GANGogh: Creating Art with GANs

Kenny Jones Jun 18, 2017 · 13 min read

Novel Generation of Flower Paintings

Introduction:

The work here presented is the result of a semester long independent research performed by Kenny Jones and Derrick Bonafilia (both Williams College 2017) under the guidance of Professor Andrea Danyluk. The code associated with this project can be

found at https://github.com/rkjones4/GANGogh. Kenny and Derrick are both heading to Facebook next year as Software Engineers and hope to continue studying GANs in whatever capacity is available to them.

Background:

Generative Adversarial Networks (GANS) were introduced by Ian Goodfellow et. al. in a 2014 paper. GANs address the lack of relative success of deep generative models compared to deep discriminative models. The authors cite the intractable nature of the maximum likelihood estimation that is necessary for most generative models as the reason for this discrepancy. GANs are thus set up to utilize the strength of deep discriminative models to bypass the necessity of maximum likelihood estimation and therefore avoid the main failings of traditional generative models. This is done by using both a generative and a discriminative model that compete with each other to train the generator. In essence, "The generative model can be thought of as analogous to a team of counterfeiters, trying to produce fake currency and use it without detection, while the discriminative model is analogous to the police, trying to detect the counterfeit currency. Competition in this game drives both teams to improve their methods until the counterfeits are indistinguishable from the genuine articles." In more concrete terms:

"To learn the generator's distribution Pg over data x, we define a prior on input noise variables Pz(z), then represent a mapping to data space as G(z; θg), where G is a differentiable function represented by a multilayer perceptron with parameters θg. We also define a second multilayer perceptron D(x; θd) that outputs a single scalar. D(x) represents the probability that x came from the data rather than Pg. We train D to maximize the probability of assigning the correct label to both training examples and samples from G. We simultaneously train G to minimize log(1 − D(G(z))). In other words, D and G play the following two-player minimax game with value function V (G, D): min G max D V (D, G) = Ex∼Pdata(x) [log D(x)] + Ez∼Pz(z) [log(1 − D(G(z)))]. (1)"

The game is played by training G and D alternately until the Nash Equilibrium is reached and the samples G produces are indistinguishable from the samples in the dataset. This formulation is immediately suspect due to the fact that the paper's authors explain that this method results in saturated gradients, and therefore they use a different formulation when actually training. Nevertheless, at the time of publication this formulation produced some of the most promising generative samples of any

model that had been released, and its general structure allowed others to tweak the design to create more nuanced instances of the problem (Goodfellow et. al. 2014).

The Auxiliary Classifier GAN(AC-GAN) was introduced by Augustus Odena et. al. in 2016. AC-GAN adds a classifier layer to the discriminator and a conditioning vector to the generator. The discriminator is then trained to minimize classification error in addition to the traditional GAN objective of real vs. fake. This approach allows labels to be leveraged and provides additional information to the GAN. In addition, it allows the generator to be conditioned to produce certain classes of samples. At present, this is probably the most effective conditional GAN and tends to do better than GANs using unlabeled data. Transfer learning and multitask learning are leveraged by this model (Odena et. al. 2016).

StackGAN was introduced by Zhang et. al. in 2016. StackGAN uses feature information retrieved from a Recurrent Neural Net and a two phase image generation process — - the first phase creates a low resolution image from a noise vector and the second phase uses an encoding of the first image to create high resolution image. The feature information is used to condition both discriminators and both generators. This model creates state of the art high resolution samples.

The Wasserstein GAN(WGAN) was introduced by Martin Arjovsky et. al. in 2017 and introduced a new theoretical framework for GANs that is grounded in theory and empirical findings. With this new framework they create the WGAN which minimizes an approximation of the wasserstein distance, which they found to have useful theoretical properties for the problem of generative modeling, as it creates nicer gradients in the discriminator that the generator can learn from more easily (Arjovsky et. al. 2017). The way they enforced some of their constraints was shown to be inconsistent and was improved upon by Ishaan Gulrajani using a gradient based penalty to constrain the slope of the discriminator. This model was found to be extremely robust and effective at training a variety of GANs ranging from the usual model to previously untrainable models (Gulrajani 2017).

Wavenet and Conditional PixelCNN are two generative models that came out of Google in 2016. Neither are GANs, but both utilize an interesting idea called global conditioning. The key to global conditioning is using the conditioning vector not as a one time addition to the noise vector at the beginning, but instead as a conditioning vector to each activation function in the neural net. This allows much stronger and more complex conditioning than the traditional method. They also both use gated

multiplicative activation functions which seem to mesh well with this global conditioning (van den Oord 2016; van den Oord 2016).

Goal:

Our primary motivation in studying GANs this semester was to try to apply a GAN-derived model to the generation of novel art. Much of the work in deep learning that has concerned itself with art generation has focused on style, and specifically the style of particular art pieces. In papers such as *A Neural Algorithm of Artistic Style*, deep learning nets learn to 1) differentiate the style of a piece of art from its content and 2) to apply that style to other content representations. By building off of the GAN model, we hoped to build a deep-net that was capable of not only learning a distribution of the style and content components of many different pieces of art, but was also able to novelly combine these components to create new pieces of art. The task of novel content generation is much more difficult than applying the style from one particular piece of art to the content of another.

Dataset and Collection:

As the generation component of a GAN can only learn to represent the real distribution that is presented to the discriminator, choosing an appropriate dataset of paintings was a crucial decision for us. We at first experimented with using a dataset of paintings that were only from one artist: Monet. After a few initial tests we found that our models on this dataset were converging poorly as the dataset with only 1200 paintings was too small. We needed to find a way of increasing the size of our dataset, and so we turned to the wikiart database, which is a collection of over 100,000 paintings all labeled on style, genre, artist, year the painting was made, etc (wikiart). From this dataset we knew that we would 1) have enough data to make it likely for our model to converge and 2) that we would be able to use the labeling information to improve the learning capabilities of our net: in GAN scenarios it is often helpful to use extra conditional information when training, especially when learning a distributions with clear sub-distributions (i.e. landscapes vs portraits). In the final version of our model, we constructed our dataset by creating python scripts to scrape over 80,000 individual images directly off of the wikiart website, while retaining both the style and genre labels of each image.

Model:

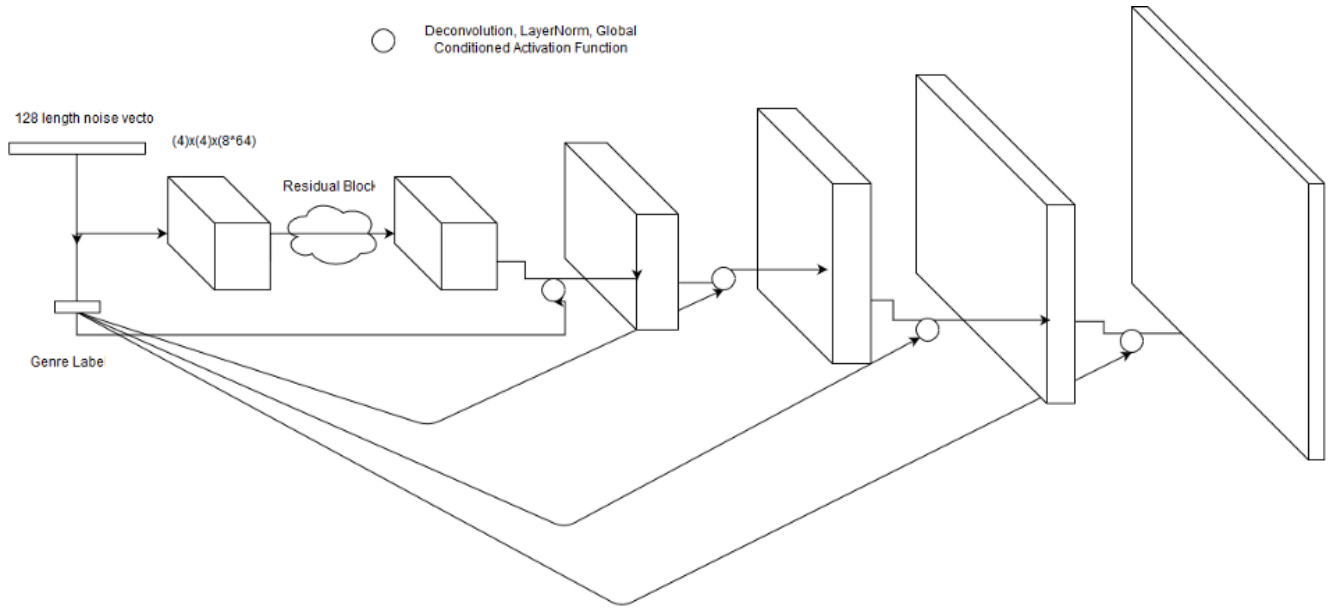
In designing our model we wanted to leverage the power of the improved wasserstein metric in an extension of the typical AC-GAN framework. At a high-level this means that we initially set up our discriminator and generator under the improved wasserstein model, but in addition we added a classifying component to our discriminator where it would also try to predict the genre of a given painting. Our generator was then augmented to take in a one-hot-vector of label data whose purpose would be to influence the genre of the produced painting. We could enforce this metric by adding a penalizing term to our discriminator's cost function that tries to minimize the cross-entropy in its prediction of genre versus the real genre of a given painting, and adding a penalizing term to our generator that tries to minimize the cross-entropy of the discriminator's prediction versus the genre it was instructed to make based on the conditioning vector.

Two other modifications we made to the standard GAN model was the inclusion of pretraining and global conditioning. Both of these additions were made after we observed a tendency for our model to focus heavily on producing 'real' versus 'fake' images in the first iterations, and only later learning to provide different kinds of representations when conditioning on genre. We hypothesized that one of the reasons for this was that when beginning training, our discriminator had not learned the difference between genres at all, and as a result the generator was unable to start differentiating as early as we would like. Our proposed solution was to pretrain our discriminator, before any generator training, and in this way give our discriminator an initial idea of the differences between genres. In the vanilla GAN formulation, this would lead to problematic behavior and a low chance of convergence, but as we use the wasserstein metric, which limits the power of the discriminator and always provides smooth gradients for the generator, we actually reach a more optimal state under this framework: as in an 'optimal' formulation of the GAN setup we would train the discriminator to convergence before each generator training step.

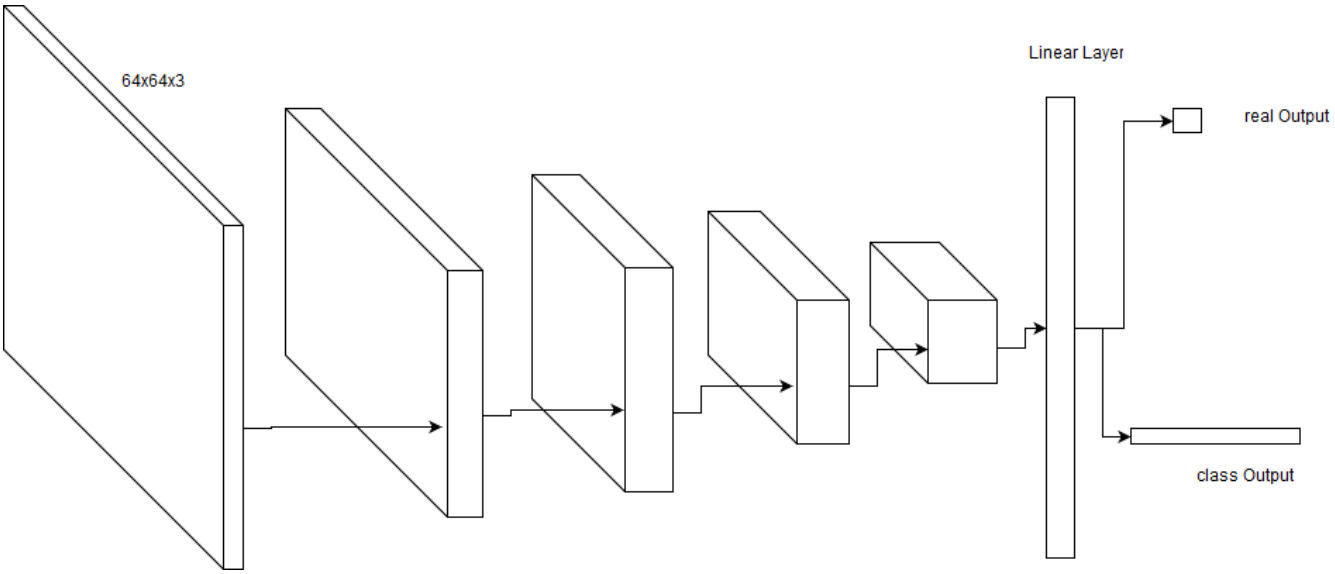
Our inclusion of global conditioning was once again included to allow our generator to better differentiate its produced content as a result of its conditioning vector. The idea behind global conditioning is that every time we perform an activation function between layers in our network, we can use the conditioning vector to influence how this activation occurs, and this conditioning vector can be used differently on a layer by layer basis. We chose to use the conditional gated multiplicative activation function from Conditional PixelCNN and Wavenet. To our knowledge this is the first time that global conditioning has been applied to the GAN framework.

Detailed Architecture:

Generator:



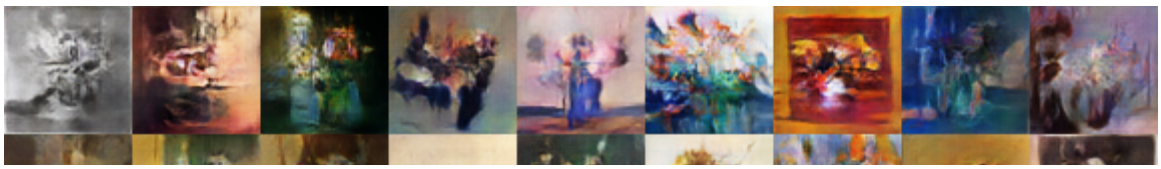
Discriminator:

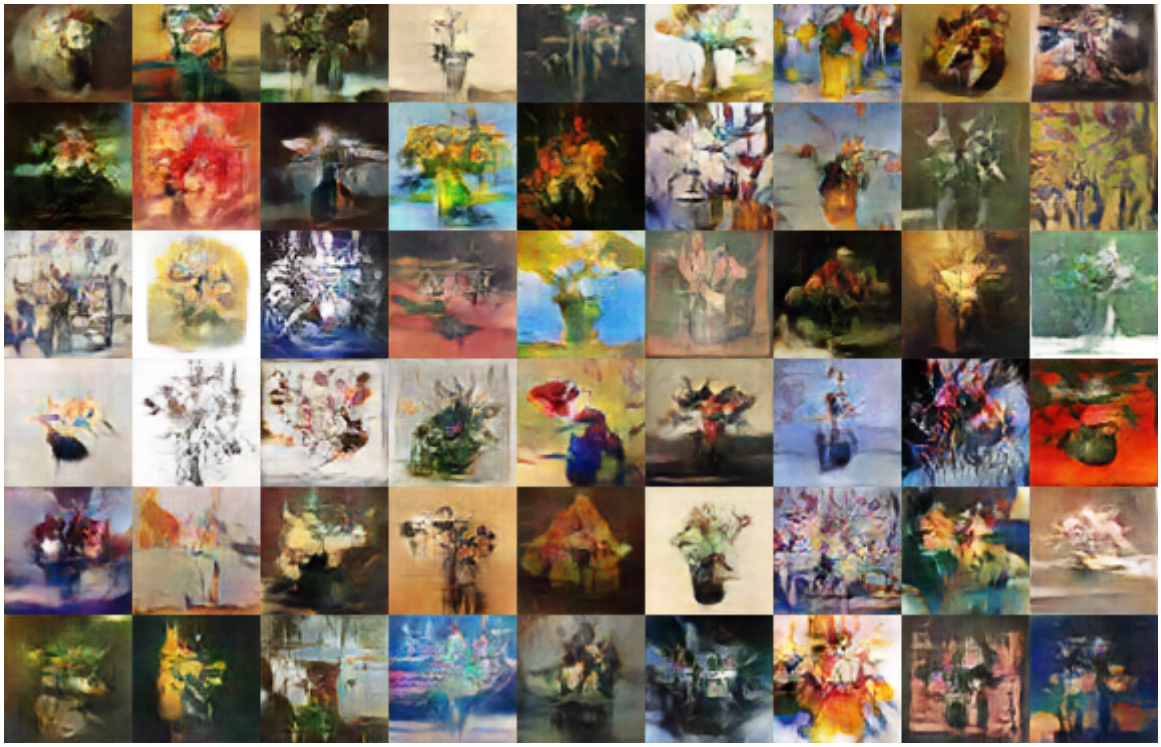


Results:

Here we present a series of discriminator selected photos from three of our genres.

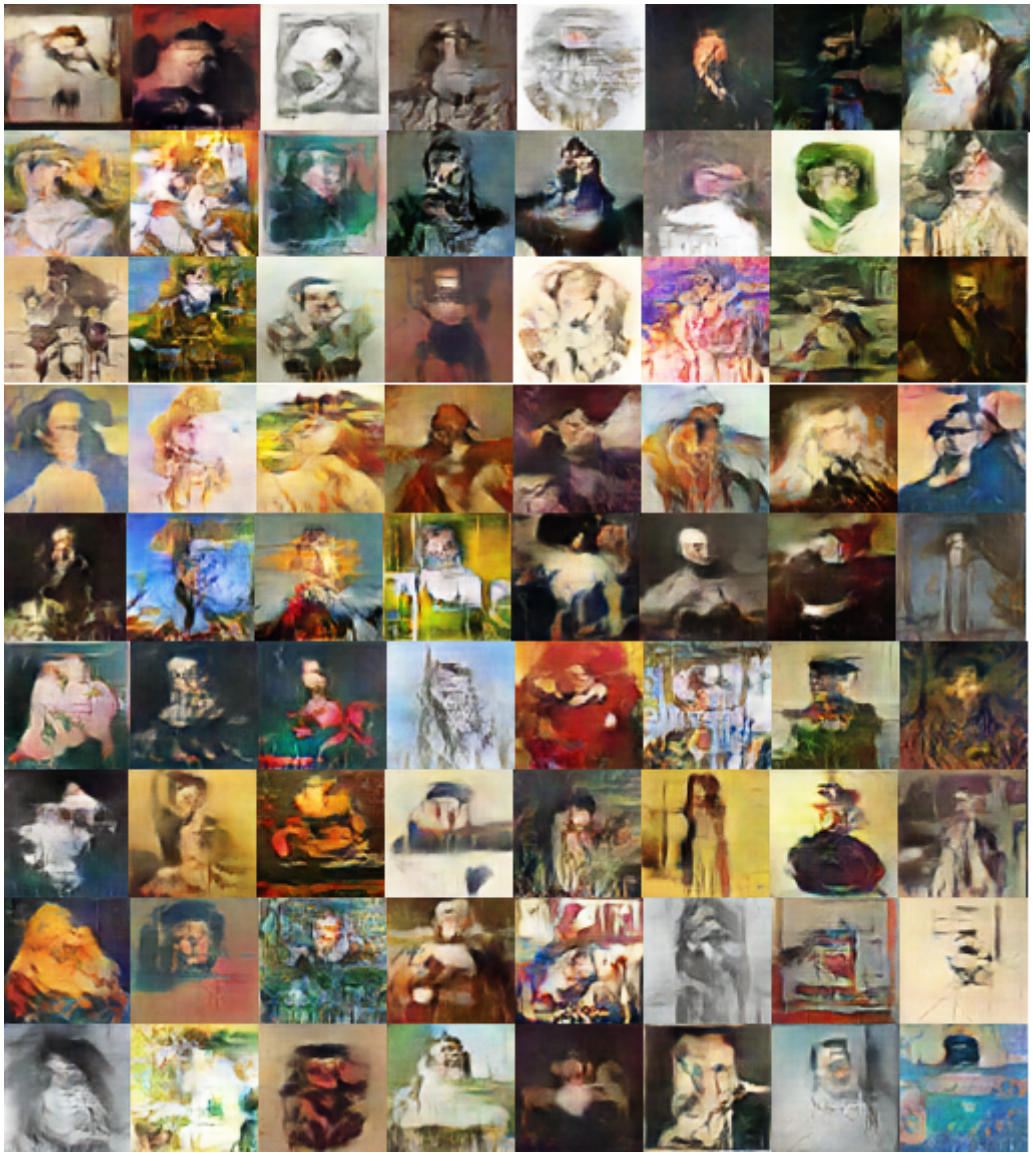
Flowers:





Landscapes:

Portraits:



Due to the nature of our discriminator, we were able to use it to select ‘good’ images that our generator produced. Specifically, as our discriminator was able to both classify and judge the realness of an image, we can produce a set of generator images for a specific genre, and from those images ask the discriminator to choose images that it 1) classifies as that genre with high confidence and 2) also judges has a high realness value. This is the method by which the above images were produced.

We can also, under a static noise vector, see how the generator did in differentiating between genres by changing the given conditioning information. These images for all genres can be seen in the Appendix. Here notice that while the generator has definitely learned to produce different kinds of genres, they are not as differentiated as we may like. Also there is a high variance in the quality of the produced art, where some images produced are much better than others, and even for a given noise vector, it is more realistically represented in specific genres. We believe the reason for this phenomenon is once again that it is much easier for our model to learn representations of ‘real’ versus ‘fake’ art rather than producing art in a given ‘genre’. In adding

pretraining and global conditioning we alleviated this problem somewhat, but clearly have not fixed it completely. Another challenge we faced is that some genres of art are much more complex or have a higher-variance than others. In particular this led to poor performance in our generator when producing certain genres such as Mythological paintings and Animal paintings, we hypothesize though that given larger datasets our model should also be able to produce realistic examples of even these genres.

Although our primary motivation in this project was the generation of images, we were also able to measure the effectiveness of our discriminator by evaluating it as a classifier. We measured our classifying ability both on our training set and a test set 1/20th the size of our training set, and managed to get a very good classifying accuracy of above 90% on genre in our test set. We note that while our classifying accuracy is not strictly rigorous it compares favorably to other classification metrics performed on the same data set, Wai Ren Tan got a classification accuracy of 74.14% on approximately the same dataset in 2016 (Our dataset is just slightly different because we scraped the wikiart website at different points in time) (Tan 2016).

Future Work:

A substantial limiting factor in our project was our access to computational resources. Working with GPU's that had only 2GB of RAM we were forced to stick to producing 64x64 pixel images, as any higher resolution forced significant reductions in both our batch size and dimensionality of the model. We noticed significant improvement in our model when comparing the generated images from our 64x64 models to our initial 32x32 models, so we hypothesize that with additional computational resources, that allow scaling to 128x128 images, the quality of the images would also substantially increase under our current architecture. In thinking about scaling the problem of art generation to even bigger image sizes such as 256x256, we believe that an architecture influenced by StackGAN, using two of our proposed GAN models could work very well. One substantial benefit that we believe such an architecture could give is that our stage I GAN could serve as a way of forcing differentiation between the different genres, and thus the stage II GAN when conditioned on stage I images will be better suited to overcome the difficulties we saw in our model where noise vectors with different conditioning information map to similar images. We also note that this two-tiered GAN model could be improved with additional conditioning information, specifically we suggest that in addition to our genre conditioning set-up described in our model, another one hot vector with style information could be used in the input to both the

stage II discriminator and the stage II generator, and then further division of generated art combinations could be explored such as the renaissance portrait versus the abstract portrait.

Conclusion:

While we do not believe that our model 'solves' the problem of art generation, we hope to have offered insights into the ways in which GANs can be used to generate novel art. Specifically, we have shown that under the right conditioning metrics, wasserstein-style GANs are able to take into account conditioning information in a useful way for image distributions that are 'hard'. Moreover, we have shown that global conditioning is a valid addition to the GAN framework. We also point to our discriminator selected images for specific genres to validate this line of research, as it is clear our generator is doing an excellent job of producing novel art for specific genres.

Works Cited:

Arjovsky, Martin, Soumith Chintala, and Léon Bottou. "Wasserstein gan." *arXiv preprint arXiv:1701.07875* (2017).

Goodfellow, Ian, et al. "Generative adversarial nets." *Advances in neural information processing systems*. 2014.

Gulrajani, Ishaan, et al. "Improved Training of Wasserstein GANs." *arXiv preprint arXiv:1704.00028* (2017).

Odena, Augustus, Christopher Olah, and Jonathon Shlens. "Conditional image synthesis with auxiliary classifier gans." *arXiv preprint arXiv:1610.09585* (2016).

Tan, Wei Ren, et al. "Ceci n'est pas une pipe: A deep convolutional network for fine-art paintings classification." *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016.

van den Oord, Aäron, et al. "Wavenet: A generative model for raw audio." *CoRR abs/1609.03499* (2016).

van den Oord, Aaron, et al. "Conditional image generation with pixelcnn decoders." *Advances in Neural Information Processing Systems*. 2016.

Wikiart. https://www.wikiart.org/

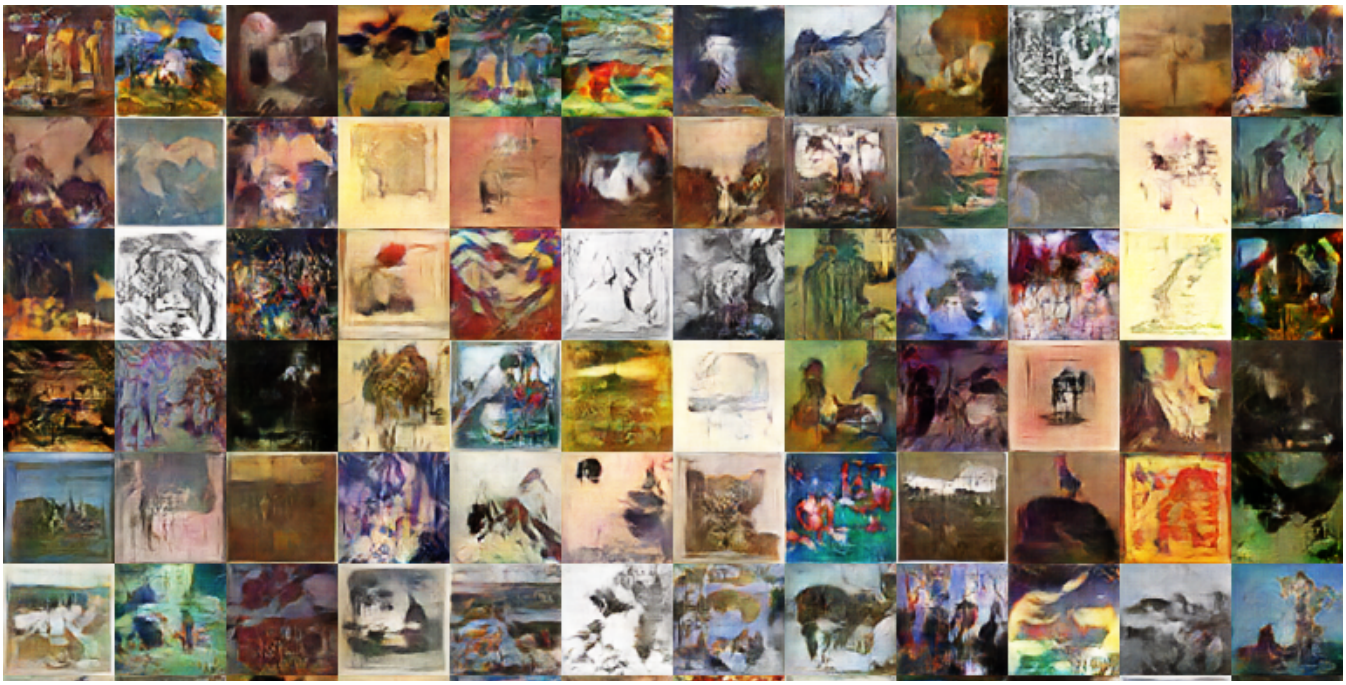
Zhang, Han, et al. “StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks.” *arXiv preprint arXiv:1612.03242* (2016).

Appendix:

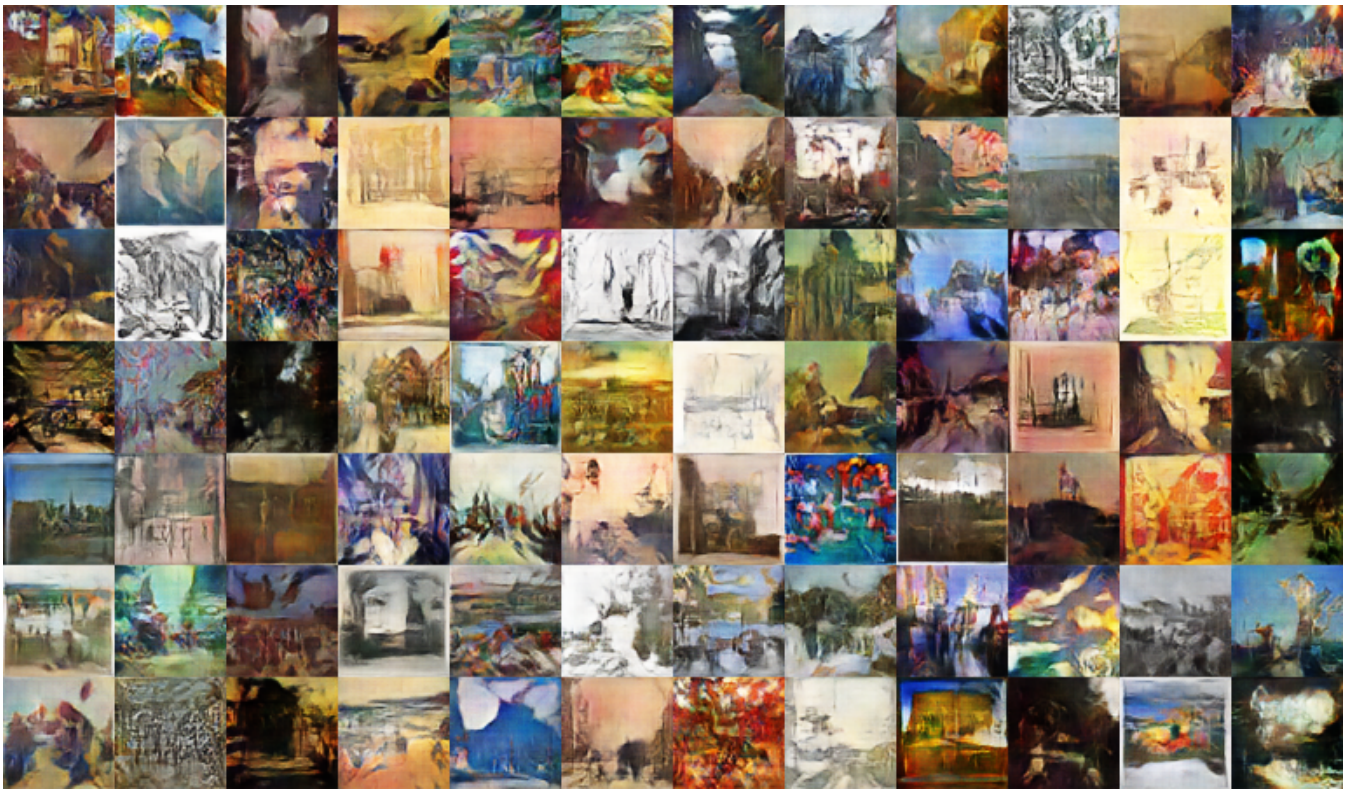
Images produced with same noise vector but different conditioning vector.

Abstract:

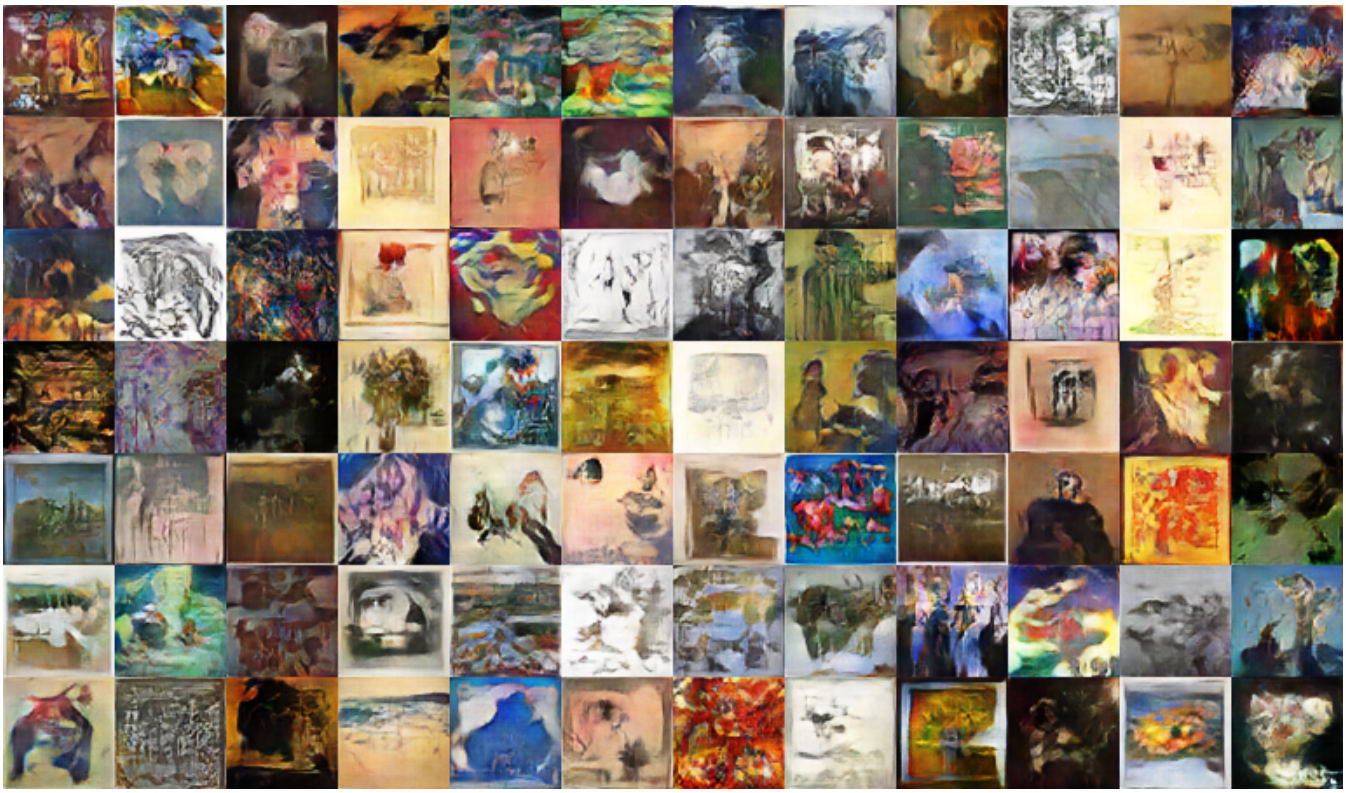
Animal-Painting:



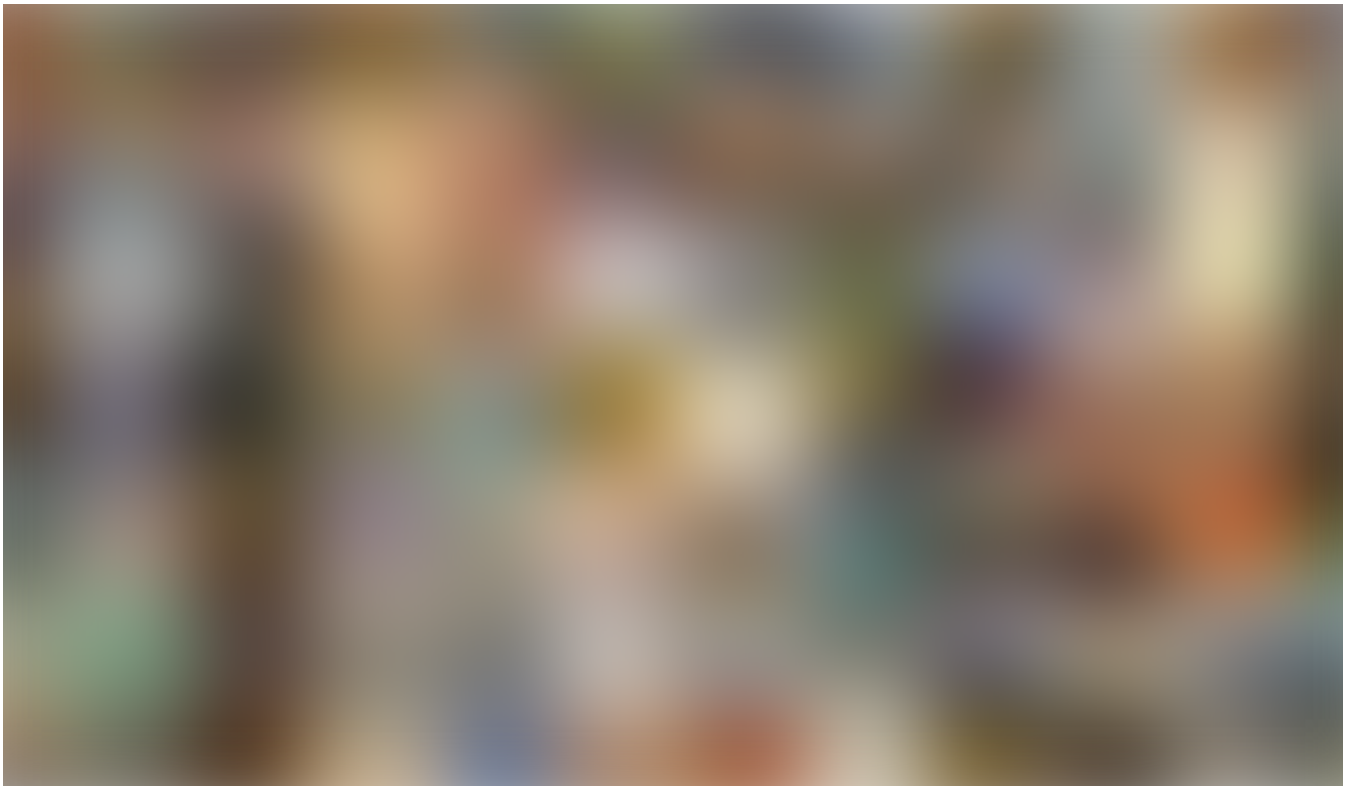
City-Scapes:



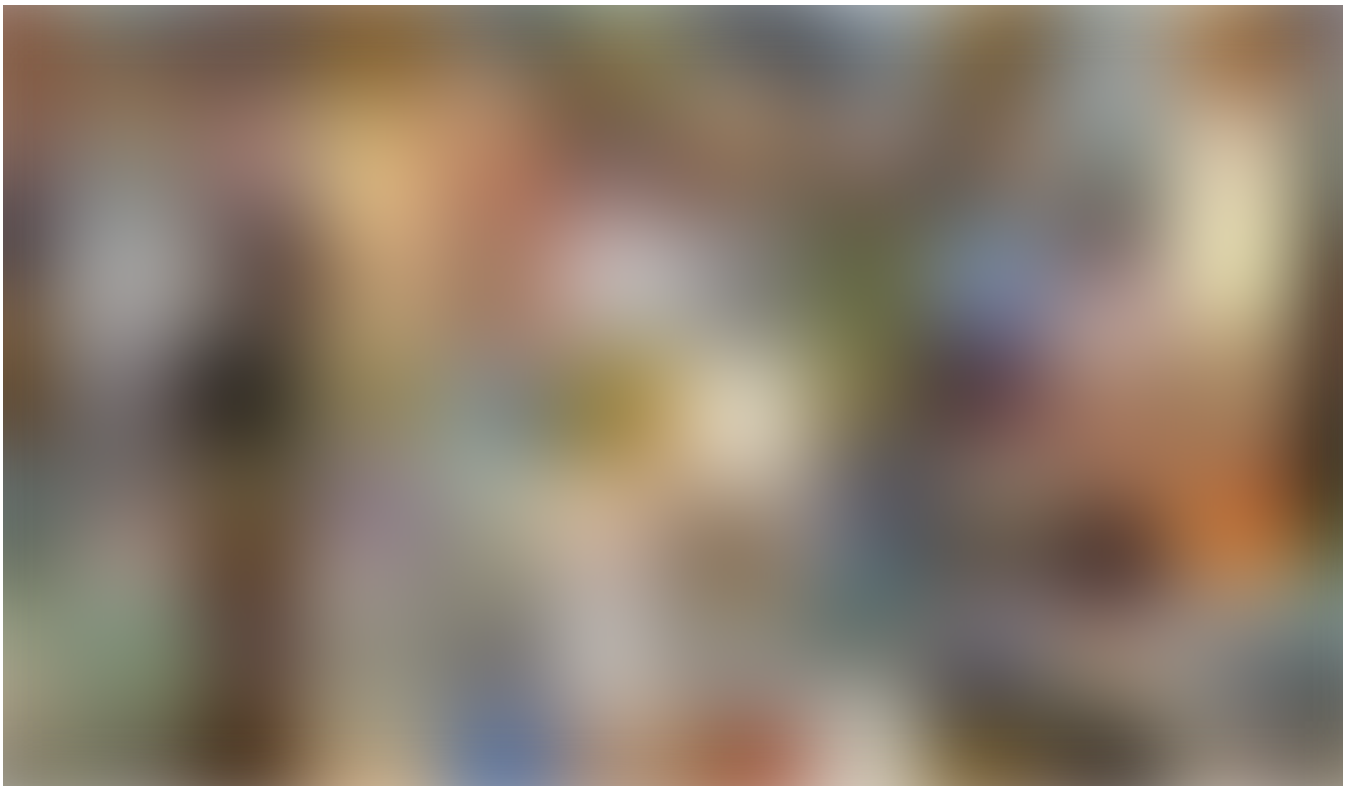
Figurative:



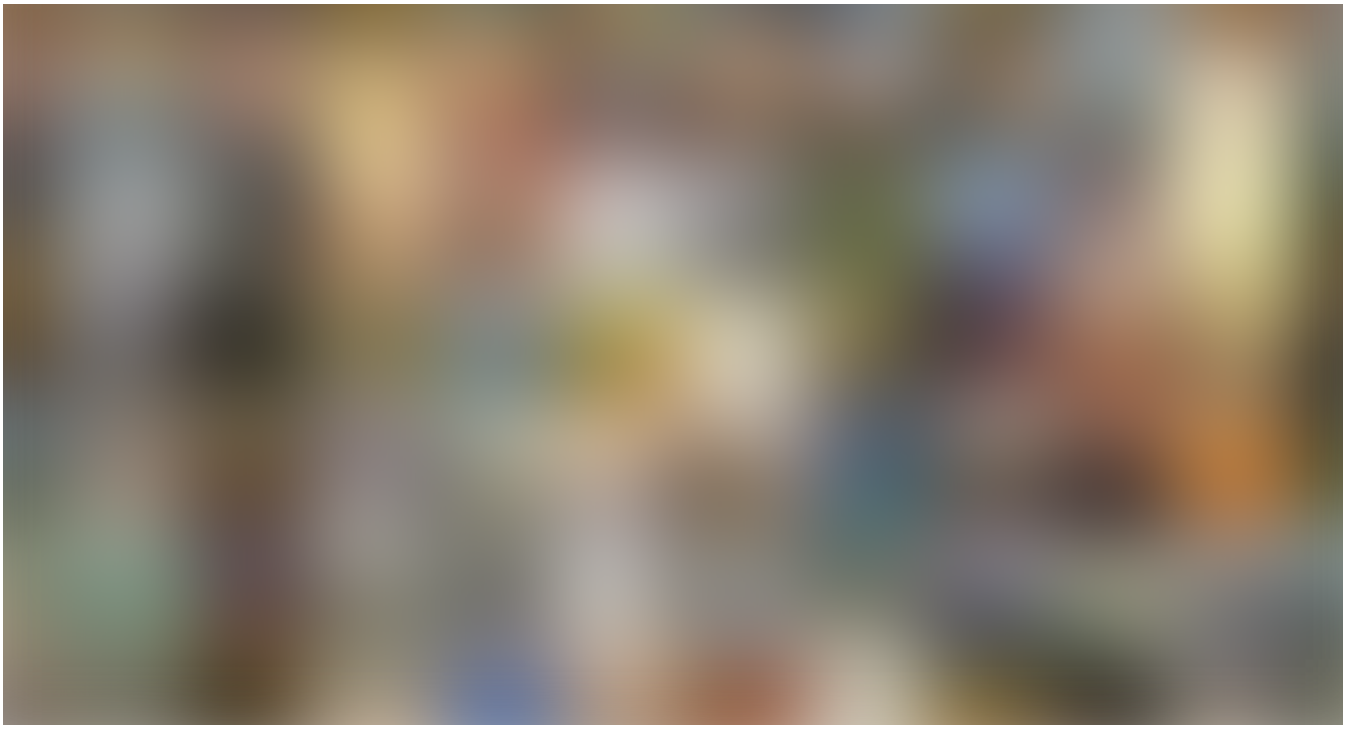
Flower Paintings:



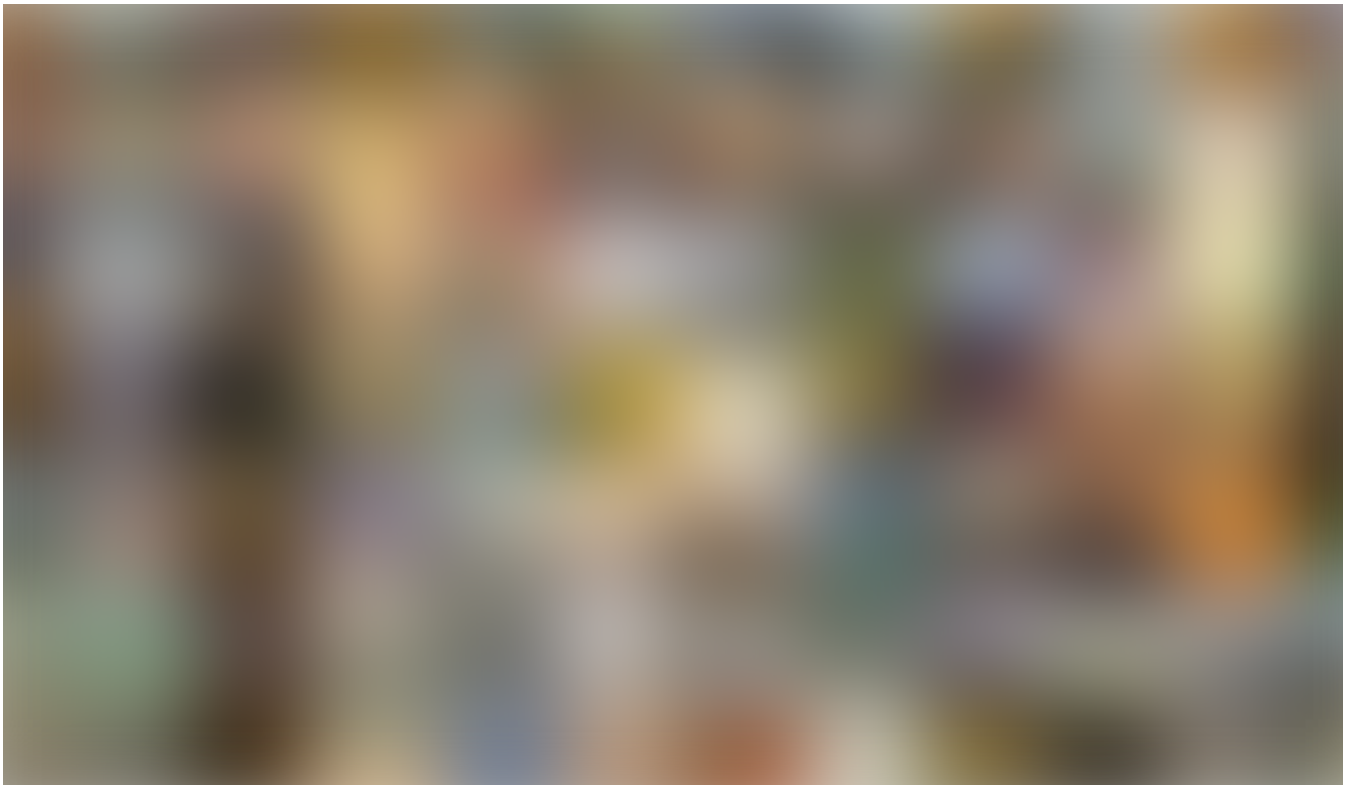
Genre-Painting:



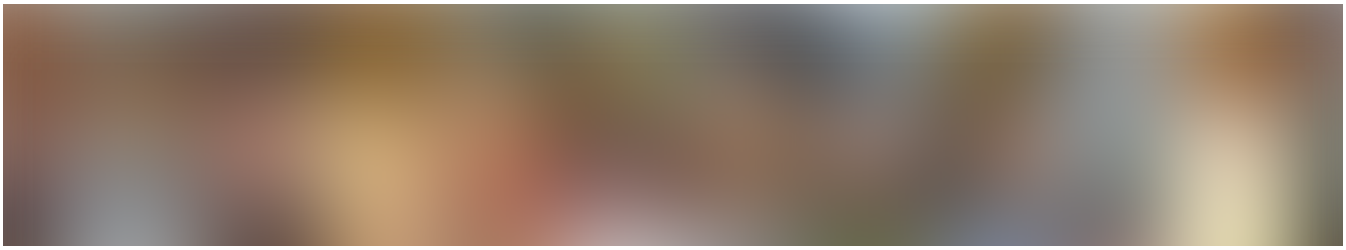
Landscape:

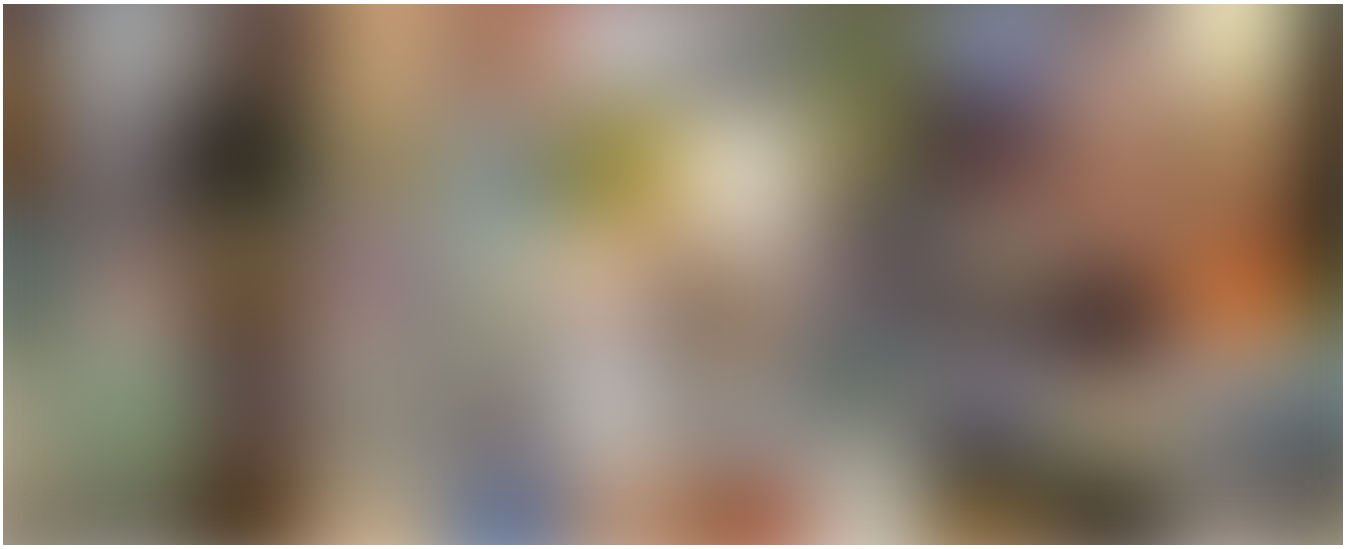


Marina:



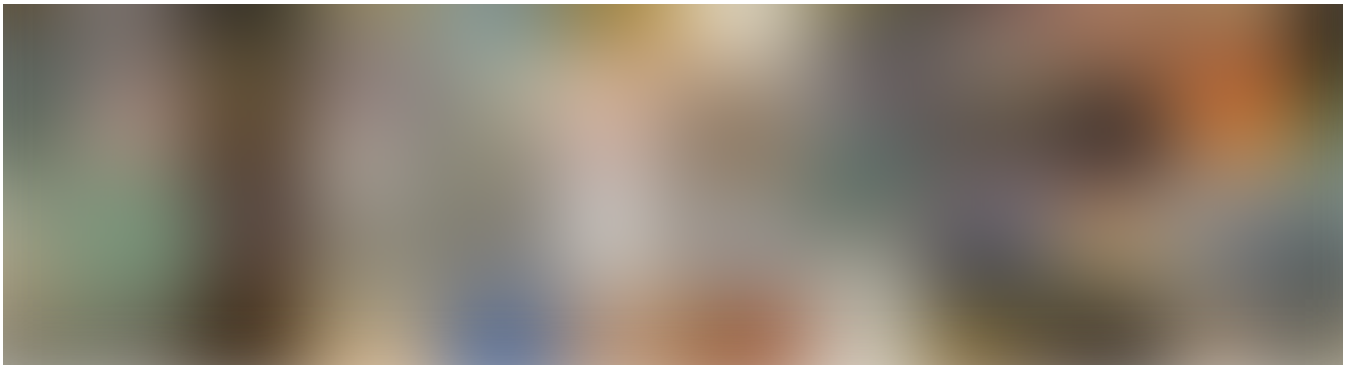
Mythological-Painting:



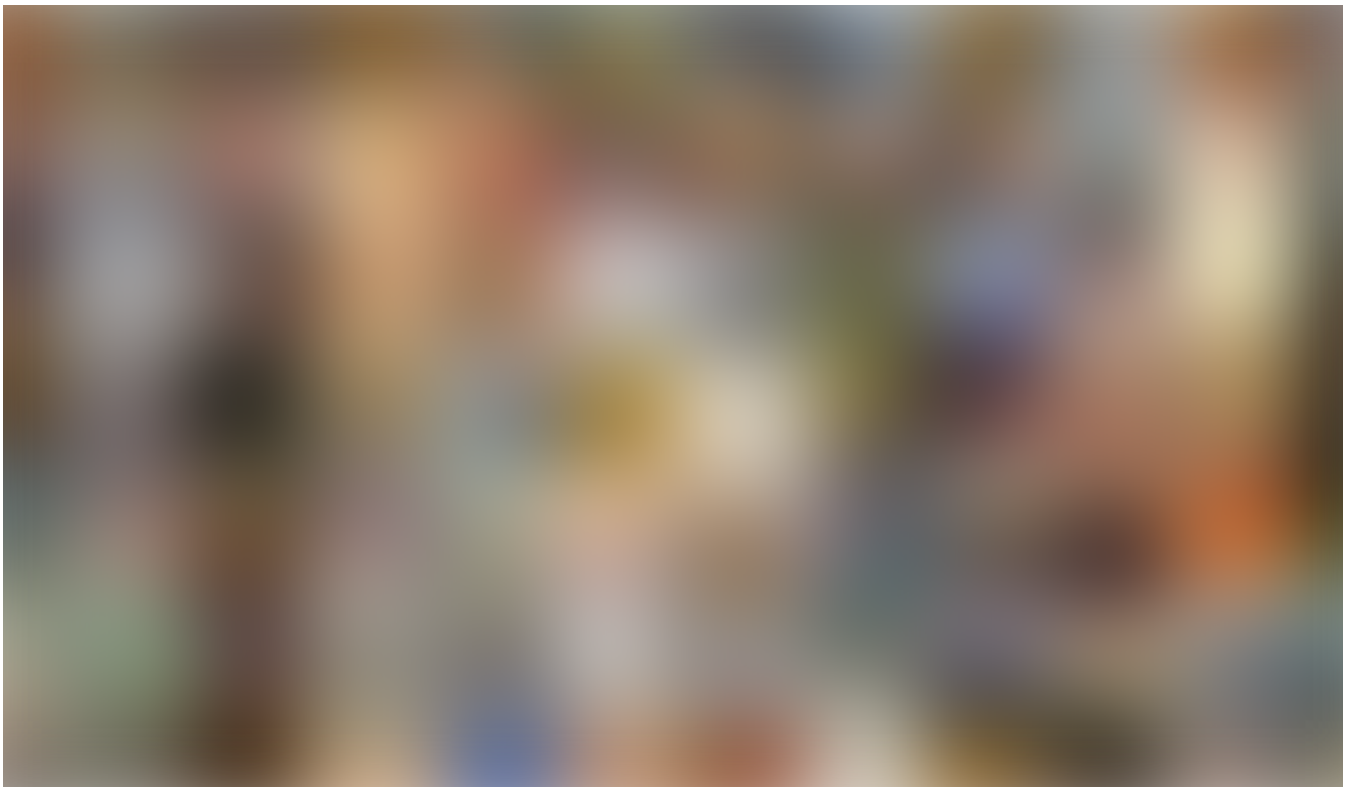


Nude-Painting:

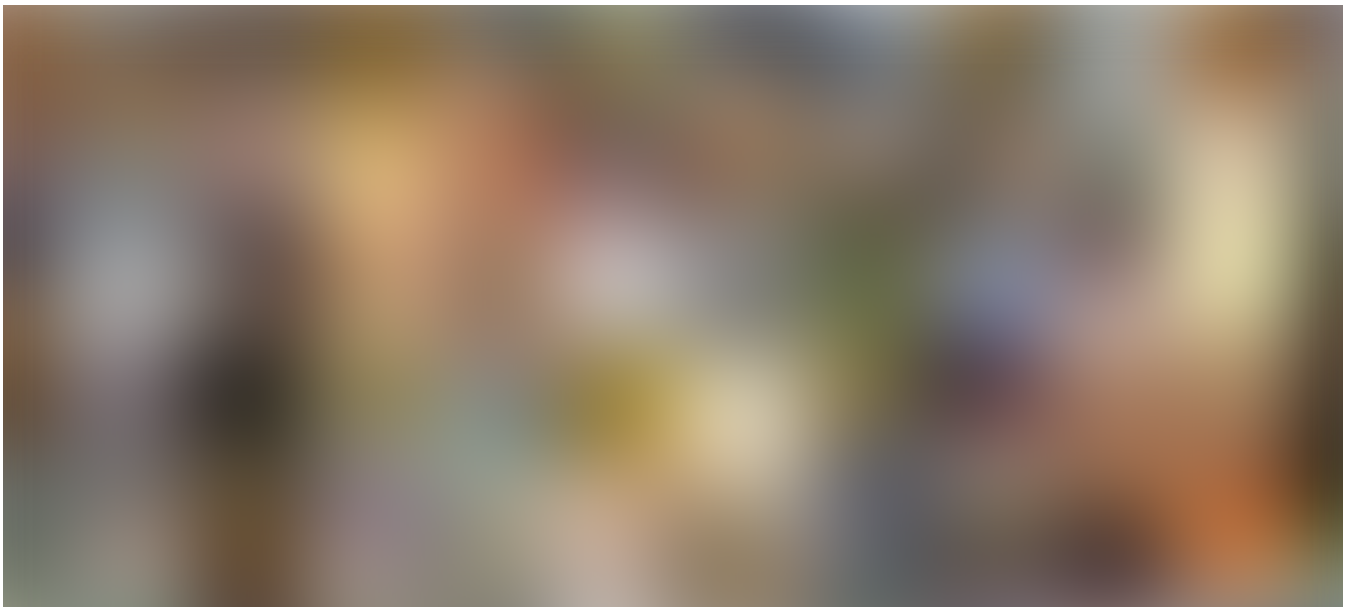
Portrait:

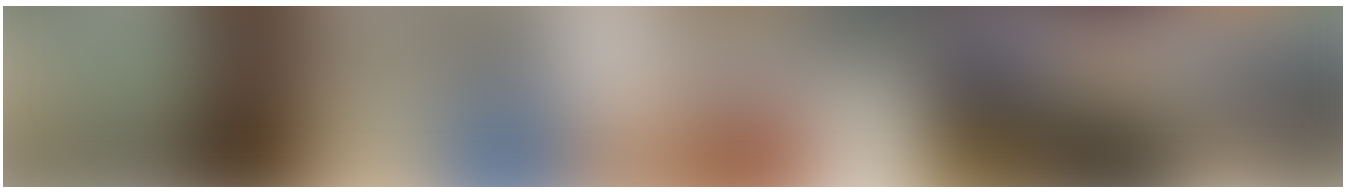


Religious-Painting:

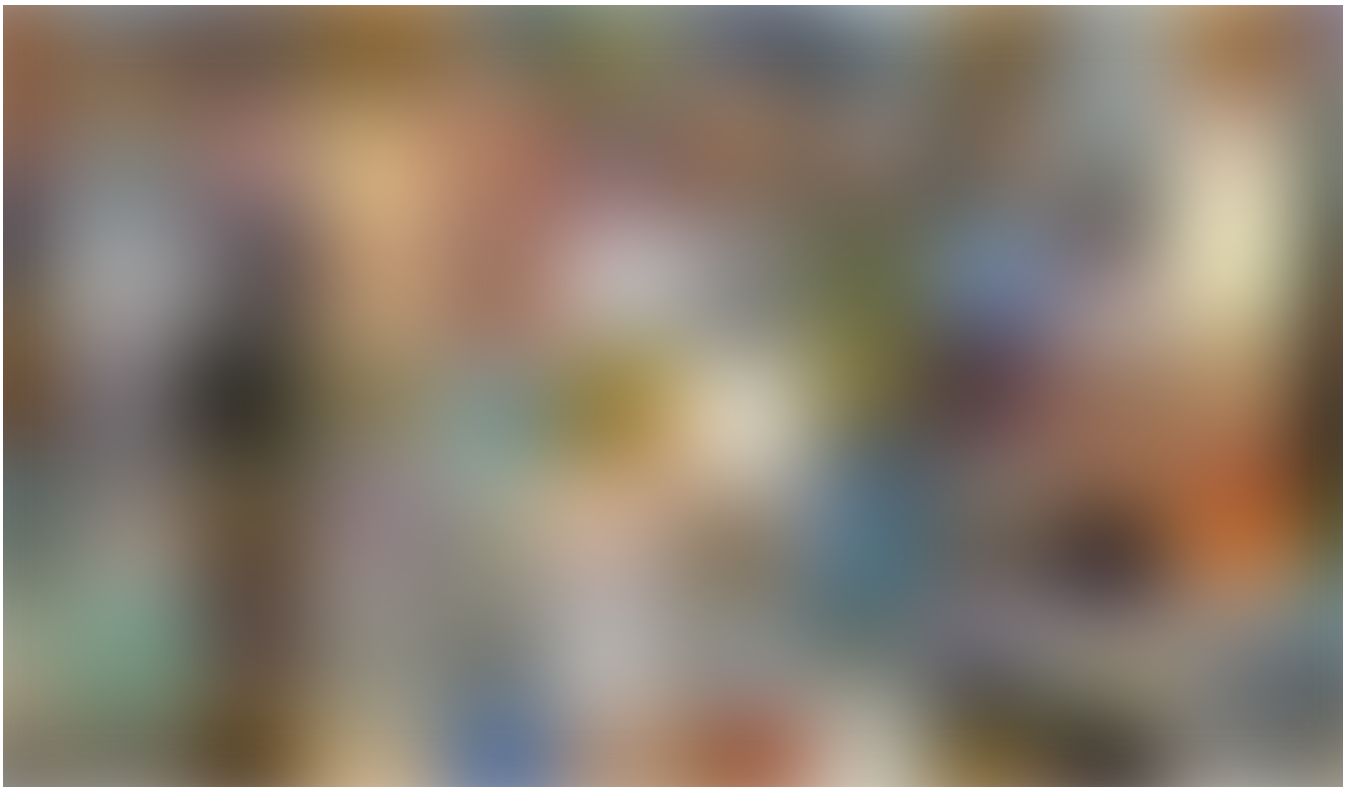


Still-Life:





Symbolic-Painting:



Machine Learning Generative Art Gans Towards Data Science Art