Evaluating CycleGANs for generating art: from realism to abstract art

David Lüdke
Exchange student, david-luedke@web.de

Mattias Ulmestrand
Chalmers Master's student, matulm@student.chalmers.se
(Dated: June 6, 2021)

Generative Adversarial Networks (GANs) have successfully been applied in many fields of image-to-image translation. Specifically, cycle-consistent GANs (CycleGANs) have shown to be widely applicable for transforming between domains even for unpaired data sets. In order to investigate strengths and limitations of CycleGANs, this study aims to assess their applicability for generating art by different painters. Through qualitative and quantitative analysis of visual results, the study may shed light on which kinds of mappings are feasible for CycleGANs. A selection of several painters was used for training, with styles ranging from realism to abstract art. Two different generator architectures were used, the first being U-Net and the second being ResNet. U-Net is also shown to perform better with concatenating layers switched to adding layers. The discriminator was kept the same PatchGAN architecture for both generator architectures, for a fair comparison between the two generators. The overall results of the study suggest that painters with styles closer to reality are easier to reproduce. When subjected to more abstract styles such as Kandinsky or Warhol, the U-Net architecture was found to struggle more in producing convincing art than ResNet. The U-Net was found to be more prone to introducing artifacts for these styles, but can at the same time be quite convincing for more realistic painters such as Monet and Munch. The ResNet architecture appears more stable, and may overall produce more convincing artworks.

I. INTRODUCTION

Humans have been creating art in many forms throughout all of known history. From the earliest cave imprints to modern paintings and music, art is innately, and arguably uniquely, human. As the craft relies upon subjectivity and invoking feelings, it is no easy task to quantify how to create a work of art and encode this information to a computer. In the world of machine learning, however, one of the most promising and recent methods of generating art is Generative Adversarial Networks (GANs) [1].

A GAN is a type of neural network architecture which aims to transform input X to output Y. Mapping input to output is done by a generator function G, $G : X \to Y$. The generator is trained in parallel with a discriminator D, which learns to distinguish real data from the Y domain and generated data. In the original GAN paper, they were used for generating images from noise [1]. Examples of usages in the paper include handwritten numbers and facial images. Since the introduction of GANs, architectures such as StyleGAN have accomplished deceptively realistic generated facial images [2]. Another usage is image completion, which GANs were successfully used for in [3]. Furthermore, GANs have been used for super resolution tasks. TecoGAN is one such GAN, which can improve image quality as well as video quality with remarkable detail [4].

GANs can either be used with paired or unpaired data. In image-to-image translation, Paired data relies on corresponding X and Y images and typically simplifies the training procedure, whereas unpaired data is easier to come by and structure as no correlation is required between X and Y. A specific type of GAN architecture suitable for unpaired images is cycle-consistent adversarial networks (CycleGANs) [5]. CycleGANs were first introduced in 2017 and have proven to be effective in translating images across domains such as photographs to paintings or zebras to horses [5]. Almahairi et al. also used CycleGANs for transforming between male and female faces, as well as generating shoes from shoe outlines [6]. CycleGANs have also successfully been applied outside the realm of image-to-image translation. Kaneko et al. used CycleGANs for speech conversion with better results than previous methods [7].

An alternative solution for generating art is to use pre-trained image classification networks with Gram matrices, introduced in [8]. This approach typically encodes the style in earlier convolutional layers with lower level of feature abstraction, whereas the content is encoded at the final convolutional layers with higher level of abstraction. The method can generate convincing results, but is most easily trained using one specific painting and photo combination, such as in [8]. One advantage of using GANs is thus that they can more generally learn painter styles rather than painting style.

A. Problem formulation

The aim of this paper is to investigate which styles and painters can successfully be reproduced in a CycleGAN setting. To this end, two models are compared, ResNet and U-Net [9],[10]. A wide selection of painters is used for training, ranging from realism to abstract art.

B. Hypothesis

Art styles which are close to reality are expected to be easier to reproduce. Since the distributions of the photo domain X and the art domain Y are similar, a small amount of processing should be sufficient in order generate a convincing work of art. Styles such as cubism and abstract art are expected to be more difficult to reproduce, since the distributions are farther apart. To evaluate the hypothesis, generator and discriminator loss will be evaluated for different painters. Results will furthermore be visually inspected.

II. METHODOLOGY

In this section, the CycleGAN is introduced with its corresponding losses and architectures. In addition, the general implementation and setup is presented.

A. CycleGAN

Instead of learning one highly unconstrained generator that maps from X to Y in a simple GAN setting, CycleGANs train two GANs in a cycle. This idea was first introduced in [11]. When training a CycleGAN, M samples $\{x_i\}_{i=1}^{M}$ are given from domain X and N samples $\{y_j\}_{j=1}^{N}$ from the Y domain. Furthermore, two generators are trained, the mapping $G : X \to Y$ and the inverse mapping $F : Y \to X$. The two discriminators D_X and D_Y distinguish between y and $G(x)$ or x and $F(y)$ by outputting a reduced representation of the image with measures indicating whether the image is real. The cycle aspect relies on the idea of generating the initial image after cycling – transferring back and forth – between the domains with the generators. In order to successfully train the CycleGAN, several different types of losses are utilized. The losses are introduced in the section below.

1. Losses

a. Adversarial loss A standard GAN is trained using what is referred to as adversarial loss. With adversarial loss, the discriminator is trained to better discriminate generated images. At the same time, the generator learns to fool the discriminator. In the case of a CycleGAN, two adversarial losses are used:

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

and

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(F, D_X, X, Y) = &\mathop{\mathbb{E}}_{x \sim p_{\text{data}}(x)} \log(D_X(x)) \\ &+ \mathop{\mathbb{E}}_{y \sim p_{\text{data}}(y)} [\log(1 - D_X(F(x))]. \end{aligned} \tag{2}$$

b. Cycle consistency loss The cycle consistency loss reduces the possible mapping functions to be cycle consistent. The cycle consistency loss aims to minimize the difference between x and $F(G(x))$ in the cycle $x \to G(x) \to F(G(x)) \approx x$. The mapping restriction is formalized by the following loss:

$$\begin{aligned}\mathcal{L}_{\text{cycle}}(G, F) = &\mathop{\mathbb{E}}_{x \sim p_{\text{data}}(x)} [\|F(G(x)) - x\|_1] \\ &+ \mathop{\mathbb{E}}_{y \sim p_{\text{data}}(y)} [\|G(F(y)) - y\|_1]. \end{aligned} \tag{3}$$

c. Identity loss The identity loss is added to ensure the preservation of original image features [12], [13]. The identity loss has also been shown to be able to preserve colors of the original image [5]. The loss acts similarly to a regularization, and takes following form:

$$\begin{aligned}\mathcal{L}_{\text{identity}}(G, F) = &\mathop{\mathbb{E}}_{x \sim p_{\text{data}}(x)} [\|x - F(x)\|_1] \\ &+ \mathop{\mathbb{E}}_{y \sim p_{\text{data}}(y)} [\|y - G(y)\|_1]. \end{aligned} \tag{4}$$

d. Full loss Combining losses $a - c$, the total loss produces a constrained mapping that is cycle consistent and preserves input features. This gives the following objective:

$$\begin{aligned}\mathcal{L}(G, F, D_X, D_Y) = &\mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) \\ &+ \mathcal{L}_{\text{GAN}}(F, D_X, X, Y) \\ &+ \lambda \mathcal{L}_{\text{cycle}}(G, F) \\ &+ \lambda \mathcal{L}_{\text{identity}}(G, F), \end{aligned} \tag{5}$$

where λ is a hyper-parameter handling the relative impact of the cycle-consistency and identity loss. When training the model, the following is aimed to be optimized:

$$G^*, F^* = \min_{G,F} \max_{D_X, D_Y} \mathcal{L}(G, F, D_X, D_Y). \tag{6}$$

B. Implementation

In order to compare differences between different types of architectures, two distinct generator architectures, U-Net and ResNet, have been implemented. The same discriminator was used for both generator architectures. The discriminator was implemented as a 30×30 PatchGAN [14]. All models were implemented in Python using Keras with TensorFlow as backend.

1. Generator

There are many possible generator architectures for GANs. Two architectures, U-Net and ResNet, have shown to be widely applicable. Hence, these two distinct architectures were used to form the basis of the networks and are juxtaposed in the result section.

a. U-Net The U-Net architecture has in recent years seen much attention, especially in the field of semantic segmentation [15]. In addition, the encoder-decoder network with skip connections has also been applied for many generative tasks such as image-to-image translation [16]. The model implemented in this report consists of eight down-sampling layers with 4×4 filters, instance normalization and leaky ReLU activation functions. The up-sampling is done by 8 transpose convolutional layers with 4×4 filter size, instance normalization and ReLU activation. The output of the i:th up-sampling layer is concatenated with the output of the i:th down-sampling layer, hence forming the U-Net structure. A summary of the generator architecture is provided in TABLE I.

b. Improved U-Net In addition to the standard U-Net implementation, a slightly different implementation was made by changing the concatenations featured in U-Net to instead adding the two contributions.

c. Residual network generator ResNet architectures have been widely applied for CycleGANs [17], [18]. For the experiments in this report, the ResNet architecture is based on the initial CycleGAN paper [5], but is altered slightly. The architecture involves three convolutions, 9 residual blocks, two transpose convolutions with stride 2 and lastly a convolutional layer generating the RGB channels from the features. The layers before the resiudal blocks have padding set to same, and the layers after the residual blocks have valid padding. The residual block first pads the edges with zeros and then performs a 3×3 2D convolutional operation followed by instance normalization. Padding, convolution and normalization are repeated once more, and finally the output is added with the input to the residual block. The architecture is summarized in TABLE II.

2. Discriminator

The discriminator is inspired by the original CycleGAN paper [5] and the pix2pix paper [19]. The discriminator down-samples the images with 5 consecutive convolutional layers to 30×30 patches. This allows to classify/critique the images bases on a down-sampled patch, while keeping the number of parameters lower and allowing different image sizes. The discriminator architecture is summarized in TABLE III.

TABLE I. The sequential layers of the U-Net generator. (L)ReLu stands for (leaky) rectified linear unit activation function and Conv2D is a 2 dimensional convolutional layer. InstanceNorm refers to instance normalization. U-Net Conv2D refers to the skip connection coupled with 2 dimensional convolution.

	U-Net Generator
1.	4×4 Conv2D with 64 filters, stride 2, LReLU
2.	4×4 U-Net Conv2D with 128 filters, stride 2, InstanceNorm., LReLU
3.	4×4 U-Net Conv2D with 256 filters, stride 2, InstanceNorm., LReLU
4.-8.	4×4 U-Net Conv2D with 512 filters, stride 2, InstanceNorm., LReLU
9.-11.	4×4 U-Net Conv2DTranspose with 512 filters, stride 2, InstanceNorm. and dropout rate 0.5 followed by ReLU
12.	4×4 U-Net Conv2DTranspose with 512 filters, stride 2, InstanceNorm., ReLU
13.	4×4 U-Net Conv2DTranspose with 256 filters, stride 2, InstanceNorm., ReLU
14.	4×4 U-Net Conv2DTranspose with 128 filters, stride 2, InstanceNorm., ReLU
15.	4×4 U-Net Conv2DTranspose with 64 filters, stride 2, InstanceNorm., ReLU
16.	4×4 Conv2DTranspose with 3 filters, stride 2 and tanh activation

TABLE II. The residual network generator uses 9 residual blocks with filter size 3×3 and a variety of convolutional layers with leaky ReLU, instance normalization and up- and down-sampling strides. (L)ReLu stands for (leaky) rectified linear unit activation function; Conv2D is a 2 dimensional convolutional layer; InstanceNorm refers to instance normalization.

	Residual Generator
1.	7×7 Conv2D with 32 filters, stride 2, LReLU
2.	5×5 Conv2D with 64 filters, stride 2, InstanceNorm., LReLU
3.	3×3 Conv2D with 128 filters, stride 1, InstanceNorm., LReLU
4.	9 Residual blocks
5.	3×3 Conv2DTranspose with 64 filters, stride 2, InstanceNorm., LReLU
6.	5×5 Conv2DTranspose with 64 filters, stride 2, InstanceNorm., LReLU
7.	6×6 Conv2D with 3 filters, stride 1, tanh activation

C. Data and painters

This work is based on the datasets “I'm something of an painter myself” and “Best artworks of all time” provided by Kaggle [20], [21]. The firstmost dataset provided 7038 photos to train on. Painters were selected from the latter dataset with the following criteria:

1. At least 50 artworks

2. Degree of correspondence to the photos

TABLE III. The discriminator network down-samples the images to 30×30. (L)ReLu stands for (leaky) rectified linear unit activation function; Conv2D is a 2 dimensional convolutional layer; InstanceNorm refers to instance normalization.

	Discriminator
1.	4×4 Conv2D with 64 filters, stride 2, LRelu layer
2.	4×4 Conv2D with 128 filters, stride 2, InstanceNorm., LRelu
3.	4×4 Conv2D with 256 filters, stride 2, InstanceNorm., LRelu
4.	4×4 Conv2D with 512 filters, stride 1, InstanceNorm., LRelu
5.	3×3 Conv2D with 1 filter, stride 1

3. Distinct style

Consequently the painters Salvador Dalí, Albrecht Dürer, Vincent van Gogh, Wassily Kandinsky, Claude Monet, Edvard Munch and Andy Warhol were selected to cover a large variety of styles to compare, see FIG. 1.

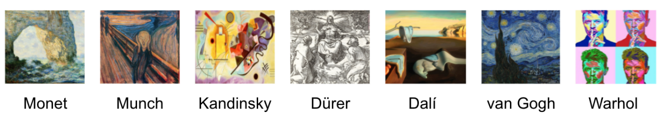

FIG. 1. Example artworks by each painter.

Our hypothesis is that bigger domain changes will lead to more complex and harder to learn mappings. Based on our subjective qualitative assessment we have ranked the painters across three criteria from least to most realistic, see TABLE IV. Correspondence refers to the general correspondence between the objects in real world settings and paintings. High numbers mean highly abstract art. Furthermore, color realism refers to natural color distributions. Lastly, consistency in the painter domain is ranked.

D. Experiment setup

To evaluate the models on different painters, two different experiments were conducted per painter. Both the ResNet and U-Net generator architectures were trained individually on each painter. The training was conducted with the Kaggle TPU cluster. The CycleGANs were trained with random training pairs of about 70 paintings and 7000 images. Training data was augmented using a random series of cropping, rotation and horizontal and vertical reflections. The augmentation was done the same way for the photo set as for the current painter set. The augmentation was in practice executed by creating an infinite TensorFlow dataset, meaning that each epoch in general sees different training data. Training was done with a batch size of 4 with λ set to 10. Training was conducted during 400 epochs for U-Net and 600 epochs

TABLE IV. Ranking of painters across correspondence in scenery, color realism and consistency. Lower numbers mean higher ranking. Note that this qualitative assessment is highly subjective.

Painter	Correspondence	Color Realism	Consistency
Salvador Dalí	3	2	5
Albrecht Dürer	6	7	1
Vincent van Gogh	4	4	6
Vasiliy Kandinskiy	7	5	4
Claude Monet	1	1	2
Edvard Munch	2	3	3
Andy Warhol	5	6	7

for ResNet induced by its slower convergence. The used optimizer was Adam with a learning rate of $2 \cdot 10^{-4}$.

III. RESULTS

In this section, the quantitative and qualitative results are presented. In addition, the results are briefly explained, to be expanded upon in the discussion session.

A. Quantitative results

Here, the quantitative results are presented in terms of the different losses. All losses are also featured with means μ over all epochs and the legend order is given according to the summed painter ranks in TABLE IV.

1. Generator and discriminator losses

With both used generator architectures, a hierarchy of generator loss magnitudes is visible. This hierarchy seems to correspond moderately well with the degree of realism in the painter styles. The losses were smoothed with Savitzky-Golay filtering with a window size of 75 and order 1 to reduce noise. See FIG. 2 for a collection of the losses attained with the ResNet architecture. When comparing ResNet with U-Net in FIG. 3, it is evident that the hierarchy is almost the same for both architectures. However, one clear difference is that Dürer gets radically higher loss than all other painters for U-Net. In contrast to the other painters, Dürer's art is almost exclusively colorless engravings. The improved U-Net was able to lower the loss considerably for Dürer and lowered losses for Kandinsky and Warhol slightly, which is shown in FIG. 4. In addition to painter generator losses, see

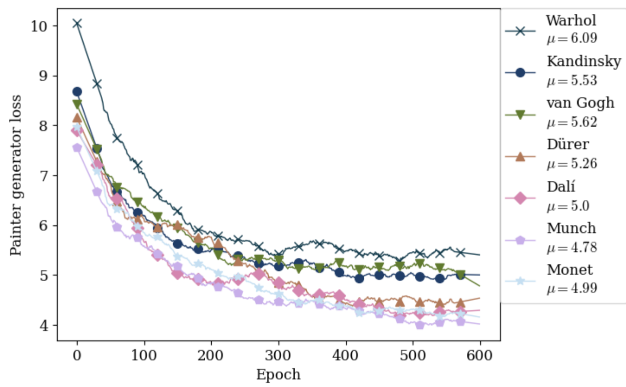

FIG. 2. The plot shows the smoothed art generator training loss with the ResNet architecture for each of the painters. Monet, Munch and Dalí attained the lowest generator losses. van Gogh, Dürer and Kandinsky have higher losses, and Warhol has the highest.

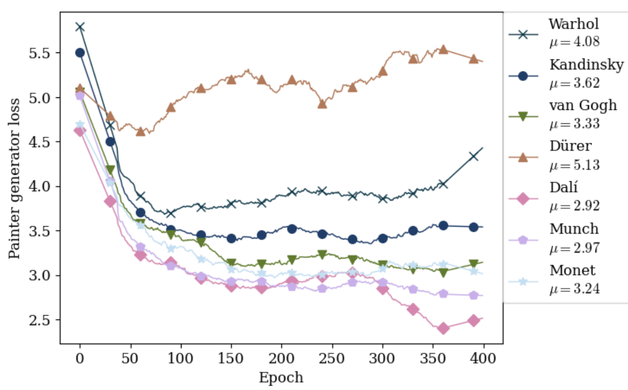

FIG. 3. The plot shows the smoothed art generator training loss with the U-Net architecture for each of the painters. Monet, Munch and Dalí attained the lowest generator losses, followed tightly by van Gogh. Kandinsky and Warhol have notably higher losses. Dürer has radically higher loss than all other painters.

FIG. 1 & 2 in Supplementary material for discriminator losses. In both cases, a tendency to negative correlation between discriminator and generator loss is seen.

2. *Image reconstruction - cycle loss*

One part of the generator loss is the cycle loss. Cycle losses are presented in the Supplementary material as FIG. 4-6. The presented cycle loss makes up approximately half of the total generator loss for both architectures. Furthermore, the same hierarchy as in the generator loss across painterss is seen in the cycle loss.

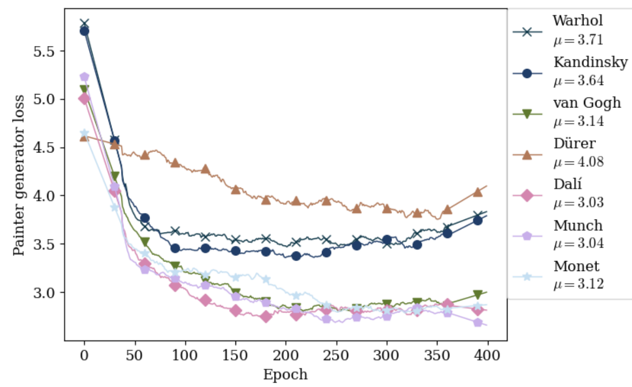

FIG. 4. The plot shows the smoothed art generator training loss with an altered U-Net architecture for each of the painters. Monet, Munch and Dalí attained the lowest generator losses, followed tightly by van Gogh. Kandinsky and Warhol have higher losses, but are slightly lower than for the standard U-Net. The loss for Dürer is lowered significantly compared with the standard U-Net.

B. Qualitative results

Generated image quality can only approximately be quantitatively assessed. Hence, to supplement the quantitative assessment, a qualitative evaluation is conducted in this section. Both generated paintings for unseen test photos and cycled test photos are evaluated. Lastly, on a more anecdotal side we show that our generated artworks are able to fool Google.

1. *Generated artworks*

FIG. 5 shows a collage of some generated artworks for all painters using the ResNet architecture. In FIG. 6 all U-Net generated images are depicted, and in FIG. 7 the improved U-net results are shown. Both architectures appear able to generate artworks with different styles from unseen photos. While the ResNet generator is able to produce transformations with stronger color changes, the U-Net generated images appear to be in closer correspondence to the photo. Furthermore, the U-Net artworks suffer more from artifacts. The improved U-Net still suffers from these artifacts, but gets better image quality for Dürer and appears to be slightly better for other painters as well. The color changes also appear to be more vivid. For more examples, see FIG. 13-16 in Supplementary material.

2. *Cycled photographs*

Cycled photographs are seen in Supplementary material, FIG. 7-9. Residual effects of the art transformation can be seen for ResNet. This is not as clear for U-Net, except for Dürer. Some improvement for the Dürer cycle can be

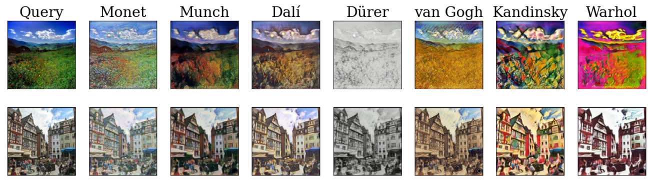

FIG. 5. ResNet-CycleGAN generated artwork for test photos by painter. ResNet captures some important features from the painter style, and does not produce many artifacts.

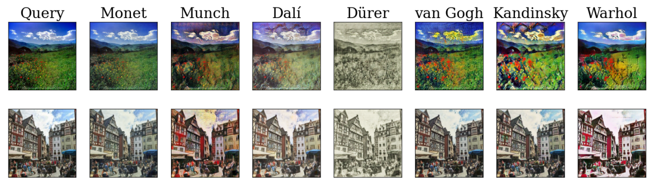

FIG. 6. U-Net-CycleGAN generated artwork for test photos by painter. U-Net produces visible artifacts in some cases, and does not give as intense transformations as ResNet.

seen for the improved U-Net, such as better sky color and less yellow tinge to the cityscape.

3. Reverse Google image search

Using one of the arguably best image classifier - the reverse Google image search - many of the generated artworks are shown to be considered fine art and very similar to the target painter domain, as Google provides art of the painter as similar images. Two examples can be seen in FIG. 10 & 11 in the Supplementary material.

IV. DISCUSSION

In this section, the brief analysis in the results section is elaborated upon. Initially, the complexity of evaluating artwork work quality is discussed. Then the assessment of image domain distances is expanded upon. Lastly, U-Net and ResNet are compared.

A. Artwork quality

Art is innately subjective. Its quality can only hardly be qualitatively or quantitatively appraised. The quantitative results have shown that both architectures are able to learn domain mappings for all painter. The generator losses show a distinct magnitude hierarchy, which loosely corresponds to painter realism. Further, ResNet has proven to be more stable in learning painter domain mappings than U-Net's. However, the quantitative quality evaluation of generated artwork is inherently bounded by the validity of used evaluation metrics. This is especially pressing as GANs generally lack an

FIG. 7. Improved U-Net-CycleGAN generated artwork for test photos by painter. Color changes are more extreme than for U-Net, and especially the Dürer art appears more gray than the slightly yellow tinge for U-Net.

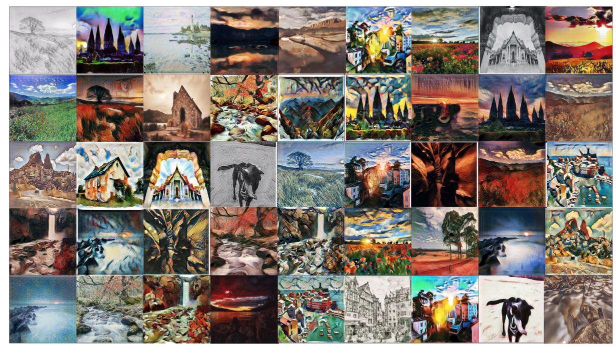

FIG. 8. Collage with an excerpt of some of the most expressive images generated by the CycleGANs. Images were selected from all painters and shows the many different settings which the CycleGAN architectures can handle. Some of the transformations were more successful than others. For example, no trained networks were found to handle the dog image particularly well (row 5 and column 8 or row 3 and column 4).

objective function to evaluate quality. Generator and discriminator losses can only be taken as indications. They strongly depend on the adversarial process.

Hence, a qualitative assessment was further conducted. Given the little training data and complex task of unpaired image-to-image translation, both architectures were able to produce convincing and subjectively good quality artworks. The ResNet architecture produced more extreme transformations, while producing less artifacts. Even though less extreme domain changes are achieving lower losses, strong changes produce very artistic and impressive artworks. Still, the qualitative assessment lacks objectivity.

In the absence of an completely objective quality assessment, we have branched out trying to find other evaluation processes. Empirically grading the artwork by asking strangers would prove to be more objective than the qualitative assessment of the two authors, but still lacks validity given the limited art expertise of everyday humans. Hard to attain experts ratings could provide validity, while suffering from more statistical uncertainty. In this experiment an arguably objective expert – Google reverse image search - was used to show that the generated images are not just considered art, but also similar to other artworks of the same painters.

B. Image domain distance

The results suggest that loss magnitude is correlated to larger image domain distribution changes. The distribution correspondence was ranked subjectively, but nonetheless appears to coincide with the results. Larger distribution changes may give rise to more artifacts. The subjective nature of the quality assessment leads to a personal preference of some of the strong domain changes with higher losses. Therefore, it is difficult to conclusively validate the hypothesis that stronger domain changes are harder to learn. Part of why the more extreme distribution changes get high loss is also that they generally appear more difficult to cycle between, but do not necessarily pose much greater difficulties for generating artworks. Because of the unsupervised nature of the CycleGAN, generated artworks are difficult to compare directly with true artworks.

To supplement the subjective assessment of domain distribution changes, histograms were produced for each painter dataset representing distribution of mean color changes in the dataset, as well as color changes relative to the photo dataset. These histograms are included in Supplementary materials, FIG. 12. The high statistics (mean and standard deviation) for Warhol and the low statistics for Monet and Munch could partially explain why the painters are on the opposite sides of the loss spectrum. Glancing back at TABLE IV, this is in accordance with the visual rankings, as Munch and Monet have good scores in all three categories in contrast to Warhol. Dürer also has a very high color distance to the photo dataset, which could explain why U-Net handles the painter poorly. However, this does not explain why Dalí, who obtained intermediate ranking and statistics, is among the painters with lowest loss. The color statistics are not sufficient in order to give the whole truth, nor is the subjective assessment.

C. Comparison of ResNet and U-Net

An obvious difference between ResNet and U-Net was that U-Net handled the colorless dataset by Dürer very poorly. The current architectures differ in latent dimension. ResNet has a latent dimension of 64×64, whereas U-Net has a meek latent dimension of 1. In an attempt to find out if this was a contributing factor for U-Net's performance on the Dürer dataset, the latent dimension of U-Net was increased to 64×64 by changing most strides to 1 instead of 2 in the up- and down-sampling layers. The change did not improve performance notably. Adding two more layers was also tested, and so was decreasing the maximal number of filters from 512 to 128, which is the same as in ResNet. This did not improve performance either. Furthermore, removing all skip connections altogether gave considerably worse results.

The improved U-Net trained on Dürer achieved a loss of 3.5-4.0 as opposed to the previous 5.0-5.5. The networks for the other painters achieved about the same loss as before. The Dürer network thus attained almost as low loss as the ones trained on Warhol or Kandinsky. Qualitatively, results were similar for most painters, and featured similar artifacts as before. The results for Dürer improved considerably, however, and some visual improvements were seen for Kandinsky and Warhol.

U-Net's poor performance on Dürer thereby appears to be partially due to the concatenations. One possible reason for this might be that much of the early layer information gets added to latter layers when doing the concatenations. There might thereby be a risk of too much emphasis being put on these early representations, which have not been processed to great extent and may restrict output to being closer to input. As result, representations may not be convincing, and adversarial training is inhibited. This is also suggested by the results in FIG. 6, which shows much milder transformations than for ResNet in FIG. 5. The adding operations might give softer contributions compared with concatenations.

The empirical difference that U-Net produces more artifacts may also be related to the multiple layers skipped in the U-Net. In ResNet, skip connections are only established at the encoded stage. The merge of an early representation with a deeply encoded representation in U-Net might therefore give rise to unwanted artifacts for more extreme distribution changes, since one part is heavily processed, and the other is only lightly processed. In addition, U-Net is generally used for applications where small changes should occur, such as segmentation masks. For extreme distribution changes, it might be better to instead deconstruct the signal, process it and reconstruct it.

The ResNet architecture has 2,974,851 parameters, and U-Net has 54,414,979 parameters. Still, U-Net appears to converge faster. When the number of filters were decreased for U-Net, the number of parameters was instead 3,417,475, which is quite close to ResNet.

V. CONCLUSION

This report has shown that CycleGANs are capable of producing art similar to many different art styles. The hypothesis that art which is more similar to reality is easier to reproduce seems to coincide with network losses quite well. However, even if abstract art may lead to higher loss, it may look convincing to the human eye as well as other more objective metrics. For the purpose of generating convincing artworks, ResNet generally appears preferable to U-Net, even if the architecture is prone to attaining higher loss. Furthermore, a potential improvement to U-Net for art generation has been developed, which may work better for extreme distribution changes.

VI. FUTURE WORK

CycleGANs are unstable in training. Since the introduction of GANs and CycleGANs, there have been attempts to improve their performance by use of different loss functions and training schemes, such as the Wasserstein loss and gradient penalties [11],[22]. Even though, initial experiments incorporating a Wassertein loss and gradient penalties have not yielded improvements they could provide possibilities for improvements. Extensive hyperparameter tuning could additionally further the generator quality. An open area of research has proven to be the development of an objective quality criteria to rate generated artworks. Lastly, other architectures and set-ups could be explored, as the cycle-consistency is restricting possible transformations to style and not generating or altering shapes to a greater extent.

VII. CONTRIBUTION STATEMENT

Both participants have contributed equally in all domains, from coding to writing the report and developing the presentation.

[1] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, Generative adversarial networks (2014), arXiv:1406.2661 [stat.ML].

[2] T. Karras, S. Laine, and T. Aila, A style-based generator architecture for generative adversarial networks (2019), arXiv:1812.04948 [cs.NE].

[3] C. Zheng, T.-J. Cham, and J. Cai, Pluralistic image completion, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[4] M. Chu, Y. Xie, J. Mayer, L. Leal-Taixe, and N. Thuerey, Learning Temporal Coherence via Self-Supervision for GAN-based Video Generation (TecoGAN), ACM Transactions on Graphics (TOG) **39** (2020).

[5] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks (2020), arXiv:1703.10593 [cs.CV].

[6] A. Almahairi, S. Rajeshwar, A. Sordoni, P. Bachman, and A. Courville, Augmented CycleGAN: Learning many-to-many mappings from unpaired data, in *Proceedings of the 35th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 80, edited by J. Dy and A. Krause (PMLR, 2018) pp. 195–204.

[7] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, Cyclegan-vc2: Improved cyclegan-based non-parallel voice conversion, in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2019) pp. 6820–6824.

[8] L. A. Gatys, A. S. Ecker, and M. Bethge, A neural algorithm of artistic style (2015), arXiv:1508.06576 [cs.CV].

[9] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition (2015), arXiv:1512.03385 [cs.CV].

[10] O. Ronneberger, P. Fischer, and T. Brox, U-net: Convolutional networks for biomedical image segmentation (2015), arXiv:1505.04597 [cs.CV].

[11] J. Cao, L. Mo, Y. Zhang, K. Jia, C. Shen, and M. Tan, Multi-marginal wasserstein gan (2019), arXiv:1911.00888 [cs.LG].

[12] F. Zhan and C. Zhang, Spatial-aware gan for unsupervised person re- identification, in *2020 25th International Conference on Pattern Recognition (ICPR)* (2021) pp. 6889–6896.

[13] F. Zhan, H. Zhu, and S. Lu, Spatial fusion gan for image synthesis, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[14] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, Image-to-image translation with conditional adversarial networks, CoRR **abs/1611.07004** (2016), arXiv:1611.07004.

[15] O. Ronneberger, P. Fischer, and T. Brox, U-net: Convolutional networks for biomedical image segmentation, (2015), arXiv:1505.04597 [cs.CV].

[16] L. Zhang, Y. Ji, X. Lin, and C. Liu, Style transfer for anime sketches with enhanced residual u-net and auxiliary classifier gan, in *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)* (2017) pp. 506–511.

[17] B. Chang, Q. Zhang, S. Pan, and L. Meng, Generating handwritten chinese characters using cyclegan, in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)* (2018) pp. 199–207.

[18] A. Almahairi, S. Rajeshwar, A. Sordoni, P. Bachman, and A. Courville, Augmented CycleGAN: Learning many-to-many mappings from unpaired data, in *Proceedings of the 35th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 80, edited by J. Dy and A. Krause (PMLR, 2018) pp. 195–204.

[19] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, Image-to-image translation with conditional adversarial networks, CVPR (2017).

[20] Kaggle, I'm Something of a Painter Myself: Use GANs to create art - will you be the next Monet? (2020).

[21] Kaggle, Best Artworks of All Time – Collection of Paintings of the 50 Most Influential Artists of All Time (2019).

[22] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, Improved training of wasserstein gans (2017), arXiv:1704.00028 [cs.LG].