# Applications of Generative Adversarial Network-Image Completion and Image Super Resolution

**Prakruthi Manjunath[1], Pragathi M I [2], Pooja S Vel[3] , S Komala[4], Dr. Sreenivasa B C[5]**

*[1-4]Student, Dept. of Computer Science Engineering, Sir MVIT, Karnataka, India*
*[5]Associate Professor, Dept. of Computer Science Engineering, Sir MVIT, Karnataka, India*

---------------------------------------------------------------------***---------------------------------------------------------------------

**Abstract** - *Deep learning and Artificial Intelligence are the technologies proven to become more accurate over years. By using these technologies now, we can learn without any human intervention or supervision and perform tasks with the precision of man himself. GAN is an evolving technology that has achieved excellent accuracy and seems to be progressing much more over time. Its various applications include image classification, image generation, image processing or enhancement, video frame generation and much more. This paper will explore the basic structure of the model, working of the model and its application. Two applications of GAN will be explained in detail.*

***Key Words*: Generative Adversarial Network, super resolution, boundless, image-inpainting, generator, discriminator, Deep Learning**

## 1. INTRODUCTION

Generative Adversarial Network, or GAN, is a method of generative modelling using deep learning techniques and is one of the most discussed subjects nowadays. It was first designed in 2014 by Ian Goodfellow and his colleagues, and has been widely studied since then. GANs consist of coming across and mastering the similarity within the data in a manner that can generate the original dataset. The basic idea was that of two models in constant competition, meaning one's loss is another's gain. This approach leads them to produce realistic, nondifferentiable protégées of the input. For example, if we were to input a video, the output would also be in the form of a video. GAN models are trained to identify patterns or similarities from the inputs, and they can create items that are very closely related to the input. GAN has proven itself in various difficult tasks such as improving resolution, generating facial expressions, and much more. We shall look at the basic working of GAN and check out a basic algorithm of the same. We will also have a peek at two widely used GAN models that help us in altering images

## 2. BASIC STRUCTURE OF GAN

### 2.1 Basics of GAN

A Generative Adversarial Network is an approach to an algorithm of generating modelling using deep learning methods such as neural networks. It works on the basis of opposition or conflict (i.e. Adversarial) and tries to better itself by a feedback loop. A GAN model is made up of two things: a generator (G) and a discriminator (D), which are both neural networks. These two neural networks try to "Outsmart" each other and thus better themselves.

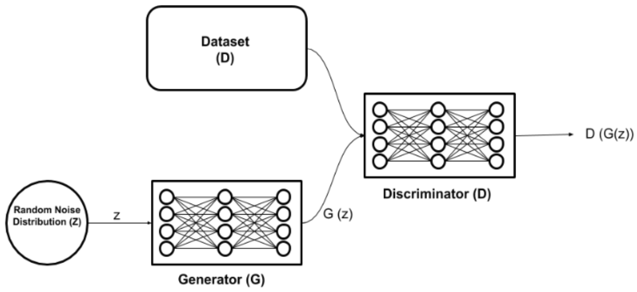


**Fig 1: A brief structure of GAN**

### A. Generator

The generator is used to create an output from a random set of inputs. The inputs are taken in the form of an array known as a **random distribution (Z).** This (i.e. random distribution), when fed into the generator, produces an output.

### B. Discriminator

There are two sets of data that are fed to the discriminator: the real and the fake. The discriminator has to differentiate between the two data. It is fed data from the generator as well as the dataset(D). The discriminator outputs a 1 (real) or a 0 (fake) and checks to see if the result is correct or not. It then proceeds to give feedback to itself and the generator and the cycle continues.

Both methods can be applied using a GAN. GANS can further be categorized into 2 sections:

### C. Supervised Translation

Here the input consists of ground-truth image pairs on which the output is based upon. The disadvantage of this technique is that the ground truth images pair datasets are difficult to create.

**D. Unsupervised Translation**

Instead of having pairs of corresponding images in the dataset, we have two datasets without any correspondence between them. We make certain assumptions that can make correspondences between different images in different domains.

Generative Adversarial Networks can be implemented in multiple leading languages such as Java and Python. GAN's can be implemented in java using the Deeplearning4java (DL4J) library. Deeplearning4java includes functions such as ActivationLayer() which applies the activation function to the neural network. Python uses libraries such as TensorFlow and PyTorch.

## 2.2 Training Model

A basic GAN model can be termed as an interaction between the generator and the discriminator. Taking this into consideration, each model has a loss function. In this particular section, we aspire to draw out the loss functions of both the generator and discriminator. Before that, there are certain terms that we need to be aware of:

z : latent vector; G (z ) : fake data; x : real data; D (x) : Discriminator's evaluation of real data; D (G (z )) : Discriminator's evolution of false data; Error (a ,b): Error between a and b.

The objective of the discriminator is to accurately name produced pictures as fake and exact information as real. Therefore, the Loss function of the discriminator can be termed as:

$$L\,D = Error\,(D\,(x),1) + Error\,(D\,(G\,(z)),0) \quad (1)$$

Here, we make use of an exceptionally nonexclusive, prevalent documentation of the error to allude to some function that foresee the distance or the differences between the two functional parameters.

The generator's objective is to trick the discriminator so that it gets confused about labelling the images as real or fake.

$$LG = Error\,(D\,(G\,(z)),1) \quad (2)$$

We have to be aware that the loss feature is to be minimized. A typical loss function that is used in binary classification issues is binary go-entropy. In category responsibilities, the random variable is discrete; therefore, the equation may be best expressed as a summation of the following:

$$H(p,q) = -\sum x \in \chi p(x) log q(x) \quad (3)$$

We are able to simplify this expression even similarly in binary cross- entropy since there are only two labels: zero and one. Binary cross entropy satisfies our aim in that it gauges how specific two appropriations are on the subject of parallel order of finding out if an input information factor is true or fake. Applying this to the loss functions shown in (1)

$$LD = -\sum x \in \chi, z \in \zeta log(D(x)) + log(1 - D(G(z))) \quad (4)$$

The same is applied for (2)

$$LG = -\sum z \in \zeta log\,(D\,(G\,(z)) \quad (5)$$

We now have two loss functions with which we can train the generator and the discriminator. The paper with the aid of Goodfellow had the equation framed as a min-max recreation wherein the discriminator seeks to maximize the given amount while the generator seeks to achieve the reverse. Mathematically,

$$\min_{G} \max_{D} log(D(x)) + log(1 - D(G(z))) \quad (6)$$

There is a need for two separate loss functions for both generator and discriminator. This is due to the fact that the gradient of function y=log(x) is found to be steeper near x than that of y=log(1-x), which implies that trying to maximize log (D (G (z ))) , or equivalently, minimizing –log (D (G (z ))) would lead to more reliable improvements to the performance of the generator than trying to minimize log (1 – D (G (z ))).

Now that we've described the loss functions for both the discriminator and the generator, we can now find the parameters for the generator and the discriminator such that the loss functions are optimized. This corresponds to training the model in sensible phrases. In GAN, the generator and discriminator are educated one after the other. The quantity of interest can be described as a function of and. Let's name this the value

$$V(G,D) = E_{x \sim p_{data}}[log(D(x))] + E_{y \sim p_g}[log(1 - D(G(z)))] \quad (7)$$

We look more into the distribution model made by the generator rather than the discriminator. Accordingly, let us rewrite the value function by substituting it with a new variable

$$V(G,D) = E_{x\sim p_{data}}[log(D(x))] + E_{y\sim p_g}[log(1-D(y))] = \int_{x\in\chi} p_{data}(x)log(D(x)) + p_g(x)log(1-D(x))dx \quad (8)$$

The task of the discriminator is to make D (x) almost equal to 1, if the input data is (x). The discriminator tries to make D (G (z )) close to 0, if the input is fake data G (z ) as seen above, whereas G tries to get it closer to 1. Training the generator,

$$V(G,D) = E_{x\sim p\ data}[log(D(x))] + E_{y\sim p_g}[log(1-D(G(y)))] \quad (9)$$

The generator must be able to learn the underlying distribution of the data from sampled training examples. Meaning, $P_{data}$ and $P_g$ should be as close to each other as possible in the range. A generator G is said to be ideal if it can mimic p so well that it can make a compelling model distribution p. The final equation keeping in mind the zero-sum game is:

$$minGmaxD\int(D,G) = nE_{x\sim p\ data}(logD(x)) + E_{y\sim p_g}(log(1-D(g(z)))) \quad (10)$$

$$minGmaxD\int(D,G) = nE_{x\sim p\ data}(logD(x)) + E_{y\sim p_g}(log(1-D(g(z))))$$

## 3. LITERATURE SURVEY

In 2018, Yang-Jie Cao et al. [1] proposed the various advancement of generative adversarial networks in computer vision. This paper elaborates over the origin of GAN and its fundamentals. The basic models of GAN and also the working of those basic models is also mentioned. There are also mentions of different models available and their applications in various fields. The architecture of a basic structure of GAN that comprises a generator and discriminator are looked at along with the mathematical representation of the loss function. The authors have also mentioned about the evolution of GAN and how far it has come with respect to its model. There is also a brief experiment with basic MNIST that is run through different GAN models and this gives us an idea of how the models differ from each other and what application it would be the best at. The key takeaways are better understanding of various GAN models along with its uses and application in Computer Vision.

In 2018, Ting-Chun Wang et al. [2] proposed a new method for synthesizing high-definition photo-sensible pics from semantic label maps with the help of conditional GAN. In this experiment, they generated 2048×1024 visually attractive results with a novel adversarial loss, in addition to new multi-scale generator and discriminator architectures. Their approach considerably outperformed current techniques in each quality and resolution of deep image synthesis.

In 2019, Xining Zhu et al. [3] proposed an image super resolution with novel quality loss model based on GAN. We explore the method of Image Super Resolution with GMGAN i.e., Gradient Map Generative Adversarial Network. This method is different than SRGAN as in GMGAN, they use an alternative method to the per-pixel loss function that is calculated during SRGAN. The model used is WGAN-DP. Instead of loss function, they have used a term called Gradient Magnitude Similarity Deviation. This is calculated slightly differently than the traditional loss function. GMSD is inversely proportional to the resolution of the image. This method is proved to be slightly more effective as compared to traditional loss function calculated using SRGAN.

In 2019, Zhaoqing Pan et al. [4] proposed a survey on the recent progress of generative adversarial network. The paper introduces us to deep generative models that form the basis of GAN, and also gives an overview about the architecture and different uses of GAN. It also tells us about the conventional GAN models and how the models were introduced with some variations to make them compatible to solve real world problems. The authors also mention the right way of training GAN models and offer methods to increase a model's stability. The applications of GAN in Natural Language Processing and also in Computer Vision is mentioned at the end. It also gives an overview of the challenges faced by GAN and the future prospects of GAN models.

In 2019, Taesung Park et al. [5] proposed spatially adaptive normalization for synthesizing photo-realistic images given an input semantic format that lets it take control over each semantic and style. They proposed that normalization layers tend to scrub away semantic facts. To address this, they used the input format for modulating the activation in normalization layers via a spatially-adaptive, learned transformation. Their approach had advantages over current approaches concerning both visible fidelity and alignment with input layouts.

In 2020, Yi Jiang et al. [6] proposed image inpainting using Wasserstein GAN loss function. CelebA dataset is used to test the model. It has an autoencoder with two discriminators and one generator. Adversarial loss functions are used that are similar to boundless GAN. This model did not acquire the expected results and thus, the paper considers this a failure. It failed to capture the neighbor-pixels to get appropriate results. The model mostly generated unaligned images as compared to the original images that were compared with. The model is almost very similar to Boundless GAN. Skip- connection was used in the generator in hopes of better results. This particular method is not mirrored in the Boundless GAN model which we have considered and that model has given satisfactory result In 2020, Yun Wu et al. [7] suggested a model for photo super-resolution and reconstruction based on GAN. It explores the use of perceptual loss, which is well-known shows a fast convergence and fantastic visible effect, to manual the generator network modelling. Wasserstein is used to distance into the discriminator network to enhance the discrimination stability and capacity of the model. This elaborates the variations among the conventional approach of super resolution and the deep CNN. It explains why GAN has a better gain in comparison to deep CNN. It takes three forms of losses: perpetual loss, pixel-wise loss and discriminator loss. Experiments were carried out on DIV2K, Set5, Set14 and BSD100 data sets. The detailed algorithm of this approach is mentioned in the paper as well. For results, the paper compares its own method with deep CNN, bicubic and HR methods. A dual generator network method ensures complete feature extraction.

## 4. MERITS AND DEMERITS

GAN's have an upper hand compared to many similar supervised and unsupervised networks.

A. Merits

1) GAN's are a good way to classify data: The trained discriminator can be used to classify objects according to the training.
2) GAN's can be trained on minimal data: unlike other networks such as CNN's which are data-hungry.
3) GAN's are an excellent manner to classify the use of semi-supervised learning strategies: It includes a dataset that carries a minute amount of categorized data and a huge amount of uncategorized data.
4) GAN's understand the internal representation of the data: and therefore, can be used with unlabeled datasets.
5) GAN produces data indistinguishable from real data: which can be used in multiple real-world applications.

B. Demerits

1) The generator suffers from a situation known as model collapse wherein the generator lacks the ability to produce different varieties of samples.

2) If the discriminator becomes too good at its job it hinders the generator's ability to learn. Thus, the GAN cannot produce the correct output.

3) GAN's are less efficient at classifying data compared to CNN when datasets are large.

4) GAN's can be affected by non-convergence. This may be due to parameter oscillation and destabilization.

## 5. METHODOLOGY

### 5.1 Boundless GAN

Conventional image augmentation models that work on assorted datasets and safeguard significant level semantics and low-level picture structures and surfaces have wide uses in image altering, and PC illustrations. While inpainting has been widely discussed, in this paper, we found that it is trying to straightforwardly apply the best in class inpainting strategies to picture augmentation as they will, in general, produce hazy or dreary pixels with conflicting semantics. We apply semantic moulding with the Generative Adversarial Network (GAN) and accomplish extraordinary subjective and quantitative outcomes on images with cognizant semantics and also outwardly satisfying tones and surfaces in the all-inclusive locales. We likewise show promising outcomes in extraordinary augmentations, including display age.

A. Model

Wasserstein GAN framework is used as a model. that has a generator network which is trained by the assistance of discriminator, which is trained accordingly. The generator, G has information comprising of the picture z with pixel esteems in the reach of [−1, 1], which has to be expanded. and a double cover M. Both z also, M comprises an area of known pixels and unknown pixels. Rather than inpainting systems, the distant locale pixels imparts a limit to the known area on just one side where, z is set to 0 in the unknown area, and M is set to 1 in the obscure district and 0 in the known district. The yield of G(z,M) of G has similar measurements as z and also, the loss of a pixel during preparation, utilizes this full yield. Notwithstanding, the last stage prior to taking care of the discriminator D is to supplant what the generator combined in the known locales and the pixels that are known.

B. Training

Reconstruction loss and Adversarial loss are combined and then the model is trained.

$$Lrec = k\, x - G\,(z\,, M\,)k \tag{11}$$

Wasserstein GAN hinge loss is used for adversarial loss that defines the coarse prediction.

$L_{adv,D} = E_{x\text{-}p(x)}\,[Re\,LU\,(1 - D\,(x\,, M, x)) + Re\,LU\,(1\ D\,(x\,, M,$

*x*))]
(12)
Here, *ReLU* is the rectified linear unit function.

$$L_{total}=L_{loss}+\lambda L_{adv,G} \quad (13)$$

Challenge dataset. These are then separated into spatial dimensions of 257x257 pixels.

A. Output

The main objective of Boundless GAN is to complete an incomplete image. Hence the output obtained after training using the given dataset is shown in figure 2.

The above derived equation is said to be the equation for the total loss function λ is set to 10. The models are trained with the datasets of the top 50 classes of the Places365-

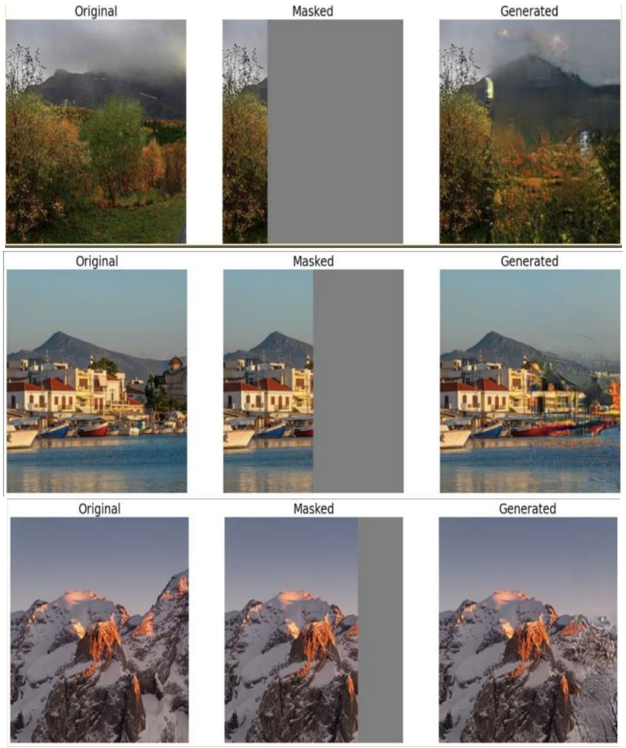


**Fig 2: Output that shows image completion for 75%, 50% and 25% of the original image.**

## 5.2 SRGAN

This design intends to recuperate better surfaces from the picture when we upscale the input image, so that quality cannot be undermined. For example, there exists several techniques such as Bilinear Interpolation, which can be utilized to play out this assignment, yet they experience the ill effects of picture data loss and smoothing. The creators proposed two structures, SRResNet (omitting GAN ) and another one as SRGAN (with inclusion of GAN). It is presumed that SRGAN has an improved accuracy and produces picture all the more satisfying to eyes than SRGAN.

A. Model

The generator design contains residual networks rather than deep convolution networks since residual organizations are anything but difficult to prepare and permits them to be significantly more profound to produce better outcomes. This is on the grounds that the residual network utilized a kind of association called skip associations. The goal of the input is expanded with both well-prepared convolution layers of sub-pixels. While preparing, a high-resolution picture (HR) is reduced to an image which is of a lower resolution (LR). The generator design then attempts to upscale the picture from a low goal to super-resolution(SR). After this process the picture is sent into the discriminator, it then attempts to recognize a super resolution and High- Resolution picture and create the antagonistic misfortune, which then back propagates into the generator design once again.

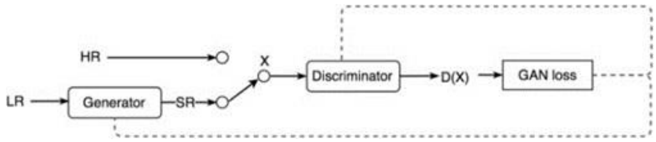


**Fig 3: Training model to implement SRGAN**

B. Training

The high-resolution pictures are accessible during training is subjected to Gaussian filter to obtain Low Resolution as output. It is then down sampled by factor. The SRGAN utilizes perceptual loss function (LSR) that constitutes the combined amount of both the loss segments: adversarial loss and content loss. This loss is significant for the exhibition of the G. The combination of this loss is significant for the exhibition of the G. There are two types of content loss which we shall be considering: pixelwise MSE loss for the SRResnet architecture

$$l_{MSE}^{SR} = \frac{1}{r^2WH}\sum_{x=1}^{rW}\sum_{y=1}^{rH}(I_{x,y}^{HR} - G_{\theta_G}(I^{LR})_{x,y})^2 \tag{14}$$

The loss of various VGG layers is also taken as one of the content losses. This VGG loss is based on the ReLU activation layers of the pre-trained 19layer VGG network.

$$l_{VGG/i.j}^{SR} = \frac{1}{W_{i,j}H_{i,j}}\sum_{x=1}^{W_{i,j}}\sum_{y=1}^{H_{i,j}}(\phi_{i,j}(I^{HR})_{x,y} - \phi_{i,j}(G_{\theta_G}(I^{LR}))_{x,y})^2 \tag{15}$$

Adversarial loss is the one that powers the G to picture close to high-resolution images with the help of a discriminator that is prepared to separate between HR and SR images.

$$l_{Gen}^{SR} = \sum_{n=1}^{N} -\log D_{\theta_D}(G_{\theta_G}(I^{LR})) \tag{16}$$

The final equation will be:

$$l^{SR} = l_{X}^{SR} + 10^{-3} l_{Gen}^{SR} \tag{17}$$

Where the first element is the context loss and the second element is the adversarial loss and both together constitutes perpetual loss.

*C.* Output

After upscaling the image four times, we obtain an output as shown in figure 4 . The data is taken from ImageNet database and around 350 hundred images are trained to acquire maximum precision. We must note that the testing images do not come from the already trained samples. The down sampling factor, *r* is taken as 4.


**Fig4: Output predicted after applying r as 4**

## 6. CONCLUSION

There are many factors that can be improved in GAN models. SRGAN can be achieved with another model called ESRGAN+. This has the basis of what we saw but only the generator loss functions are improvised to provide better results. Boundless GAN can be improved to make the output a lot better. Similarly, GAN is a growing field and there will always be scope for improvement and betterment. There are many fields in which GAN models are applied as seen above. This can be further taken care of.

## ACKNOWLEDGEMENT

We are simply ecstatic to convey our sincere gratitude towards the management of **Sir M Visvesvaraya Institute of Technology**, Bangalore for providing us the opportunity and the support to accomplish our paper in their premises.

This paper couldn't have been possible without his thoughtful guidance and patience. We would like to thank our guide **Dr. Sreenivasa B C**, Associate Professor of Computer Science and Engineering Department.

**Prof Dr. G.C. Bhanu Prakash**, HOD, Department of Computer Science and Engineering for his invaluable support and guidance. Heartfelt thanks to **Prof Dr V R Manjunath**, Principal, Sir M Visvesvaraya Institute of Technology, for his support and encouragement.

## REFERENCES

[1] Yang-Jie Cao, Li-Li Jia, Yong-Xia Chen, Nan Lin, Cong Yang, Bo Zhang, Zhi Liu, Xue-Xiang Li, And Hong-Hua Dai, "Recent Advances of GAN in Computer Vision" in 2018 IEEE Access (volume7) , pp. 14985-15006, doi: 10.1109/ACCESS.2018.2886814

[2] T. Wang, M. Liu, J. Zhu, A. Tao, J. Kautz and B. Catanzaro, "High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018, pp. 8798-8807, doi: 10.1109/CVPR.2018.00917.

[3] Xining Zhu, Lin Zhang, Lijun Zhang, Lijun Zhang, Xiao Liu, Ying Shen, and Shengjie Zhao, "GAN-based Image Super Resolution with a Novel Quality Loss" in 2019 International Symposium on Intelligent Signal Processing and Communication Systems, doi: 10.1109/ISPACS48206.2019.8986250

[4] Zhaoqing Pan; Weijie Yu; Xiaokai Yi, Asifullah Khan, Feng Yuan, Yuhui Zheng "Recent Progress on Generative GANs: A Survey" in 2019 IEEE Access(volume7), eISSN:2169-3536, doi: 10.1109/ACCESS.2019.2905015

[5] T. Park, M. Liu, T. Wang and J. Zhu, "Semantic Image Synthesis With Spatially-Adaptive Normalization," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 2332-2341, doi: 10.1109/CVPR.2019.00244.

[6] Yi Jiang, Jiajie Xu, Jing Xu, Baoqing Yang and Junwu Zhu, "Image Inpainting Based on Generative Adversarial Networks" in 2020 IEEE Access(volume8), pp:22884-22892, doi: 10.1109/ACCESS.2020.2970169.

[7] Yun Wu, Lin Lan, Huiyun Long, Guangqian Kong, Xun Duan, and Changzhuan Xu, "Image Super-Resolution Reconstruction Based on a Generative Adversarial Network" in 2020 IEEE Access(volume8), pp: 215133-215144, doi: 10.1109/ACCESS.2020.3040424.