



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

IJIRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 11, Issue 8, August 2022

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.118

9940 572 462

6381 907 438

ijirset@gmail.com

www.ijirset.com



A Deep Convolutional Generative Adversarial Network based Framework for Creation of Images

Akhila Sahiti Racherla, Vandana Bhattacharjee

Department of Computer Science and Engineering, Birla Institute of Technology, Mesra
Ranchi, Jharkhand, India

Department of Computer Science and Engineering, Birla Institute of Technology, Mesra
Ranchi, Jharkhand, India

ABSTRACT: In the last few years, Generative Adversarial Networks (GANs) have geared up and attained great recognition for various applications including creation of images when trained on bulky datasets. This research work includes the collaboration of deep convolutional neural networks with Generative Adversarial Networks. In Generative Adversarial Networks, feed-forward neural networks used usually contain a single hidden layer. With the use of convolutional networks for GANS it leads to the route of DCGANs which take convolutional and convolutional-transpose layers in generator and the discriminator respectively that are then trained. The responsibility of the generator is to generate the fake samples and the discriminator gets trained with real samples to compare with the fake samples. This paper also concludes that with each increase in epochs, the clarity and the development of image dataset improves.

KEYWORDS: Generative Adversarial Networks, image generation, Deep convolutional networks

I. INTRODUCTION

The ability to generate some new matter leads to believing that Generative Adversarial Networks (GANs) are some kinds of magic for someone who is new to GANs. However, there is always a reason, logic, mathematics and modeling involved behind these models. Generative Adversarial Networks are the models in deep learning which have the ability to produce new pieces of information or content. These neural networks generate images, texts, music, etc. The first key point is that there exists a probability distribution which describes the kind of data that needs to be generated. For this generation process to happen from the target distribution, an inverse transformation method is used. Once the training is done, the generative network takes points that follow a simple distribution as input and turns them into the points following the target distribution.

The second key factor is related to the adversarial training that determines the learning of the generator. It defines how the generator learns the transformation of a simple distribution into the correct target done by the function. A discriminative network is trained to take true data as input to classify them. In other words, during the training process of generative networks, the target and the generated distributions are not compared directly. Both the networks, generator and the discriminator, have opposite goals. On one hand, the role of the generator always tries to produce fake data that resembles the true data. On the other hand, the role of the discriminator is to classify the data as accurately as possible.

Generative Adversarial Networks are trained by the use of gradient descent techniques and methods that are structured to get a cost function that is of low value.

Generative Adversarial Networks consist of two models which are termed as generator and discriminator. These two compete with each other to analyse the variations of the required dataset input. These two models are termed as Generator and Discriminator and are symbolized as Generator $G(x)$ and Discriminator $D(x)$. The generator creates some fake data by incorporating feedback from the discriminator. The discriminator in GAN is considered to be a classifier that identifies original data from fake data which is built by the generator. The generator learns the distribution of data and is trained to increase the probability of the discriminator network to make mistakes.

There are some types of Generative Adversarial Networks such as-

- 1) Vanilla GANs which is considered to be the simplest kind of GAN where the Generator and the Discriminator both are simple multi-layer perceptron.
- 2) There is another such type of Generative Adversarial Network named as Deep Convolutional GAN that comprises ConvNets. These are considered to be more stable and produce higher quality images.
- 3) Conditional GANs (CGANs) use extra label information to generate better results.
- 4) Super Resolution Generative Adversarial Networks (SRGANs): It produces the output which is photorealistic high-resolution image when given a low-resolution image.

When it comes to conditional GANs with respect to image- to-image translation, that includes transformation from day to night, the discriminator is fed with real night-time images as well as generated (conditioned on) day-time images as input. Conditional GANs as indicated by Goodfellow (The Father of GANs) presents:

1. Image Super-Resolution: Producing high resolution versions of the given input sample images.
2. Creating Art: Creating artistic, images, paintings, sketches, etc.
3. Image-to-Image Translation: Transforming images in a particular domain, like day to night, summer to winter, etc.

As this paper is based on DCGANs, DCGANs are not only used for generating images that are out of the scope of human perspective imagination but DCGANs are also for generating faces of anime, cartoon characters. These GANs can also be trained on the images of humans to generate realistic faces. They are also used for text to image translation, that is, it can be used for building realistic images from textual descriptions of simple objects like animals, birds, buildings. Generative Adversarial Networks can also generate 3-dimensional models using 2-dimensional pictures of objects from multiple perspectives. These Generative Adversarial Networks can be used in image-to-image translation or text-to-image translation. Further insight about DCGANs is covered in the later part of this paper.

II. RELATED WORK

The section showcases about the literature review related to generative adversarial networks, in particular, focusing mostly on creation of images, applications of GANs, innovative techniques to improve the training involving GANs and various types of GANs that evolved around.

Goodfellow in [1] presented a new framework in which two models namely generator and discriminator are trained simultaneously. A generative model G takes up the data distribution whereas another one that is the discriminative model D works on the estimation of the probability of a sample that is taken from the training data rather than a generative model. In other words, it is the estimation process that happens for generative models via an adversarial process. The entire system is trained using backpropagation. Goodfellow has also mentioned about the disadvantage of this framework, that is, there is no explicit representation and that D must be well synchronized with G during training and also G cannot be too trained.

In this paper [2], the authors presented a hierarchical model to show the learning done on image decompositions. The model that was built was trained on various natural images. The images for training were taken from Caltech-101 and 256 datasets were used by the model. The pre-processing done on each image included the conversion of each image to a Gray-scaled image which was resized to 150 x 150. The model architecture used 4 layers with 7 x 7 filters and the features extracted from this outperform SIFT and other representations of feature learning approaches.

The authors of the paper in [3] introduced the conditional version for generative adversarial networks. This was built by feeding the data to a condition and this condition was applied to both the nets, that is, the generator as well as the discriminator. This extra information, y , is considered to be an auxiliary information, for example class labels or data from other modalities. The model that generated MNIST digits was conditioned on class labels. The work was done for unimodal as well as multimodal.

In paper [4], the authors have shown new architectural features, several methods and procedures for training that could be applied to the GANs framework. The primary goal did not focus on common implemented techniques rather the techniques focused more on achieving state-of-the-art results in semi-supervised classification. This confirmation was done by a visual Turing test. The author pointed out one of the main failure modes for GAN, that is, the generator could not meet up with the parameter setting where it emits the same point.

Though it is evidently seen that supervised learning with convolutional networks (CNNs) has gained much popularity in computer vision applications [5]. On the other hand, unsupervised learning with CNNs is less visible comparatively. The gap between these two was bridged by the authors in this paper. The training was done on a variety of datasets of images and the experiments have shown that a hierarchy of representations have been learnt by the deep convolutional

adversarial pair. It has been noted that when the models are trained for a longer duration of time, they collapse a subset of filters to a single oscillating mode which has been put in future work.

The authors of paper [6] have proposed an alternative architecture for generative adversarial networks. The new architecture is automated learning and unsupervised separation of high-level attributes. It also leads to much better interpolation properties, along with better disentanglement. To quantify these two, interpolation and disentanglement, two new automated methods are proposed. And finally, they introduce a new highly varied and a dataset of high-quality human faces. The properties of the style-based generator discussed in this paper includes style mixing where a given percentage of images are generated with the use of two random latent codes instead of one during training. Switching from one latent code to another is termed as style mixing.

Class of vision consists of image-to-image and also the graphic problem's goal is to learning the mapping of an input image with an output image which is possible using the training set of image pairs [7]. But this cannot be the scenario in each and every task. Therefore, the authors of this paper have come up with an approach to learn the conversion of an image from X , that is a source to a target, Y_n image from domain X to domain Y . It is represented as a mapping $G : X \rightarrow Y$.

This paper [8], covers one of the challenging problems in computer vision. The authors synthesize high-quality images from text descriptions which have many practical applications. The authors have proposed Stacked Generative Adversarial Networks (Stack GAN). The model generates 256×256 photo-realistic images that are based on some condition given some text descriptions. A sketch-refinement technique is used for the decomposition of the given problem into sub-problems. In the Stage-I, the role of the GAN is to sketch the shapes and colours of the provided object with respect to the description of the text. In Stage-II, GAN uses the results received from the Stage-I as well as it uses the text descriptions as inputs followed by generating images with high-resolution and photo-realistic details. This stage rectifies faults obtained in results of Stage-I and then adds other specific details using refinement process.

In [9], the conditional adversarial networks are considered to be a solution to image to image translation. The learning of the networks is dependent on the mapping of an input image with an output image which is possible using the training set of image pairs along with the learning of a loss function which is used while training of this mapping. The way GANs learn a generative model of data, in the similar way, conditional GANs (cGANs) learn a conditional generative model. The approach demonstrated in this paper depicts that this technique is very effective and informative at reconstructing objects from edge maps, synthesizing photos from label maps and colorizing images, among other tasks. These conditional adversarial networks are really considered to be promising with images specially that involve highly structured graphical-outputs.

A new training methodology has been discussed for generative adversarial networks [10]. In this paper, both the generator and discriminator grow progressively. The layers are added to that model in an increasing manner with fine and specific details as the training progresses. This adds upon in increasing the speed of the training and also this helps in stabilization and producing the images of unprecedented quality. A simple way has been implemented to increase the variation in generated images. CIFAR10 dataset has been used. A new metric method is also discussed in this paper by the authors for the evaluation process of the GAN results. This has been done with respect to the quality of the image as well as the variation. Also, a higher and better quality of the CELEBA dataset has also been mentioned as an additional contribution.

Though the progress in generative image modelling has successfully generated high-resolution images but as there is diversification in samples that are collected from complex datasets (for example ImageNet) it turns out yet to be an elusive goal [11]. In this, the authors train GANs (Generative Adversarial Networks) at a bigger scale to study the instabilities in particular to the given scale. The orthogonal regularization technique has been applied to the generator. The dataset used is ImageNet dataset. The training was done on this dataset at 128×128 resolution. They have tried to improve the scalability and modify the regularization to improve conditioning, boosting performance.

It has been discussed in this paper [12] by the authors that few of the discovered models that are trained on some images fail to generalize to real images. To overcome this, a work that has already been mentioned previously in some of the research related to unsupervised domain adaptation algorithms has been discussed that maps the representations between the two domains that is, in other words, learn of extraction of the features that are considered to be domain-invariant. With respect to this, the authors presented a work in which a new approach learns in a manner that is unsupervised. The presented generative adversarial network takes up the source-domain sample images to make it look as if the images are taken from the target domains.

In [13], the model generated by the authors presents the discriminator as an energy function in this Energy-based Generative Adversarial Network (EBGAN). This releases low energies to the regions of the data manifold and attributes higher energies to some other specific regions. The way a generator in probabilistic Generative Adversarial Networks is trained to generate samples with minimal energies and on the other hand, the discriminator is trained to



produce high energies to these generated samples. This has shown that the form and way in which the EBGAN exhibits this paper has gained and resulted in more stability than those of the regular ones during the training process in GANs. Researchers have also shown high resolution generated images. The stability of this framework has been shown on MNIST digit generation in addition to a fully connected network.

The synthesis of realistic images automatically from text is one of the primary goals in the field of current Artificial Intelligence. It has also been mentioned in the paper [14] by the researchers that recurrent neural network architectures are being developed. This is useful for learning the discriminative text feature representations. On the other hand, deep convolutional generative adversarial networks have depicted compelling images of various areas, for example, faces, room interiors, album covers, etc. The researchers have therefore tried to bridge the gap between these two arenas, that is, bridging between text and image modeling (also, translation of visual concepts from characters to pixels). This is done with the help of deep architecture Generative Adversarial Networks formulation. In this paper, they have shown their model by generating images of birds and flowers through meticulous detailed text description.

In [15], Generative Adversarial Networks are proven evidently to be successful when dealing with Unsupervised learning. Thus, it has been observed that there is a possibility of loss function leading to vanishing gradient problems during the learning process. The paper is focused on dealing with this issue. They have proposed a framework named Least Squares Generative Adversarial Networks (LS-GANs) which deals with the least square's loss function. Researchers have shown that the Pearson χ^2 divergence can be minimized by minimizing the objective function of LS-GAN. When comparing the LS-GANs with the regular ones, it has been shown that the LS-GANs generate a higher quality of images. This has been done by performing experiments by designing two network architectures. In these two architectures, the first one is based on generation of image with 112×112 resolution. The other network architecture is for tasks which contain a lot of classes.

In paper [16], Markovian Generative Adversarial Networks (MGANs) has been showcased. This is a technique which is used for the training of generative neural networks for synthesis of texture. As it is known that deep neural network approaches present results of synthesis quality yet led to computational costs for low resolution images. This paper describes the above-mentioned efficiency issue. Precomputation of feed-forward, stride convolutional network that grabs the feature statistics of Markovian patches then followed by generating results of any arbitrary dimensions. This mentioned network can decode from photo to artistic sketched or paintings. Using the adversarial training, the quality obtained can be compared with that of those obtained from neural texture synthesis methods. In this, no optimization is needed any longer during the time-generation which hence leads to an increase in the run time performance, that is, 0.25M pixel images at 25Hz. This has been compared and concluded that it surpasses the previously mentioned neural texture synthesizers (500 times faster). This strategy is applied to style transfer, texture synthesis, and video stylization. The model is trained by using 100 selected images that were randomly selected from the ImageNet dataset.

The researchers of this paper [17], focused on learning a joint distribution of multi-domain images using coupled generative adversarial networks (CoGAN). Unlike other approaches of generative adversarial networks which have a requirement of image sample tuples in various domains during the process of training for training set, CoGAN learns a joint distribution regardless of need of tuples of corresponding image samples. This is done using a constraint of weight sharing that restricts the network capacity. This CoGAN has been applied to various joint distribution learning tasks that includes such as joint distribution of color and depth images, facial images samples with different attributes, etc. And for each and every task, it learns the joint distribution regardless of any need of tuples of the corresponding images. The authors have also demonstrated and extended its application to image transformation and domain adaptation.

The authors of paper [18] have built up a model for depicting the demanding task of estimating an image with high resolution given its low-resolution is considered as super resolution (SR). Though the recent work succeeded in decreasing the mean squared reconstruction error but they lack high-frequency details. Therefore, to deal with this shortcoming, they have come up with a GAN for image super-resolution. It is considered to be the first framework by the researchers that is capable of producing photo-realistic natural sample images for $4\times$ upscaling factors. It has been done using a loss function that has an adversarial loss and a content loss. A discriminator network has been used to deal with the adversarial loss which is trained to distinguish between the highly resolved images and original photo realistic images. Using Super Resolution Generative Adversarial Network (SRGAN), a Mean Opinion Score (MOS) test has been conducted that shows huge significant gains. These scores that were obtained through Mean Opinion Score are considered to be closer to those of the original super-resolution images in comparison to the scores obtained from other methods.

In [19], a cascade of convolutional networks is used to produce a high quality sample of natural images. This is done within a Laplacian pyramid framework. This generative parametric model produces in a manner of coarse-to-fine fashion while generating images. With the use of Generative Adversarial Nets (GAN) approach, a separate generative convnet model is trained at each level of the pyramid. With experiments, it has been shown that the samples drawn



from this model when compared to samples drawn from some other existing approaches, differ significantly in terms of quality (the discussed model presenting a higher quality image). Also, in the qualitative assessment, the human evaluators assumed the CIFAR-10 image samples generated by this model to be the real images and this happened around 40% of the time. They also show samples from models that used LSUN scene dataset while training to obtain higher resolution images.

III. OVERVIEW OF DEEP CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORKS

In this, the generator network competes against an adversary. It is the job of the discriminator network to differentiate between the samples obtained by the generator network and the samples obtained from the training data.

The generator model uses an input which is a fixed-length random vector to generate a sample in the domain. This vector is used as a generative process. This vector space is known to be the latent space consisting of latent variables (also called as hidden variables). These hidden variables or latent variables are the variables that are not directly observable. In Deep convolutional Generative adversarial networks, the new points taken out of this latent space can be feeded as input and also the generator model applies meaning to points chosen in latent space.

In the Discriminator Model, the sample (real or generated) is taken as input from the domain and predicts a binary class of real or generated (Here the original/real samples are obtained from the training dataset and the generated samples are output obtained from the generator model).

A good quality image can be represented using DC- Generative Adversarial Networks (DCGANs) training and later using the generator and discriminator networks parts again as feature extractors.

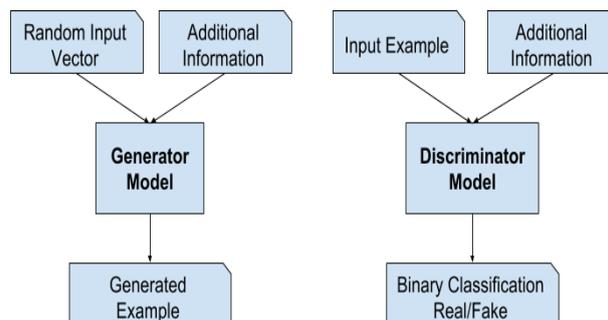


Fig. 1. Generator and Discriminator model

The generator can be again repurposed as it will be helpful to learn the extracted features from the problem domain examples. A few features extraction layers can be reused in transfer learning applications with the help of similar kind of input data.

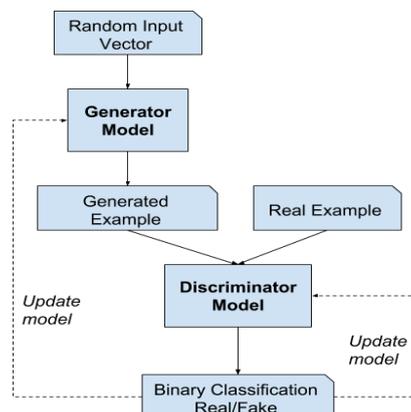


Fig. 2. Flow of the network



GANs with Deep Convolutional Neural Networks (DCGANs) are basically used for image Data and utilize Convolutional Neural Networks in the form of models of generator and discriminator. Usually, constructing image data means that the input to the generator feeds a compressed representation of the image datasets which get used for training purposes. As the generator presents newly build images which then can easily be seen and analysed by the users of the model, this has led the focus towards applications of Convolutional Neural Networks (CNNs). So basically, the networks evolve over the iteration. So, every step is an iteration. The input of the signal is a random signal into our generator. This is done as it needs to start to generate some images. In the beginning the generated images are completely random images. Both the generator as well as the discriminator are not trained in the very beginning.

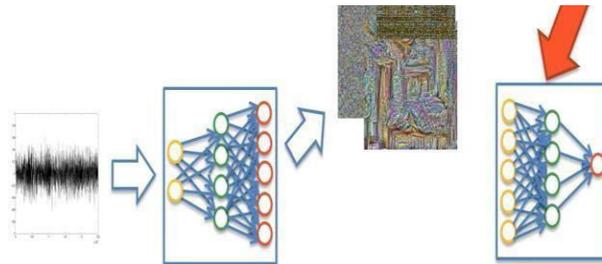


Fig. 3. Flow of the network

While training the discriminator, it is feeded with the random images that are generated by the generator and along with these randomly generated images, the discriminator is feeded with the real images. For instance, the role of a GAN is generating images of a dog. So, in this case, the generator generates random images which may or may not be with respect to an image of a dog. And on the other hand, the discriminator is feeded with the random images constructed by the generator along with images of dogs. Following this, after every epoch, the probability of each iteration result is noted. Generally, in the first epoch, it is a 0% likelihood. So basically, in every single case, the cumulative error will be calculated and will back propagate through the network of the discriminator. Hence, with each iteration during the training, the discriminator has learned which is actually the learning process and learns from the errors and next time it improves itself.

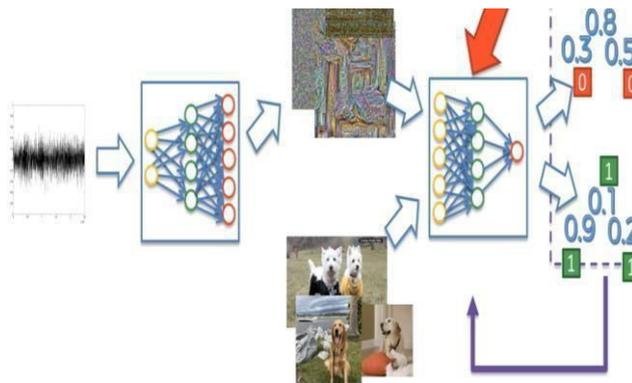


Fig. 4. Back propagation through the network while training phase

Now while training the generator, the generator part learns to build fake data by feeding whatever feedback it receives from the discriminator. Generator training needs a tighter integration of the generator as well as the discriminator as compared with that of the discriminator training requires. Training for both the generator as well the discriminator is done together and, in each iteration, both tend to improve themselves.

In the upcoming section, we have shown the procedure, that is, the methodology of the work. The steps of the working of a deep convolutional generative adversarial network model are also mentioned in the next section.

IV. METHODOLOGY

In this section the procedure of this work is presented briefly.

In the first step the data has been collected from one of the most widely used dataset which consists of over 60,000 images. Since these DCGANs require to have an input which persists heavy dataset, the dataset should include all kinds of images (so that the neural networks are feeded with all varieties of images) and also for training and generating the image, the neural networks need a huge bulk of images and one of the bulky datasets that covers a variety of images is CIFAR-10 dataset. It contains all sets of images which are 32X 32 in size, and consists of 10 classes of 6000 images per class (classes such as airplane, bird, dog, automobile, deer, cat, etc). These images are colored images which are batched into 1000 randomly selected images from each of the present class. After loading the data, a neural network, say m, will initialize all its weights.

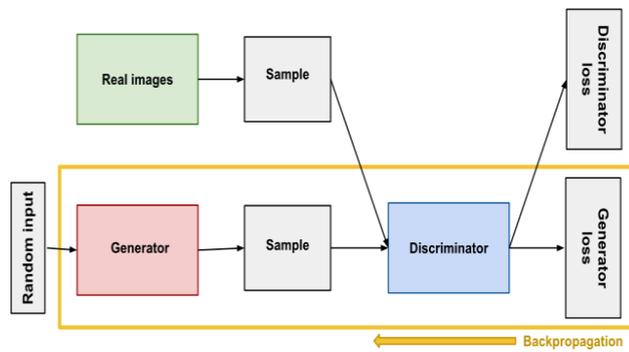


Fig. 5. Flow of the entire process with respect to generator loss

The next step includes defining the architecture of the generator. In this, a meta module of a neural network has been created that contains a sequence of modules (convolutions, full connections, etc.). The building generator neural network starts with an inverse convolution, which means the inputs are vectors and the outputs are images which is followed by normalization and Relu rectification respectively. Also, Tanhrectification has been used to drop the linearity and stay between -1 and +1.

In discriminator, unlike the building of a neural network for a generator, the neural network consists of a convolution, that is, inputs are images and on the other hand outputs obtained are vectors. Here LeakyRelu has been used as a discriminator and also sigmoid rectification has been applied to break the linearity.

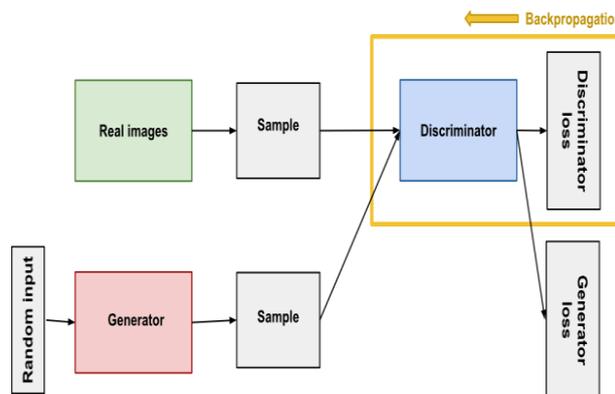


Fig. 6. Flow of the entire process with respect to Discriminator loss

For training the DCGANs, it includes several steps. The first step is to update the weights of the neural network of the discriminator. Followed by training the discriminator with a real image of the dataset which is then forward propagated into the neural network of the discriminator to get the prediction (a value between 0 and 1). Then the loss is computed for real images using a criterion object(errD_real). Then the discriminator is trained with a fake sample image generated

by the generator which is forward propagated into the neural network of the generator to get some fake generated images and prediction (a value between 0 and 1) respectively. Again, the loss is computed and noted but this time the loss is computed from fake images (errD_fake). Both the computed errors are added for backpropagation and optimization of the discriminator. The loss function used overhere is Binary cross entropy or BCE loss function:

$$L(\hat{y}, y) = [y \cdot \log \hat{y} + (1 - y) \cdot \log(1 - \hat{y})]$$

Original data
Reconstructed data

Now, as the objective of the discriminator is to properly present the fake and real dataset, it should be maximized whereas on the other hand, the generator is competing against the discriminator, it will try to minimize. Hence, the loss function of generator and discriminator could then be written as:

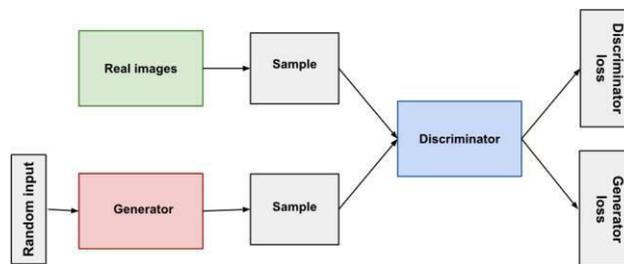
$$L^{(D)} = \max[\log(D(x)) + \log(1 - D(G(z)))]$$

$$L^{(G)} = \min[\log(D(x)) + \log(1 - D(G(z)))]$$

Combining both the equations, it could be written as:

$$L = \min_G \max_D [\log(D(x)) + \log(1 - D(G(z)))]$$

Similar steps have been followed for generators as well. The only difference is that it is done only for fake generated images which are feeded into the neural network of the discriminator to get the prediction. Once the training has been done, the result is stored as images into a created folder.



Briefly the generalized steps mentioned are as follows:

- 1) Initializing the weights that are applied while building the architecture.
- 2) Choose the architecture for generator and discriminator which consists of a meta module of a neural network that has a sequence of modules, that is, the architecture includes Sequential meta model of convolutional, full connections in the sequence of convTranspose2d, batch normalization, Relu for building up the network of generator and conv2d, leakyRelu followed by batch normalization for building up the network of discriminator.
- 3) Training the discriminator with a real image of the dataset.
- 4) Training the discriminator with a fake sample image generated by the generator.
- 5) Backpropagating the total error.



- 6) Updating the weights of the neural network of the generator.
- 7) Saving the real images and the generated images of the minibatch every 100 steps.

V. EXPERIMENTS

The training was done for 24 epochs and the below images show the output achieved after each epoch. To generate these images, it took overnight to do the training.

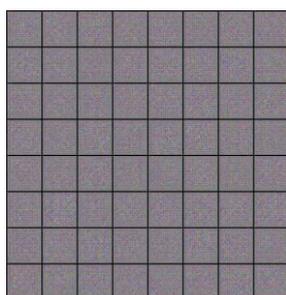


Fig 8. Fake sample (epoch 0)



Fig 9. Fake sample (epoch:1)



Fig 10. Fake sample (epoch:2)



Fig 11. Fake sample (epoch:3)

VI. RESULTS

From the images obtained, it can be seen that the first image is completely a random image. In the initial epochs, the images obtained are blurry images yet the objects are visible to some extent. As the epochs continue, detailing and specifications can be observed in the images. Though, finding the difference between the adjacent images might be difficult in each epoch, a specific detailing modification take place. As the epoch reaches to 15th or 20th epoch, more detailing and clarity in images as well as objects present in the images could be seen.

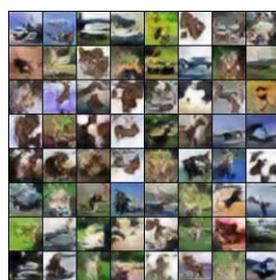


Fig 12. Fake sample (epoch:4)



Fig 13. Fake sample (epoch:5)



Fig 14. Fake sample (epoch:6) Fig 15. Fake sample (epoch:7)

The process of generation of these fake samples continues for next epochs as well. The below Figures that show the generated images are fake samples of epochs of later training. The Fig. 18 presents the real sample images which are used.



Fig 16. Fake sample (epoch:23) Fig 17. Fake sample (epoch:24)



Fig 18: Real sample

TABLE 2

epoch/25	iteration / length (data loader)	Loss of Discriminator (Loss_D)	Loss of Generator (Loss_G)
0/25	0/782	1.3426	7.8215
0/25	1/782	1.0275	4.8336
0/25	2/782	1.1784	6.4559
⋮	⋮	⋮	⋮
0/25	781/782	1.1784	6.4559



TABLE 3

epoch/25	iteration / length (data loader)	Loss of Discriminator (Loss_D)	Loss of Generator (Loss_G)
.	.	.	.
.	.	.	.
.	.	.	.
24/25	777/782	0.0355	4.8818
24/25	778/782	0.4841	4.3262
24/25	779/782	0.1730	6.5592
24/25	780/782	0.1336	5.5777
24/25	781/782	0.0849	5.0557

The Table 2 shows the loss of generator and discriminator in each epoch run based on the batch size (formed by randomlyclubbing the images) which is indicated as data loader in the table. As it can be seen that the loss of the generator is way higher than the loss of the discriminator which is obviously due to the creation of new images. On the other hand, the discriminator loss lies in between 0.3965 and 1.1784. In the similar way, Table 3 shows the last epoch details.

VII. CONCLUSION AND FUTURE WORK

The use of Deep convolutional generative adversarial Networks is becoming popular at present. The generation of images using deep convolutional neural networks enables a transparent solution based on the needs of the given problem domain. The formation of these networks with images has widespread use all across. Generating images with the help of many convolutional layers along with generators and discriminators where the images that are created are beyond the imagination of human perspective. Image creation itself is a huge task and is rapidly taking a huge interest in people. The increase in the resolution of images is one of the demanding fields these days. This is because any viewer expects the image to have clarity, specificity and detailing. Some of the on-demanding technologies which serve as an add-on to the advancement of image generation are Machine Learning, Computer Vision, Deep Learning, etc. The extension of this work could be towards a domain such as video creation. Video creation would include the frame prediction concept. A network architecture could be presented based on time points for frame generation. Each adjacent frame should have similarities because then only the video generated would be proper and would contain some meaning. Another possible extension could be text description where we aim to make a more robust, efficient and highly potential tool. This can be an add-on to the above-mentioned idea of future work.

REFERENCES

1. Ian J. Goodfellow, Jean Pouget-Abadie* , Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair† , Aaron Courville, Yoshua Bengio, Generative Adversarial Nets.
2. Matthew D. Zeiler, Graham W. Taylor and Rob Fergus Dept. of Computer Science, Courant Institute, New York University, Adaptive Deconvolutional Networks for Mid and High Level Feature Learning.
3. Mehdi Mirza, Conditional Generative Adversarial Nets, arXiv:1411.1784v1 [cs.LG] 6 Nov 2014
4. Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, arXiv:1606.03498v1 [cs.LG] 10 Jun 2016, Improved Techniques for Training GANs.
5. Alec Radford & Luke Metz indico Research Boston, MA, Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv:1511.06434v2 [cs.LG] 7 Jan 2016.
6. Tero Karras, Samuli Laine, Timo Aila NVIDIA, A Style-Based Generator Architecture for Generative Adversarial Networks, arXiv:1812.04948v3 [cs.NE] 29 Mar 2019.



7. Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros, Berkeley AI Research (BAIR) Laboratory, UC Berkeley, Image-to-Image Translation with Conditional Adversarial Networks, arXiv:1703.10593v7 [cs.CV] 24 Aug,2020.
8. Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, XiaoLei Huang, Dimitris Metaxas; Rutgers University, Lehigh University, The Chinese University of Hong Kong, Baidu Research, StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks.
9. Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros, Berkeley AI Research (BAIR) Laboratory, UC Berkeley, Image-to-Image Translation with Conditional Adversarial Networks arXiv:1611.07004v3 [cs.CV] 26 Nov 2018.
10. Tero Karras, Samuli Laine, Timo Aila NVIDIA, arXiv:1710.10196v3 [cs.NE] 26 Feb 2018, Progressive growing Of GANs for improved Quality, Stability and Variation.
11. Andrew Brock, Heriot-Watt University, Jeff Donahue, Karen Simonyan† DeepMind, Large Scale GAN Training For High Fidelity Natural Image Synthesis, arXiv:1809.11096v2 [cs.LG] 25 Feb 2019.
12. S.Kannadhasan and R.Nagarajan, Development of an H-Shaped Antenna with FR4 for 1-10GHz Wireless Communications, Textile Research Journal, DOI: 10.1177/00405175211003167, March 21, 2021, Volume 91, Issue 15-16, August 2021
13. Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, Dilip Krishnan, Unsupervised Pixel-Level Domain Adaptation with Generative Adversarial Networks arXiv:1612.05424v2 [cs.CV] 23 Aug 2017.
14. Junbo Zhao, Michael Mathieu and Yann LeCun, Department of Computer Science, New York University Facebook Artificial Intelligence Research, Energy-Based Generative Adversarial Networks, arXiv:1609.03126v4 [cs.LG] 6 Mar 2017.
15. Scott Reed, Zeynep Akata, Xinchun Yan, Lajanugen Logeswaran REEDSCOT, AKATA, Bernt Schiele, Honglak Lee, Generative Adversarial Text to Image Synthesis, arXiv:1605.05396v2 [cs.NE] 5 Jun 2016.
16. Xudong Mao, Qing Li, Haoran Xie, Raymond Y.K. Lau, Zhen Wang and Stephen Paul Smolley , Least Squares Generative Adversarial Networks, arXiv:1611.04076v3 [cs.CV] 5 Apr 2017.
17. Chuan Li and Michael Wand Institut for Informatik, University of Mainz, Germany, arXiv:1604.04382v1 [cs.CV] 15 Apr 2016, Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Network.
18. Ming-Yu Liu, Oncel Tuzel, Mitsubishi Electric Research Labs (MERL), Coupled Generative Adversarial Networks, arXiv:1606.07536v2 [cs.CV] 20 Sep 2016.
19. Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, Wenzhe Shi, Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, arXiv:1609.04802v5 [cs.CV] 25 May 2017.
20. Emily Denton, Soumith Chintala, Arthur Szlam, Rob Fergus, Dept. of Computer Science Courant Institute New York University, Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks, arXiv:1506.05751v1 [cs.CV] 18 Jun 2015
21. Kirill Eremenko, Data Scientist, Hadelin de Ponteves, AI Entrepreneur, udemy course of deep learning and computer vision
22. <https://developers.google.com/machine-learning/gan>, Google developers site



INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor: 8.118



ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com



www.ijirset.com

Scan to save the contact details