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 potential of image-to-image feature learning and translation in the absence of paired training data. We also explore three cutting-edge models that represent state-of-the-art techniques in style mapping and translation. The purposed models are trained, tuned, and evaluated. At this stage, the findings of this study demonstrate that the Cycle-Consistent adversarial network outperforms both DiscoGAN and neural style transfer model.**

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

I. INTRODUCTION

Major breakthroughs of deep learning model 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. First reason is that there's no 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 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 to classify. 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 interact with each other that make the overall performance of the models unstable and harder to train.

Nevertheless, we do see a big potential in image generation algorithms. Many applications can be developed based on image generation algorithms. For example, it 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 algorithm 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 things 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. However, it was only until Zhu el al. [3] groundbreaking paper, the paired data limitation was finally lifted. Our findings discover mainly three 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 the result of its inverse mapping Y->X should produce similar results, that is X->Y->X' $\approx$ X. Therefore, the model involves two generators-discriminator pairs.

DiscoGAN aims to discover relations between different domains also in the absence of paired data. Cross-domain relations could be similar color across two images, or similar objects. DiscoGAN adopts cycle-consistency loss but it has two reconstruction loss, one for both domains, 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 taken 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 85% and 15%, respectively. The data set is batched with batch size of 1. The final training set contains 1014 instances, while the test set have 179 samples.

C. Data Preprocessing

A set of preprocessing operations are implemented as they were proven to improve accuracy [3, 8, 12]. Noticeably, the training set is preprocessed with all the following operations, while the test set should only 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 bilinear 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 normal 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 regional. In contrast, colors in Monet's paintings are mixed and show large variations regionally.

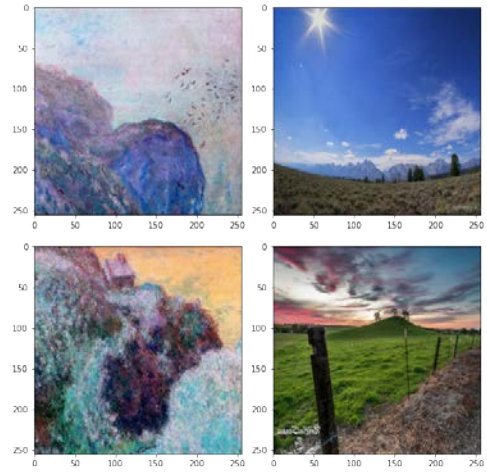

Fig 1: Visualization of Monet (left) vs. Photo (right)

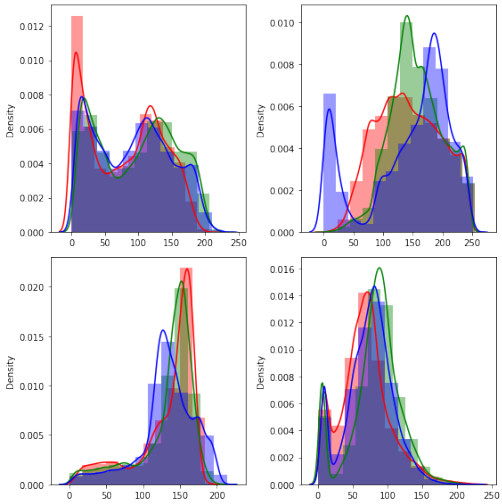

Fig 2: Color Channels distribution plots of Monet (left) vs. Photo (right)

E. Modeling

The models were implemented using Keras machine learning library with a TensorFlow GPU backend [9, 10, 11]. As mentioned earlier, CycleGAN, DiscoGAN, and neural style transfer models were developed to compare effectiveness. Early stopping was implemented to interrupt training when the monitored loss stops decreasing 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 generators and two discriminators. The two generators are responsible for generating images from domain X to domain Y, and Y to X, respectively. Accordingly, the two discriminators will try to distinguish real X from fake X, and real Y from fake Y, respectively. 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 conventional image classification

CNN architecture. Noticeably, ReLU activation function is used in the generator, where 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: 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]. However, it also uses a reconstruction loss, which is a measure of how well the original image is reconstructed after the two translations from domain X to domain Y and back to domain X. In some scenarios, the reconstructed images may encounter mode collapse problems. Mode collapse refers to scenarios that different input images may be mapped to the same output image by the discriminator. This extra loss function can solve this issue notably.

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 intermediate layers of pertained image classification network, for instance, VGG19.

F. Hyperparameters Tuning

A grid search is scheduled 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 4. The combination that yields the lowest photo-to-Monet generator loss will be selected and the corresponding model will be used to evaluate on the test set. Hyperparameters planned for each algorithm are summarized in Table I.

TABLE I
HYPERPARAMETERS AND PARAMETER DISTRIBUTIONS

Model	Hyperparameter	Parameter Distributions
CycleGAN	Number of down-sampling & up-sampling blocks in generator	2, 3
CycleGAN	Number of down-sampling block in discriminator	3, 4
DiscoGAN	Number of iterations	200,500,1000
DiscoGAN	Learning rate	0.0002,0.0004
Neural Style Transfer	Number of iterations	1000, 2000, 3000

G. Reverse Scaling and Evaluation Process

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

IV. PRELIMINARY RESULTS AND DISCUSSION

Table II provides the current hyperparameters settings of each model. At this stage, preliminary image outputs were obtained but implementation of performance metrics are yet to be finished.

TABLE II
CURRENT HYPERPARAMETER SETTINGS

Model	Current Hyperparameters
CycleGAN	Down-sampling & up-sampling blocks in generator = 2
CycleGAN	Down-sampling block in discriminator = 3
DiscoGAN	Number of iterations = 500
DiscoGAN	Learning rate = 0.0002
Neural Style Transfer	Number of iterations = 3000

A. Performance Measures

1) Memorization-informed Fréchet Inception Distance (MiFID): MiFID is a modified version of Fréchet Inception Distance (FID). FID is a metric to measure the distances between feature vectors calculated for real and generated images [6, 13]. The MiFID metrics is also the evaluation metric used by the Kaggle competition. The quality of generate images is inversely related to the MiFID score, that is smaller MiFID indicates better image quality.

2) Inception Score (IS): Inception score is another popular metric used in studies to measure GAN performance. This metric tries to address two aspects of image: quality and diversity. It is also found that the metric is highly correlated with human subjective evaluation [14]. In contrary to MiFID, higher inception score is desirable.

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

B. Cycle-Consistent Adversarial Network

Fig. 3 shows the translated photos produced by CycleGAN. It is observed that the model is capable of detecting color segments. Original object boundaries are maintained and properly translated.

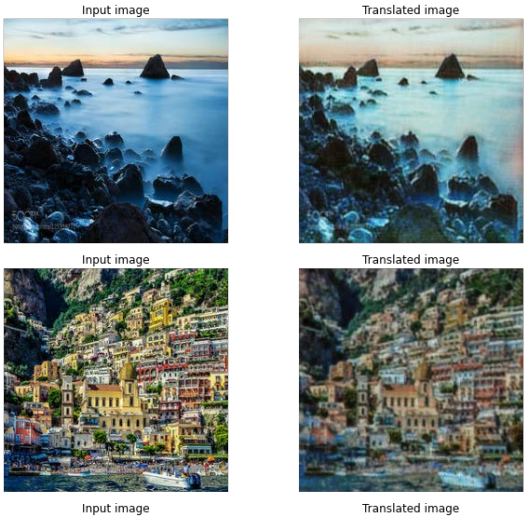

Fig. 3: CycleGAN Photos vs. Monet-Style Photos

C. Discover Cross-Domain Adversarial Network

As we can see in Fig. 4, DiscoGAN can learn some of the features from Monet's paintings. The translated images are visually different from those original photos. Although there are some certain colors tend to be much darker than the input images.

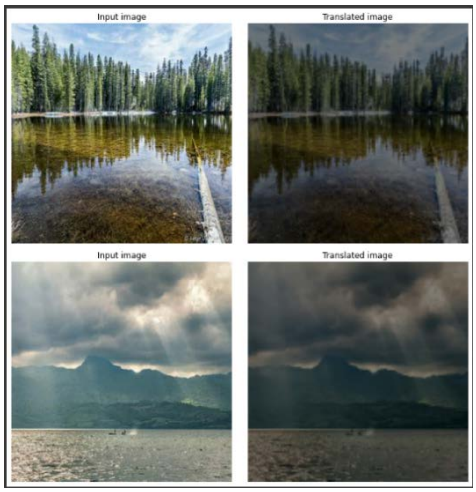

Fig. 4: DiscoGAN Photos vs. Monet-Style Photos

D. Neural Style Transfer

In the result of the neural style transfer seen in Fig. 5, the produced images are visually different than CycleGAN. The model is able to blend the style reference image into the content image. However, original color segments and object boundaries are also blended which result in slightly worse visual effects than CycleGAN.

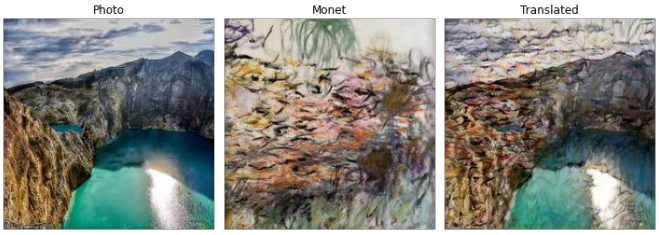

Fig. 5: Neural Style Transfer Photos vs. Monet-Style Photos

V. Next Step

In the next step, we plan to perform hyperparameter tuning for all three models and conduct evaluations using the metrics outlined in the performance measures section. Expected schedules of deliverables are summarized below in Table III.

TABLE III
SCHEDULES OF DELIVERABLES

Deliverables/Tasks	Expected Time	Due Day
Hyperparameter Tuning	Mar 26	NA
Evaluation and Comparison	Mar 28	NA
Project Presentation	Mar 30	Apr 1st
Final Report	Apr 12th	Apr 19th

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.