

***DEPARTMENT OF COMPUTER SCIENCE ENGINEERING,***

***SCHOOL OF ENGINEERING AND TECHNOLOGY,***

***SHARDA UNIVERSITY, GREATER NOIDA***

**Image super-resolution using GAN**

***A project submitted***

***in partial fulfillment of the requirements for the degree of***

***Bachelor of Technology in Computer Science and Engineering***

**by**

**Darshan Parekh (2018013147)**

**Ankita Maiti (2018010641)**

**Supervised by:**

**Dr. Vishal Jain**

**Associate Professor**

**May, 2022**

**CERTIFICATE**

This is to certify that the report entitled “**Image super-resolution using GAN**” submitted by Mr. / Ms. Names of the students(s) having **Ms. Ankita Maiti (2018010641) and Mr. Darshan Parekh (2018013147)** to Sharda University, towards the fulfillment of requirements of the degree of Bachelor of Technology is record of bonafide final year Project work carried out by him/her in the Department of Computer Science and Engineering, School of Engineering and Technology, Sharda University. The results/findings contained in this Project have not been submitted in part or full to any other University/Institute for award of any other Degree/Diploma.

Signature of Supervisor

Name:

Designation:

Signature of Head of Department

Name:

(Office seal)

Place:

Date:

# Signature of External Examiner

# Date:

# ACKNOWLEDGEMENT

A major project is a golden opportunity for learning and self-development. We consider ourselves very lucky and honored to have so many wonderful people lead us through the completion of this project.

First and foremost, we would like to thank Dr. Nitin Rakesh, HOD, CSE who allowed us to undertake this project.

My thanks to Dr. Vishal Jain for his guidance in my project work, **Image super-resolution using Gan**, who despite being extraordinarily busy with academics, took time out to hear, guide, and keep us on the correct path. We do not know where we would have been without his help.

CSE department monitored our progress and arranged all facilities to make life easier. We choose this moment to acknowledge their contribution gratefully.

Name and signature of Students

Ankita Maiti – 2018010641

Darshan Parekh – 2018013147

# ABSTRACT

*Reconstructing low-resolution images to high-resolution images by building a neural network is quite challenging but can be used in many applications like medical imaging, public surveillance, or old photo recovery. In comparison with previous methods, deep learning has a breakthrough in high resolution in accuracy and speed. This project aims towards enhancing low-resolution images by applying a deep network with an adversarial network (Generative Adversarial Networks) to produce high resolutions images. Our main target is to reconstruct the high-resolution image by developing the low-resolution image such that the main details in the reconstructed images are not lost.*

# Table of Contents

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  | [Certificate](#_Toc69759028) 2  [Certificate 3](#_Toc69759029)  [Acknowledgment 4](#_Toc69759032)  [Abstract 4](#_Toc69759031)  [Table Of Contents 6](#_Toc69759033)  [1. Introduction 8](#_Toc69759034)  [1.1 Problem Definition …………….. ………………………………………](#_Toc69759035)9  [1.2 Problem Specification………………………………………………………………1](#_Toc69759036)0  [1.2.1 Project Goals](#_Toc69759037) 10  1.2.2 Project Deliverables……………………………………………………………..10  1.2.3 Cost of Project…………………………………………………………………...10  1.2.4 Deadlines of the Project………………………………………………………..10  1.2.5 Risk and Assumptions…………………………………………………………10  1.2.6 Use Cases………………………………………………………………………11  1.2.7 Requirement Specification……………………………………………………11   1.2.8 Potential Challenges…………………..………………………………………11  1.3 Hardware Specification……………………………………………………………12  1.4 Software Specification…………………………………………………………….13  2. Literature Survey……………………………………………………………………… 14  2.1 Existing System……………………………………………………………………14  2.2 Proposed System…………………………………………………………………..20  2.3 Feasibility Study…………………………………………………………………..25  3. System Analysis & Analysis…………………………………………………………..28  3.1 Requirement Specification………………………………………………………..28  3.1.1 Product Prespective…………………………………………………………….28  3.1.2 Product Functions……………………………………………………………….28  3.1.3 User Characteristics…………………………………………………………….28  3.1.3.1 Large Organisations……………………………………………………….28  3.1.3.2 Academic Organizations.………………………………………………….29  3.1.4 Design Constraints..…………………………………………………………….29  3.1.5 Assumptions and dependencies………………………………………………..29  3.1.6 Requirement specifications…………………………………….……………….29  3.1.6.1 User Interfaces….………………………………………………………….29  3.1.6.2 Hardware Interfaces……………………………………………………….29  3.1.6.3 Software Interfaces…………………………………………………..…….30  3.1.7 System Features…..…………………………………………………………….30  3.1.7.1 Introduction…..…………………………………………………………….30  3.1.7.2 Input………….…………………………………………………………….30  3.2 Flowchart…………………………………………………………………………..31  3.3 Design & Test Steps……………………………………………………………..33  3.3.1 Design…………………………...……………………………………………..32  3.3.2 Dataset Used…………………… ……………………………………………..33  3.3.3 Breakdown structure…………………………………………………………..35  3.3.4 Test Steps………………………………….…………………………………..37  3.4 Algorithm & Pseudocode…..……………………………………………….…...38  4. Results of SRGAN…………………………………….………………………………….41  4.1 During Training………………………………….…………………………………41  4.2 During Testing……………………………………………………………………..44  4.3 Graphical Representaion…………………………………………………………...44  4.3.1 Discriminator vs generator loss graph………...…………………………………45  4.3.2 Structural Similarity Index Measure Graph… ...……………………………….. 46  4.4 Output of the model……………………………………………………………….. 47. 5.Conclusion……………………………………………………………..…………………..48  5.1 Future Development………………………………………………………………..48  REFERENCE………………………………………………………………………………...47 |  |
|  |  |  |

**LIST OF FIGURES**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Fig No.** | **Description** | **Page No.** |
| 1 | Fig 1.1 | Architecture of Model | 10 |
| 2 | Fig 2.2.1 | Composition of image super resolution | 21 |
| 3 | Fig 2.2.2 | Architecture of GAN | 23 |
| 4 | Fig 3.2 | Flow chart of SRGAN | 32 |
| 5 | Fig 3.3.1.1 | Training of Discriminator model | 34 |
| 6 | Fig 3.3.1.2 | Training of Generator model | 34 |
| 7 | Fig 3.3.2.1 | Images of BSD300 | 35 |
| 8 | Fig 3.3.2.2 | Images of URBAN100 | 35 |
| 9 | Fig 3.3.2.3 | Images of ImageNet | 36 |
| 10 | Fig 3.3.3 | Breakdown structure of the system | 37 |
| 11 | Fig 3.4.1 | Convolutional Block | 39 |
| 12 | Fig 3.4.2 | Upsample Block | 39 |
| 13 | Fig 3.4.3 | Residual Block | 40 |
| 14 | Fig 3.4.4 | Generator Block | 40 |
| 15 | Fig 3.4.5 | Discriminator Block | 41 |
| 16 | Fig 4.1.1 | Output from training | 43 |
| 17 | Fig 4.1.2 | Output from training | 43 |
| 18 | Fig 4.1.3 | LR, HR and SR images from training | 44 |
| 19 | Fig 4.2.1 | Output from testing | 45 |
| 20 | Fig 4.2.2 | LR, HR and SR images from testing | 46 |
| 21 | Fig 4.3.1 | Discriminator vs generator loss graph | 46 |
| 22 | Fig 4.3.2 | Similarity measure graph | 47 |
| 23 | Fig 4.4 | Final Output | 48 |

1. **INTRODUCTION**

Generating high-resolution photos from low-resolution photos is a challenging task and has huge applications in the real world, including photo reconstruction, surveillance cameras, and computer-aided design. Recently, generative adversarial networks have gained extensive recognition from the computer group because of their promising results in synthesis of real-world images.

Generative Adversarial Network (GAN) is a type of Artificial Intelligence algorithm used for unsupervised ML(Machine Learning). GAN is a deep NN (Neural Network) architecture made of two networks, the first network being a generator and another being discriminator, which compete with each other (hence the name "adversarial"). GAN is about creation, such as drawing portraits or creating symphonies. The principal focal point of GAN is to produce information without any preparation.

GANs consist of a Generator and Discriminator. Consider it like a game where Generator produce data via probability distribution and Discriminator acts like a classifier. Discriminator decides whether input is from true training dataset of fake generated data. Generator optimize data so that it can match true training data. Or consider it like discriminator guides generator to produce realistic data.

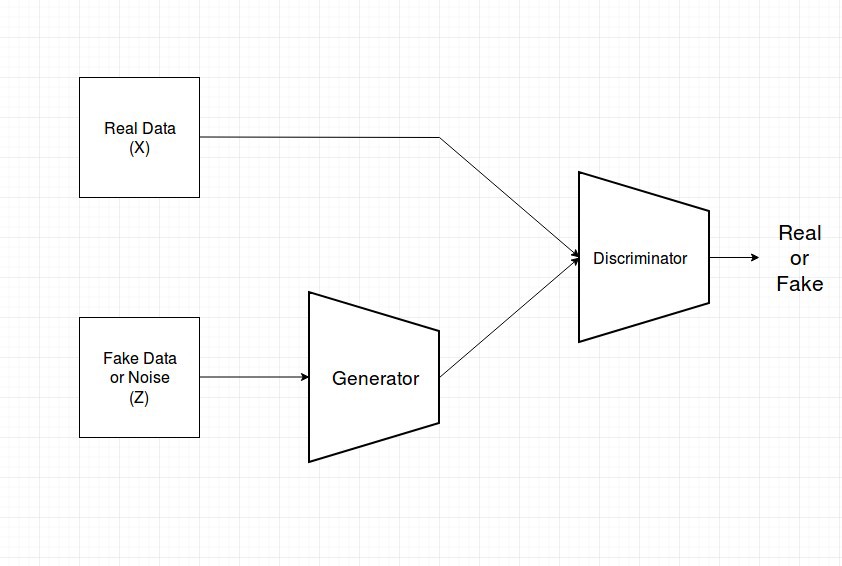
In the Generative Adversarial Network (GAN) suggested by Goodfellow et al. [7] which was introductory in the field of Super-Resolution, the generative model is pitted to confront the opponent which is to fool the discriminator and train two models at the same time: the generator G that catches the distributed data and discriminator D evaluates that the sample comes from training dataset instead of generated image by the generator G. The process training of G is to improves the likelihood of D mistake. The proposed frame correlates with a very small and huge game with two-player.

For example we want to generate random person face, we will select dataset which consist of images of facial details and fake data consist of random noise. Generate will produce images from random noise which will be classified by Discriminator. Both these networks are trained simultaneously. One distinct feature of GANs is generator produced images which contain features of original data images but may or may not be same. Generators produce face image with heterogenous mixture of features.

Super-Resolution (SR) is the procedure of reconstructing one or more low-resolution images to high-resolution images of the same instance [26]. The difficult task of evaluating high-resolution (HR) images only from their corresponding low resolution (LR) is called Single Image Super-Resolution (SISR). For high magnification factors, the compositional nature of the undetermined super-resolution is particularly obvious. The texture details in the reconstructed SISR image are usually absent. The enhancement mission of Supervised Single Image Super-Resolution algorithms is usually

(a) minimizing the MSE i.e. mean square error between the restored HR image and also maximizing the PSNR i.e. peak signal-to-noise ratio or

(b) Maximizing the SSIM i.e., Structural Similarity Index or both.



*Fig 1.1 Architecture of Model*

* 1. **Problem Definition**

The project’s goal is to enter, process, and train deep Neural network in order to generate super resolution images using low resolution images.

To generate Super-resolution images from a low-resolution image by using a deep network with an adversarial network (GAN) to construct super-resolution images. To reconstruct high-resolution images from low-resolution images. One of the most impressive applications of generative adversarial network is that it can generate new/distinct human images which doesn’t belong to primary human images dataset used for training.

The most well-known application of generative adversarial network is text to image generation. Model takes text input and generates images scene depending on text received from input.

* 1. **Problem Specification**

* + 1. **Goals of Project**
* Create a GAN model that can reconstruct low-resolution images to Super-resolution.
* Make a system that is unique with the current state-of-the-art designs.
* Also, come up with a novel method that can initiate a new paradigm for studying training data.

**1.2.2 Project Deliverables**

* A model that can be used in various devices such as surveillance cameras, medical, media, etc.

**1.2.3 Cost of Project**

* Insignificant on the grounds as most executions should be done in more significant level registering gear, greater expenses can be applied because of innate power utilization and use when working with AI projects. These can likewise be dropped utilizing on the web stages like Google Colab.

**1.2.4 Deadlines of the Project**

* Requirements and Feasibility Study: 4 Weeks
* Design: 2 Weeks
* Development: 18-20 Weeks
* Integration and Testing: 4 Weeks
* Publishing the work

**1.2.5 Risk and Assumptions**

The main risks that our products may come across are:

1. Economic Limitation

Because of the expensive setup requirements for efficient and fast operation of the product and the high-power consumption of our systems.

1. Market Competition

In order to create sustainability of the product, the entry cost of the system should be negligible.  
  
**1.2.6 Use Cases**

* Surveillance: Detect, perceive and execute facial acknowledgment on low-goal pictures acquired from surveillance cameras.
* Medical: Spatial inclusion, Output time, and SNR i.e signal to noise ratio proportion, catching high-goal MRI pictures can be difficult. Super-Resolution tackles this issue by producing high-goal MRI from low-goal MRI pictures.
* Media: Super-resolution may be applied to reduce server costs because media data can be transferred at a low resolution and upgraded instantly.

**1.2.7 Requirement Specification**

* The projects need to enhance the image to high resolution from low resolution.
* In special use cases, it needs to not lose the characteristic feature of the low resolution while reconstructing the image.
* The project should be light as software, as it is targeted to run on embedded systems like surveillance cameras, medical, media, etc.

**1.2.8 Potential Challenges**

* The problem of inter-class similarity occurs when images to be classified are physically different but give similar characteristics.
* To reduce the effects of this risk, we have to extensively train the generator model of the GAN to create high-resolution images.
  1. **Hardware Specification**

|  |  |
| --- | --- |
| **Minimum Requirements** | **Windows** |
| Operating System | Windows 7 or above |
| Processor | Minimum Intel Core i3.  Recommended Intel Core i5.  (For the processing of the system) |
| Disk space | Minimum 500 GB HDD.  Recommended 500GB SSD (Fast storage and access) (Storing legacy data) |
| Graphics | Minimum 2GB NVIDIA MX940.  Recommended 8GB RTX 3050.  (For faster processing of data and recognition) |
| Wired LAN network | Minimum 100mbps bandwidth.  Recommended 300mbps bandwidth.  (For Local Deployment) |
| Wireless Network (Wi-Fi) with an active internet connection. | Minimum 50mbps Wi-Fi internet.  Recommended 100mbps Wi-Fi internet.  (Remote/Cloud deployment) |

* 1. **Software Specification**

|  |
| --- |
| Tensorflow (Deep learning Tools) |
| Keras ( backend for Tensorflow) |
| Numpy |
| Google Colab/ Jupiter notebook ( IDE) |

1. **LITERATURE SURVEY**

**2.1 Existing System**

Author Tzu-An Song et al[1] has suggested the method self-supervised super-resolution (SSSR) which is PET dependent on dual Generative Adversarial Networks (GAN) that trains the combinedly to generate SR PET images from unpaired PET inputs in a self-supervised manner. For supervised training, the images have been used from the BrainWeb database. The proposed network receives low- resolution PET. The research shows the result that, SSSR appears to be weaker in comparison with VDSR it is better than classic deblurring. The last is true in light of the fact that VDSR is fully supervised and based on paired training data sets.

Author [2] has suggested the model, conditional generative adversarial network for developing high resolution images from semantic label maps. It generates for combining 2048×1024 images as a result with new adversarial loss. It also produces new multi- scale generator and discriminator architectures. By incorporating information from object instance segmentation which empowers object control for example, eliminating/adding object and changing the classification of the object. Also, the suggests a way to generate diversification given similar info, permitting users to alter the display of the objects. The outcomes recommends that conditional GANs can combine high-resolution images with no user created loss or pre-trained network.

Author Christian Ledig et al[4] have suggested the model, super-resolution generative adversarial network (SRGAN) with the help of deep residual network (ResNet), for generating an image of super-resolution using GAN. The proposed network can generate photo-realistic images by upscaling factor ×4. They developed a loss function that comprises a content loss and an adversarial loss.

Author Xin Yu et al[5] develop a network that uses attribute-embedded upscaling which comprises two types of network, one being upsampling network and the other one being a discriminative network. The purpose of the upsampling network is to skip connection with the help autoencoder. On the other hand, the purpose of the discriminative network is to check if the image of faces after being super-resolved consists of the expected attributes, and after that, the loss is used by upsampling of the network for updation.

Author Xining Zhu et al[6] have suggested the method Gradient Map Generative Adversarial Network (GMCAN) which generates images about HVS (Human Vision System) to design a loss function by combining IQA(Image Quality Assessment). The type of GAN used in this method is Wasserstein GANs(WGAN-GP) which is an improved version to control the instability of the initial GAN. It consists of two parts, the first being network architecture and the second being the loss function. It also focuses majorly on training the network with different datasets which are Part of the ImageNet dataset, DF2K (DIVIK + Flickr2K) dataset, DF2K + OST (OutdoorSceneTraining) dataset, and DIVIK dataset. Also to test the network, experiments were performed on different datasets including Set5, Set14, BSD100, and Urban100. But GMCAN is unable to show adequate outcomes when handling strong repetitive shapes.

Author [7] focusses on training two models the evaluating generative models by an adversarial process. The generative model is the first model which captures the data and discriminative model, the second model, whose purpose is to evaluate that the data was sent by the training data not generative model. The training process is for generative model is to magnify the chance of the committing mistake by the discriminative model.

Author William T. Freeman et al[8] has suggested an algorithm that accesses a dataset of the image to train to generate a high-resolution image when the image is being zoomed in. To create the training data, high-resolution images are used and breakdown every image in such a way that fixing the breakdown of the images can be done later in the procedure. Normally, the image is been blurred and subsample them to make a low-resolution image of ½ the number of existing pixels in every dimension. The algorithm performs its best to evaluate the low-resolution image of vocabulary or letters of a text i.e. when text is zoomed in high-resolution characteristic is formed.

Author J. Kim et al[11] proposed an extremely precise super-resolution method with the help of a deep convolutional network which is been used for ImageNet Classification motivated by VGG-Net. Relatable data over an enormous image domain is taken advantage of productively by dropping small filters repeatedly in a deep network. For training, high-resolution images are directly modeled by SRCNN.

Author Jiwon Kim et al[12] proposed a method of super-resolution with the help of a deeply-recursive convolutional network (DRCN). Expanding recursion depth can further develop execution without presenting new boundaries for extra convolutions. Two approaches have been proposed to facilitate the struggle of training. Firstly, supervision of all the recursion is required. The reconstruction method is something very similar for all recursions. As every recursion prompts an alternate HR forecast, we join all predictions coming about because of various levels of recursions to convey a more precise last prediction. The subsequent proposition is to utilize a skip-connection from the input to the reconstruction layer. In the super-resolution, an LR image (input) and an HR image (output) share similar data overall.

Author Ishan Durugkar et al[15] have suggested the model, a generative multi-adversarial network (GMAN) which is a framework for extending GAN to various discriminators. GMAN can use earlier, unmodified targets for reliable training. The image generation task that compares the proposed framework with the standard GAN shows that when measured by paired GAM type indicators, GMAN produces higher quality samples in a small number of iterations. To train the network, the data sets used are MNIST data set, CelebA, and CIFAR-10 dataset. The goal is to create multiple discriminators in the GAN framework and study the role of the discriminators. Allowing the generator to automatically adjust its learning plan is better than GAN using a single discriminator on MNIST.

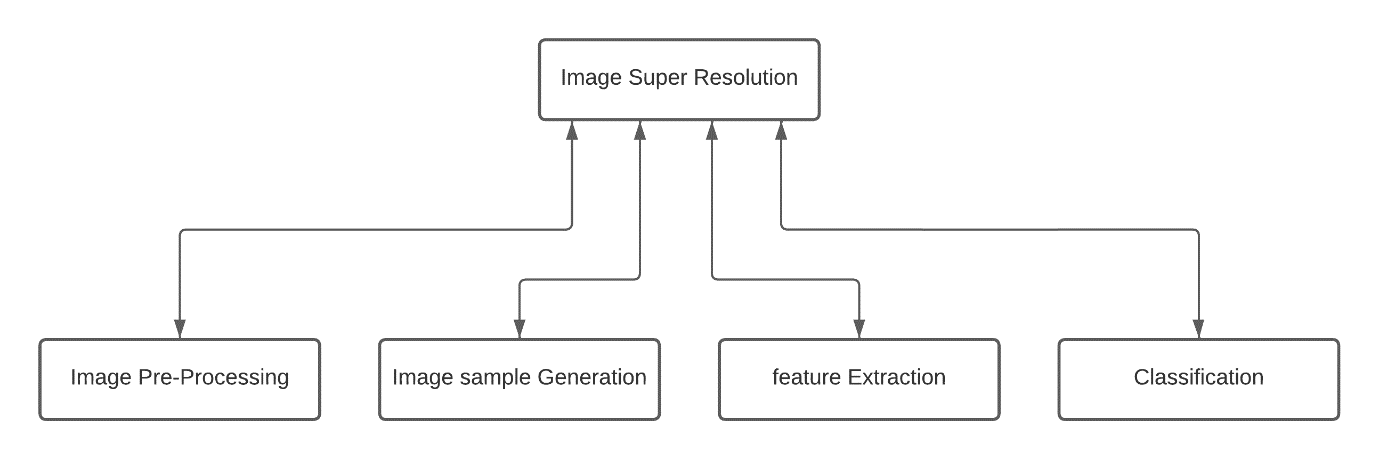
Author [21] demonstrates several new techniques considered to improve convergence of the GANs which lead to improvised semi-supervised training and sample generation. For semi-supervised experiment purposes CIFAR-10, MNIST and SVHN datasets are used and for sample generation experiment SVHN, CIFAR-10, ImageNet and MNIST datasets are used. The technique includes Feature matching, historical averaging, minibatch discrimination virtual batch normalization, and one-sided label smoothing. With these techniques, the training of GANs becomes stabilize which helps to train even the untrainable model.

*Table 2.1 Comparison of methods by various researchers*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method | Description | Dataset Used | | Ref No |
| Training Dataset | Testing  Dataset |
| Super-resolution generative adversarial network (SRGAN) using ResNet | This network can generate photo-realistic images by upscaling factor × 4 using the deep residual network. | Set5, Set14, BSD100 | BSD300 | [4] |
| Gradient Map Generative Adversarial Network (GMCAN) | It generates images about HVS (Human Vision system) to design a loss function by combining Image quality Assessment. | DF2K (DIVIK + Flickr2K), DF2K + OST (OutdoorSceneTraining), and DIVIK. | Set5, Set14, BSD100, and Urban100 | [6] |
| GAN | It focuses on training two models the evaluating generative models by an adversarial process. | MNIST, the Toronto Face Database  (TFD), and CIFAR-10. | MNIST, the Toronto Face Database  (TFD) | [7] |
| Coupled generative adversarial network (CoGAN) | It is constructed to study the joint distribution in two different domains which include a pair of GANs. | MNIST, Celeb Faces Attributes, RGBD, and NYU dataset |  | [9] |
| Conditional generative adversarial network (CGAN) | Construction of GAN by adding some data to condition on both the generator and discriminator | MNIST,  Flickr, ImageNet, YFCC100M2, and MIR Flickr 25000 dataset | MNIST | [10] |
| Super-resolution method with a deep convolutional network | It overcomes the drawback of SRCNN by increasing models network depth and achieving better accuracy. | Berkeley Segmentation Dataset | Set5, Set14, Urban 100, and B100 | [11] |
| Super-resolution with DRCN | The super-resolution model with a deep recursive layer improves the overall performance without the need of adding a new parameter. |  | Set 5, Set 14, B100, Urban100 | [12] |
| SimGAN | Stimulated and Unsupervised learning where the job is to get familiar with a model to work on the authenticity of a stimulator’s output utilizing unlabeled genuine information while saving the comment data from the stimulator | UnityEyes, NYU hand pose dataset and MPIIGaze | NYU hand pose and MPIIGaze | [13] |
| Style and Structure Generative Adversarial Network (S2-GAN) | It includes- the Structure GAN that creates a surface normal map and secondly, the Style GAN which takes input as the surface normal map and creates the 2D image. | NYUv2, RGBD, SUN RGBD, Places, and ImageNet dataset. | NYUv2, SUN RGB-D, | [14] |
| Generative multi-adversarial network (GMAN) | It is a framework for extending GAN to various discriminators to produce higher quality samples in a small number of iterations. | MNIST, CelebA and CIFAR-10 |  | [15] |
| Conditional adversarial network | Image-to-image conversion problem which is effective in synthesizing pictures from label maps, coloring images, and recreating objects. | Cityscape, UT Zappos50K, CMP Facades, Google Maps, and Paris Street View dataset. | UT Zappos50K | [16] |
| Perceptual Generative Adversarial Network (Perceptual GAN) | It improves detection performance of small through reducing the difference of representation of small objects with the large objects. | Tsinghua-Tencent 100K benchmark and Caltech benchmark | Tsinghua-Tencent 100K benchmark and Caltech benchmark | [17] |
| StackGAN | With Conditioning Augmentation to come up with 256×256 photo-realistic images constrained on text representation. | CUB, Oxford-102, and MS COCO | CUB and Oxford-102 | [18] |
| Energy-based GAN | It consists of two method- GANs and auto-encoders in which the discriminator act as the energy function that allocates the low energy to the area around the data while the high energy goes to the remaining areas. | MNIST, ImageNet, LSUN bedroom, and CelebA |  | [22] |
| GAN | With techniques like Feature matching, historical averaging, minibatch discrimination virtual batch normalization, and one-sided label smoothing to improve the convergence of the GANs | MNIST, CIFAR-10, ImageNet and SVHN |  | [21] |

**2.2 Proposed system**

In this work for generating super-resolution images, we used GANs as the framework. GANs deep network combined with adversary network to produce HR images is more appealing as they are more detailed compared to similar design SRResNet. These are measured with the amplification factor measured by PSNR ResNet. We propose a generator network with optimized perceptual loss. This loss is based on the feature map of the VGG network rather than the content loss based on MSE.



*Fig 2.2.1 Composition of Image super resolution*

For Super-resolution Generator Network is fed with an interpolated low-resolution image *ILR* as input. Here *ILR* is the low-resolution image of its High-resolution image *IHR*. LR images are obtained by downsampling images via various methods such as Bicubic, bilinear, etc. with downscaling operation factor r. High-resolution images *IHR* are only present for training. Images with C color channels, *ILR* can be stated as *W* X *H* X *C* and *IHR*, *ISR* by r*W* Xr*H* X *C.*

Our primary aim is to train a Generator G that accepts LR images and outputs corresponding HR images. We train the generator as a feedforward neural network GΘG. For training *IHR*, n= 1,…N with corresponding to *ILR* , n= 1,…N

we use the perceptual loss *ISR* as a weighted sum of most loss components.

1. **Adversarial network architecture**

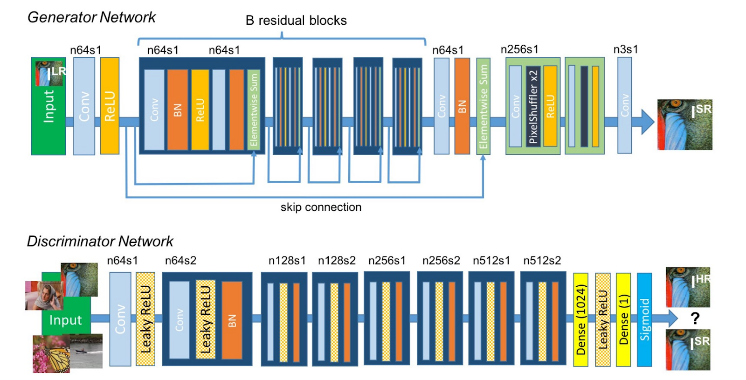
After setting up the generator, we define a discriminator model DΘD with a generator network GΘG to resolve the minimum/maximum problem.

The purpose behind this architecture is to train a generator network with the purpose of deceiving the discriminator network, that is specially trained to tell the difference between real and super-resolution images. Now the discriminator decision generator is used to learn to create an image of HR image similar to a real image, which makes it more difficult for the discriminator to determine which is a super-resolution or a real image.

The generator network shown in the figure has B residual blocks of the same design. 64 feature maps with batch normalization layers along with two 3x3 kernels, and input image parameters. The resolution of the ReLu activation function is increased by two sub-pixel convolutional layers.

The generator model shown in figure 1 has B residual blocks of the same design. 64 feature maps with batch normalization layers along with two 3x3 kernels, and input image parameters. The resolution of the ReLu activation function is increased by two sub-pixel of the convolutional layers.

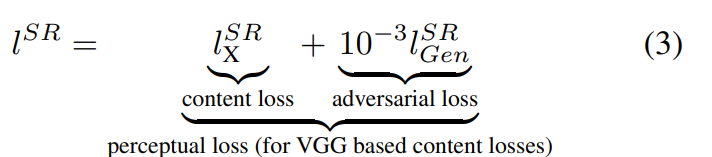
A trained discriminator was used to tell the difference between real HR images and generated super resolution images. Use Leaky ReLu activation instead of max pooling because it helps the gradient to pass through the architecture more easily. It contains 8 convolutional layers, and the number of 3x3 cores keeps increasing. When the features are doubled, 64 to 512 kernels in the VGG network. The stride convolution reduces the frame resolution, so the generate feature map is changed by 2 dense layers to ensure.



*Fig 2.2.2: (Source* *- Ledig et al.[4]) Architecture of GAN*

1. **Perceptual loss**

This is crucial for the execution of the generator network. The loss function *ƖSR* examines the solutions related to related characteristics. Then the perceptual loss is formulated as a weighted sum of the adversarial loss and content loss components.



1. **Content loss**

In previous implementations, this method is widely used this gives high PSNR values but usually the lack of detail in the image leads to a smooth texture, instead of relying on pixel-to-pixel loss, the loss function we use is based on a weighted combination. This loss is implemented in ReLu activation layers. VGG loss feature representation of super-resolution images and the reference image.

1. **Adversarial loss**

Along with content loss, we combine the generative network to perceptual loss. The support network tries to trick the discriminator network to make the solution close to the real image. The loss of the generator *ƖSR*is explained on the basis of the probabilities of the discriminator DΘD(GΘG(ILR)) is the Probability that the reconstructed image GΘG(ILR)) is a original HR image.

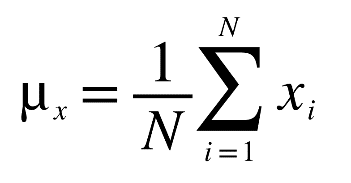
1. **Structural Similarity Metric**

To compare quality between two images relies on estimated errors between truth image and super resolute image. Most common metric is to estimate the difference between values of each corresponding pixels between truth image and super resolute image. Human visual perception can easily identify dissimilarities between images hence, Structural Similarity Metric replicates this behavior of differentiating images.

Structural Similarity Metric calculation ranges between -1 to 1. +1 value indicates both images are very similar and -1 indicates images are every different.

This is based on three key features

* ***Luminance*** : which can be defined as average of all pixels values and formula,



*Where,*

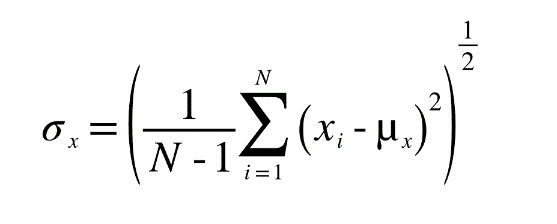
*x is image*

*N is total number of pixels*

*is ith pixel value*

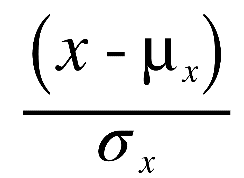
Luminance comparison is function of *l(x,y)* is then function of and

* ***Contrast*** : which can be defined as standard deviation of all pixel values and formula,



Contrast comparison *c(x,y)* is then function of and

* ***Structure*** : can be defined as unit standard deviation



* 1. **Feasibility of our project**

**2.3.1 Operational and Technical Feasibility**

-The motive of this study is to verify if the viability of the system being developed is up to the mark and is compatible with the skill set of the developers. If the concepts tackled in this project are far too difficult to comprehend for the people developing the system, there can be an added burden taking extra time and learning efforts to make sure that the project is up to the mark.

**-** The proposed system of this research is highly feasible in technical regards as it doesn’t require any major resources. Every resource introduced in this research is easily available for use publicly and the proposed system is also very much operational despite being a novel method in this research field.

-The developers of the system are also well versed with the concepts involved with this system being students in the field of Machine Learning. Although some topics are new to the developers it is still ensured that the development of the system is merely a learning opportunity and not a burden on the developers. The following are the experiences of the developers with the concepts that were required in the development of this system:

GAN(Generative Adversarial Networks) :This topic was completely new to the developers and in fact motivational, considering complexity of this topic developers faced many challenges in implementing and achieved good outcomes despite of challenges in training and testing =.

CNN : Although conventional Convolutional Neural Network models have been explored in depth by the developments over the course of their academic career, the concept of 3D CNNs is relatively new to the developers

Working with image datasets : Up until this point, the developers have worked with datasets consisting of data-points, but never RGB images, this made the integration of said images a new learning experience during the development of this system,

The dataset used in the development of this system is widely known as one of the best datasets for applications of this nature. Since many other benchmarks are clearly and efficiently defined in the research space, making sure that comparing our system with the state of the art is relatively easy.

The system thus developed is also aimed to be operational with relatively less hardware requirements, however since it is a machine learning application, it is inherently inclined to rely on graphical memory to perform optimally. Therefore, systems which are equipped with a dedicated graphics card tend to perform faster and more efficiently than ones without said graphic cards. This was verified firsthand when the model was trained on a beefy computer after initial development on a notebook laptop and it was significantly faster to train the model on the other machine.

**2.3.2 Economic Feasibility**

- The following analysis is carried out to check whether the development of the system is not adversely affecting the financial situation of the developers initially and the users of the system after the development has been complete.

- The initial developments of the system indicated that the economic constraints are next to nil save for the cost of a machine capable enough to code and some graphical memory. But further developments made it apparent that a high end system with generous graphical memory is required for the most optimal and swift training of the model. Systems like these can be quite expensive and can create a financial burden on the developers. But since this is not a strictly needed requirement it can be overlooked.

- In conclusion, keeping the amount of competition on the market, the cost of the project must be kept at the minimum so it can appeal to the maximum number of users. This can be achieved with given that the model takes a somewhat long time to train and process data.

**2.3.3 Legal Feasibility**

* Privacy of the people may be violated if they are unaware of their images is being used for the testing or training.
* The aim of this study is to make sure that the development and deployment of the project does not cross any legal bounds and is not for concern of any legal action by the users.
* Since the project uses images that are available in public dataset, a situation may arise in the future development of this model when we use the human images to train and test the model for achieving super resolution images from low degraded images. The images of people might not be aware that their images are being used which may cause legal concern.
* The system must not be employed casually as it is a matter of privacy and utmost care and consideration must be observed when deploying the system in a public space.

**2.3.4 Environmental Feasibility**

* The environmental feasibility study deals with the effects that the system being developed may have on the environment. As responsible citizens we must make sure that no adverse effects are being caused on the environment due to our system.
* Carbon emissions may be an impact on the environment because all its physical components use electricity.
* The environmental effects of this system are negligible at best when a relatively low powered machine is being used to train the model.

**2.3.5 Scheduling Feasibility**

* In order to maintain a smooth workflow and avoid irregular work load on the developers, the scheduling feasibility study is carried out to realize the scope of the project and come up with a sustainable timeline
* As discussed in section 1.3.4. An efficient timeline was come up with beforehand to ensure that no irregularities arise when developing the system.
* This timeline takes into account each integral part of development and also allocates time for testing and publishing.

1. **SYSTEM ANALYSIS AND DESIGN**

**3.1 Software Requirement Specification**

**3.1.1 Product Perspective**

In view of its composition, the Super resolution image system proposed here, will be the best and certainly by the implementation of a novel approach to interpret the activities performed by an individual it will open up new domains of research in this field. The aim of the panel will be to add to a system that can be efficiently used by organizations everywhere, such as surveillance, medical field and media to generate images that has low resolution and which can converted into super resolution image and provide better usage to the mentioned field.

**3.1.2 Product Functions**

Super resolution using GAN is aimed at generating super resolution images with the help of low-resolution images and based on the trained model, and provide better usage of the images required by the different categories.

Those categories can be diverse and its capabilities purely depend on the dataset it's trained on, which makes it a very versatile model to use.

* + 1. **User Characteristics**

The targeted users here will be all the fields that require surveillance including Airport, Banks and similar places with very high footfall. Moreover the user base will include places that requires with high quality images like medical fields or media. Also, it can be used in schools or universities for security reasons. For example, places like these are heavily populated with similar appearances, so to determine a specific student for security reasons can’t be done surveillance cameras only. It will require a high quality image to determine by whom a nuisance has been created.

* + - 1. **Large Organizations**

When fully developed, the system can be used as part of the law/security office of banks, malls and airports, and the detection of suspicious activities. It will also help security agency of any organisation to prevent or protect the organisation.

* + - 1. **Academic Organizations**

In order to scrutinize a large area such as a university campus or school class rooms, this system can be implemented easily as mostly the basic surveillance infrastructure is already installed in these places but places like these are heavily populated with similar appearances, so to determine a specific student for security reasons can’t be done surveillance cameras only. It will require a high-quality image to determine by whom a nuisance has been created.

* + 1. **Design and Implementation Constraints**

The organizations that are not willing to make a huge investment in laying the foundation for this system and also creating an open UI which users can use with even their day to day uses considering the demanding nature of the processing required for it to work efficiently.

* + 1. **Assumptions and Dependencies**

-There should be an Internet link that is secure and fast in order to transmit data fast and realtime.

- The device's operating system should be up to date and at least meeting the minimum requirements of this system.

* + 1. **Requirement Specification**
       1. **User Interfaces**

Currently the system is not ready to be used by the user. But it can be implemented by providing an interface where the user has to input a image and it will get super resolution image from a low resolution image.

* + - 1. **Hardware Interfaces**

The system would require a high-end computing device along with a powerful GPU capable of running inference on real-time video frames.

* + - 1. **Software Interfaces**

The system would be running using any sub-version of Python 3 on any operating system.

* + 1. **System Features or Functional Requirements**
       1. **Introduction**

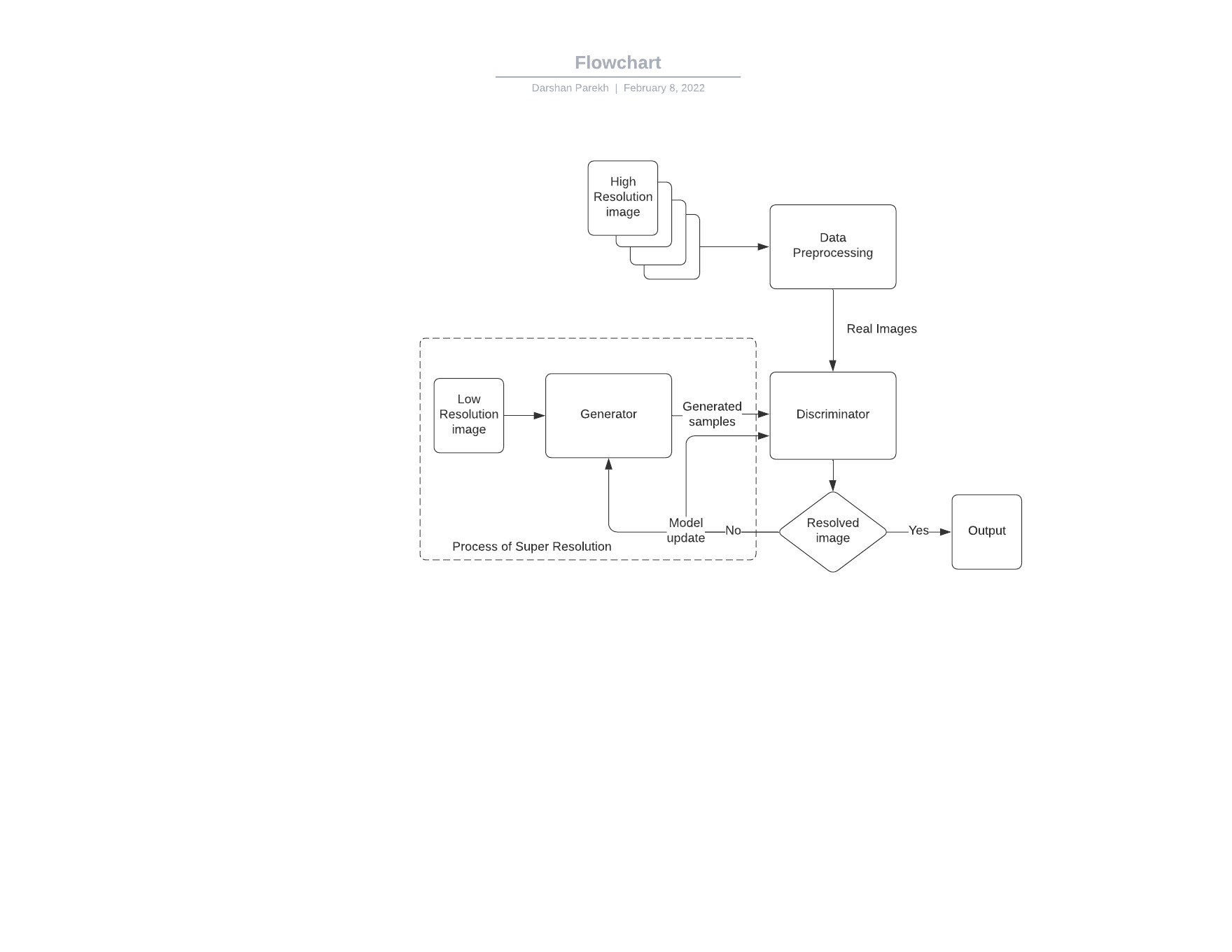
The aim of this system is to generate super resolution images with the help of high-resolution images using deep neural network. To understand this easily, suppose you have an image whose quality is degraded over time and you want to preserve the image. This system will help you in such a manner that it will take that image as an input and on basis of its trained model, it will generate a super resolute image in which the features of the input image will be enhanced.

In the proposed system, the important component is Generative adversarial network (GAN), ,which consist of two models generator and discriminator. The generative model is pitted to confront the opponent which is to fool the discriminator and train two models at the same time: the generator G that catches the distributed data and discriminator D evaluates that the sample comes from training dataset instead of generated image by the generator G. The process training of G is to improves the likelihood of D mistake. The proposed frame correlates with a very small and huge game with two-player.

* + - 1. **Input**
  + Images

As a user, only input that it has to provide is images as the entire system is based on the generating super resolution image with the help of low-resolution images. The model is trained with various datasets of different category which makes it capable to deal with any kind of image the user will provide. The dataset will contain a high resolution image which helps the model to be trained. So with the help of high resolution image, the model is able to generate a super resolution image.

* 1. **Flowchart**



*Fig:3.2 Flow chart of SRGAN*

* 1. **Design and Test Steps**
     1. **Design**

**-** The project focusses to increase the small size images to high-resolution images while keeping drop quality to a minimum. This has various applications from satellite image analysis to medical image processing etc.

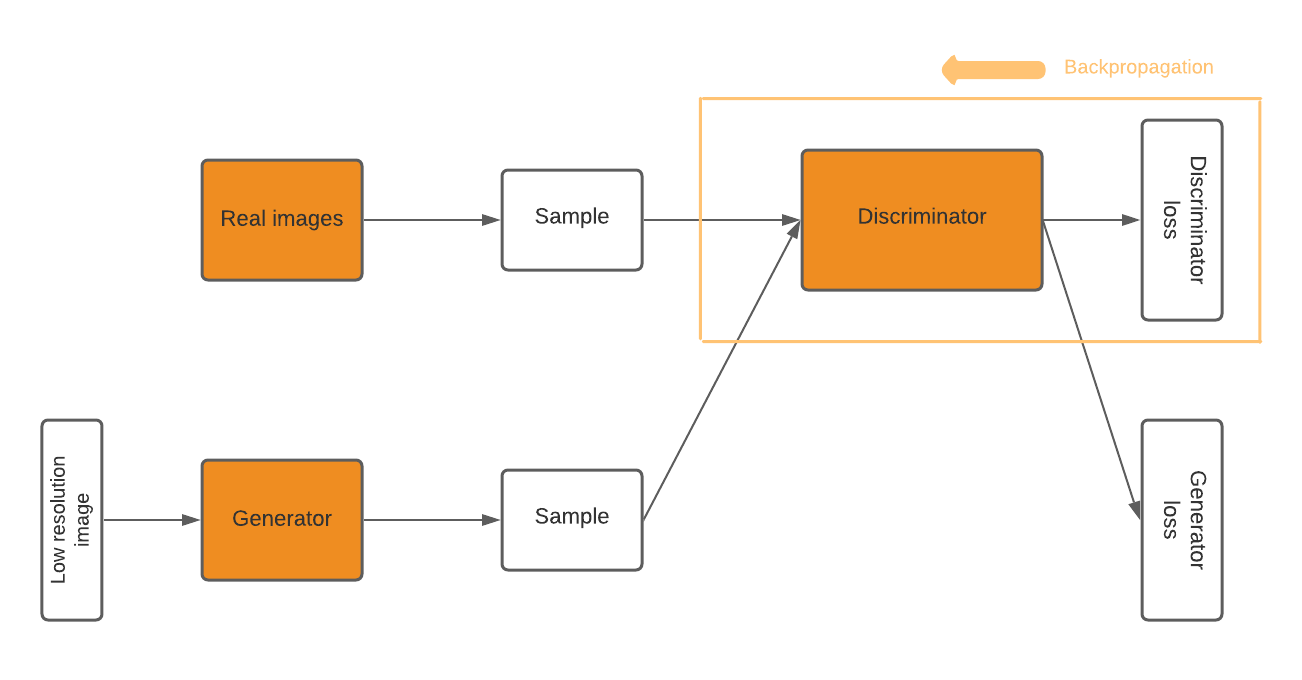
- SRGAN: Super Resolution GAN consists of a deep network with an adversary network to generate high-resolution images. The architecture consists of two Neural Networks,

* Generator that generates plausible data
* Discriminator is simply a classifier and tries to distinguish real data and generator’s data

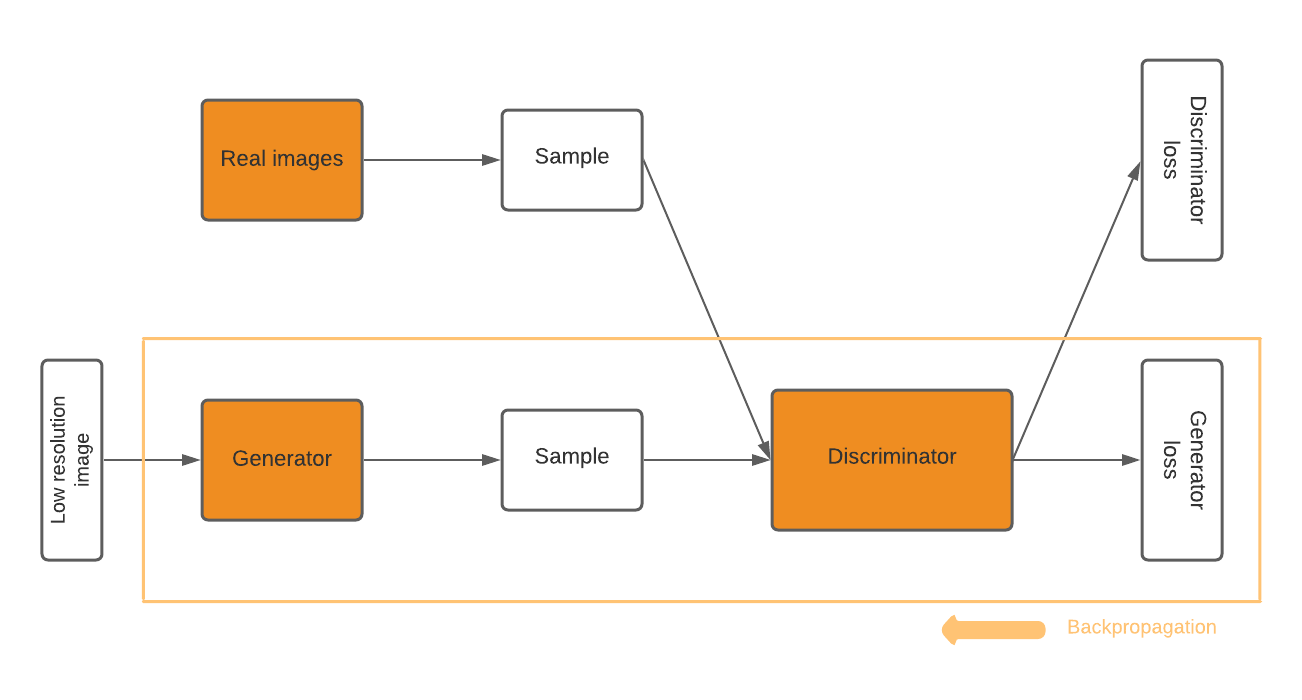
- This architecture uses the alternating training method which means that both Generator and Discriminator models have distinguished training processes.

1. Discriminator is trained for one or more epochs.
2. Generator is trained for one or more epochs.
3. Step 1 and 2 are repeated.

* GAN tries to replicate probabilistic distribution, which means the loss function of both networks reflects the distribution distance between the generated and real data. Both of these networks have their own loss functions, Generator loss function penalizes the generator for generating data that the discriminator distinguishes as fake. Through backpropagation, each weight is adjusted in the valid direction by evaluating the weight’s impact on the output. Backpropagation starts at the output to flows back to the discriminator to the generator. During this generator training discriminator is constant.
* Simultaneously, the training of the discriminator continues with the training phase of the generator. Any other way, the generator would be attempting to hit a moving objective and may never combine.



*Fig 3.3.1.1 Training of Discriminator model*



*Fig 3.3.1.2 Training of Generator model*

* This back-and-forth approach of the model allows GANs to attempt to solve difficult generative problems to simple classification problems.
  + 1. **Dataset Used**
* **BSD (Berkeley Segmentation Dataset)**
* There is total 300 images which includes 200 for training and 100 for testing purposes.
* It include huge variety of images ranging from natural images to object specific images for example, plants and foods etc.
* There are color images as well as grayscale images.

A collage of pictures

Description automatically generated with low confidence

*Fig 3.3.2.1 Images of BSD300*

* **Urban 100**
* It contains 100 images.
* Images are of urban scenes.
* All the images are colorful.
* It is used for training purposes.
* A collage of buildings

  Description automatically generated with low confidence

*Fig 3.3.2.2 Images of URBAN100*

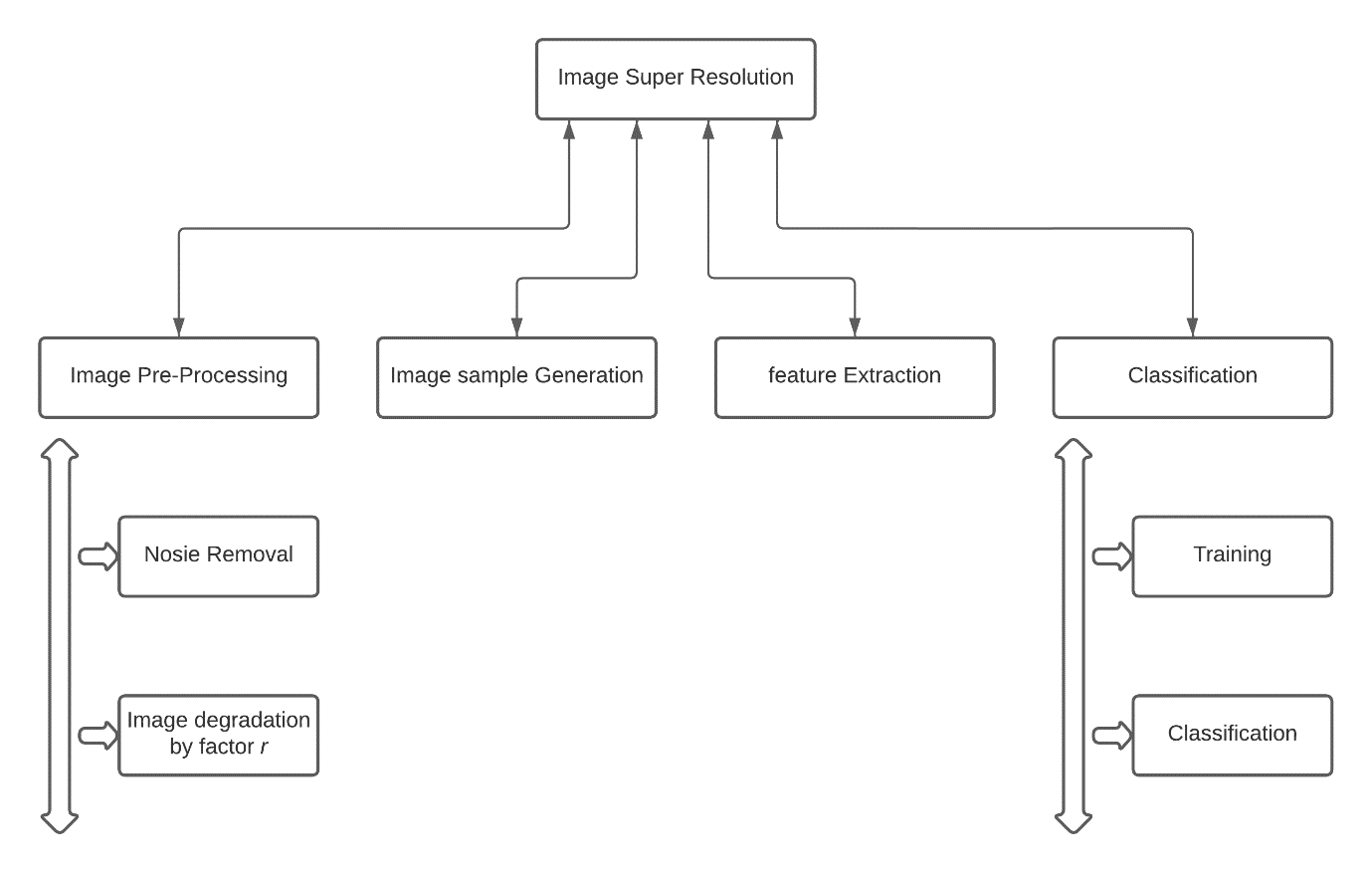
* ImageNet dataset
* It contains 1 million images and 1,000 object classes
* It includes 50,000 for a validation dataset and 150,000 for a test set.
* It is set of manually annotated training images
* A picture containing shop, sale

  Description automatically generated

*Fig 3.3.2.3 Images of ImageNet*

* + 1. **Breakdown structure**

The purposed work is split into following modules as shown in Fig:



*Fig 3.3.3 Breakdown structure of system*

1. Image Preprocessing

The goal of pre-processing is to improve the image's quality so that we can analyse it more effectively. We can suppress unwanted distortions and enhance some features that are required for the specific application we are working on by preprocessing. These characteristics may differ depending on the application.

For example, set of algorithms are used to process image, such as image inversion, reshaping, thinning etc. so that it is easier to train the model

1. Image sample generation

In the process, the goal is to provide the generator with data as images so to train itself. The generator model takes degraded image or pre-processed as input and generates samples.

1. Feature extraction

Feature extraction is a step in the data preprocessing process that divides and reduces a large set of raw data into smaller groups. As a result, processing will be simpler. The fact that these large amounts of data have a large set of variables is the most essential characteristic of them. These variables necessitate a significant amount of computing power to process. As a result, feature extraction assists in retrieving the best features from large data sets by choosing and incorporating variables into features, significantly lowering the amount of data. These features are simple to process while accurately and uniquely describing the actual data set.

To implement feature extraction, we have used VGG19 which is a CNN (convolutional neural network) with 19 layers deep.

1. Classification

The process of categorising and labelling clusters of pixels or vectors inside a photo related to specific rules is known as image classification. One or more spectrograph or textural characteristics could be used to develop the classification law.

* + 1. **Test Steps**

-Test the consistency of the data set.

-The test system meets our minimum requirements.

-Test whether the compiler is installed correctly. (Python in our case)

-Test that all required libraries are installed in the system.

-The test code is being executed and no errors were encountered.

-Test whether the dataset is imported correctly.

-Test whether all preprocessing is successfully implemented on the data set.

-Pass our algorithm/model test to ensure the shortest training time.

-Test whether the model is successfully trained.

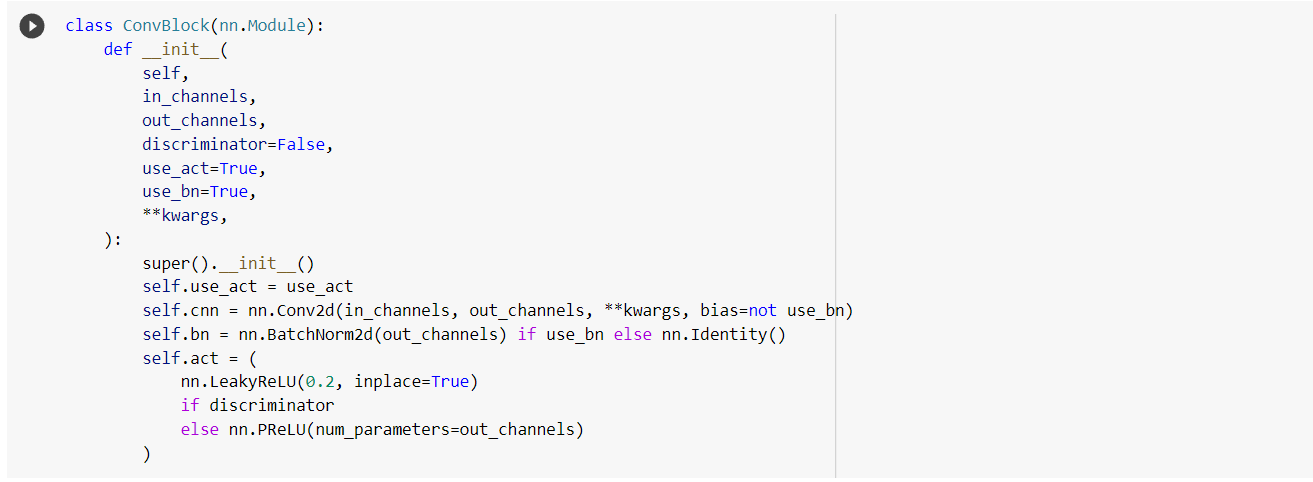
-Test training loss.

-Test to verify the loss.

-Test the accuracy of our proposed model on sample data.

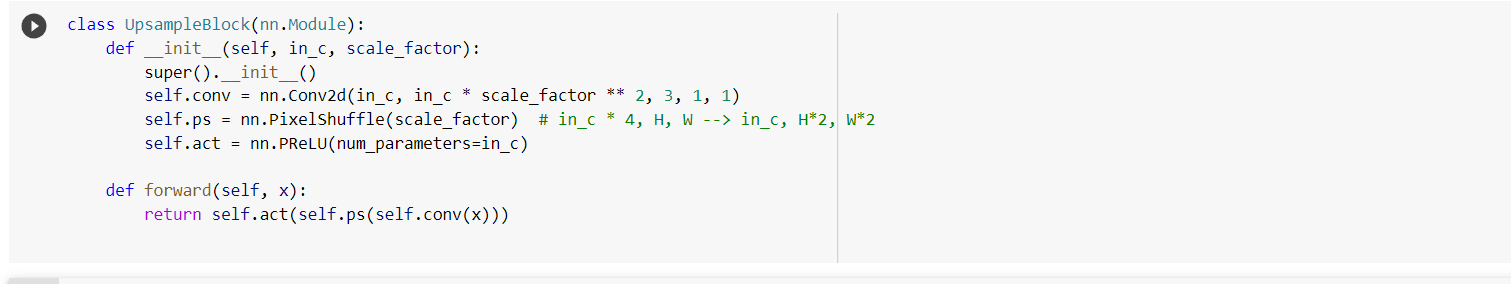
-Any system abnormalities encountered during test execution

* 1. **Algorithm and pseudocode**



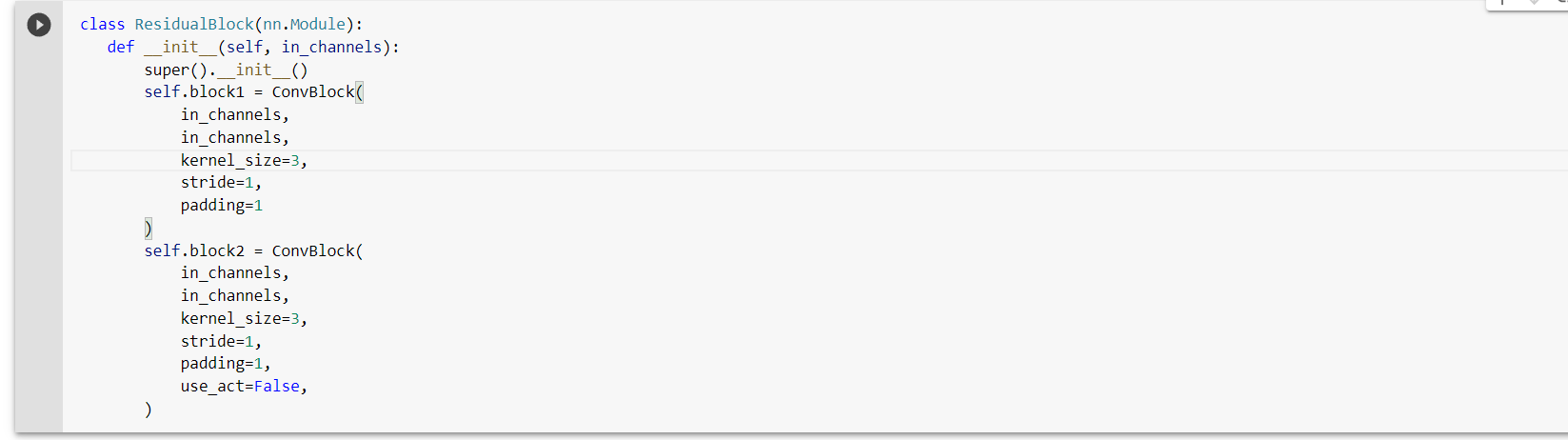
*Fig 3.4.1 Convolutional Block*

* In case of Generator, it consists of convolutional layer, Batch Normalization, and PReLu activation function.
* In case of Discriminator, it consists of Convolutional layer, Batch Normalization, and LeakyReLu activation function.



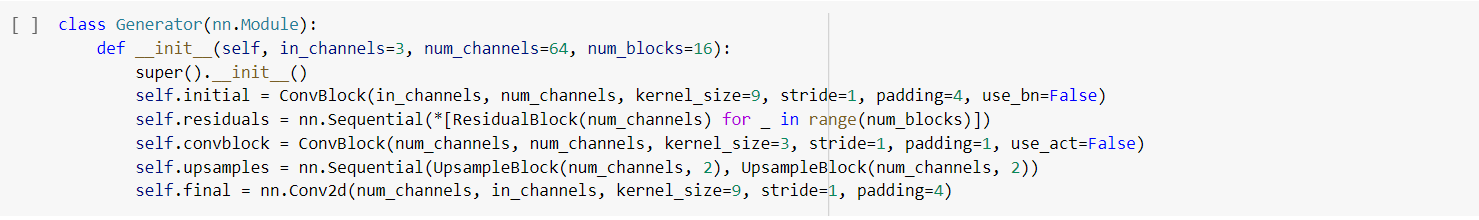
*Fig 3.4.2 Upsample Block*

* This block upscales image \* 4 times, height \*2 and width \*2.



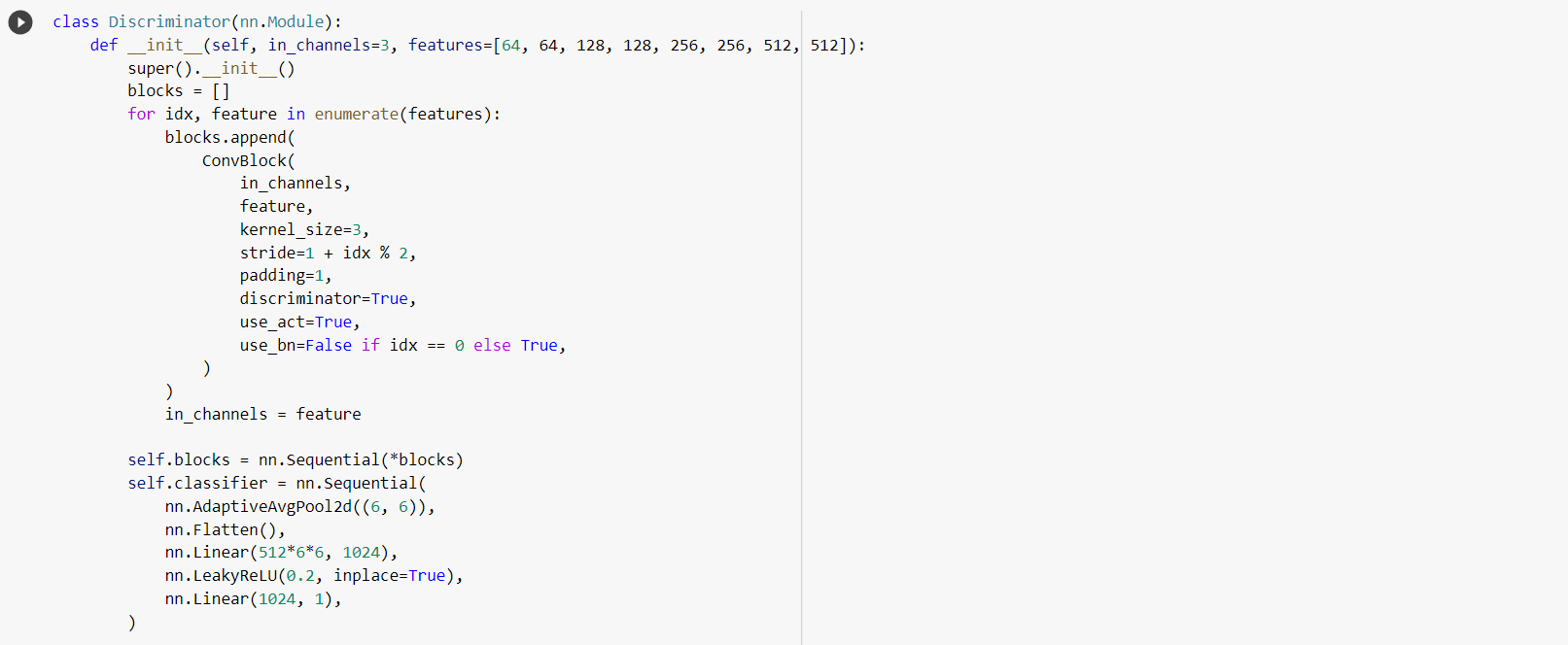
*Fig 3.4.3 Residual Block*

* There are two blocks in Residual block defining kernel size, padding, stride and activation function with a skip connection.



*Fig 3.4.4 Generator Block*

* We define Generator model with initial parameters (kernel, padding, strides and activation) followed by Residual blocks in sequencial which is 16 times and convolutional block and upsampling.



*Fig 3.4.5 Discriminator Block*

* In this we define features and block list which contains kernel size, padding and activations for discriminator followed by adaptive pool and LeakyReLu activation function.

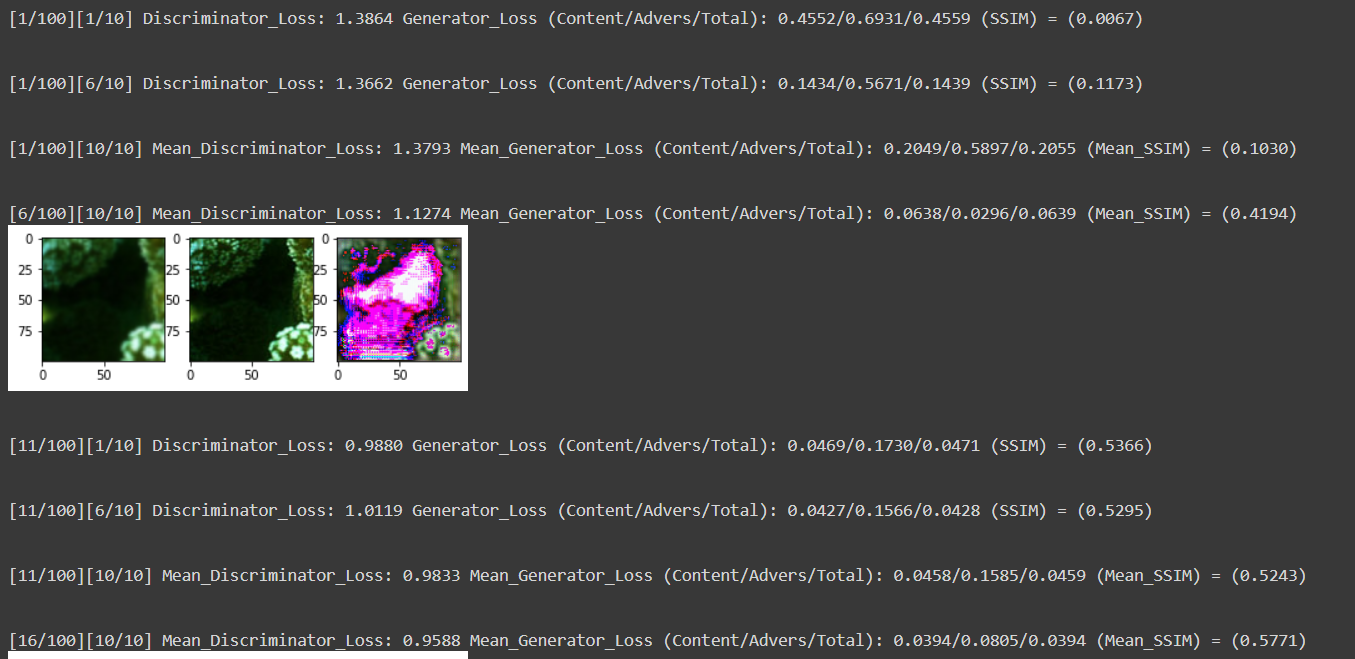
1. **RESULTS OF SRGAN**

**4.1 During Training:**

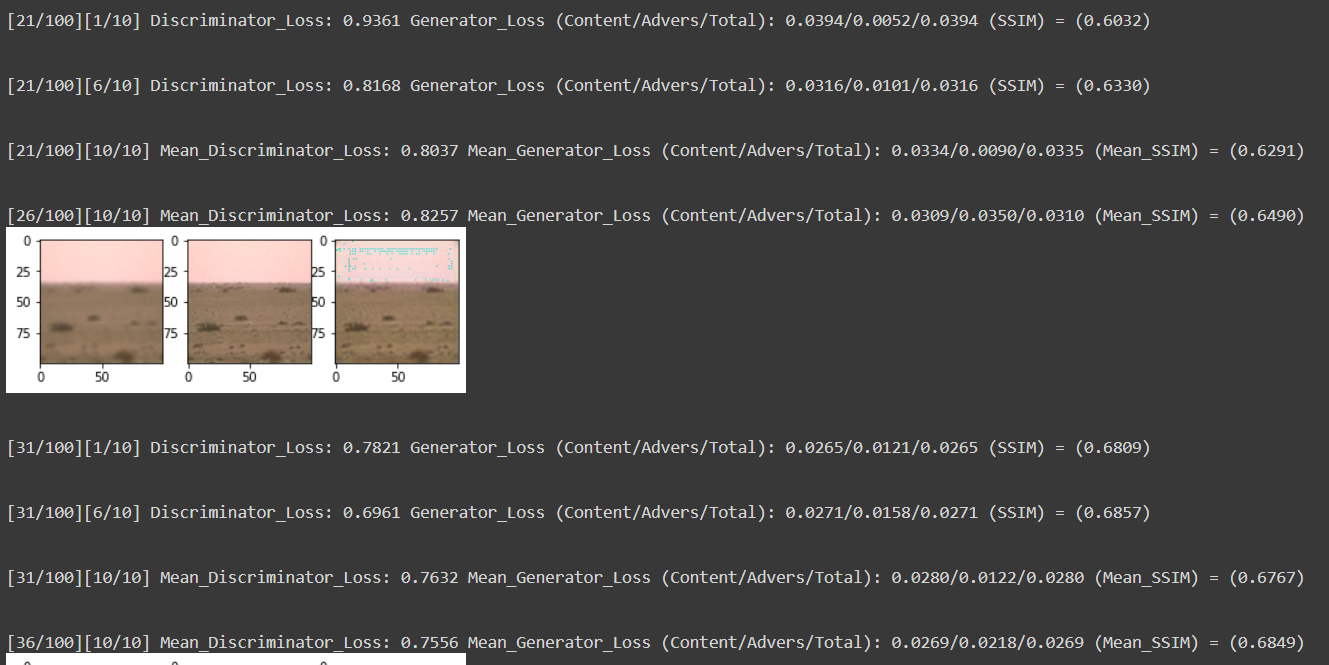
Result of the SRGAN is determined by 3 factors:

* + 1. Discriminator loss
    2. Generator loss
    3. SSIM
* Discriminator loss: The loss function which describes the probability that the generator correctly classifies the real image.
* Generator loss : This function is maximised by the generator. To put it another way, it tries to maximise the discriminator's outcome for its fake occurrences. It includes Content loss, Adversial loss and Perceptual loss.
* Content loss: This loss function determines VGG loss which is the Euclidean distance between the feature representations of a reconstructed image and the reference image.
* Adversarial loss: The loss function is based on the probabilities of the discriminator’s overall training samples.
* Percetual loss: The loss function examines the solutions associated with related characteristics.It is the weighted sum of both perceptual loss and adversarial loss.
* SSIM: To perform comparative analysis quality between two images relies on estimated errors between truth image and super resolute image.  
  The Structural Similarity Metric calculation is based on three factors:
* Luminance
* Contrast
* Structure

As the training of the GAN proceeds, it is observed that the discriminator loss and generator loss is gradually reduced which means that the generator model is able to generate images that is able fool the dicriminator. Also the value of SSIM increases gradually which indicates that similarity between the truth image and super resolution image also increases.



*Fig 4.1.1 Output from training*



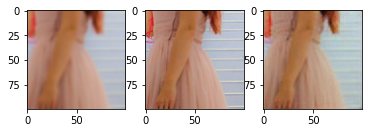
*Fig 4.1.2 Output from training*

As show in the figure, it observed at a certain point during the training, the values are as follows

* Discriminator loss is 0.7010
* Generator loss is 0.0269
* Similarity measure 0.6849

**Visualizing predictions**

The following are the 6 photos from the training stage, along with their predictions:-



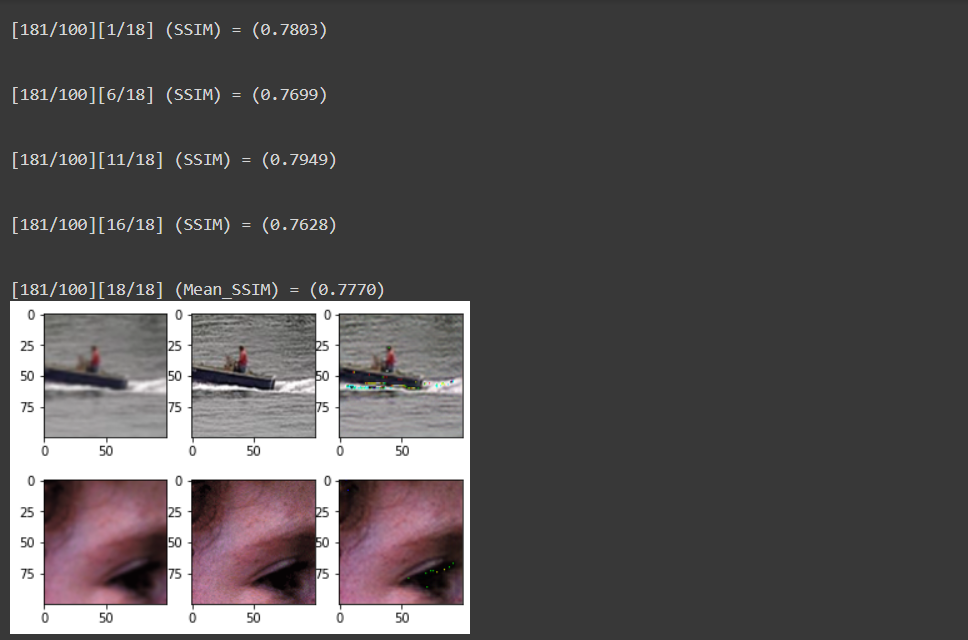
1. *Low resolution image ii) High Resolution image iii) Super resolution image*
2. A picture containing indoor

   Description automatically generated*Low resolution image ii) High Resolution image iii) Super resolution image*
3. *Low resolution image ii) High Resolution image iii) Super resolution image*

*Fig 4.1.3 LR, HR and SR images from traing*

* 1. **During Testing**

Similarity measure was ranging from 0.6969 to 0.8786 on average 0.770 through testing process.



*Fig 4.2.1 Output from testing*

**Visualizing predictions**

The following are the 6 photos from the Test stage, along with their predictions:-



1. *Low resolution image ii) High Resolution image iii) Super resolution image*



1. *Low resolution image ii) High Resolution image iii) Super resolution image*

*Fig 4.2.2 LR, HR, and SR images from testing*

* 1. **Graphical Representation of result**

**4.3.1 Discriminator vs generator loss graph**

* Discriminator loss: The loss function which describes the probability that the generator correctly classifies the real image.
* Generator loss : This function is maximised by the generator. To put it another way, it tries to maximise the discriminator's outcome for its fake occurrences.

Graphical user interface

Description automatically generated with medium confidence

*Fig 4.3.1 Discriminator vs generator loss graph*

* Generative loss includes Content loss, Adversial loss and Perceptual loss.
* Content loss: This loss function determines VGG loss which is the Euclidean distance between the feature representations of a reconstructed image and the reference image.
* Adversarial loss: The loss function is based on the probabilities of the discriminator’s overall training sample.
* Percetual loss: The loss function examines the solutions associated with related characteristics.It is the weighted sum of both perceptual loss and adversarial loss.

As the training of the GAN proceeds, it is observed that the discriminator loss and generator loss is gradually reduced which means that the generator model is able to generate images that is able fool the dicriminator.

* + 1. **Structural Similarity Index Measure Graph**
* SSIM: To perform comparative analysis quality between two images relies on estimated errors between truth image and super resolute image.  
  The Structural Similarity Metric calculation is based on three factors:
* Luminance
* Contrast
* Structure

A picture containing text, electronics, screenshot

Description automatically generated

*Fig 4.3.2 Similarity measure graph*

As observing the graph, the value of SSIM increases gradually with respect to increase in epoch which indicates that similarity between the truth image and super resolution image also increases.

**4.4 Output of the model**

In the given figure, it shows that after 100 epoch the following are the result

- mean discriminator loss = 0.7166

In generator loss ,

-Content loss = 0.0176

-Adversarial loss = 1.9451

-Perceptual loss = 0.0217

-SSIM = 0.7599

Text

Description automatically generated

*Fig 4.4 Final Output*

1. **CONCLUSION**

In this paper we have devised image super resolution architecture via Generative Adversarial Networks (GAN) based on a combination of existing reliable Neural Network architectures, namely Residual Neural Networks (RNNs). The spatial features of an RGB image are extracted using VGG network which is the Euclidean distance between the feature representations of a reconstructed image and the reference image. In this model, the addition Structural Symmetry measure ensures similarity between generated data image and real image. Though out the training and testing phase our model attained plausible numbers (Training phase 89% and testing phase average 78%) which is comparable with the state-of-the-art models.

**5.1 Future Improvement**

To improve our model even further, we can introduce the addition of extra training data which in most cases have shown to increase accuracy. Further improvements can also be made by adding other popular datasets and combining them, adding custom data and tuning hyperparameters. The developers of this project eventually wanted to add real-life use case for this architecture but for now model is purely for research purpose.

# REFERENCE

1. T.-A. Song, S. Chowdhury, K. Kim, K. Gong, G. El Fakhri, Q. Li, and J. Dutta, “Super-resolution PET using a very deep convolutional neural network,” in Proc IEEE Nucl Sci Symp Med Imag Conf. IEEE, 2018.
2. Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, Bryan Catanzaro, “High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs” in CVF Conference on Computer Vision and Pattern Recognition, IEEE, 2018.
3. Karthika Gopan,and Kumar G.S, “Video Super Resolution With Generative Adversarial Network” in 2nd International Conference on Trends in Electronics and Informatics (ICOEI) ,2018.
4. C. Ledig, L. Theis, F. Huszar, J. Caballero, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, “Photo-realistic single image super-resolution using a generative adversarial network,” CoRR, vol. abs/1609.04802, 2016.
5. Xin Yu, Basura Fernando, Richard Hartley, and Fatih Porikli, “Super-Resolving Very Low-Resolution Face Images with Supplementary Attributes” in Conference on Computer Vision and Pattern Recognition, IEEE, 2018
6. Xining Zhu , Lin Zhang , Lijun Zhang , Xiao Liu , Ying Shen , and Shengjie Zhao, “GAN-Based Image Super-Resolution with a Novel Quality Loss” Mathematical Problems in Engineering, Hindawi, vol. 2020
7. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. “Generative adversarial nets” in Advances in Neural Information Processing Systems (NIPS), pages 2672–2680, 2014.
8. W. T. Freeman, T. R. Jones, and E. C. Pasztor. “Example-based super resolution” in IEEE Computer Graphics and Applications, 22(2):56–65, 2002.
9. M.-Y. Liu and O. Tuzel. “Coupled generative adversarial networks”. In *Advances in Neural Information Processing Systems* *(NIPS)*, 2016.
10. M. Mirza and S. Osindero. “Conditional generative adversarial nets”. *arXiv preprint arXiv:1411.1784*, 2014.
11. J. Kim, J. Kwon Lee, and K. Mu Lee, “Accurate image super-resolution using very deep convolutional networks,” in *Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit*, 2016, pp. 1646–1654.*.*
12. J. Kim, J. K. Lee, and K. M. Lee. “Deeply-recursive convolutional network for image super-resolution” In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
13. A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. “Learning from simulated and unsupervised images through adversarial training” In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
14. X. Wang and A. Gupta. “Generative image modeling using style and structure adversarial networks” In *European Conference on Computer Vision (ECCV)*, 2016.
15. I. Durugkar, I. Gemp, and S. Mahadevan. “Generative multiadversarial networks” In International Conference on Learning Representations (ICLR), 2016.
16. P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. “Image-to-image translation with conditional adversarial networks”. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
17. J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan. “Perceptual generative adversarial networks for small object detection”. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
18. H. Zhang, T. Xu, H. Li, S. Zhang, X. Huang, X. Wang, and D. Metaxas. “StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks”. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
19. J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. “Unpaired image-to-image translation using cycle-consistent adversarial networks”.In IEEE International Conference on Computer Vision (ICCV), 2017.
20. A. Radford, L. Metz, and S. Chintala. “Unsupervised representation learning with deep convolutional generative adversarial networks”. In International Conference on Learning Representations (ICLR), 2015.
21. T. Salimans, I. Goodfellow,W. Zaremba, V. Cheung, A. Radford, and X. Chen. “Improved techniques for training GANs”. In Advances in Neural Information Processing Systems
22. (NIPS), 2016.J. Zhao, M. Mathieu, and Y. LeCun. “Energy-based generative adversarial network”. In International Conference on Learning Representations (ICLR), 2017.
23. Z. Yi, H. Zhang, P. T. Gong, et al. “DualGAN: Unsupervised dual learning for image-to-image translation”. In IEEE International Conference on Computer Vision (ICCV), 2017.
24. Guim Perarnau, Joost van de Weijer, Bogdan Raducanu,and Jose M. Álvarez “Invertible Conditional GANs for image editing” In Workshop on Adversarial Training, NIPS 2016, Barcelona, Spain.
25. X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley. “Least squares generative adversarial networks”. In IEEE International Conference on Computer Vision (ICCV), 2017.
26. Wenming Yang, Xuechen Zhang, Yapeng Tian, Wei Wang, and Jing-Hao Xue. “Deep learning for single image super-resolution” IEEE Transactions on Multimedia, 2019