD
Methodology (CycleGAN)		FD & MR	
Methodology (DiscoGAN)	FD & MR		
Methodology (NST)			FD & MR
Methodology (other sections)	FD	FD & MR	FD
Result and Discussion	FD	FD & MR	FD
Conclusion	FD	FD & MR	
Final Editing and Formatting	☑		
Final Review		☑	

E. Repository

As Bitbucket does not provide preview of Jupyter Notebook files, we provide the following GitHub repository as viewing alternative. However, Bitbucket repository was created as well.

https://github.com/jye64/ECE9039-Project

https://bitbucket.org/RickyXu/i-am-a-painter-myself/src/master/