GAN-ART

MAJOR TECHNICAL PROJECT (DP 401P)

to be submitted by

AJ R LADDHA(B16004)

KAUSTUBH VERMA(B16021)

for the

**END-SEMESTER
EVALUATION**

under the supervision of

Dr. ARNAV BHAVSAR

SCHOOL OF COMPUTING AND ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY MANDI

KAMAND-175005, INDIA

July, 2020

MTP 2019-20 Project Form
School of Computing and Electrical Engineering

Name: ....Kaustubh Verma., Aj R Laddha................................

Roll No.: ....B16021 , B16004...

Branch: ....CSE...

Project Title: ....GAN-ART..

Supervisor Name: ....Dr. ARNAV BHAVSAR...

Objectives (mention in bullet points)

Original	**Revised**
• Artistic image generation based on multiple genre and style at good resolution.	• Artisitic image generation based on single genre and multiple styles at reasonable resolution. • Use some method of Super Resolution.

Signature of Supervisor

Contents

Abstract

Ever since the rise of artificial intelligence and its strong future as an unstoppable force in human life has been stated,questions started floating around among scientists of machines' ability to generate human level creative outcomes such as jokes,music, art,etc. The ability of machine to be creative is vital to show that AI is in fact intelligent.

Art by the definition can be described as - "*The expression or application of human creative skill and imagination, typically in a visual form such as painting or sculpture, producing works to be appreciated primarily for their beauty or emotional power.".*[8] The definition which in itself has terms such as expression,imagination,creative skills,etc. have strong link to emotional intelligence and is widely regarded by the critics of AI which machines simply doesn't do, and will never be able.

To tackle the ability to be creative by computers, different algorithms have been proposed to explore the creative space. Many approaches have used an evolutionary process, generating candidates and evaluating them using a fitness function,improving on each iteration. Others have used interactive systems, where the AI explores the creative space, while humans plays the role of an observer giving back his feedback to drive the process.

An important component of art-generating algorithms is to relate their creative process of generating art to the process adopted by humans to do the same over their lifetime. A human for its creative process utilizes prior experience of and exposure to art .A theory is therefore needed to model how to integrate exposure to art over a time to creation of art. Colin Martindale(a professor at the University of Maine) who studied creativity and artistic processes for around 35 years proposed a theory explaining new art creation. He suggested that at any point in time, *"Creative artists try to increase the arousal potential of their art. However, this increase has to be minimal to avoid negative reaction by the observers also called as principle of least effort".*[9] He also suggested that style break and thereby generation of new art happens when an artist tries to increase the arousal potential. Above ideology is an important way to look forward the task of creative artwork.

Deep neural networks have excelled in advancing AI across various domains. The ability of Deep neural nets in generating novel images has been heavily guided by Generative Adversarial Networks (GAN). GAN's ability to take a foothold in generative field has been prominent over the years and with upcoming modifications which have helped in stabilizing training of GANs, they are powerful than ever before.

Initial Research

1.1 Background and Literature Survey

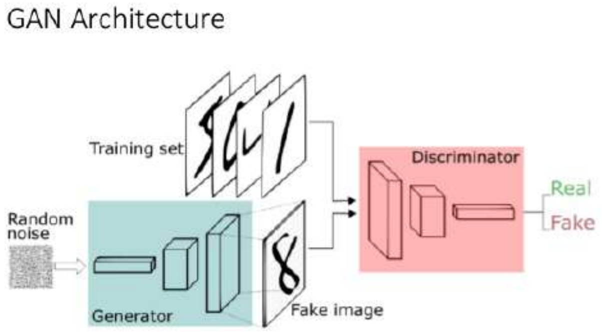
Introduction of GAN to solve the problem of generating artwork has been fairly recent with architectures and techniques that have enhanced and stabilized GAN training. Most of these work have relied on following datasets- Wikiart[10] dataset , Painter By Number (PBN) dataset, Behance Artistic media (BAM) dataset. Here we look at brief introduction to GAN's followed by some of the work done for generating art using adversarial networks.

1.1.1 GAN (Generative adversarial network)[1]

A generative adversarial network is a deep learning model in which two neural networks compete against each other to generate accurate predictions. GANs was published in a paper by Ian Goodfellow and at the University of Montreal,2014.

GANs are an exciting and rapidly emerging field having the ability to generate realistic images and data that can span across many domains, most notably in image-to-image translation tasks such as and in generating photo realistic photos of objects, scenes and going as far as generating fake images of human faces.

GAN composes of two deep networks, the generator, and the discriminator.



As can be seen in the above diagram the basic structure of GAN consists of two networks,where a Generator generates images from random noise data(fake image),while the discriminator tries to discriminate the fake image from real image in training dataset. Then the same discriminator will provide feedback to the generator to create images that look like the real.

Following equation guides GAN training -

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

Therefore GAN can be described as a mini-max game in which Generator wants to minimize V while Discriminator wants to maximize it.

Here we are using this GAN's ability to generate novel artwork based on different approaches guided by works mentioned below.

1.1.2 Pix2Pix[2]

The pix2pix model also referred as Image to Image translation using Conditional GANs. It has two neural networks as in all GANs that is Generator and Discriminator.

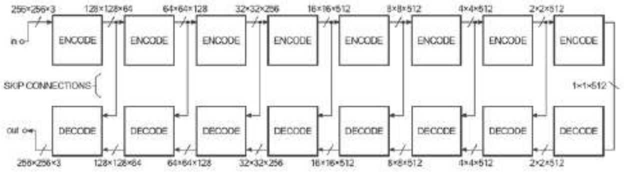


Figure 1.1: PIX2PIX Generator architecture

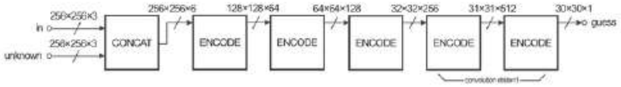


Figure 1.2: PIX2PIX Discriminator architecture

The generator has an encoder-decoder type of structure but with modification of skip connections(Direct connections b/w Encoder and Decoder Layers). This architecture is known as U-Net Architecture. It first downsample/compress the image into low dimensional vector and again upsample it into original dimension. The discriminator is PatchGAN Discriminator which classifies patches of image instead of whole image as real or fake.

The objective/loss function is combination of L1 loss and Conditional GAN lose.

1.1.3 MSG-GAN(Multi-Scale Gradients for Generative Adversarial Networks)[3]

Multi Scale Gradients[3] GAN proposes a new simple but effective architecture that deals with the issue of adapting GAN architectures to various different data-set due to susceptibility to hyper-parameters and instability in training. A large part of the reason of this instability in GAN training occurs when there isn't enough overlap between real and fake distribution, whereby the gradient flow from discriminator to generator through multiple intermediate layers becomes highly uninformative.

MSG-GAN addresses this by allowing the flow of gradients from discriminator to generator at multiple intermediate scales. Similar to highly popular Progressive GAN, this ability allows MSG-GAN to generate high resolution image on various datasets but with similar sets of hyper-parameters and better training time. Being a simple architecture that has a single generator and a discriminator gives a further advantage of having total less parameters(not having discriminator at each resolution) and no need to color regularization term at multiple scales as in StackGAN.

MSG-GAN framework builds upon two basic architecture of ProGAN and Style-GAN. These two architecture are therefore called as MSG-ProGAN and MSG-StyleGAN. There is no progressive growing in either of them and therefore ProGAN without any progressive growing is just DCGAN. While MSG-StyleGAN uses certain modification of StyleGAN on vanilla DCGAN.

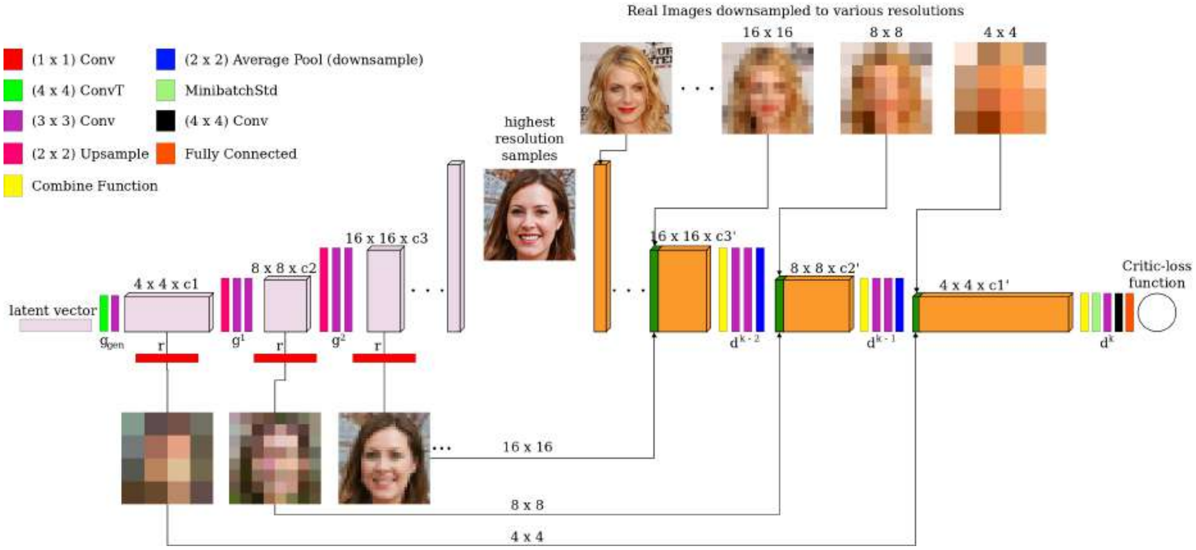


Figure 1.3: MSG-GAN architecture[3]

As can be seen in Fig1.5, the architecture includes connection from multiple layers of generator to the corresponding layers in the discriminator. At intermediate generator layers, a 1x1 convolution converts the activation volume to images which are then concatenated to the corresponding activation's in the discriminator coming from the main convolution path.

On the discriminator side, the loss function is now not just the function of final output of highest resolution image of the generator, but also the function of outputs from the intermediate layers. WGAN-GP and Non saturating GAN loss with 1-sided GP are two loss functions of the critic that are experimented with. Also the gradient penalty is taken as the average over each intermediate layer.

1.1.4 PixelBrush: Art Generation from text with GANs[4]

PixelBrush[4] is a tool proposed by Stanford University(Jiale Zhi) which can generate art from text descriptions using generative adversarial networks. PixelBrush's main contributions are -

- Using text as a conditional input to GAN and generating painting images that are consistent and enhanced in details due to added text information.
- Using different generators for same discriminator and how that affects image quality.
- PixelBrush also provides another viewpoint to look at how generated image's entropy varies with the entropy of real image.

The basic model of PixelBrush is based on Conditional Generative Adversarial Network, which is a variant of GAN that accepts additional information y(here in form of text), so that both generator G and discriminator D are conditioned and trained in a better manner on that condition.

To use text as condition input to conditional GANs,text is first trained and converted to text embedding vector using skip-through vectors.The Skip-Thoughts model is a sentence encoder which encodes input sentences into a fixed-dimensional vector. PixelBrush used recurrent neural network where encoder is GRU,while the decoder is a conditional GRU for training skip thoughts. A trained Skip-Thoughts model will encode similar sentences nearby each other in the embedding vector space. Overall the architecture uses skip-thought text embedding network to convert text into embedding vectors and then feeding these text embedding with some random noise z to generator network that generates artwork which have been conditioned on the text provided. These generated images are then feeded to the discriminator network together with their text embedding(again

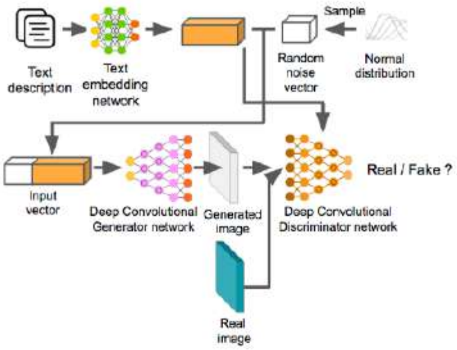


Figure 1.4: PixelBrush Architecture[4]

as a conditional input).Discriminator then differentiates between the real and fake image.
Oxford paintings dataset which contains 10 categories of images such as birds,chair,etc. was used for PixelBrush. Oxford painting dataset contains only short painting titles which don't capture the whole scene on the image. Therefore to provide descriptive captions for the images a two-step solution was adopted by the authors where image captioning network Neuraltalk2 was used generate captions for all the painting images.These captions were then improved by humans to generate better descriptions for the images.
To evaluate the results, humans where asked to tell the difference between the artwork generated from GAN and real artwork, as well as using a image classifier to verify the same. The ability to use text as a condition in PixelBrush's model makes it a great way to add theme information which we can use in our project to generate art according to a theme.

1.1.5 GAN-Gogh[5]

GAN-Gogh is model made by researchers at Williams College.[?] It is a model which uses basic GAN to generate novel art. They use the knowledge of deep network so that it was capable of both learning the style-distribution and content components from different segments of art, and also able to combine such components to generate new art. The novel content formation is very difficult than just emulating the style from one art to the content of other art which was done in most of the earlier researches.
The dataset used by them is Wikiart data with more than a million paintings labeled on basis of Styles, Genres, Artists and Year of making that painting.

Talking about architecture the authors have initially form discriminator with generator on the basis of earlier Wasserstein[11] model. The concept of Wasserstein metric[11] was also enforced via a penalizing term in the cost function for discriminator which in turn tries to minimize the cross-entropy while predicting genre v/s the real labelled genre for a painting, and they also added a penalizing term on the generator which will tries to minimize the cross-entropy that the discriminator predicted v/s the genre that was formed using the conditioning vector.
They even added concept of Global conditioning[5] on GAN which was new on that period of time. Global conditioning will perform an activation function between different layers in deep-network model, than they had used the condition vector which influences the activation to occur, and this condition vector is used separately on each layers using layer by layer basis. Global conditioning highly helps generator so that it can better differentiate between its produced content on the basis of that condition vector.
Their results from GAN-gogh model are comparable to best results obtained in terms of artwork generation.

1.1.6 Neural Algorithm of Artistic Style(N.A.A.S)[6]

Neural Algorithm of Artistic Style[?] proposed by Leon A. Gatys. The model uses two images to create an output image : A Content Image(On which style will be transferred) and A Style Image(Style to Transfer).

The model use the Convolutional Neural Networks(CNNs) mostly used in image processing task. The author uses the VGG-19 model trained on object recognition as it hierarchically learns the object information. It helps the model in learning the content of the image. The image can be re-visualised by regenerating image from feature maps.The higher layers of Feature responses(Feature Maps) in the network are used as the content representation. The higher layers does not contain the smaller level pixel value yet it contains the major content of the image.

The style of the image is obtained from the feature space placed on the responses by filter at each layer. The method use correlation between filter responses at different layers. For an image a style is represented by color and textures and content is shapes in that image. If the value of correlation matrix is high that means the image has lot of style and color. One of the most important finding through this paper is that the style representation and the content representation in CNNs are separable. We can tweak both the representations independently so as to produce new images.

In the process of style transfer,model is not exactly being trained. VGG-19 model is only used to minimize two defined loss. The input image is initialized by random noice and then passed through different layers of VGG-19. The losses that are computed named as Style loss and Content Loss which will ensure that the output image will contain both the style and content.

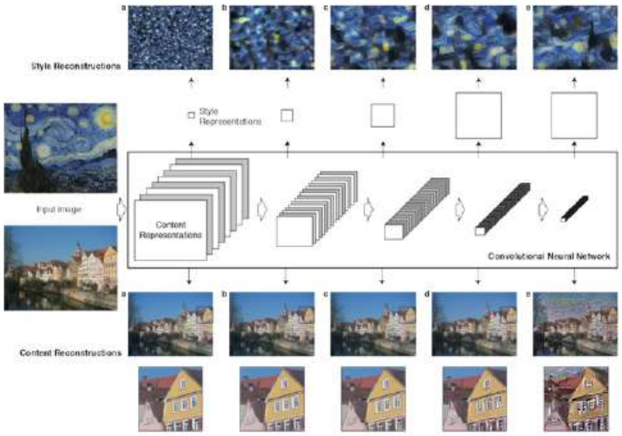


Figure 1.5: N.A.A.S architecture[6]

The content loss is calculated as the input image(random noice) and content image are passed through various layers and the euclidean distance is calculated in between all the images generated after each layers.

For calculating style loss the Gram Matrices of the images are compared. The calculation of gram matrix is done by multiplying a matrix and its transpose. While finding gram matrix we multiply every column with every row it will give us non localized information like texture, weights etc. The euclidean distance is calculated from respective pair of Gram Matrix at each layer. Both the losses are given weights so that we can tweak the style and content in the output image.

For preserving color in the image we use Luminance-only-transfer.[12] It performs style transfer only in the luminance channel. First the algorithm will get the luminance channel from both style and content image and the NAAS is applied on these channels. Then the output luminance image mixed with YIQ color model to get the final image.

1.2 Objective and scope of the Work

The objective of our work is to do a thorough study and application of previous methods of generating novel art and come up with a novel approach based on GAN that can either improve upon previous artworks or combine their methodologies to better script the process of generating art. Two major pillars of our artwork are genre and art,whereby we look to generate artwork for a particular genre and multiple styles. We aim to have a final methodology that is capable enough to generating artworks that could be used across multiple genres and styles with the availability of required datasets.

However it must be noted that the evaluation of our work is largely subjective rather than objective. Same artwork can have different evaluation by different individuals. Also creative art can vary from being very specific to very abstract, therefore we would like our generated artwork to fall somewhere in the middle of both extremes.

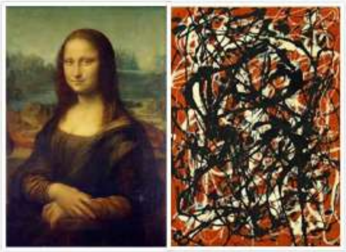


Figure 1.6: Extremes of realistic vs abstract art

As mentioned in abstract, generating novel work is heavily dependent on previously observed artwork(dataset), even in the case of humans. Therefore scope of work is also heavily lead by available dataset and their corresponding artists. Wikiart is the most prominent and heavily used artistic dataset available. In accordance with the resources available to us and the dataset we could gather, we restrict our approach to generating artwork for the genre ”Landscape”, and then transferring multiple styles to wrap up of artistic image.

Work done - Previous Semester

2.1 Brief Approach

Previously, we worked on designing a model architecture based on Conditional GAN that uses style and genre as conditions for our network to generate desired output of a particular genre and style.

Dataset: We have used Wikiart dataset[10] for our task. Wikiart dataset is a collection of thousands of visual artworks maintained by wiki.org. The artworks are partitioned well into the style, genre and artists. Most of the artwork available on wikiart is of particular artist,genre and style. We have maintained with ourself a dataset of around 27 prominent style and 10 genre. But because of large discrepancy in the number of images available for each style and genre, we have selected 5 style and genre for training our model which are distinct with human eye and have considerable number of images. Even though we picked small number of styles and genre, still there were lots of difference in number of images and many artwork in the distinct style and genre looked similar,which impacted our results.

2.2 Sample Images from Wikiart

Style These are the style we have used for training and their corresponding sample images :

1. Cubism
2. Expressionism
3. Impressionism
4. Realism
5. Symbolism

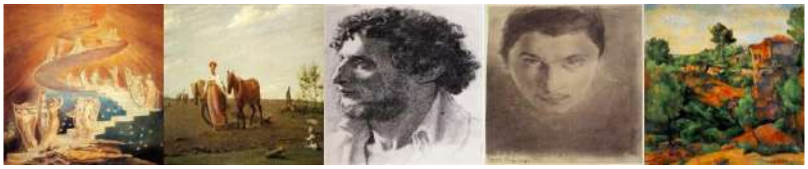


Figure 2.7: Symbolism, Realism, Impressionism, Expressionism, Cubism

Genre

These are the genres we have used for training and their corresponding sample images :

1. CityScape
2. Landscape
3. Portrait
4. Still life

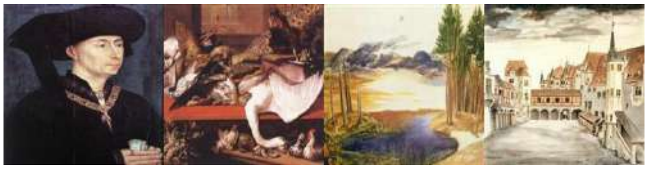


Figure 2.8: Portrait, Still-Life, Landscape, CityScape

Initially we started with the DCGAN[7] architecture and made it conditional and trained on style and genre to produce the required results but the results were not up to the mark and we found that even on further training the results were not improving and Generator was unable to improve the results. Following are the results we obtained when training conditional DCGAN[7] on style and genre. Each of the blocks from the below image belongs to a particular class of style or genre.

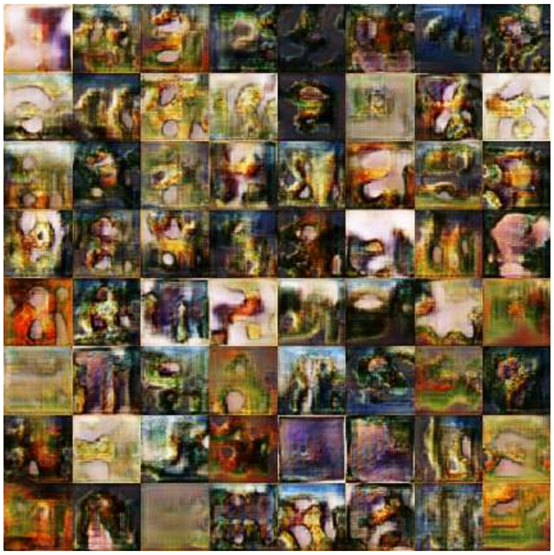


Figure 2.9: Sample Image obtained when trained on Style.

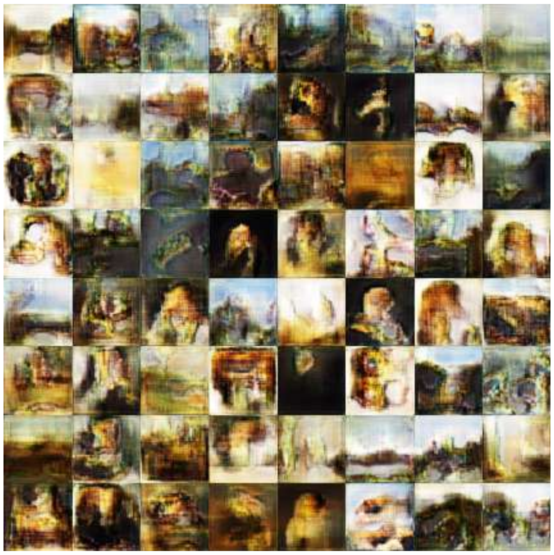


Figure 2.10: Sample Image obtained when trained on Genre.

We found that our discriminator was unable to differentiate between the various genre and therefore we moved on a new GAN architecture inspired from AC-GAN(Auxiliary Classifier GAN). Auxiliary Classifier Generative adversarial network[13] is a network architecture that improves GAN by changing discriminator of GAN to also predict the class label(such as particular style or genre in our case.This modification of standard GAN helps in stabilizing training and most importantly makes discriminator aware of what style or genre class exactly is.The discriminator architecture therefore looks like this:

Besides the basic architecture of AC-GAN, another addition was adding Global conditioning(inspired by GAN-gogh paper) whereby whenever there is an activation function between the layers of the generator archi-

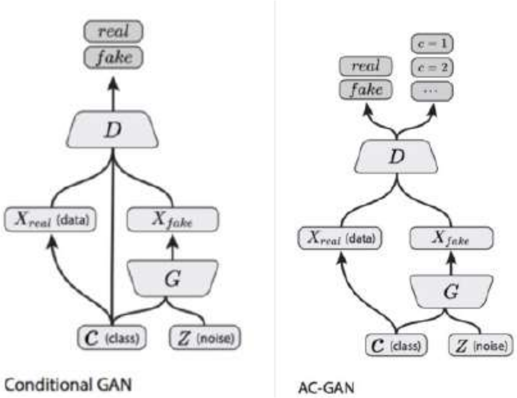


Figure 2.11: Conditional GAN vs AC-GAN[13]

tecture, we can use conditioning vector to influence this activation. For this task conditional gated multiplicative activation was used from Conditional Pixel CNN paper. Global conditioning helps both in making our model conditional for particular style and genre, and at the same time influence the layer in our model to get conditioned according to that specific genre or style.

As already mentioned since initially the discriminator was unaware what particular style or genre of image exactly is, therefore generator is able to fool the discriminator easily and discriminator therefore was initially only focusing on whether the image produced by generator is real or fake(actual GAN loss). To improve upon this we initially pre-trained the discriminator for few iteration on actual real data and real labels(style and genre classes).

2.2.1 Results

2.2.2 Genre

1. Cityscape

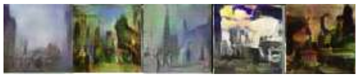


Figure 2.12: Result on Cityscape Genre

2. Landscape

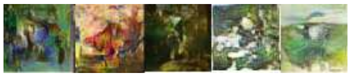


Figure 2.13: Result on Landscape Genre

3. Portrait

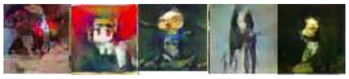


Figure 2.14: Result on Portrait Genre

4. Still life

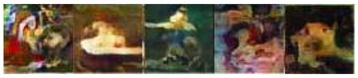


Figure 2.15: Result on Still Life Genre

2.2.3 Style

1. Impressionism

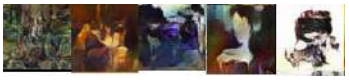


Figure 2.16: Result on Impressionism Style

2. Realism

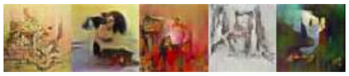


Figure 2.17: Result on Realism Style

3. Symbolism

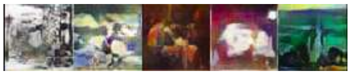


Figure 2.18: Result on Symbolism Style

4. Expressionism

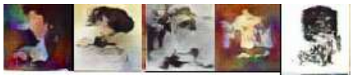


Figure 2.19: Result on Expressionism Style

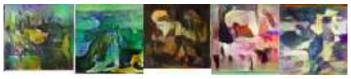


Figure 2.20: Result on Cubism Style

5. Cubism

The results we obtained here seemed to have elements of particular genre or style but still they were not convincing enough.

Work done

3.1 Introduction

Being unable to obtain to obtain good results when training previous mentioned Auxiliary Conditional GAN architecture that generated multiple style and genre, we shifted to a more clear approach of picking a particular genre and then applying styles to it. For this purpose we picked ”Landscape” as our genre and trained generative model for that. We started with Vanilla DCGAN[7] for 128x128 landscape images.

3.2 Generative Models

3.2.1 DCGAN(Deep convolutional generative adversarial networks)[7]

DCGAN[7] is one of the most popular and successful architecture for GAN. It composes of convolution layers without max pooling.Instead of max pooling,it uses convolution stride for downsampling and transposed convolution for upsampling. We trained DCGAN architecture for 128x128 images, being restricted by the resources.

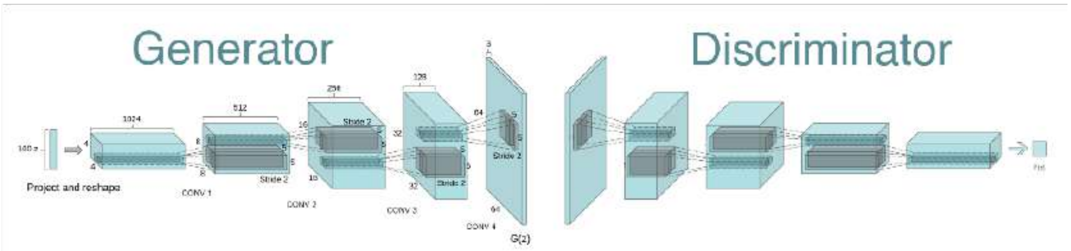


Figure 3.21: DCGAN architecture[7]

3.2.1.1 Results

When trained vanilla DCGAN for a single class such as ”Landscape”, the results where definitely much better than what we obtained previously. However to improve upon these results we improved upon the architecture of DCGAN inspired from Multi Scale Gradient GAN.

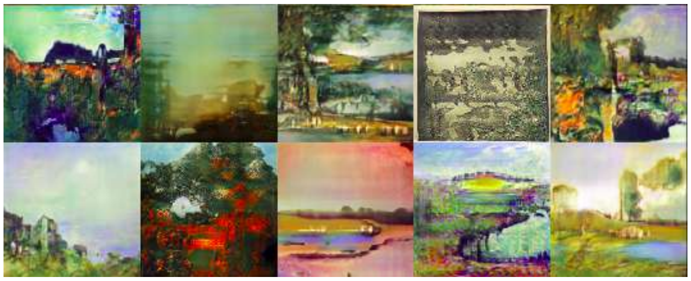


Figure 3.22: Images produced by DCGAN architecture

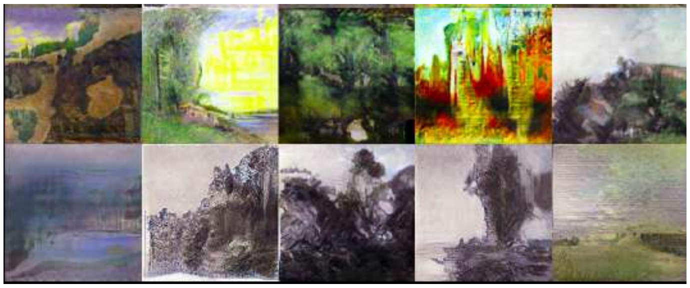


Figure 3.23: Images produced by DCGAN architecture

3.2.2 Multi Scale Gradients GAN[3]

As mentioned previously in the literature survey about MSG-GAN,[3]it a simple but robust architecture that is less susceptive to hyperparameters and can be used across many domains. With connections directly from generator to discriminator, the instability of GAN training is also handled well.

Our architecture for MSG-GAN looks as shown below. We trained the similar architecture for 128x128 and 256x256 variant(restricted by resources).

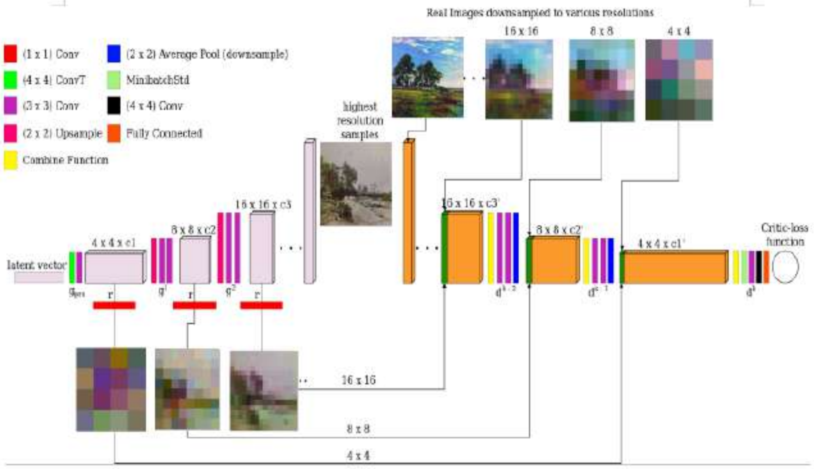


Figure 3.24: Architecture for our model

Since there are images generated even at intermediate layer of the generator, we could also observe how the images are generated from lower to higher resolution, while training.

3.2.3 Results

We can see from the below results, the addition of multi scale gradient architecture over the DCGAN architecture has significantly improved the results. Also the stability of training can be observed from fig of epoch vs resolution whereby the generator only makes small incremental improvements for a fixed latent points,as training progresses.

Result on 128x128 variant

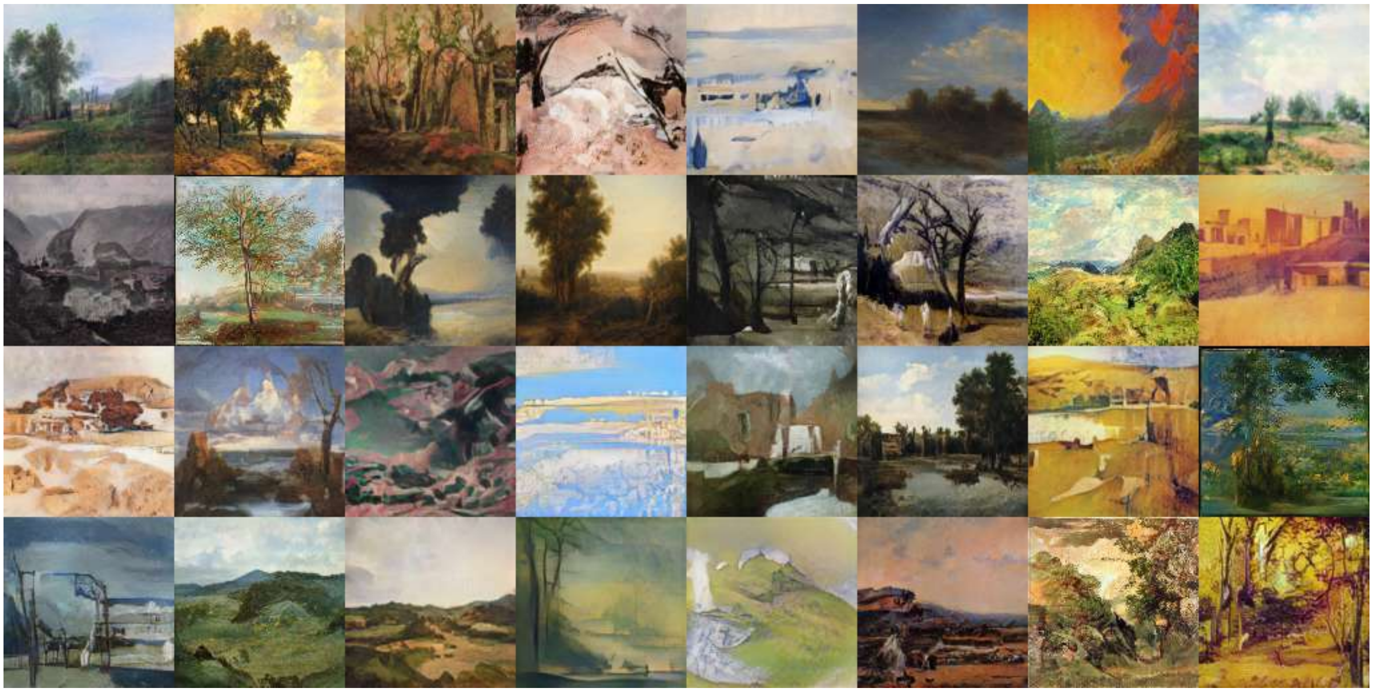


Figure 3.25: Result on 128x128 variant

Result for for 256x256 image

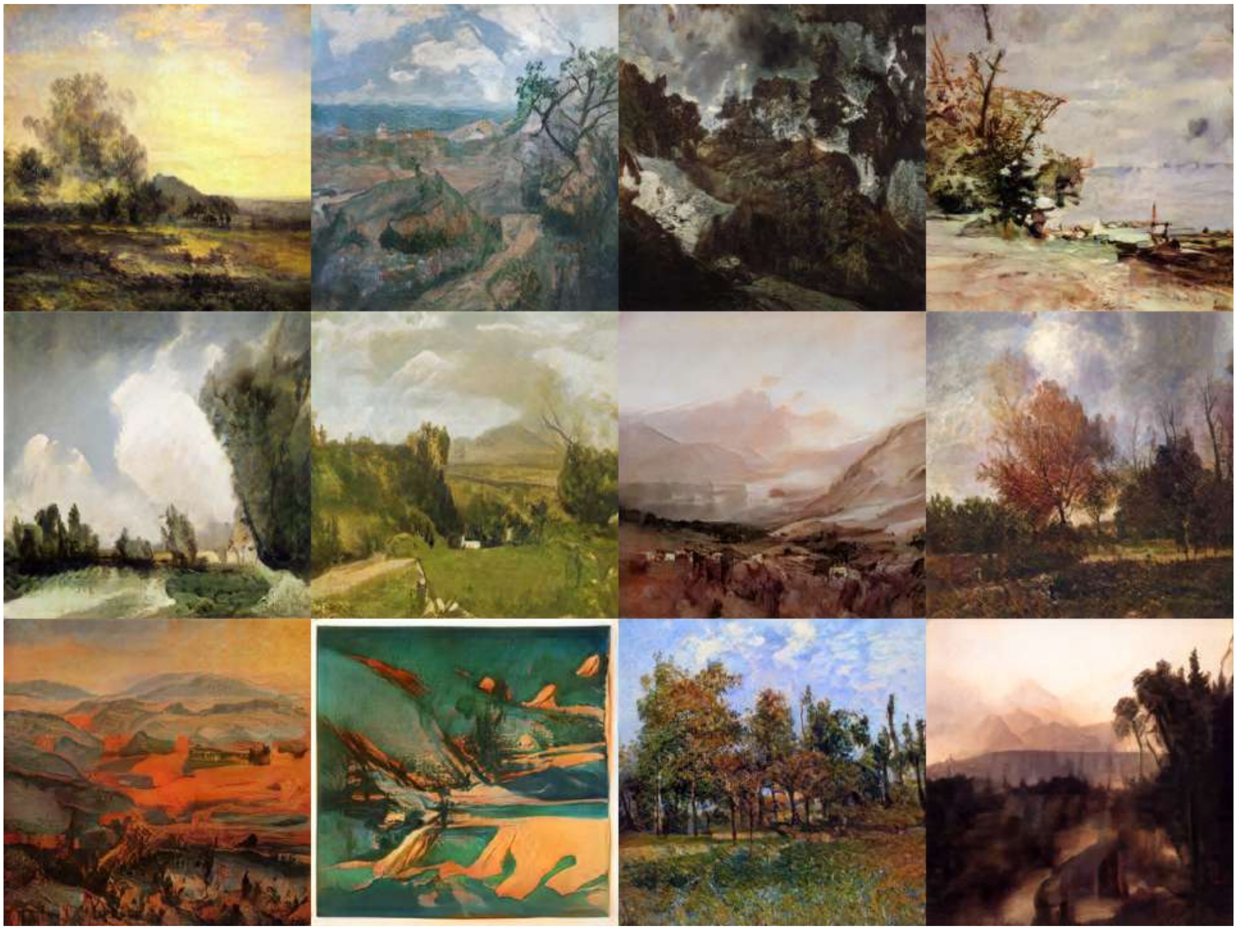


Figure 3.26: Result on 256x256 variant

Training Epochs vs Resolution image

As can be seen from the below plot of training epochs vs resolution, which shows how the different intermediate layers from 4x4 to 256x256 train and hence the quality of image improves at all scales.
We can see how the generator makes incremental changes for a fixed latent point during the training process.

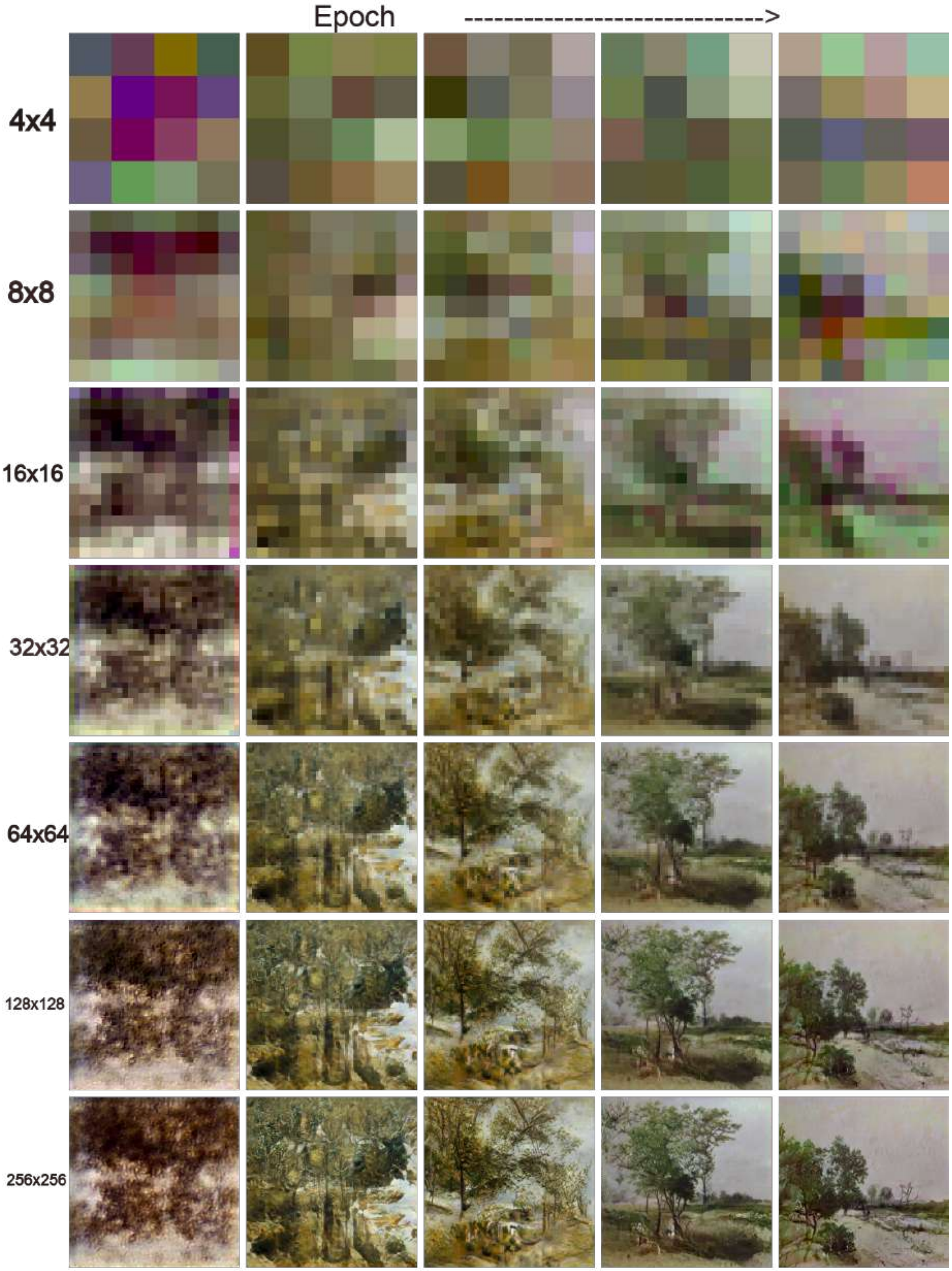


Figure 3.27: Epochs vs Resolution , Example 1

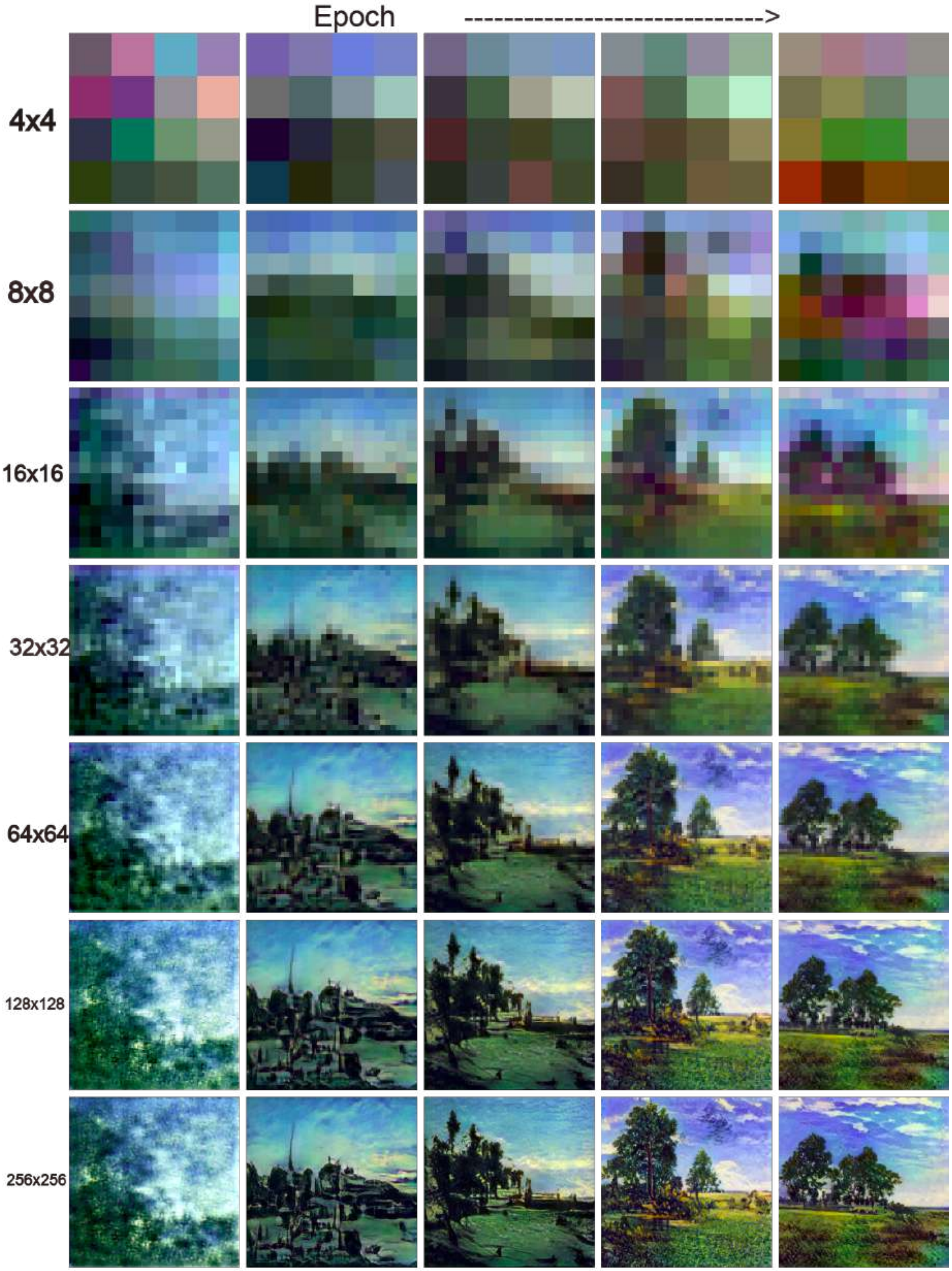


Figure 3.28: Epochs vs Resolution, Example 2

3.3 Image Super Resolution

Being restricted by the resources and training time to allow experimentation, we propose a novel GAN architecture that builds upon the basic conditional GAN architecture of pix2pix and improves it significantly for super resolution with less parameters and training time. This model allowed us to obtain higher resolution art images from the landscape images generated by the previous model with far less resources.

The task of super resolution involves increasing the resolution of the image to twice or more of the original image, but preserving the fine texture which would otherwise be missing when using methods such as nearest neighbor interpolation or bicubic interpolation.

3.3.1 Architecture

Generator We replaced the conventional U-Net architecture for generator with 54-Layer Tiramisu. The tiramisu[14] architecture has seen state of the art performance in semantic segmentation. The encoder-decoder architecture has been used across several image to image translation problems. In such a architecture the image is down sampled using convolutions until a bottleneck and then brought back up to its full resolution using transposed convolution. Since there are a lot of structural similarity between the encoder and decoder, skip connections are used in U-Net architecture for the same.

Densely Connected Convolutional Networks (DenseNets) which use dense blocks that joins every intermediate layer's output with its input in a feed forward manner has achieved better results in image classification. The Tiramisu architecture remains similar to U-Net except that it uses dense blocks. This method has proved to be highly parameter efficient.

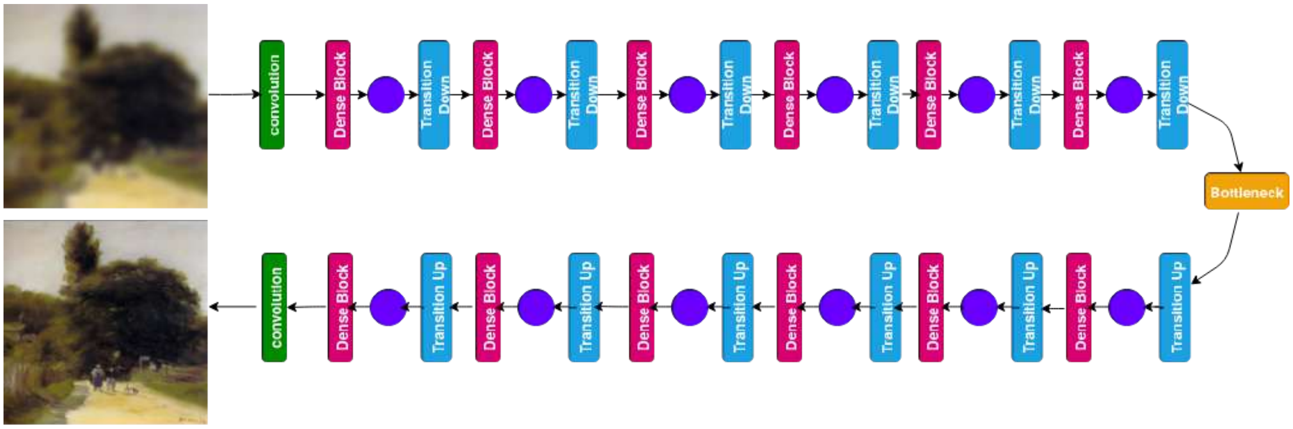


Figure 3.29: Generator Architecture

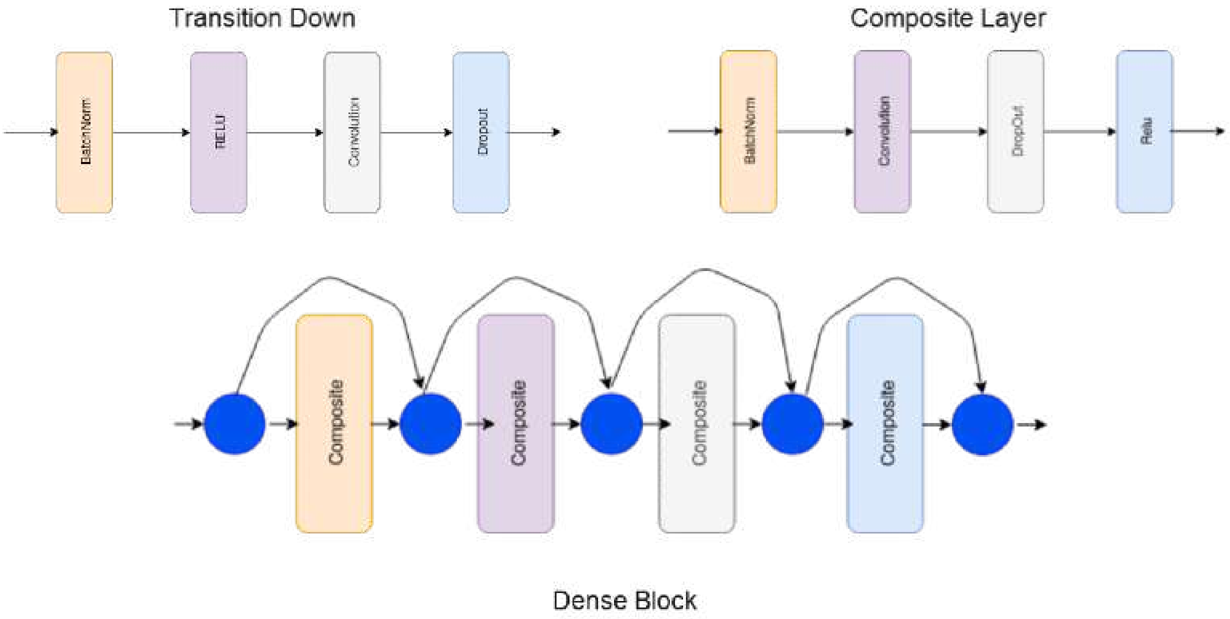


Figure 3.30: Various components in generator. **Transition Down Layer**; **Composite Layer**; **Dense Block**. The concatenation operation shown by curved arrows ending on blue circles

For 512x512 model the model had 6 layers of dense block on either side of bottleneck in the encoder-decoder architecture. After each dense block on the encoder side there is a transition down layer that consisted of convolution, batchnorm and relu. While on the discriminator side after each dense block is a transition up layer using transposed convolution.

Generator	
Operation	Output Shape
Input Conv	512,512,48
DB Encoder 1	512,512,96
TD1	256,256,48
DB Encoder 2	256,256,96
TD2	128,128,48
DB Encoder 3	128,128,96
TD3	64,64,48
DB Encoder 4	64,64,96
TD4	32,32,48
DB Encoder 5	32,32,96
TD5	16,16,48
DB Encoder 6	16,16,96
TD6	8,8,48
DB Bottleneck	8,8,228
TU6	16,16,48
DB Decoder 6	16,16,192
TU5	32,32,48
DB Decoder 5	32,32,192
TU4	64,64,48
DB Decoder 4	64,64,192
TU3	128,128,48
DB Decoder 3	128,128,192
TU2	256,256,48
DB Decoder 2	256,256,192
TU1	512,512,48
DB Decoder 1	512,512,192
Output Conv	512,512,3

Discriminator

For the discriminator we used Patch Discriminator from Conditional GAN.[2] The receptive field at the output is larger than a single pixel as in case of DCGAN[7] discriminator and therefore when doing pixel wise comparison, the receptive field is a patch and helps in reducing artifacts in images.

Discriminator	
Operation	Output Shape
Layer1	256,256,32
Layer2	128,128,64
Layer3	64,64,128
Layer4	32,32,256
Layer5	31,31,512
Layer6	30,30,1

3.3.2 Loss Function

The loss function in our case is a weighted combination of adversarial, L1 and perceptual loss.

$$Loss_{t}otal = W_{gan}\ x\ L_{Adv} + W_{L1}\ x\ L_{L1}\ + W_{vgg}\ x\ L_{vgg}$$

Adversarial loss The adversarial loss is the normal Conditional GAN loss -

$$L_{A}dv = E_{(x,y)}[log(D(x,y)]\ + E_{(}x,z)[log(1 - D(x,G(x,z))]$$

Smooth L1 loss The smooth L1 loss function between the target image and generated image is better than L2 loss as it helps reduce pixelization and artifacts.

$$L_{L1}\ = E_{x,y,z}[||y - G(x,z||_{1}]$$

Perceptual Loss

Different from the pixel wise mean square loss between two images, the perceptual loss compares the loss between the generated and real image in high level feature space which provides visually better results. For our case we used VGG-19 network to obtain the high feature maps at higher dimensions. The generated and real images are passed through the VGG-19 network and the loss and L2 loss is calculated at pool 5 layer just before the fully connected layers.

3.3.3 Dataset

We designed two model for super resolution at 256x256 images and 512x512 images. All images from our landscape dataset which were greater than or equal to 512x512 were resized directly to 256x256 and 512x512 for generating the good resolution real samples. While for conditional input to generator these images were first resized to 128x128 and then resized to 256x256 and 512x512 using bicubic interpolation. This gave us a scale factor of 2x and 4x on a 128x128 image. During the training process, images were also randomly flipped to generalize the model better. This gave us around 11,500 images for training the model.

3.3.4 Training

We used weighted values of W(L1)=100, W(adv)=2, W(per) = 10. Adam optimizer with a learning rate of 0.001 was used for training. At each iteration one update per discriminator and generator is performed.

3.3.5 Results

Super Resolution to 256x256, while training

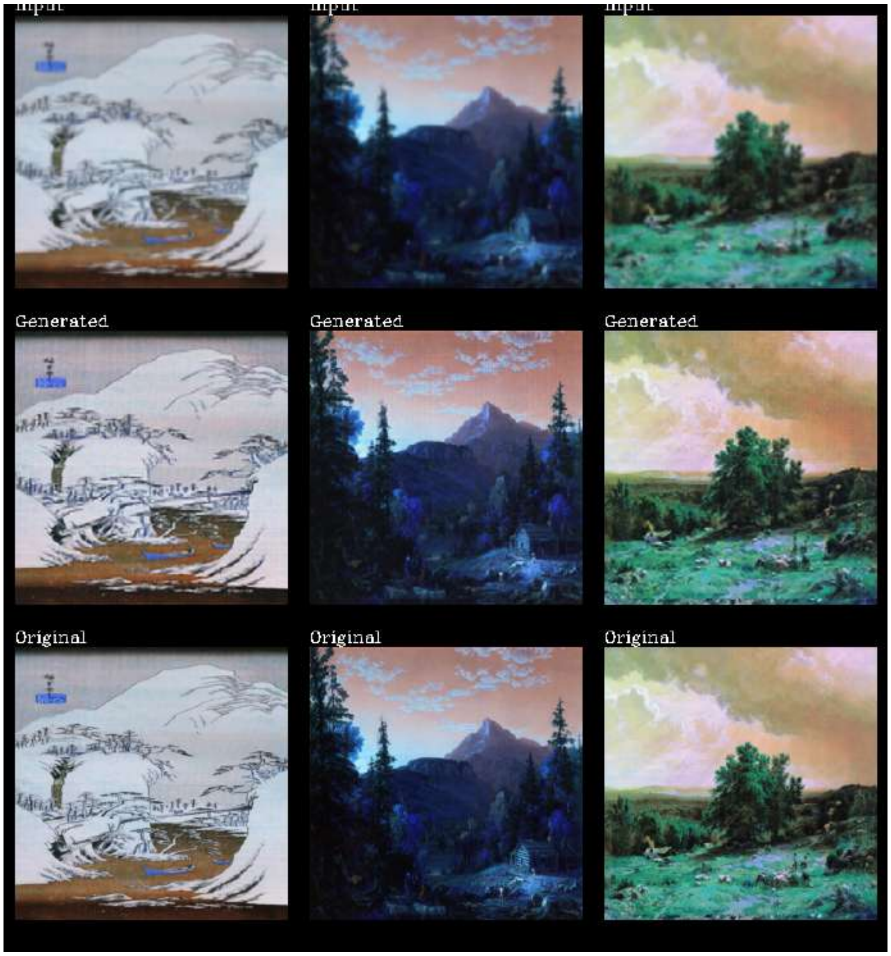


Figure 3.31: **Super Resolution to 256x256 :**

Top row: 256x256 image achieved by bicubic interpolation on 128x128 image, *Middle row:* 256x256 Image generated by our model, *Bottom row:* Original 256x256 image

Super Resolution to 512x512, while training

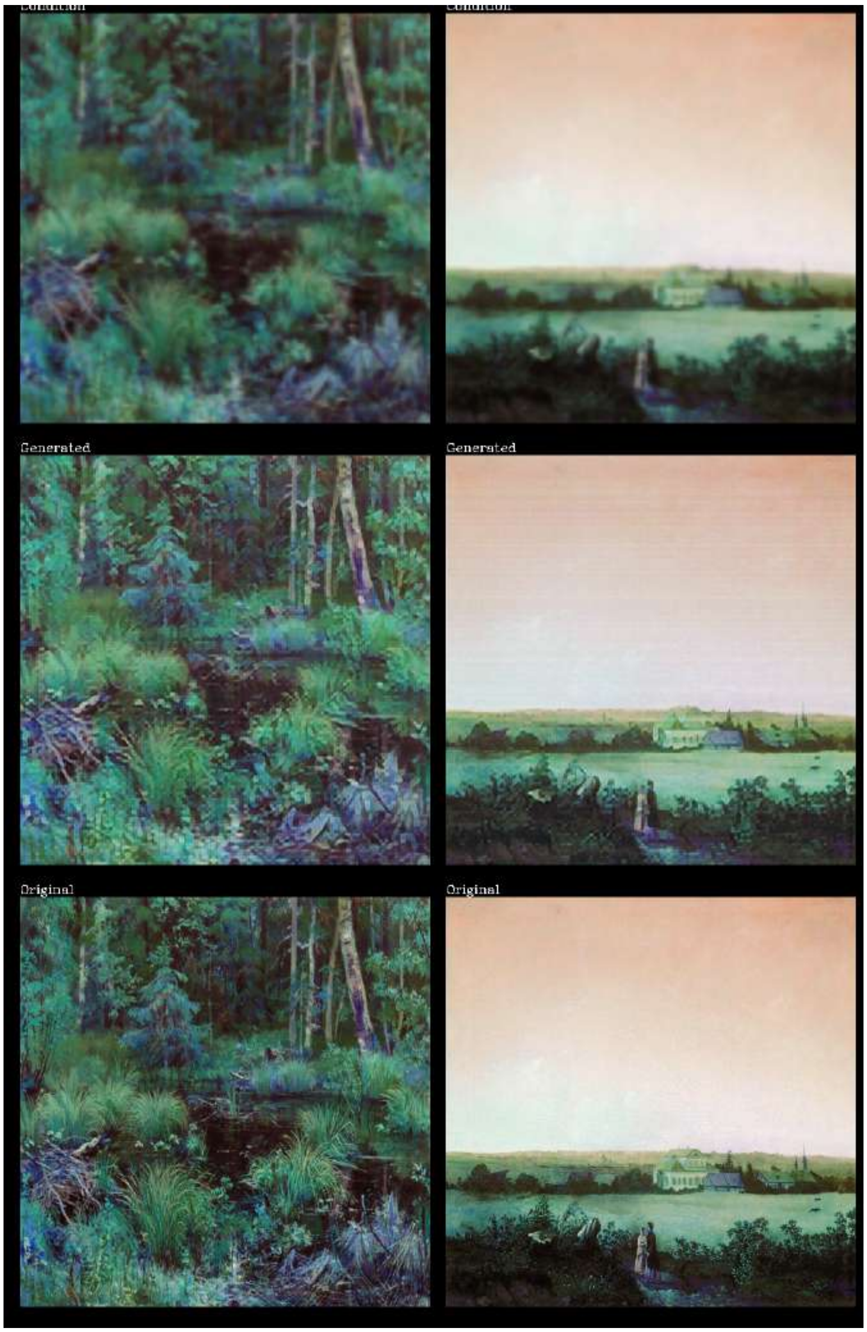


Figure 3.32: **Super Resolution to 512x512 :**

Top row: 512x512 image achieved by bicubic interpolation on 128x128 image, *Middle row:* 512x512 Image generated by our model, *Bottom row:* Original 512x512 image

From the results seen above it is clear that our model is able to perform well and produce good super Resolution images. The results for 2x scale factor are definitely better than 4x. However compared to bicubic interpolation the model does quite well. As our landscape-generation model is able to produce descent results at smaller resolutions, our Super Resolution architecture gives us the opportunity of achieving better resolution images, maintaining high level features when restricted by resources and time to train.

3.4 Style Transfer

Having now obtained good resolution images for a particular genre, the next attribute of an artistic image is the style of the image. Therefore the task is to transfer styles to the image generated by our model. For this we use the architecture of ”A Neural Algorithm of Artistic Style”,[6] as mentioned in the literature survey.

Since the method requires only a single style image and a content image without any prior image, we could experiment with multiple styles for a single image. For the styles we again referenced wikiart.org that contains several styles from various artists such as Pablo Picasso, Vincent van Gogh, etc.

With the ability to control the content:style ratio while transferring style on a image, we could experiment and observe varied style weighted images.

Preserving Color Using the method described in ”Preserving Color in Neural Artistic Style Transfer”, we could also ensure that color of the content image is preserved while transferring style. For this we used luminance only transfer approach.

3.4.1 Results

Content Image - 256x256

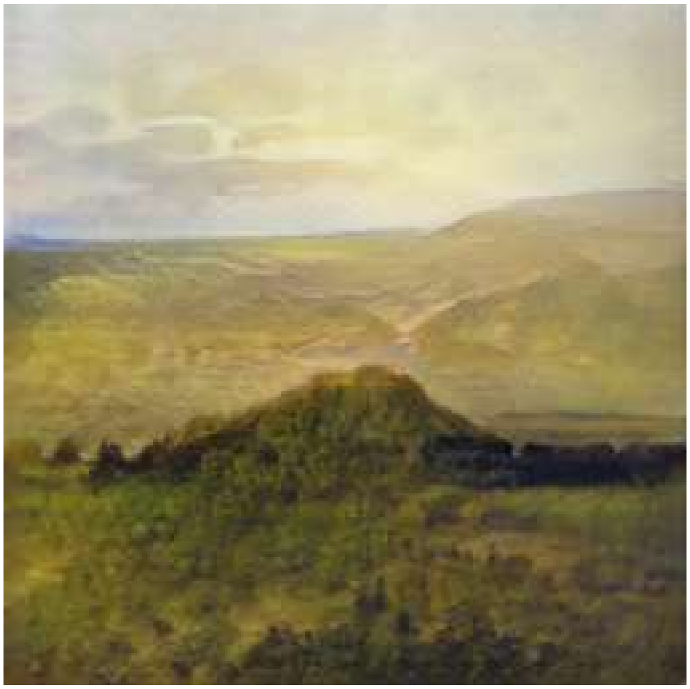


Figure 3.33: Image generated by our model

Style transfer

- **Fauvism -**

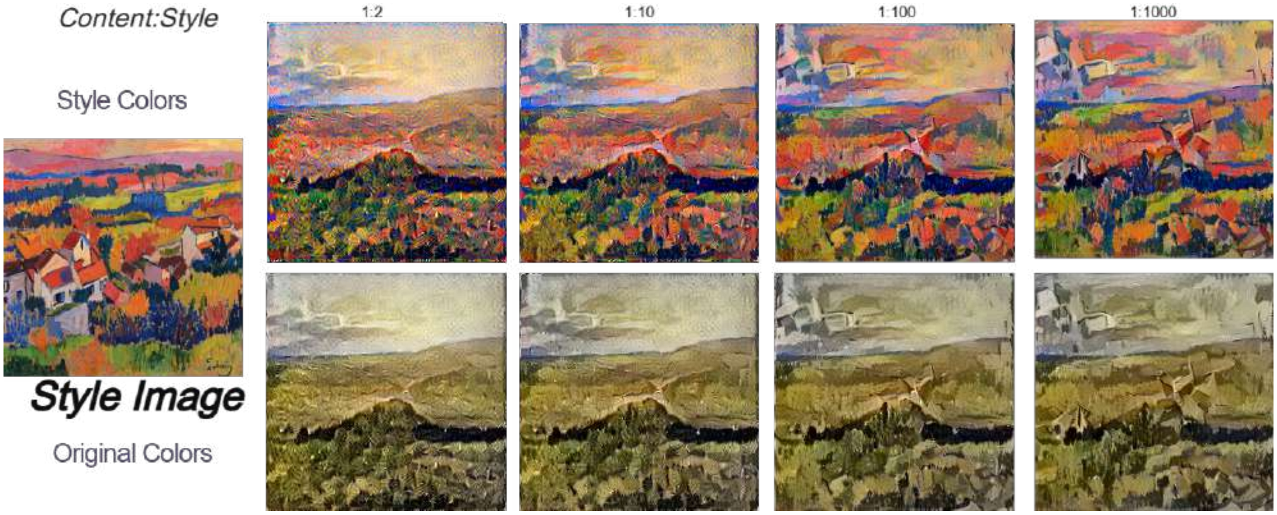


Figure 3.34: Style: Fauvism, Artist: Adre Derain

- **Expressionism -**

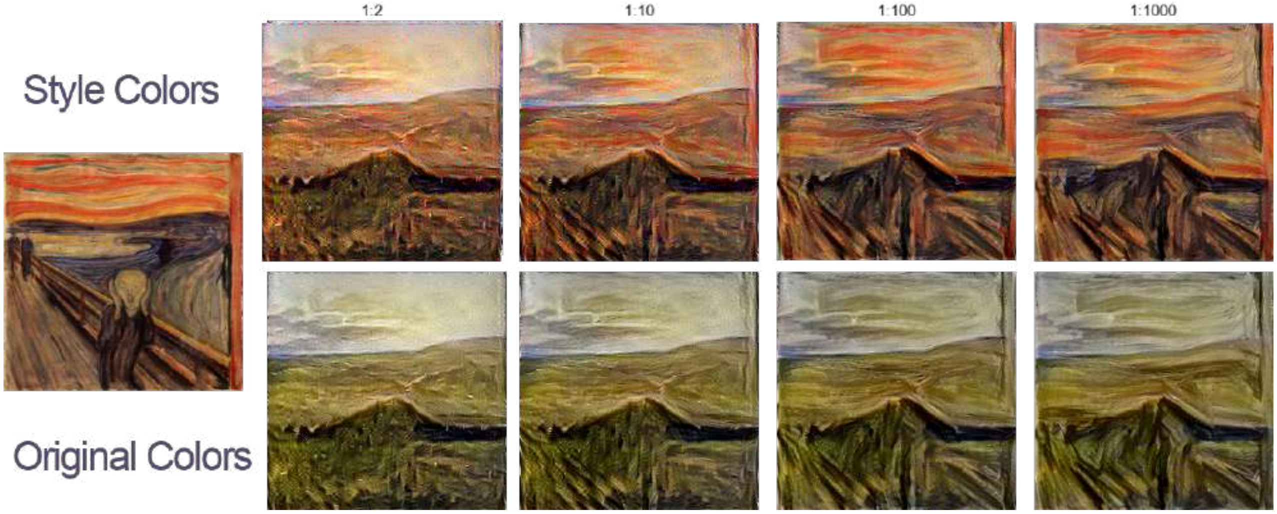


Figure 3.35: Style: Expressionism, Artist: Edvard Munch

- **Oil Painting**

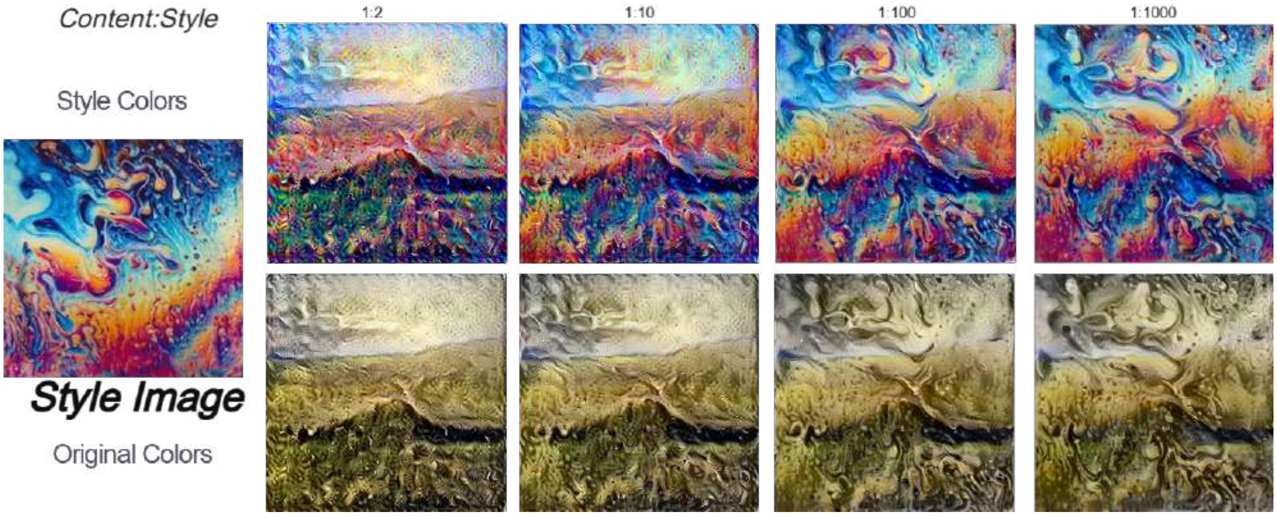


Figure 3.36: Style: Oil Painting, Artist: Steeve Wheeler

- **Indian Space**

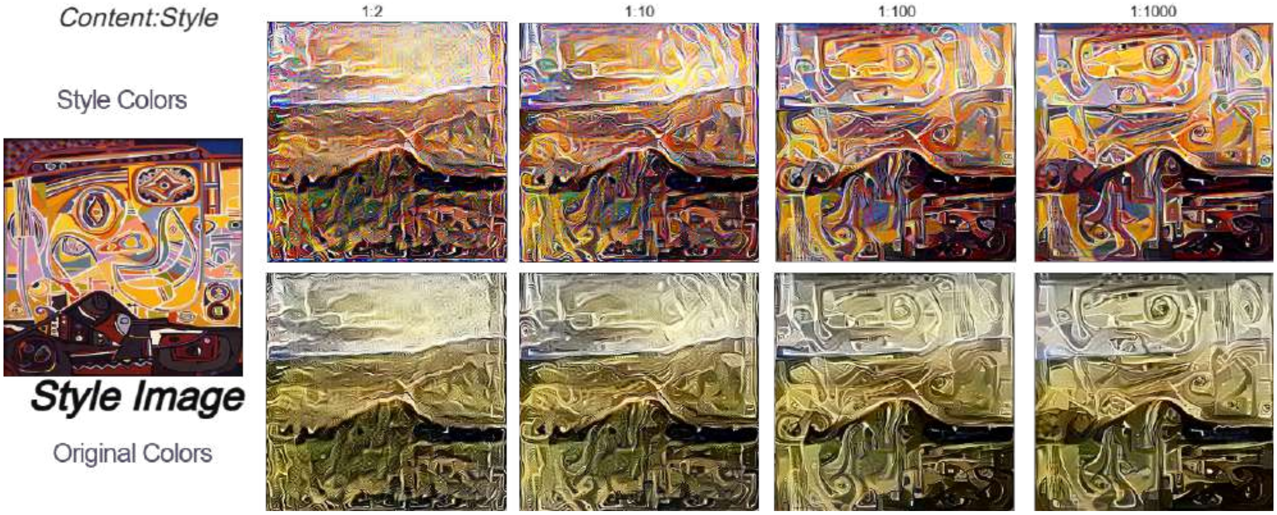


Figure 3.37: Style: Indian Space, Artist: Edvard Munch

Content Image - 512x512

Obtained from applying Super Resolution to 256x256 generated image from the model.

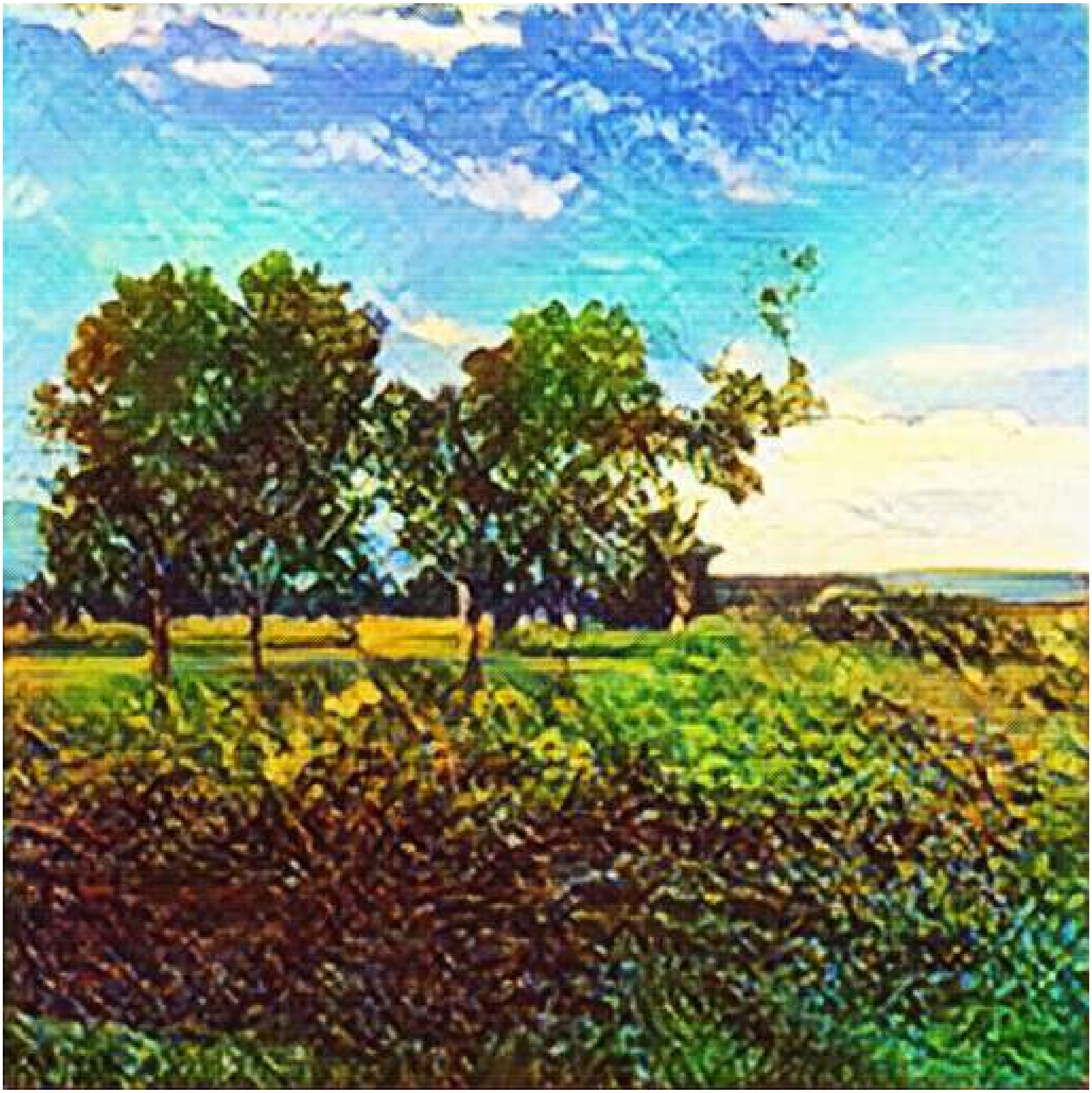


Figure 3.38: Super Resolution on 256x256 image

Style transfer

- **Analytic Cubism**

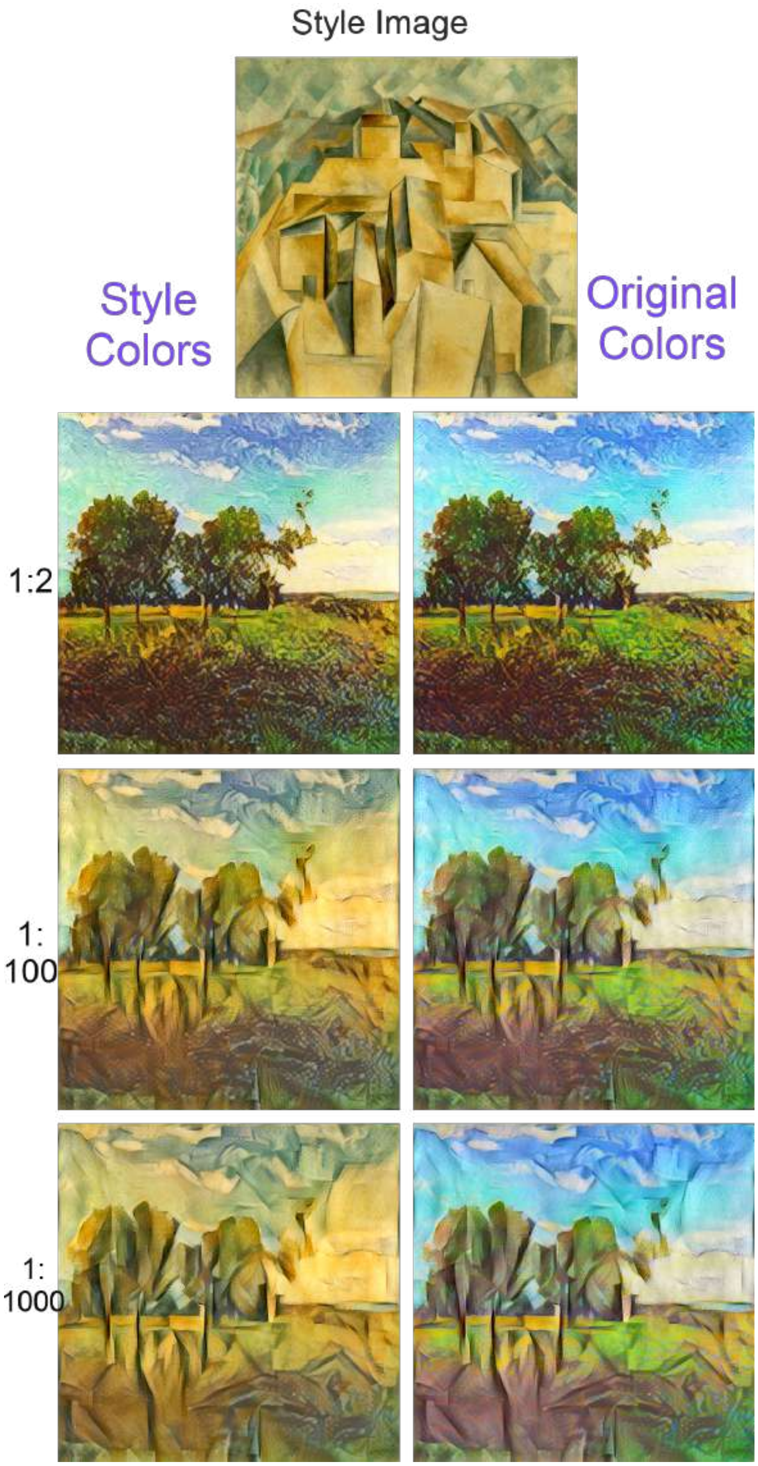


Figure 3.39: Style: Analytic Cubism, Artist: Pablo Picasso

- **Abstract**

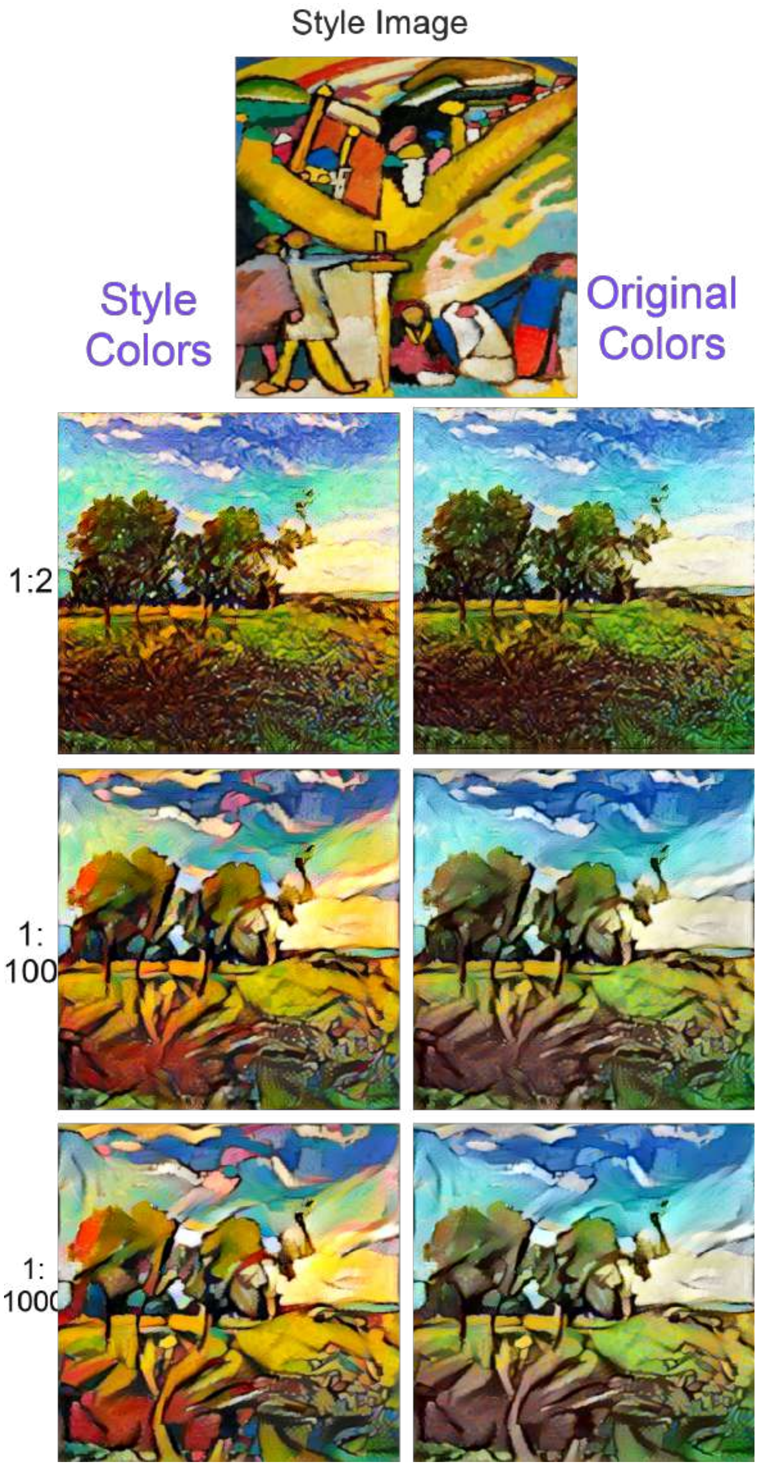


Figure 3.40: Style: Abstract, Artist: Wassily Kandinsky

3.5 Further Improvement

The approach we described above provides us good results on producing Artistic image for a particular genre and multiple styles in parameter and time efficient way. However there is still a possibility of further improvements in following manners -

- With better availability of GPU resources we can train our MSG-GAN model for higher dimensions such as 512x512 and 1024x1024. Since it is apparent that increasing resolution implies better results(as we saw from 128x128 to 256x256), we can then obtain High quality images with better details.
- There is a scope of improving the single image super resolution by experimenting with different weights in the loss function.
- Besides transferring single style on the content image, multiple styles on a single content image could be experimented with few modification in style transfer approach

3.6 Conclusion

Tackling the problem of generating creative artwork using deep adversarial networks is a rather new approach. Although there have seen several research and novel ideas applied to it, the results are far from what could really be classified as *human quality art* . Therefore such a problem has indeed a long way ahead of it and there are several improvements that can be made. We are making one such small step among an ocean of possibilities trying to create artistic image with two of its basic attributes - genre and style. Besides using the multi-scale gradient variant over the basic DCGAN architecture, we also propose an effective approach of image super resolution using losses such as perceptual loss, which improves the quality of image at higher dimension. Multiple style transfer on the generated artwork wraps up our approach of creating novel artworks.

Bibliography

[1] Mehdi Mirza Bing Xu David Warde-Farley Sherjil Ozair Aaron Courville Yoshua Bengio Ian J. Goodfellow, Jean Pouget-Abadie. Generative adversarial networks, 2014.

[2] Phillip IsolaJun-Yan ZhuTinghui ZhouAlexei A. Efros. Image-to-image translation with conditional adversarial networks, 2018.

[3] Oliver Wang Animesh Karnewar. Msg-gan: Multi-scale gradients for generative adversarial networks, 2019.

[4] Jiale Zhi. Pixelbrush-art generation from text with gans, 2017.

[5] Kenny Jones and Derrick Bonafilia. Gangogh: Creating art with gans, 2017.

[6] Matthias Bethge Leon A. Gatys, Alexander S. Ecker. A neural algorithm of artistic style, 2015.

[7] Soumith Chintala Alec Radford, Luke Metz. Unsupervised representation learning with deep convolutional generative adversarial networks, 2015.

[8] Wikipedia the free encyclopedia. Art.

[9] Mohamed Elhoseiny Marian Mazzone Ahmed Elgammal, Bingchen Liu. Can: Creative adversarial networks generating “art” by learning about styles and deviating from style norms, 2017.

[10] wikiart.org. Wikiart:visual art encyclopedia.

[11] Léon Bottou Martin Arjovsky, Soumith Chintala. Wasserstein gan, 2017.

[12] Aaron Hertzmann Matthias Bethge Eli Shechtman, Leon A. Gatys. Preserving color in neural artistic style transfer, 2016.

[13] Soumith Chintala Alec Radford, Luke Metz. Conditional image synthesis with auxiliary classifier gans, 2016.

[14] Venkateswaran N. Bharath Raj N. Single image haze removal using a generative adversarial network, 2019.