Single Image Super-Resolution Using a Generative Adversial Network

Reethu Jahnavi Veeramallu
Computer Science and Engineering
Indian Institute of Technology BHU
Varanasi, India
vreethu.jahnavi.cse21@itbhu.ac.in

Neerudu Nikhil
Computer Science and Engineering
Indian Institute of Technology BHU
Varanasi, India
neerudu.nikhil.cse21@itbhu.ac.in

***Abstract*—In this report, we explore through an implementation of a model with remarkable capabilities of super-resolving of fine texture details while up scaling through large factors. This edge-cutting Super-Resolution Generative Adversarial Network *(SRGAN)* model elevates low-resolution images to greater dimensions using sophisticated generator network that produces indistinguishable images. For accomplishing such objective, we implement it using a perceptual loss function which is combination of content loss as well as adversarial loss. Features of minute detailings in the image are recovered from highly down scaled images using the residual network implemented. This model can potentially be applied in fields such as medical imaging, remote sensing, and video processing, where high-resolution images are critical for accurate analysis and interpretation.**

***Index Terms*—Generative Adversarial Networks, Super-resolution, Perceptual loss, Adversarial loss.**

I. Introduction

Revolutionizing the world of convolutionary neural networks (CNN), Super-Resolution stands tall as the greatest game-changer yet, elevating image resolution to unprecedented heights! Yet up scaling the low resolution images that could recover the finer texture details is typically a challenging task. For an efficient super-resolution of down scaled images is monitoring the mean squared error that should be minimised. Peak-Signal to noise ratio (PSNR) and Structural Similarity Index Method (SSIM) are the duo of image comparison metrics, which pack a punch, interpreting unparalleled precision and accuracy in quest of vision perfection. The perceptual difference between the super-resolved and original image means that the recovered image is not photo realistic as defined by Ferwerda [1].

Revitalizing low-quality images has never been easier with our implementation of a Generative Adversarial Network. By leveraging a deep residual network with skip-connections, we achieve unprecedented levels of detail and resolution. Instead of using only pixel-loss to improvise our generator, we also use feature-loss to make the generated images as realistic as possible. The combination of these two losses is called perceptual loss as it considers human perception in terms of a loss. This feature loss or content loss is obtained by comparing the features of the original image and that of the generated image. The features are obtained by using a feature map extracted from a pre-trained CNN model. In this project, we extract the feature map from VGG-19 model pre-trained over ImageNet dataset.

II. Background

In this section, we review background information related to our project.

A. Convolutionary Neural Network (CNN) Model:

CNN is a special type of neural network which is mostly applied to extract relevant features for the image content [2]. A CNN model learns the valuable features automatically for better classification rather than extracting handcrafted explicit features by classical image recognition system [3].

Generally, a CNN model consists of input, output, and numerous hidden layers. Each hidden layer has three main components - the convolutional layer, the pooling layer, and the fully connected layer. In the convolutional layer, raw pixel data is converted into a feature map that detects all different patterns in the input data. While in the pooling layer, the feature map is being downsampled to select the most valuable information. In the fully connected layer, this joint information content is used to classify the given input image into various classes.

B. Generative Adversial Network(GAN):

Generative Adversarial Network(GAN) is a class of machine learning/deep learning models in which two neural networks compete among each other. One network's gain becomes the other network's loss [4]. These have proven to be useful under unsupervised learning, semi-supervised learning, fully-supervised learning and reinforcement learning.

A GAN model consists of two networks - Generator and Discriminator. The generative network generates candidates which the discriminator evaluates. The generative network learns to map from a latent space to a data distribution of interest, while the discriminative network learns to distinguish candidates produced by the generator from the real data distribution. Generative networks are trained to fool discriminator networks by producing novel candidates that are as real as possible, such that the discriminator is unable to distinguish between real and synthetic samples. Discriminative networks are trained to be able to distinguish between real and synthetic samples as accurately as possible.

C. VGG:

VGG stands for Visual Geometry Group. It is a standard deep CNN architecture with multiple layers. The "deep" refers to the number of layers with VGG-16 or VGG-19 consisting of 16 and 19 convolutional layers. The VGG architecture is the basis of ground-breaking object recognition models. In this project, we have used the results of VGG-19 [5].

VGG-19 consists of 19 weight layers in the network. Its network is constructed with very small convolutional filters. The architecture consists of four types of layers - input layer, convolutional layer, hidden layers, fully-connected layers. This model is used in deep learning image classification problems. In this project, we use the features obtained by this model to improvise the generator's performance so that the images are as realistic as original images [6].

D. Residual Neural Network(ResNet):

Deep Residual Neural Networks are used to solve the degradation problem of the deep neural networks. The degradation problem is that when the network depth increases, accuracy gets saturated and then degrades rapidly [7]. The idea of the residual networks is that instead of letting layers learn the underlying mapping, let the network fit the residual mapping. The approach to implement this is to add a shortcut or a skip connection that allows information to flow more easily from one layer to the next's next layer, i.e., we bypass data along with normal CNN flow from one layer to the next layer after the immediate next.

By adding new layers, because of the "Skip connection" or "residual connection", it is guaranteed that performance of the model does not decrease but it could increase slightly. The skip-connections help to address the Vanishing Gradient problem. They also make it easy for a ResNet block to learn an identity function. Two main types of blocks are used in a ResNet, depending mainly on whether the input/output dimensions are same or different - Identity block and Convolution block. Very deep Residual Networks are built by stacking these blocks together [8].

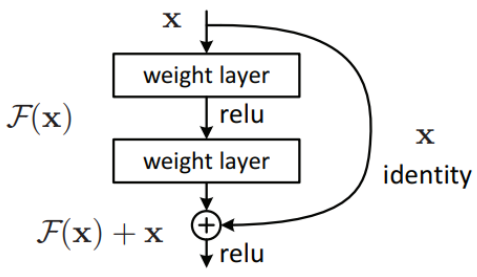


Fig. 1: Residual Neural Network

III. Model

Our aim is to estimate a super-resolution image from a low-resolution input image which is similar to the original high-resolution image. We only have high-resolution images during training. We generate low-resolution images from these high-resolution training images. In training, low-resolution images I^{LR} are obtained by a down sampling operation with a down sampling factor *r* applied on high-resolution images I^{HR}. For an image with C color channels, if the high resolution image is a real valued tensor of size $W \times H \times C$, then I^{LR} is given by $rW \times rH \times C$. The super-resolution image will also have the same size as that of the high-resolution original image.

Our goal is to train a generating function G that estimates a HR image for a LR image which is given as input. To achieve this, we train a generator network as a feed-forward CNN G_{θ_G} parameterised by θ_G. Here, $\theta_G = \{W_{1:L}; b1 : L\}$ denotes the weights and biases of a L-layer deep network which is obtained by optimising a SR-specific loss function denoted by l^{SR}. For training images $I_n^{HR} n = 1, 2, ..., N$ with corresponding $I_n^{LR} n = 1, 2, ..., N$, we solve the equation:

$$\hat{\theta_G} = \arg\min_{\theta_G} \frac{1}{N} \sum_{n=1}^{N} l^{SR}(G_{\theta_G}(I_n^{LR}), I_n^{HR}) \tag{1}$$

In this project, we design a perceptual loss l^{SR} as a weighted combination of several loss components that model distinct desirable characteristics of the recovered SR image so that the recovered image is similar to the original image both in terms of pixels and features [9].

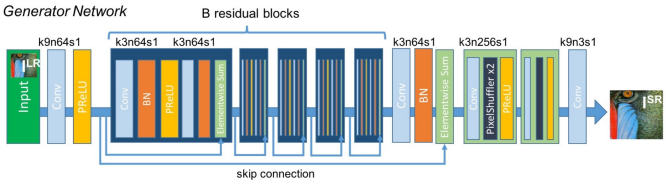
A. Adversial Network Architecture

We define a discriminator network D_{θ_G} which is optimised in an alternating manner along with the generator in order to solve the adversial min-max problem. The basic principle behind this formulation is that it allows us to train a generative model G with the goal of fooling a differentiable discriminator D that is trained to distinguish super-resolved images from real images. With this approach our generator can learn to create solutions that are highly similar to real images and thus making it difficult for D to classify. This encourages perceptually superior solutions to be generated more. This is in contrast to SR solutions obtained by minimizing pixel-wise error measurements, such as the Mean Square Error(MSE).

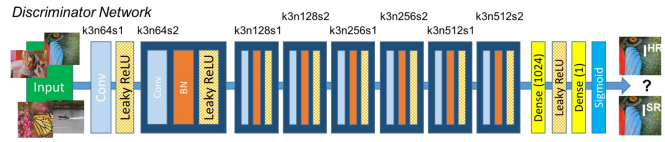
The core of our generator consists of B residual blocks each having identical layout. We use two convolutional layers with small 3×3 kernels and 64 feature maps followed by batch-normalization layers and we use ParametricReLU as the activation function. Furthermore, we increase the resolution of the input image with two trained sub-pixel convolution layers [10].

We train the discriminator network to be able to distinguish real HR images from generated SR images as accurately as possible. In this project, we use LeakyReLU activation ($\alpha = 0.2$) and avoid max-pooling throughout the network. The discriminator network contains eight convolutional layers with an increasing number of 3×3 filter kernels, increasing by a factor of 2 from 64 to 512 kernels as in the VGG network. Strided convolutions are used to reduce the image resolution each time the number of features is doubled. The resulting 512 feature maps are followed by two dense layers and a final

sigmoid activation function in order to obtain a probability for sample classification.



(a) Generator Network



(b) Discriminator Network

Fig. 2: Architecture of Networks with corresponding kernel size (k), number of feature maps (n) and stride (s) indicated for each convolutional layer

B. Perceptual Loss

The definition of our perceptual loss function l^{SR} is critical for the performance of our generator network. While l^{SR} is generally modeled based on the MSE, we improve it and design a loss function that assesses a solution with respect to perceptually relevant characteristics. We formulate the perceptual loss as the weighted sum of a content loss (l_X^{SR}) and an adversarial loss component. It is given as:

$$l^{SR} = l_X^{SR} + 10^{-3} l_{Gen}^{SR} \tag{2}$$

Here, l_X^{SR} is content loss which is based on VGG-19 in our project and l_{Gen}^{SR} is the adversial loss. Together they make the perceptual loss.

1) Content Loss: The most widely used optimisation target for image SR on which many approaches lie is the pixel-wise MSE loss. However while trying to improve the image quality, solutions of MSE optimisation problems often lack high-frequency content which results in perceptually unsatisfying solutions with overly smooth textures. Thus, instead of pixel-wise losses we use a loss function that is closer to perceptual similarity [11].

We define the VGG loss based on the ReLU activation layers of the pre-trained 19 layer VGG network which is pre-trained on the Imagenet dataset. We obtain the feature map of a convolution after activation and before maxpooling layer. We then define the VGG loss as the euclidean distance between the feature representations of a reconstructed image and the reference image.

2) Adversial Loss: In addition to the content loss, we also add the generative component of our GAN to the perceptual loss. This encourages our network to favor solutions that reside on the manifold of natural images, by trying to fool the discriminator network. The generator loss l_{Gen}^{SR} is defined based on the probabilities of the discriminator over all training samples.

IV. Results

We trained our model on the Cattle Face data-set, which contains 3800 images belonging to approximately 340 different classes. We initially start with randomly sized images which are then converted into 32×32 size for the low resolution images set and 128×128 for the high resolution images. The super resolution images are also obtained of the size 128×128. After 225 epochs of training, we achieved satisfactory results. The generator loss was significantly reduced from 110 to a value of 3, indicating that our approach was successful in producing high-quality super-resolution images.

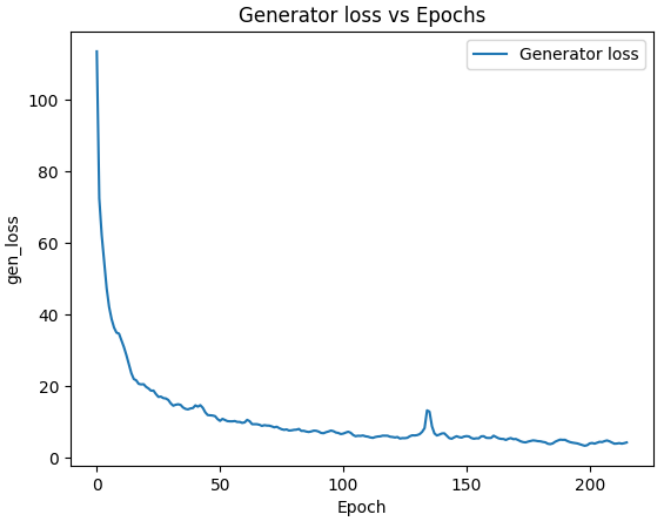


Fig. 3: Graph representing generator loss decrement on training for about 225 epochs

In order to evaluate the quality of the generated images, we conducted a thorough comparison with their corresponding high-resolution counterparts using two essential metrics:

A. Peak Signal-to-Noise Ratio(PSNR):

PSNR is a metric used to evaluate the quality of a compressed or reconstructed image or video signal by comparing it to the original signal. In the case of images, PSNR is calculated by comparing the original uncompressed image to the compressed image.

$$PSNR = 10 \times log_{10} \left(\frac{(max\_pix)^2}{mse} \right)$$

- The higher the PSNR value, the less distortion there is in the compressed image, and the closer the compressed image is to the original uncompressed image in terms of quality.
- With an impressive average value of 69.02dB, the estimated metrics reveal the remarkable closeness between the generated super resolution images and the original high-resolution images.

Fig. 4: Results that are generated using the implemented SRGAN model

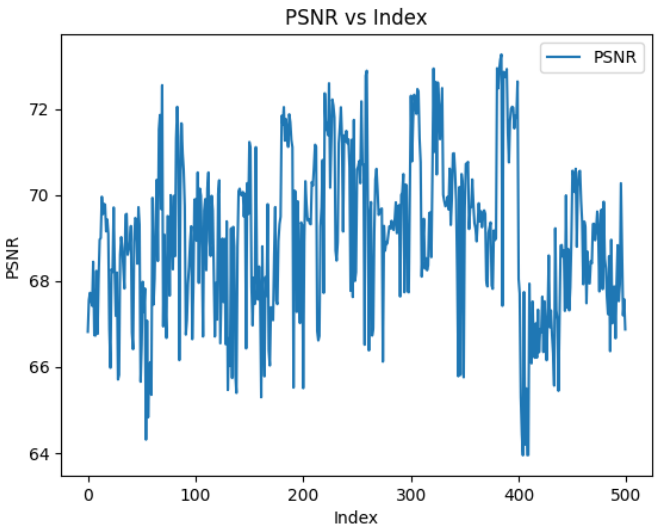


Fig. 5: Peak Signal-to-Noise ratio for the reconstructed super-resolution images and original high resolution images

B. Structural Similarity Index Measure (SSIM):
SSIM is a widely used metric for measuring the similarity between two images. The SSIM index measures the structural similarity between two images by comparing three aspects of the images: *luminance, contrast, and structure*. It is designed to take into account the human visual system's sensitivity to these aspects of the image.

$$SSIM(x, y) = (l(x, y) \times c(x, y) \times s(x, y))^w$$

where $l(x, y), c(x, y), s(x, y)$ are luminance, contrast and structure similarity between two images while w is a weighting factor that adjusts the relative importance of the three components.

- SSIM value usually ranges between -1 and 1 indicating perfect similarity between the two images.
- As the SSIM approaches 1, the images exhibit a remarkable similarity, reflecting a striking resemblance between the generated and original high-resolution images.
- By evaluating the metric over the generated high-resolution images and their corresponding original images, the values ranged between 0.55 and 0.90, showcasing an impressive level of similarity.
- The average SSIM value of 0.7499 signifies a commendable level of efficient similarity achieved. The results highlight the effectiveness of the approach in capturing and preserving the essence of the original images with remarkable fidelity.

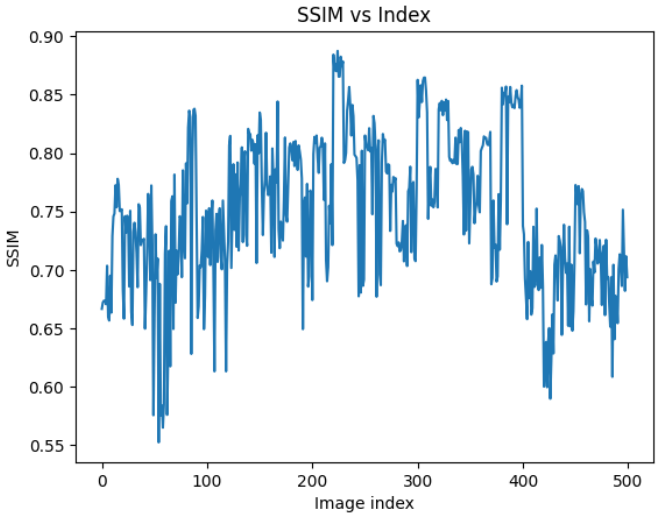


Fig. 6: Structural Similarity Index Method for the reconstructed super-resolution images and original high resolution images

V. Conclusion

Utilizing the latest libraries and technologies including Keras from TensorFlow, numpy, matplotlib and more, our model generates stunning high-resolution images when trained over large dataset and more epochs with consistent SSIM and PSNR values that rival the originals. Our implementation of SRGAN allows for images to be upscaled by four times while preserving key features and fine textures from the original image. This technology extracts essential visual details to generate high-quality images with clarity and detail. Our

approach is an effective solution for enhancing low-quality images and improving their visual appeal.

VI. Acknowledgment

With sincere effort and seamless coordination, we successfully accomplished our project goals. Our dedicated teamwork paved the way for a fruitful completion, where every challenge was overcome. Together, we achieved our objectives with satisfaction. However, it would not have been possible without the support and help of many individuals. We would like to express our sincere gratitude to our mentor Niraj sir and the management for providing us with the guidelines and supervision required to complete our project.

We are grateful to our respective teachers, and our convener Dr. S.K.Singh for their guidance and constant supervision as well as for providing necessary information regarding the project and also for their support in completing the project. Their willingness to share their vast knowledge made us understand this project and its manifestation helped us to complete the project on time.

Our sincere thanks to my friends in developing our project and to the people who have willingly helped us with their potentials and with their blessings.

References

[1] J. A. Ferwerda, "Three varieties of realism in computer graphics," in *Human vision and electronic imaging viii*, vol. 5007. SPIE, 2003, pp. 290–297.

[2] M. I. Ibrahem, M. M. Badr, M. M. Fouda, M. Mahmoud, W. Alasmary, and Z. M. Fadlullah, "Pmbfe: Efficient and privacy-preserving monitoring and billing using functional encryption for ami networks," in *2020 international symposium on networks, computers and communications (ISNCC)*. IEEE, 2020, pp. 1–7.

[3] D. Ulybyshev, I. Yilmaz, B. Northern, V. Kholodilo, and M. Rogers, "Trustworthy data analysis and sensor data protection in cyber-physical systems," in *Proceedings of the 2021 ACM Workshop on Secure and Trustworthy Cyber-Physical Systems*, 2021, pp. 13–22.

[4] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[6] L. Wen, X. Li, X. Li, and L. Gao, "A new transfer learning based on vgg-19 network for fault diagnosis," in *2019 IEEE 23rd international conference on computer supported cooperative work in design (CSCWD)*. IEEE, 2019, pp. 205–209.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[9] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4681–4690.

[10] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1874–1883.

[11] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 694–711.