

Super Resolution: A Simplified Approach Using GANs

Dr. Naveena C¹, Thanush M², Vinay N.B.³, Yaser Ahmed N.⁴

¹ Professor, Department of Computer Science and Engineering, SJBIT, Bengaluru, 560060

^{2,3,4} Student, Department of Computer Science and Engineering, SJBIT, Bengaluru, 560060

Abstract:- The project entitled “Super-Resolution: A simplified approach using GANs”, aims to construct a Higher Resolution image from a Lower Resolution image, without losing much detail. In other words, it is a process of up-sampling of an under-sampled image. In the current scenario, there is a high reliance on hardware improvements to capture better high-resolution images. Although many digital cameras provide enough HR-imagery, the cost to construct and purchase such a high-end camera is high. Also, many computer vision applications like medical imaging, forensic and many more still have a strong demand for higher resolution imagery which is likely to be exceeded by the capabilities of these HR digital cameras we have today. To cope with this demand, a method to generate an HR image is shown using Generative Adversarial Networks (GANs).

Keywords:- Super resolution, Deep learning, GANs, Image enhancement.

I. INTRODUCTION

Quality images are required in every field of research to study aspects accurately. With the advancement in Deep Learning, it has made estimation operations more reliable. This advancement can be leveraged on to estimate a High-Resolution image from a Low-Resolution image, which is generally called Super Resolution process. To better understand the need for Super-Resolution, let's talk about the times when 1-megapixel or 2-megapixel cameras were integrated into our PDAs and were extremely famous. When we look back, the images captured with them have significantly fewer details and clarity when compared to today's PDA cameras. The image might look desirable when viewed on a smaller screen. The problem arises when we want to project the same image on a bigger screen. The distortions in the image get enhanced when the small low-resolution image is zoomed or projected to fit a bigger screen. The output is a blurry and an undesirable image. This is where Super Resolution comes into play, wherein we take the small image and make it larger without losing much of the detail.

On a wider aspect, Super-Resolution can be used in research wherein a small single image or a small part of a bigger image can be used as input to generate a larger high-

resolution image having better clarity and help in studying the image more accurately. This can be valuable in medical research in endoscopic treatments, in satellite imaging, etc. This can also be a valuable integration into the Number Plate detection system, wherein the vehicle of interest is a small region of a large image captured by CCTV, super-resolution can be applied to the region of interest for accurate identification.

A question that may be lingering here is, “Why not use better cameras to capture high-quality images?”. Yes, you can use a better camera but there are certain factors that play a critical role in different use cases. In a normal smartphone, better capability cameras are already being integrated but anyone would notice that the cost, on the whole, is significantly increased. For every smartphone manufactured, it's camera component must also be manufactured and integrated. Imagine an alternative scenario, where instead of a high-end camera, a relatively lesser capability camera is integrated into the smartphone, and if the image is required to be displayed on a bigger screen or zoomed it can be enhanced using Super-resolution and displayed. The cost of manufacturing here is significantly brought down for nearly the same user experience. The same analogy can be applied to other systems where high-end cameras are of high importance.

An arguable debate can be made here about the processing power required for super-resolution. Today's smartphones are powerful enough to run super-resolution on its own, while other sectors, like number plate detection can put up a client-server architecture with a Super-Resolution server and calling it through an API to enhance the given image.



Fig 1:- Low Resolution (LR) Image(left) is enhanced to a high-resolution image(right) using super-resolution (SR). It can be closely observed that the SR image has more texture and clarity as compared to that of the LR image.

II. CHALLENGES

- Computational Efficiency – Just like any other GAN implementation, Super-resolution also requires to undergo a training phase with enormous amount of data. This will take indefinite amount of time, plus supervision is required during training to avoid overfitting of data.
- Robustness aspects – Images usually contain noise or outliers and the model generated must minimize the up-sampling of such. This requires the training images to have sufficient noise just so that the neural network can differentiate outliers and provide a better model.
- Performance – The prediction or estimation of HR image from a LR image must be fast enough to satisfy the user. Long waiting time in generation of output might result in a user losing interest.

III. OBJECTIVES

- To provide an economical method to deliver a high-resolution image.
- Lower the burden on hardware improvements.
- Provide a better scope for research.

IV. METHODOLOGY

One of the more common ways used to tackle the defined problem statement is using the bicubic interpolation technique. However, the formerly mentioned approach fails to give results that appear to be natural when perceived by human senses. Therefore, to address this issue, we have used Generative Adversarial Networks (GANs), a deep learning technique to get the super-resolution (SR) image from a given low resolution (LR) image. The methodology implemented is described in the following sections.

➤ *GANs:*

Generative Adversarial Networks are a class of advanced neural networks that was introduced by Ian J. Goodfellow in 2014. These neural networks consist of mainly two underlying neural networks, namely a Generator and a discriminator.

The role of the generator network is to “generate” the result desired by the user and the role of the discriminator networks is to monitor whether the generator network is producing the results as expected.

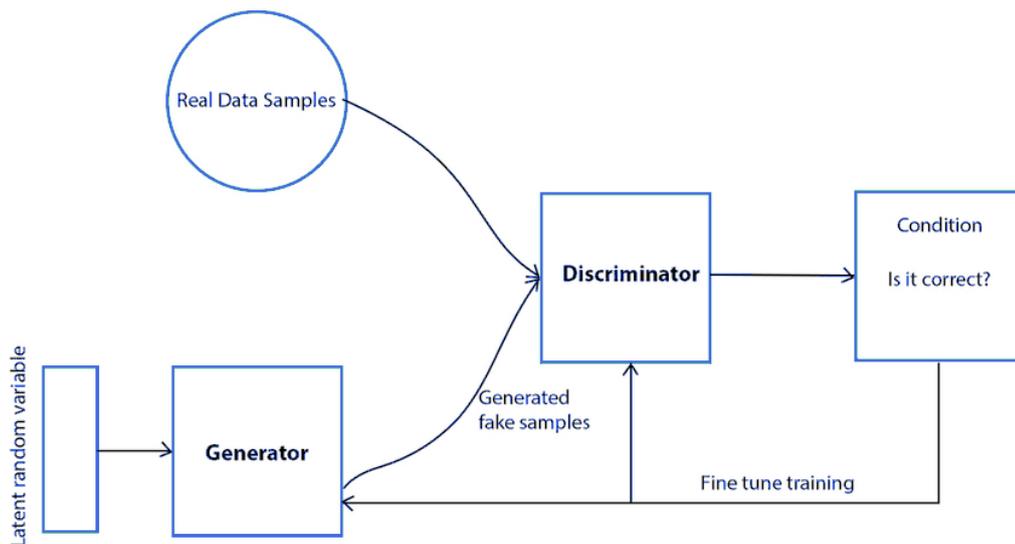


Fig 2:- Block diagram of a basic GAN network

In other words, the discriminator “discriminates” between the actual data and the result that has been generated by the generator. Fig 2 is a basic block diagram showing the various components in a GAN network.

➤ *Using GANs for Super resolution:*

• *Training the model:*

We have used the Div2k dataset for the project. Div2k Dataset is a large dataset of 1000 RGB images with a large variety of contents. The dataset is itself divided into train

data, validation data and test data. The train dataset consists of images of size two-thousand resolution each. For the purpose of this project, 600 images were used to train the model and 200 to test the generated model. The images were initially down sampled to size of 384px by 384px for our model.

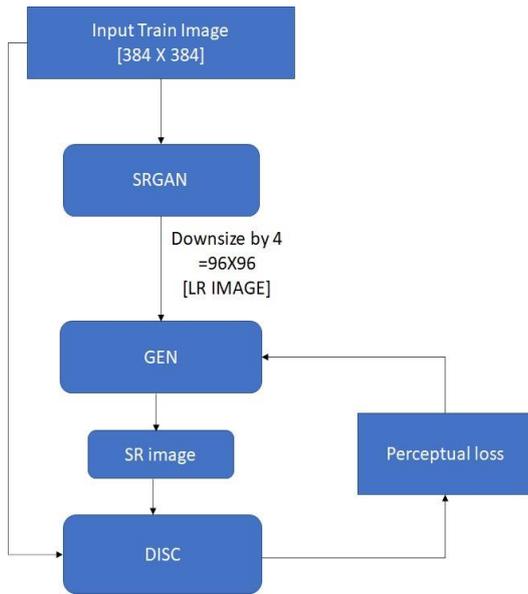


Fig 3:- Training the GAN network

Fig 3. shows the basic block diagram of the training phase for the designed GAN network as described in this section.

• *The GAN neural network:*

Residual blocks are used in the building of the generator network since the training difficulty increases with the increase in the number of deeper layers in the networks. 16 residual blocks are uses in the Generator network. Keras library APIs are used in the implementation of various activations like Parameterized Relu (PRelu). PRelu helps in the adaptive learning of the negative part off the coefficient.

For the loss function, Perceptual loss is being used. Perceptual loss function which is a weighted sum of both the content loss and adversarial loss. The content loss is of two types. The first being Pixel-wise MSE loss which calculates how different the generated image is from the actual image. The second is the VGG loss. VGG19 is a popular deep neural network that is dominantly used for image classification. The pretrained VGG networks are used for the feature extraction of the real and the generated images. For our implementation of the concept, VGG loss function is utilized. The adversarial model is implemented to monitor the images generated by the generator. By minimizing the overall perceptual loss, we can make the generator start to produce more realistic looking images.

• *Testing the Generator Model:*

After the model described in the section 4.2.2 has been sufficiently trained with the dataset described in the section 4.2.1, we can start to test the model that is generated. Figure 4 is a flow chart diagram that shows the basic steps involved in the testing of the trained generator model.

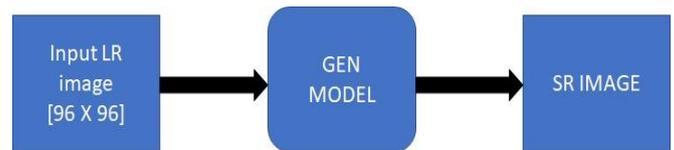


Fig 4:- Testing the generator model

The following image (fig 5) show some of the results obtained. The right most image of the three show the real image which is fed into the model (size 384px by 384px). The left most image in fig 4. is the down sampled version of the original image. The image is down sampled by a factor of 4 resulting in the image to be of the size 96px by 96px. The image in the middle is the image produced by the generator network for the input image of size 96px by 96px. The resultant image is of the size 384px by 384px with enhanced details compared to the low-resolution input image. It is clearly observed that the generator model is generating the image that is the enhanced version of the input image and is looking in close similarity to the expected original image.



Fig 5:- Testing the generator model

V. CONCLUSION

The described network and training processes were designed keeping a generic purpose in attention. However, the model can be scaled appropriately to different specific domains by training it on a more accurate and larger dataset with higher number of epochs. Using a generic model for a specific use case might not yield expected results. Instead, the model can be trained for a specific use case and dataset.

Doing so will help the generator to better predict the output for a given input image of a specific kind by increasing the accuracy in the estimation for the High-Resolution image.

REFERENCES

[1]. Christian Ledig, Lucas Theis, Ferenc Husz’ar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, Wenzhe Shi Twitter. “Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network.”, 2017 IEEE Conference on Computer Vision and Pattern Recognition.

- [2]. Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. “ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks.” ECCV 2018 workshop paper.
- [3]. Bobby Lv, Yaozhong Liu, Sherine Zhang, Hansi Zeng, Guangfei Zhu. “Super Resolution with Generative Adversarial Networks”.
- [4]. Generative Adversarial Networks Projects: Build next-generation generative models using Keras and tensorflow. A book by Kailash Ahirwar
- [5]. Alind Gupta, Delhi Technological University (DTU), Building a Generative Adversarial Network using Keras <https://www.geeksforgeeks.org/building-a-generative-adversarial-network-using-keras/>