Proceedings of 15th IOE Graduate Conference
Peer Reviewed
Year: 2024 Month: May Volume: 15
ISSN: 2350-8914 (Online), 2350-8906 (Print)

Design of SRGAN using RRDB on Generative Model and PatchGAN on Discriminative Model

Raju Poudel [a], Bishnu Hari Paudel [b]

[a, b] *Department of Electronics and Computer Engineering, Pashchimanchal Campus, IOE, Tribhuvan University, Nepal*
[a] poudelraju510@gmail.com, [b] bishnuhari@wrc.edu.np

Abstract
Super-Resolution Generative Adversarial Networks (SRGANs) have come up as a promising solution to improve the resolution of low-resolution images while preserving important details. This thesis explores designing and implementing an advanced SRGAN architecture employing Residual-in-Residual Dense Blocks (RRDB) in the generator part and PatchGAN in the discriminator part. The objective is to upscale input images of size 64x64x3 to high-resolution counterparts of size 256x256x3. The RRDBs capture intricate features and long-range dependencies within the image, facilitating the generation of high-quality pictures. Additionally, the PatchGAN discriminator aids in effectively distinguishing between real and generated images at the patch level, promoting finer-grained adversarial learning. A combined loss function comprising content loss (VGG loss) and adversarial loss is applied to achieve this. The proposed SRGAN architecture undergoes comprehensive experimentation and evaluation against existing methods, focusing on perceptual quality metrics, such as Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM), as well as subjective assessment by human evaluators. Results demonstrate the effectiveness of the proposed approach in generating visually pleasing high-quality images from low-quality inputs, thereby contributing to advancements in image super-resolution techniques.

Keywords
Super Resolution, Residual in Residual Dense Block, PatchGAN, VGG Loss, Adversarial Loss

1. Introduction

Super Resolution improves [1] an image's resolution and quality beyond its starting point. Enhancing the image's visual features helps to generate high-quality images from low-quality input photos, which also aids in better analysis. There are various methods of achieving super-resolution. Using GAN is one of them.

GAN (Generative Adversarial Network) consists of two components generator and discriminator. These two components are adversarial. To trick the discriminator, the generator attempts to produce an image that is exactly the same as the original. The discriminator looks for differences between authentic and false pictures. The discriminator and generator's to and fro motion assists the model in producing more accurate visuals.

There is a need for effective super-resolution techniques for applications like video and image processing because low-resolution images and videos can feel blurry and lack details, hindering our ability to fully appreciate the content. Super-resolution techniques hold the key to unlocking sharper, richer experiences, it is difficult to create high-quality images from low-quality images.

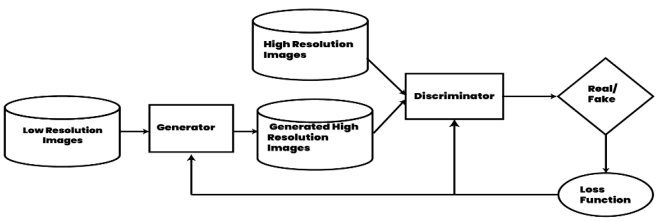

Figure 1: Basic GAN

2. Related Work

Numerous studies have been conducted in the area of super-resolution imaging. Ledig et al.[1] employ a Deep Convolution layer as a discriminator and a Residual block as a generator to achieve Photo Realistic Single Image Super Resolution using GAN. The input image is upscaled by the residual block using the convolution layer and subpixel convolution layer. The image is categorized as fake or real by the discriminator layer.

To eliminate artifacts and produce high-quality images, Wang et al [2] proposed an enhanced super-resolution generative adversarial network that combines residual in residual dense block (RRDB) as a generator and relativistic average GAN as a discriminator. With the help of its residual and dense connections, the RRDB model can produce high-quality images from input features. The chance that the generated image is more realistic or not is indicated by the relativistic GAN.

Image-to-Image Super Resolution with Conditional Network, proposed by P. Isola[3] uses PatchGAN as a discriminator and UNET as a generator. In order to make it easier to capture spatial information during the transformation process, UNET adopts an encoder-decoder structure. Instead of classifying the full image as authentic or false, the patchGAN classifies the image patchwise.

Patch-Based Visual ResNet, a generator that includes several

components like downsampling, residual block, and upsampling to fill in the missing portion of the image, is used in Ugur Demir and Unal's[4] inpainting using the GAN technique. The PG-GAN discriminator is employed, which is a hybrid of global GAN (which determines whether the entire image is real or not) and patch-GAN (which assesses local texture details and determines whether the image is real or fake).

Haichuan Ma[5]has employed the discriminator for a new purpose, recycling it by Yunan Zhu. To assess the quality of artificially super-resolved images, they employ the discriminator. Two types of discriminators are used to train a super-resolution WGAN: one discriminator evaluates the entire image, and the other works on small patches. In doing so, they assess the image quality.

Based on the literature, the coupling of RRDB as generator and PatchGAN as discriminator is done and carefully tested here, with this model to produce better results. The output of the overall model is satisfactory and the output is formulated and validated as below.

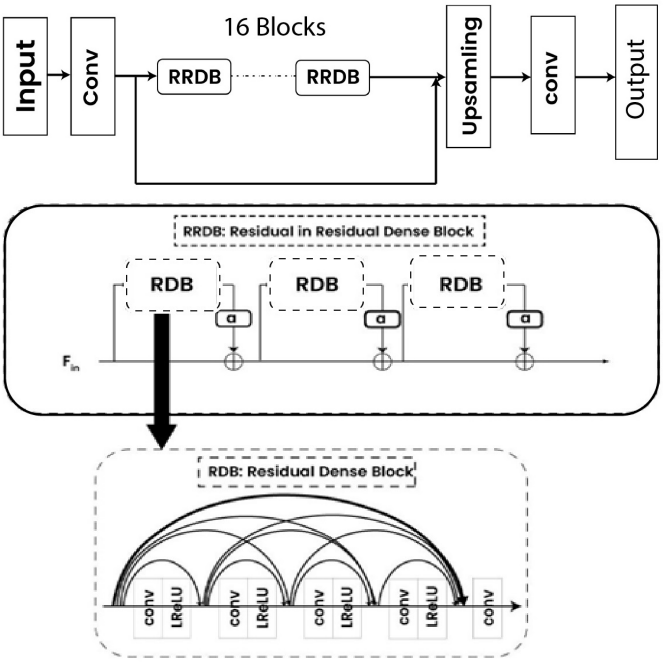

Figure 3: RRDB Generator[6]

3. Methodology

Using GAN, this block diagram facilitates the creation of high resolution images from low resolution images. The discriminator compares the generated high-resolution image to the genuine image using the generate function to determine whether it is real or fake.

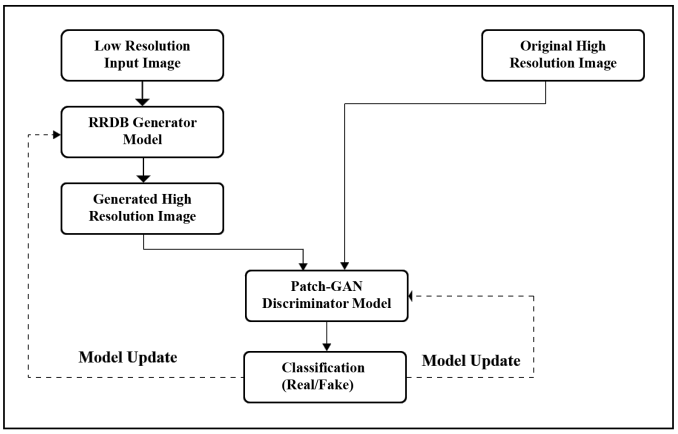

Figure 2: Standard GAN Architecture

3.1 Generator Network

The generator has a Residual in Residual Dense Block (RRDB) component where there is a residual and dense connection in the network.

The input to the generator is the low-resolution image of size 64×64×3. It is passed to through the generator network to generate the high-quality image of size 256×256×3. There are sixteen RRDB blocks in the generator network. Eight RDB Blocks make up each RRDB Block. Leaky ReLU activation functions and four convolution layers are present in every RDB block. An upsampling and convolution layer are added to the RRDB block's output to create a high-resolution image with dimensions of 256×256×3. When the input image of size 64×64×3 is passed to the RDB block it goes through the series of convolution and pooling layers along with activation functions and generates the output. This output will concatenate with the initial input and will act as the input to the next RDB layer. This process will go through 8 RDB blocks and generate the output of the RRDB block. Important low-level features are integrated with high-level features while being kept and reused by the network by concatenating feature maps from earlier layers with those from subsequent layers within the block. For the super-resolution tasks, this aids in preserving full details about the input image across the layers of the network. The 16 RRDB blocks will work similarly and finally generate the output of size 256×256×3.

3.2 Discriminator Network

The output of the generator and the original high-resolution input are fed to the discriminator. The PatchGAN discriminator is being used in the model.

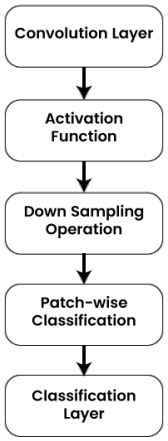

Figure 4: PatchGAN

Patchwise, the entire picture[7] is evaluated by the patchGAN. It has a downsampling layer and an activation function after five convolution layers. Utilizing an activation function such as the sigmoid, which gives a value between 0 and 1, patch-wise classification is carried out, indicating whether a patch is real or fake. To get the discriminator's final output for the full image, the probabilities are then averaged. The discriminator's confidence in determining whether the entire image is real (or fraudulent) is indicated.

The patchwise classification and determination of the receptive field in the patchGAN are as:

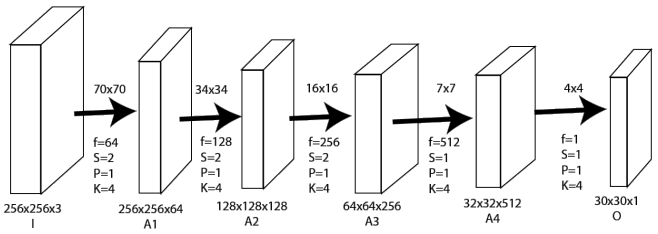

Figure 5: Determining Receptive Field for a PatchGAN[8]

From the above figure, the image of 256×256×3 is converted to the image of 256×256×64 by applying the convolution layer given by the formula:

Output Image (O):

$$O = [(N - K + 2 \times P)/S] + 1 \tag{1}$$

where,
N= Size of the input image
K= Kernel Size
P= Padding
S= Strides

This is carried out across five layers to produce a 30×30×1 image. The backtracking method is used to determine the receptive field/patch size because the model uses patch GAN. This model uses a 1×1 patch for a 30×30×1 image. To determine the size of the image patch of 256×256×3, backtracing is used to obtain it. We employ the equation of Receptive field ,

$$K_{x-1} = K + S(K_x - 1) \tag{2}$$

where,
K_{x-1}= Receptive Field of (x-1)th layer
K= Kernel size
S= Stride Value
K_x= Output receptive field of xth layer

Finally, the receptive field of the image 256×256×3 is found to be 70×70 which will be classified as real or fake using the sigmoid function.

3.3 Loss Function

To train the GAN network, the model has to minimize the objective function, which is also referred to as the loss function. The weighted sum of content loss and adversarial loss is the perceptual loss function, which serves as the model's goal function.[9].

$$L^{SR} = L^{SR}_{\text{VGG}} + 0.001 \times L^{SR}_{\text{Gen}} \tag{3}$$

The content loss[10] is given by VGG loss. Unlike traditional MSE loss, it does not calculate the difference between the pixel of the generated image and the original image. In VGG loss, we calculate how similar they are in terms of feature/pattern (i,j). The formula gives the VGG loss:

$$L^{SR}_{VGG/i,j} = \frac{1}{W_{i,j} H_{i,j}} \times \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} \left(\phi_{i,j}(I_{\text{HR}})_{x,y} - \phi_{i,j}(G_\theta G(I_{\text{LR}}))_{x,y}\right)^2 \tag{4}$$

where $W_{i,j}$ and $H_{i,j}$ are the dimensions of the feature maps inside the network, and $\phi_{i,j}$ is the feature map generated by the VGG19 network. It is the Euclidean separation between the created image's feature maps and the original image's feature maps.The created image is represented by $G_{\theta G}(I^{LR})$. The generator's loss as viewed from the discriminator's end is known as the adversarial loss.The formula gives it :

$$L^{\text{SR}}_{\text{Gen}} = \sum_{n=1}^{N} (-\log D_{\theta_D}(G_{\theta_G}(I_{\text{LR}}))) \tag{5}$$

4. Dataset

CelebAMask-HQ, a sizable face image data collection of 30,000 high-resolution face photos chosen from the CelebA data set in accordance with CelebAHQ, is utilized for the dataset. Random low-resolution images with a size of 64×64×3 pixels are fed into the generator throughout the GAN's training process with the goal of producing high-resolution images with a size of 256×256×3.

5. Result and Discussion

5.1 Training Result

The loss of the discriminator and generator for 30000 iterations for batch size 1 is given below:

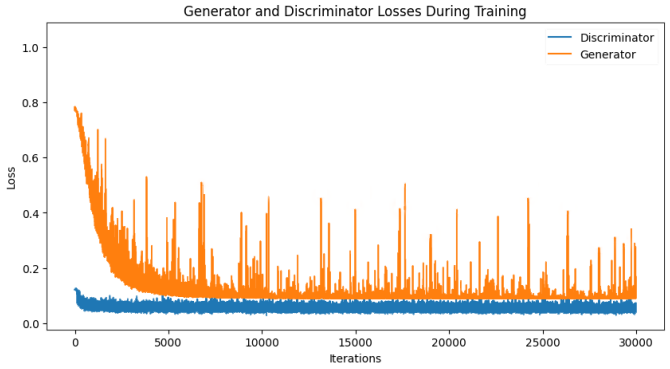

Figure 6: Generator and Discriminator Loss for Batch Size 1

At the beginning of the iteration, the generator loss is significant and progressively decreases with an increase in iterations, but there may be some variation. This suggests that as the number of iterations increases, the generator is producing better images. The discriminator's loss is nearly constant, which means that it can determine the authenticity

of an image at a consistent rate. Generator and discriminator losses vary somewhat, indicating that they are in competition with one another to outperform one another.

The loss of the generator and discriminator for 7500 iteration for batch size 4 is given below:

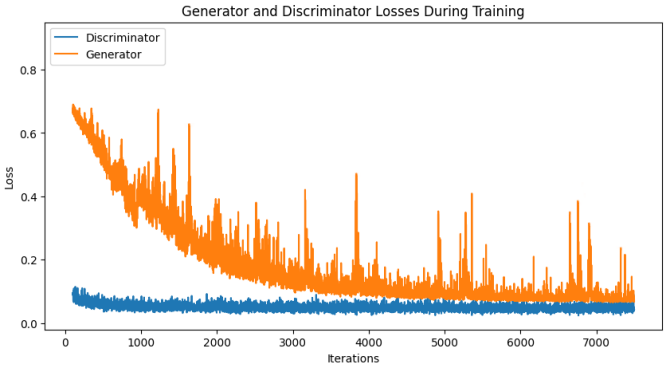

Figure 7: Generator and Discriminator Loss for Batch Size 4

The generator loss is high at the beginning of the iterations and gradually decreases with some variance. The discriminator loss is constant with some variance. Nearly all deep learning and machine learning models are trained to reduce loss to improve model performance. With GAN, this might not be the case. The discriminator and component generator of a GAN are the cause. They are opposite to each other. Rather than reducing the loss, they behave as though they compete with one another. Their competition is what led to the loss. Therefore, even with a large loss, the image quality could still be high. In the same way, the visual quality could be poor even with a small loss. This is because GAN is a complicated and dynamic system.

5.2 Output and Evaluation

In the successful iterations of the model learning, the generator model is only used to test the new sample image. The test dataset is "Nepali Celeb localized face dataset" [11] and the results of the test are:

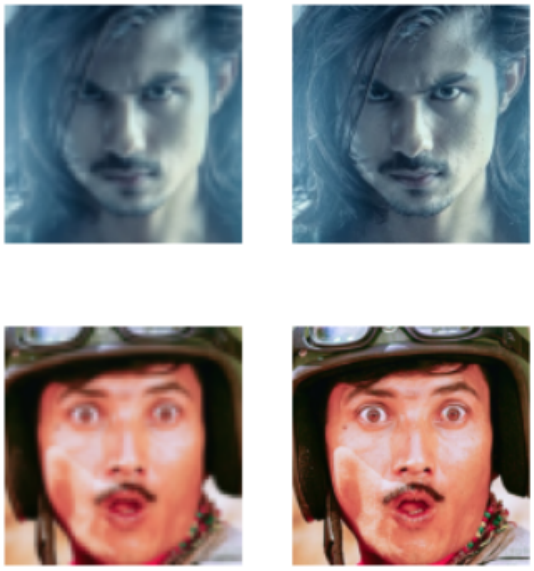

Figure 8: Input Low Resolution Image vs Generated Output Image

The left image is the low-resolution image and the right one is the generated image. The PSNR and SSIM score[12] for each image is calculated. A statistic called PSNR is used to assess how well an original image compares to one produced by a generator.. Higher PSNR means less noise. SSIM takes mainly 3 factors Lumiscence, Contrast, and Structure. The higher score means better quality of the image.

Table 1: PSNR and SSIM

Image	PSNR	SSIM
1	28.324	0.792
2	29.324	0.854
3	27.961	0.871
4	29.711	0.854
5	24.685	0.772

The overall PSNR score of the model is 27.992 dB and the SSIM is 0.8286. The comparison with the other model is given below:

Table 2: Comparison with other Model

Model	PSNR(dB)/SSIM
SR-ResNET	28.49/0.8184
SR-CNN	27.18/0.7861
Proposed Model	27.922/ 0.8286

In the above table, SR-CNN generates high-resolution images using extraction, nonlinear mapping, and reconstruction in

the model. Similarly, in the SR-Resnet Model, Resnet serves as the generator and is made up of many convolution layers, residual blocks, and upsampling layers. Normal CNN serves as the discriminator, classifying images as authentic or fraudulent.

To get result better, the discriminator in the proposed model is a patch GAN, which is used for patchwise image classification to capture fine-grained information. The RRDB generator has residual blocks, and each residual block's output is fed into another residual block to preserve the key features while generating high images which has led to the generation of better image quality than the above two models.

The proposed model obtains a PSNR score of 27.922 dB, significantly higher than the SR-CNN's 27.18 dB however slightly lower than the SR-Resnet's 28.49 dB. But here it can be clearly observed that perceived image quality and PSNR don't always match up perfectly. As we go through SSIM, the proposed model is far better than both SR-ResNet and SR-CNN, indicating that it retains more structural information and produces more aesthetically pleasing outcomes than the other two models.

6. Conclusion and Future Work

The RRDB generator employed in this model allows a 4x upscale of the image. To ensure that the crucial features are not lost during the generation process, the generator employs an RDB block that uses convolution, a pooling layer that gathers the essential features and concatenates them with the input low picture. A PatchGAN is used as the discriminator to do the patchwise classification of the picture to determine its authenticity.

The adversarial feature mapping loss is not being implemented in this research. It helps to extract the intermediate feature during the training of the discriminator from various intermediate layers and incorporate it into the generator to generate a higher-quality image, which is part of future work.

Acknowledgements

The program coordinator of the Department of Electronics and Computer Engineering, IOE, Pashchimanchal Campus is deeply acknowledged by the authors for all of the support, time, effort, and guidance. Additionally, the authors sincerely thankful to all of the professors who have provided constant support and direction throughout. In closing, the first author expresses gratitude to his friends for sharing their opinions and suggestions for the research.

References

[1] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan 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*, pages 4681–4690, 2017.

[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. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018.

[3] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[4] Ugur Demir and Gozde Unal. Patch-based image inpainting with generative adversarial networks. *arXiv preprint arXiv:1803.07422*, 2018.

[5] Yunan Zhu, Haichuan Ma, Jialun Peng, Dong Liu, and Zhiwei Xiong. Recycling discriminator: Towards opinion-unaware image quality assessment using wasserstein gan. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 116–125, 2021.

[6] Andreas Lugmayr, Martin Danelljan, Radu Timofte, Kangwook Kim, Younggeun Kim, Jae-young Lee, Zechao Li, Jinshan Pan, Dongseok Shim, Ki-Ung Song, et al. Ntire 2022 challenge on learning the super-resolution space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 786–797, 2022.

[7] Yuanfu Gong, Puyun Liao, Xiaodong Zhang, Lifei Zhang, Guanzhou Chen, Kun Zhu, Xiaoliang Tan, and Zhiyong Lv. Enlighten-gan for super resolution reconstruction in mid-resolution remote sensing images. *Remote Sensing*, 13(6):1104, 2021.

[8] Sahil. Understanding patchgan, May 2020.

[9] Anhiti Mandal and Olivia Rose. Replication of characteristic visual motifs of indian rural art forms using a generative adversarial network. *Journal of Student Research*, 12(1), 2023.

[10] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[11] Amit Pant. Nepali celeb localized face dataset, 2020.

[12] Abdul Rehman. Ssim-inspired quality assessment, compression, and processing for visual communications. 2013.