Image Super-Resolution Using a Generative Adversarial Network

Le Dang Khoa; Tran Minh Huy
khoaldse150847@fpt.edu.vn – huytmse150803@fpt.edu.vn
Scholar, FPT University, Thu Duc City, Ho Chi Minh – Viet Nam

November 2023

Abstract

Despite advancements in single-image super-resolution using faster and deeper convolutional neural networks, a significant challenge remains: how to restore fine texture details when upscaling images by large factors. Current optimization-based methods primarily rely on minimizing mean squared reconstruction error, resulting in high peak signal-to-noise ratios but lacking high-frequency details and perceptual satisfaction at higher resolutions. In our research, we introduce SRGAN, a generative adversarial network (GAN) for image super-resolution. This innovative framework can generate photorealistic natural images at 4× upscaling factors, a feat not achieved before. We achieve this by proposing a novel perceptual loss function comprising an adversarial loss and a content loss. The adversarial loss guides the solution toward the natural image space using a discriminator network trained to distinguish between super-resolved images and original photorealistic ones. Additionally, our content loss is based on perceptual similarity rather than pixel space similarity. Our deep residual network excels at recovering photorealistic textures from heavily downsampled images in public benchmarks. Extensive mean-opinion-score (MOS) tests demonstrate significant improvements in perceptual quality with SRGAN. The MOS scores obtained with SRGAN are much closer to those of the original high-resolution images compared to any existing state-of-the-art method.
Keywords: GAN, SRGAN, MSE, SRResNet, CNN,..

1 Introduction

Super-resolution (SR) involves the difficult task of predicting a high-resolution (HR) image from a low-resolution (LR) version. This area of research, widely known in the computer vision community, has garnered significant interest and finds applications across various fields [1, 13, 22].

Figure 1: Super-resolved image (left) is almost indistinguishable from original (right).[4× upscaling]

The challenge of super-resolution (SR), especially for high upscaling factors, often results in a lack of texture details in the reconstructed images. Most supervised SR methods aim to minimize the mean squared error (MSE) between the recovered high-resolution (HR) image and the ground truth, which also maximizes the peak signal-to-noise ratio (PSNR) – a common metric for evaluating SR

algorithms. However, MSE and PSNR have limitations in capturing perceptually relevant differences, such as intricate texture details, as they focus on pixel-wise differences. This limitation is evident in cases where the highest PSNR does not necessarily correspond to perceptually superior SR results, as illustrated in Figure 2. The perceptual distinction between the super-resolved and original images means that the recovered image may not be photorealistic, as defined by Ferwerda [16].

In our research, we introduce a super-resolution generative adversarial network (SRGAN) that utilizes a deep residual network (ResNet) with skip-connections and moves away from relying solely on MSE for optimization. Unlike previous approaches, we introduce a novel perceptual loss based on high-level feature maps from the VGG network [12, 33, 5], coupled with a discriminator that encourages solutions that are perceptually challenging to differentiate from HR reference images. Figure 2 showcases an example of a photorealistic image super-resolved with a 4× upscaling factor using our proposed method.

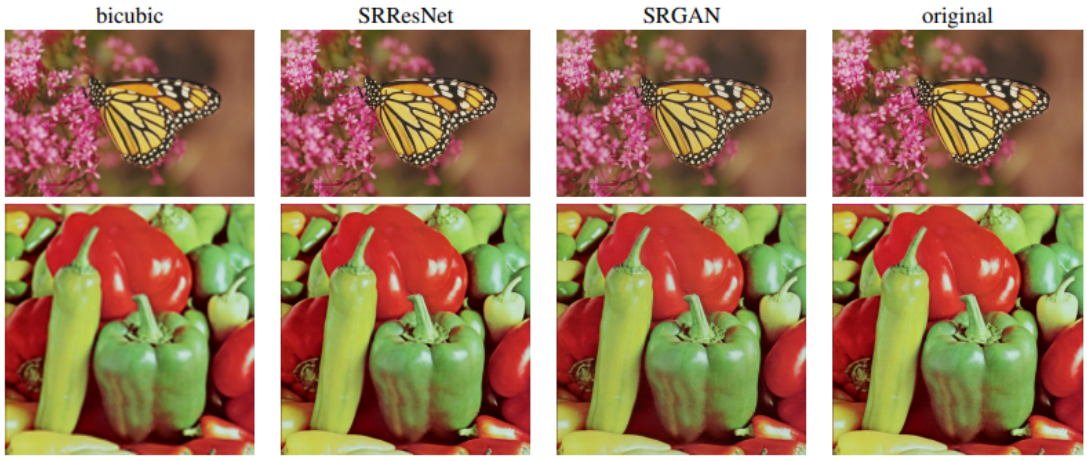


Figure 2: From left to right: bicubic interpolation, deep residual network optimized for MSE, deep residual generative adversarial network optimized for a loss more sensitive to human perception, original HR image. Corresponding PSNR and SSIM are shown in brackets. [4× upscaling]

2 Related work

2.1 Image super-resolution

Recent comprehensive articles on single image super-resolution (SISR) include works by Nasrollahi and Moeslund [33] as well as Yang et al. [21]. This discussion will specifically concentrate on single image super-resolution and will not delve into methods that involve recovering high-resolution images from multiple images [4, 15]. Early methods in SISR involved prediction-based approaches, such as linear, bicubic, or Lanczos filtering, which were fast but often resulted in overly smooth textures due to their oversimplified nature. Techniques focusing on edge-preservation were introduced to address this issue [1, 39]. More advanced methods aimed to establish intricate mappings between low- and high-resolution image data, often relying on training data. Some of these methods utilized LR training patches with known corresponding HR counterparts.

Freeman et al. [18, 17] presented pioneering work in this area. Related approaches emerged from compressed sensing techniques [62, 12, 19]. For instance, Glasner et al. [21] exploited patch redundancies across scales within the image to drive the super-resolution process. This concept of self-similarity was also explored in Huang et al. [31], where self-dictionaries were extended to include small transformations and shape variations. Gu et al. [25] proposed a convolutional sparse coding approach that enhanced consistency by processing the entire image instead of overlapping patches.

To reconstruct realistic texture details while avoiding edge artifacts, Tai et al. [22] combined an edge-directed SR algorithm based on a gradient profile prior [10] with learning-based detail synthesis. Zhang et al. [70] introduced a multi-scale dictionary approach to capture redundancies in similar image patches at different scales. For landmark images, Yue et al. [17] retrieved HR images with similar content from the web and proposed a structure-aware matching criterion for alignment.

Neighborhood embedding methods achieved upsampling of LR image patches by finding similar

LR training patches in a low-dimensional manifold and combining their corresponding HR patches for reconstruction [24, 2]. Kim and Kwon [35] emphasized the tendency of neighborhood approaches to overfit and formulated a more general mapping of example pairs using kernel ridge regression. The regression problem was also solved with Gaussian process regression [27], trees [46], or Random Forests [27]. Dai et al. [6] learned multiple patch-specific regressors and selected the most appropriate ones during testing.

In recent years, convolutional neural network (CNN) based SR algorithms have demonstrated outstanding performance. Wang et al. [9] incorporated a sparse representation prior into their feed-forward network architecture based on the learned iterative shrinkage and thresholding algorithm (LISTA) [13]. Dong et al. [9, 10] employed bicubic interpolation to upscale an input image and trained a three-layer deep fully convolutional network end-to-end, achieving state-of-the-art SR performance. It was later discovered that allowing the network to learn upscaling filters directly could further enhance performance both in terms of accuracy and speed [11, 8, 27]. Kim et al. [34] introduced a highly performant architecture, the deeply-recursive convolutional network (DRCN), which considered long-range pixel dependencies while maintaining a small number of model parameters. Particularly relevant to this paper are the works by Johnson et al. [33] and Bruna et al. [5], which utilized loss functions closer to perceptual similarity, resulting in visually convincing HR image recovery.

2.2 Design of convolutional neural networks

The current state of the art in many computer vision problems is defined by specially crafted CNN architectures, following the groundbreaking work by Krizhevsky et al. [37]. Research has demonstrated that training deeper network architectures can be challenging, but these architectures have the potential to significantly enhance accuracy by allowing modeling of highly complex mappings [9, 11]. To facilitate the training of these deeper networks, batch normalization [32] is commonly used to mitigate internal co-variate shift.

Deeper network architectures have also proven to be beneficial for single image super-resolution (SISR). For instance, Kim et al. [34] devised a recursive CNN, achieving cutting-edge results in SISR. Additionally, a design choice that has greatly simplified the training of deep CNNs is the introduction of residual blocks [29] and skip-connections [30, 34]. Skip-connections relieve the network from modeling the trivial identity mapping, which is challenging to represent using convolutional kernels. In the context of SISR, studies have shown that learning upscaling filters improves both accuracy and speed [11, 18, 27]. This represents an advancement over previous methods like Dong et al. [10], where bicubic interpolation was used to upscale the low-resolution observation before feeding the image to the CNN.

2.3 Loss functions

Loss functions based on pixel-wise measures like Mean Squared Error (MSE) struggle when it comes to handling the inherent uncertainty in recovering lost high-frequency details such as texture. Minimizing MSE tends to encourage finding pixel-wise averages of potential solutions, leading to overly smooth and perceptually poor results. This problem is demonstrated in Figure 3, where multiple solutions with high texture details are averaged to create a smooth reconstruction, showcasing the limitations of minimizing MSE.

Researchers have addressed this issue by turning to Generative Adversarial Networks (GANs) [22]. Mathieu et al. [42] and Denton et al. [7] used GANs for image generation. Yu and Porikli [6] augmented pixel-wise MSE loss with a discriminator loss to train networks for super-resolving face images with large upscaling factors ($8\times$). GANs have also been applied to unsupervised representation learning [44]. Li and Wand [38] and Yeh et al. [64] utilized GANs for style transfer and inpainting tasks, respectively.

In addition, researchers like Bruna et al. [5] and Dosovitskiy and Brox [13] proposed loss functions based on distances computed in feature spaces of neural networks, combined with adversarial training. These approaches enable visually superior image generation and are effective in solving ill-posed inverse problems related to decoding nonlinear feature representations. Johnson et al. [33] and Bruna et al. [5] utilized features extracted from a pretrained VGG network instead of low-level pixel-wise error measures.

Specifically, they formulated loss functions based on the Euclidean distance between feature maps extracted from the VGG19 network [29]. This approach has yielded more convincing results in both

super-resolution and artistic style-transfer tasks. Recent research by Li and Wand [38] further explored the effect of comparing and blending patches in both pixel and VGG feature space.

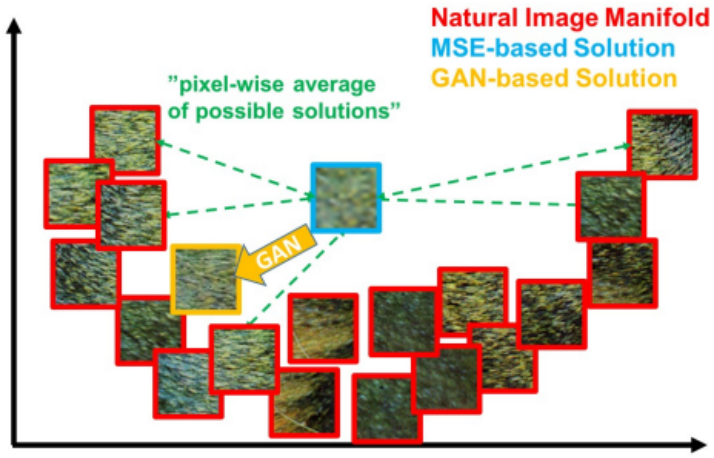


Figure 3: Illustration of patches from the natural image manifold (red) and super-resolved patches obtained with MSE (blue) and GAN (orange). The MSE-based solution appears overly smooth due to the pixel-wise average of possible solutions in the pixel space, while GAN drives the reconstruction towards the natural image manifold producing perceptually more convincing solutions.

2.4 Contribution

This study leverages Generative Adversarial Networks (GANs) to generate highly realistic natural images. The GAN framework guides the reconstructions toward regions in the search space likely to contain photorealistic images, approaching the natural image manifold, as depicted in Figure 3. This paper introduces a novel approach by combining the concept of GANs with a very deep ResNet architecture [29, 30] to create a perceptual loss function for photo-realistic Single-Image Super-Resolution (SISR). The key contributions of this work are as follows:

Setting a New Image Super-Resolution Benchmark: The authors achieve a new state-of-the-art performance in image Super-Resolution (SR) with high upscaling factors (4×). This accomplishment is measured using traditional metrics like Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) by employing a 16-block deep ResNet (SRResNet) optimized for Mean Squared Error (MSE).

Introduction of SRGAN: The paper introduces SRGAN, a GAN-based network optimized for a new perceptual loss function. Unlike conventional MSE-based content loss, SRGAN utilizes a loss calculated on feature maps extracted from the VGG network [49]. These feature maps exhibit higher invariance to changes in pixel space [38], enhancing the perceptual quality of the generated images.

Validation through Extensive Mean Opinion Score (MOS) Testing: The study conducts a comprehensive MOS test on images sourced from three widely used benchmark datasets. The results affirm that SRGAN significantly outperforms existing methods, establishing itself as the new state-of-the-art solution for estimating photo-realistic SR images, especially with high upscaling factors (4×).

3 Method

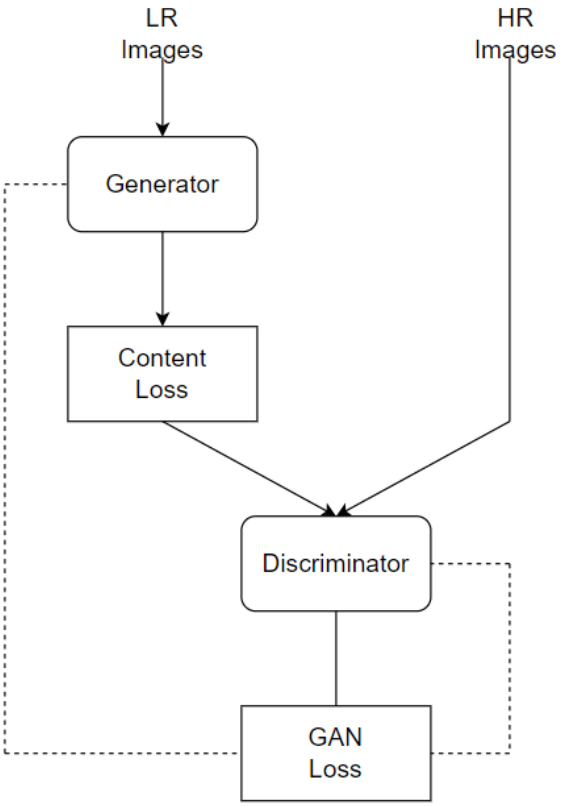
In SISR the aim is to estimate a high-resolution, superresolved image I^{SR} from a low-resolution input image I^{LR}. Here I^{LR} is the low-resolution version of its highresolution counterpart I^{HR}. The high-resolution images are only available during training. In training, I^{LR} is obtained by applying a Gaussian filter to I^{HR} followed by a downsampling operation with downsampling factor r. For an image with C color channels, we describe I^{LR} by a real-valued tensor of size $W \times H \times C$ and I^{HR}, I^{SR} by $rW \times rH \times C$ respectively.

Our ultimate goal is to train a generating function G that estimates for a given LR input image its corresponding HR counterpart. To achieve this, we train a generator network as a feed-forward

$CNNG_{\theta_G}$ parametrized by θ_G. Here $\theta_G = \{W_{1:L}; b_{1:L}\}$ denotes the weights and biases of a L-layer deep network and is obtained by optimizing a SR-specific loss function l^{SR}. For training images $I_n^{HR}, n = 1, \ldots, N$ with corresponding $I_n^{LR}, n = 1, \ldots, N$, we solve:

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

In this work we will specifically design a perceptual loss l^{SR} as a weighted combination of several loss components that model distinct desirable characteristics of the recovered SR image. The individual loss functions are described in more detail in Section 2.2.



4 Data Preparation

The data file used for this project is a data set provided by Kaggle with 855 samples. After exploratory data analysis (EDA), we found that the image set is 256x256 pixels. The photos are taken in many different styles from color, object, brightness, etc. We also note that the names of the images in the dataset have been labeled with a sequence number which helps provide a better solution for the next steps.

After exploratory data analysis, we use the Gaussian Blur function to blur the image for the purpose of creating "low resolution" images. After execution, we find that the data file includes two sets "train" and "val", the "train" set contains 1370 samples and the "val" set contains 340 samples for a total of 1710 samples. Both sets contain two The "high" and "low" photo sets are divided equally. With the "high" photo set, the photo has a quality of 256x256 pixels and the "low" photo set has a quality of 128x128 pixels.

4.1 Understanding Data and Similarity Measures

We perform experiments on three widely used benchmark datasets Set5 [3], Set14 [69] and BSD100, the testing set of BSD300 [40]. All experiments are performed with a scale factor of 4× between low-

and high-resolution images. This corresponds to a 16× reduction in image pixels. For fair comparison, all reported PSNR [dB] and SSIM [58] measures were calculated on the y-channel of center-cropped, removal of a 4-pixel wide strip from each border, images using the daala package [1]. Super-resolved images for the reference methods, including nearest neighbor, bicubic, SRCNN [9] and SelfExSR [31], were obtained from online material supplementary to Huang et al. [2] [31] and for DRCN from Kim et al. [3] [34]. Results obtained with SRResNet (for losses: l^{SR}_{MSE} and $l^{SR}_{VGG/2.2}$) and the SRGAN variants are available online [4]. Statistical tests were performed as paired two-sided Wilcoxon signed-rank tests and significance determined at $p < 0.05$.

The reader may also be interested in an independently developed GAN-based solution on GitHub. However it only provides experimental results on a limited set of faces, which is a more constrained and easier task.

4.2 Training Specifics and Parameters

We trained all networks on a NVIDIA Tesla M40 GPU using a random sample of 350 thousand images from the ImageNet database [45]. These images are distinct from the testing images. We obtained the LR images by downsampling the HR images (BGR, $C = 3$) using bicubic kernel with downsampling factor $r = 4$. For each mini-batch we crop 16 random 96×96 HR sub images of distinct training images. Note that we can apply the generator model to images of arbitrary size as it is fully convolutional. We scaled the range of the LR input images to $[0, 1]$ and for the HR images to $[-1, 1]$.

The MSE loss was thus calculated on images of intensity range $[-1, 1]$. VGG feature maps were also rescaled by a factor of $\frac{1}{12.75}$ to obtain VGG losses of a scale that is comparable to the MSE loss. This is equivalent to multiplying Equation 5 with a rescaling factor of ≈ 0.006. For optimization we use Adam [36] with $\beta_1 = 0.9$. The SRResNet networks were trained with a learning rate of 10^{-4} and 10^6 update iterations. We employed the trained MSE-based SRResNet network as initialization for the generator when training the actual GAN to avoid undesired local optima. All SRGAN variants were trained with 10^5 update iterations at a learning rate of 10^{-4} and another 10^5 iterations at a lower rate of 10^{-5}. We alternate updates to the generator and discriminator network, which is equivalent to $k = 1$ as used in Goodfellow et al. [22]. Our generator network has 16 identical ($B = 16$) residual blocks. During test time we turn batch-normalization update off to obtain an output that deterministically depends only on the input [32]. Our implementation is based on Theano [53] and Lasagne [8].

5 System Design

So we created a system to solve this problem. Our pipeline summary is divided into two stages: Adversarial network architecture and Perceptual loss function.

5.1 Adversarial network architecture

Following Goodfellow et al. [22] we further define a discriminator network D_{θ_D} which we optimize in an alternating manner along with G_{θ_G} to solve the adversarial min-max problem:

$$\min_{\theta_G} \max_{\theta_D} E_{I^{HR} \sim p_{train}(I^{HR})} \left[\log D_{\theta_D} \left(I^{HR} \right) \right] + E_{I^{LR} \sim p_G(I^{LR})} \left[\log \left(1 - D_{\theta_D} \left(G_{\theta_G} \left(I^{LR} \right) \right) \right) \right]$$

The general idea behind this formulation is that it allows one 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 difficult to classify by D. This encourages perceptually superior solutions residing in the subspace, the manifold, of natural images. This is in contrast to SR solutions obtained by minimizing pixel-wise error measurements, such as the MSE.

At the core of our very deep generator network G, which is illustrated in Figure 4 are B residual blocks with identical layout. Inspired by Johnson et al. [33] we employ the block layout proposed by Gross and Wilber [24]. Specifically, we use two convolutional layers with small 3×3 kernels and 64 feature maps followed by batch-normalization layers [32] and ParametricReLU [28] as the activation function. We increase the resolution of the input image with two trained sub-pixel convolution layers

as proposed by Shi et al. [18]. To discriminate real HR images from generated SR samples we train a discriminator network. The architecture is shown in Figure 4. We follow the architectural guidelines summarized by Radford et al. [14] and use LeakyReLU activation ($\alpha = 0.2$) and avoid max-pooling throughout the network. The discriminator network is trained to solve the maximization problem in Equation 2. It 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 [9]. 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 to obtain a probability for sample classification.

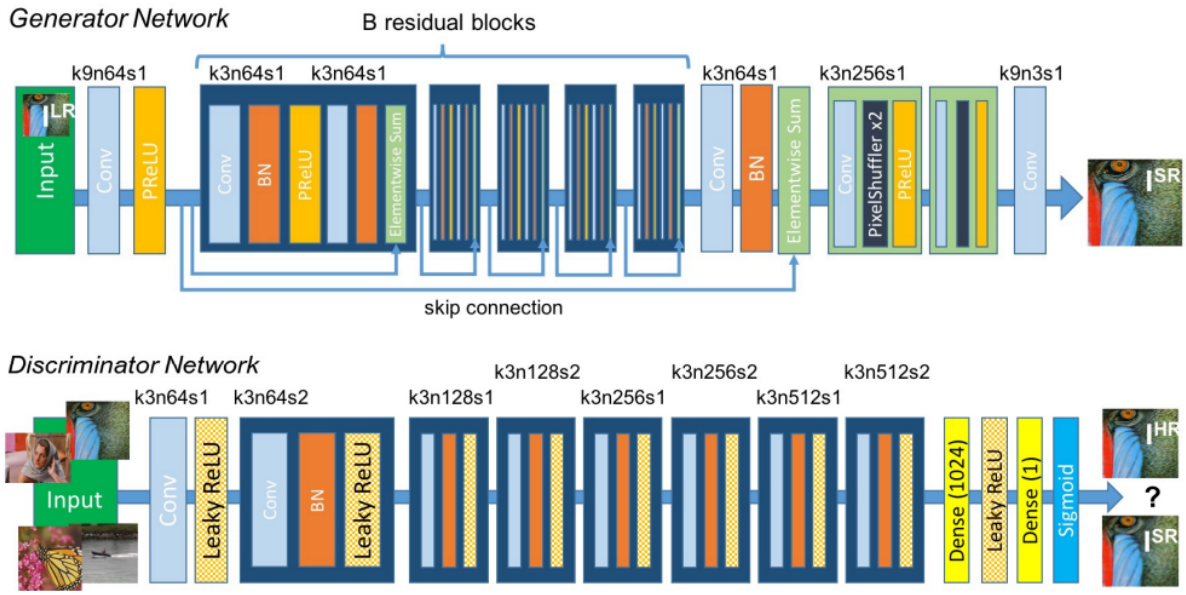


Figure 4: Architecture of Generator and Discriminator Network with corresponding kernel size (k), number of feature maps (n) and stride (s) indicated for each convolutional layer.

5.2 Perceptual loss function

5.2.1 Perceptual loss function

The definition of our perceptual loss function l^{SR} is critical for the performance of our generator network. While l^{SR} is commonly modeled based on the MSE [10, 18], we improve on Johnson et al. [33] and Bruna et al. [5] 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 $\left(l^{SR}_{\mathrm{X}}\right)$ and an adversarial loss component as:

$$l^{SR} = \underbrace{\underbrace{l^{SR}_{\mathrm{X}}}_{contentloss} + \underbrace{10^{-3} l^{SR}_{Gen}}_{adversarialloss}}_{perceptualloss(forVGGbasedcontentlosses)}$$

In the following we describe possible choices for the content loss l^{SR}_{X} and the adversarial loss l^{SR}_{Gen}.

5.2.2 Content loss

The pixel-wise MSE loss is calculated as:

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

This optimization approach is widely adopted in image Super-Resolution (SR) and serves as the foundation for many state-of-the-art techniques [10, 48]. Despite its ability to achieve high Peak Signal-to-Noise Ratio (PSNR), solutions obtained through Mean Squared Error (MSE) optimization often lack high-frequency content, resulting in visually unsatisfactory outcomes with excessively smooth textures (see Figure 2). To address this limitation, we deviate from pixel-wise losses and draw inspiration from the work of Gatys et al. [19], Bruna et al. [5], and Johnson et al. [33]. Instead, we employ a loss function that aligns more closely with perceptual similarity.

Our approach involves defining the VGG loss based on the Rectified Linear Unit (ReLU) activation layers of the pre-trained 19-layer VGG network introduced by Simonyan and Zisserman [49]. With $\phi_{i,j}$ we indicate the feature map obtained by the j-th convolution (after activation) before the i-th maxpooling layer within the VGG19 network, which we consider given. We then define the VGG loss as the euclidean distance between the feature representations of a reconstructed image $G_{\theta_G}\left(I^{LR}\right)$ and the reference image I^{HR} :

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

Here $W_{i,j}$ and $H_{i,j}$ describe the dimensions of the respective feature maps within the VGG network.

5.2.3 Adversarial loss

In addition to the previously mentioned content losses, we incorporate the generative component of our Generative Adversarial Network (GAN) into the perceptual loss. This inclusion prompts our network to prioritize solutions that align with the natural image manifold, striving to fool the discriminator network. The generative $lossl^{SR}_{Gen}$ is defined based on the probabilities of the discriminator $D_{\theta_D}\left(G_{\theta_G}\left(I^{LR}\right)\right)$ over all training samples as:

$$l^{SR}_{Gen} = \sum_{n=1}^{N} -\log D_{\theta_D}\left(G_{\theta_G}\left(I^{LR}\right)\right)$$

Here, $D_{\theta_D}\left(G_{\theta_G}\left(I^{LR}\right)\right)$ is the probability that the reconstructed image $G_{\theta_G}\left(I^{LR}\right)$ is a natural HR image. For better gradient behavior we minimize $-\log D_{\theta_D}\left(G_{\theta_G}\left(I^{LR}\right)\right)$ instead of $\log\left[1 - D_{\theta_D}\left(G_{\theta_G}\left(I^{LR}\right)\right)\right]$ [22].

6 Discussion and future work

The study demonstrated the superior perceptual performance of SRGAN through Mean Opinion Score (MOS) testing. Traditional metrics like PSNR and SSIM were found inadequate in assessing image quality according to human visual perception. Unlike previous works, this study focused on the perceptual quality of super-resolved images rather than computational efficiency. The presented model was not optimized for real-time video super-resolution, but preliminary experiments suggested that shallower networks could offer efficient alternatives with a slight reduction in quality. Contrary to some prior research, deeper network architectures were found beneficial, with ResNet design significantly impacting their performance. Extremely deep networks enhanced SRResNet's performance but required longer training and testing times, and deeper SRGAN variants proved challenging to train due to high-frequency artifacts.

The choice of content loss was crucial for achieving photorealistic solutions in super-resolution, with lSRVGG/5.4 yielding the most convincing results. Deeper network layers were capable of representing higher abstraction features, focusing on content, while the adversarial loss concentrated on texture details, distinguishing super-resolved images without adversarial loss from photorealistic ones. The ideal loss function depended on the specific application, and further research was needed to address challenges in reconstructing text or structured scenes convincingly. Developing content loss functions describing image spatial content but being more invariant to pixel space changes could enhance photorealistic image super-resolution results.

7 Conclusion

The authors performed experiments on three widely used benchmark datasets known as Set 5, Set 14, and BSD 100. These experiments performed on 4x up sampling of both rows and columns.

	SRResNet-		SRGAN-	
Set5	MSE	VGG19	MSE	VGG19
PSNR	32.05	30.51	30.64	29.84
SSIM	0.9019	0.8803	0.8701	0.8468
MOS	3.37	3.46	3.77	3.78
Set14				
PSNR	28.49	27.19	26.92	26.44
SSIM	0.8184	0.7807	0.7611	0.7518
MOS	2.98	3.15*	3.43	3.72*

In the above layer MSE means we take simple mean squared pixelwise error as content loss, VGG22 indicate the feature map obtained by the 2nd convolution (after activation) before the 2nd maxpooling layer within the VGG19 network and we calculate the VGG loss using formula described above. This loss is thus loss on the low-level features. Similarly VGG 54 uses loss calculated on the feature map obtained by the 4th convolution (after activation) before the 5th maxpooling layer within the VGG19 network. This represents loss on higher level features from deeper network layers with more potential to focus on the content of the images. The above image shows MOS scores on dataset. For each method 1710 samples were assessed. Mean shown as red marker, where the bins are centered around value i.

The main contributions of this paper is:

- This paper generates state-of-the-art results on upsampling (4x) as measured by PNSR (Peak Signal-to-Noise Ratio) and SSIM(Structural Similarity) with 16 block deep SRResNet network optimize for MSE.
- The authors propose a new Super Resolution GAN in which the authors replace the MSE based content loss with the loss calculated on VGG layer
- SRGAN was able to generate state-of-the-art results which the author validated with extensive Mean Opinion Score (MOS) test on three public benchmark datasets.

8 Refernces

References

[1] D. Martin, C. Fowlkes, D. Tal, J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics", Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, vol. 2, pp. 416-423, July 2001.

[2] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity", IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, April 2004.

[3] J. Yang, J. Wright, T. S. Huang, Y. Ma, "Image Super-Resolution Via Sparse Representation", IEEE Transactions on Image Processing, vol. 19, no. 11, pp. 2861-2873, Nov. 2010.

[4] C. Dong, C. C. Loy, K. He, X. Tang, "Learning a Deep Convolutional Network for Image Super-Resolution", European Conference on Computer Vision (ECCV), pp. 184-199, 2014.

[5] J. Kim, J. K. Lee, K. M. Lee, "Accurate Image Super-Resolution Using Very Deep Convolutional Networks", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1646-1654, 2016.

[6] C. Ledig et al., "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 105-114.

[7] K. Zhang et al. "Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising", IEEE Transactions on Image Processing, vol. 26, no. 7, pp. 3142-3155, July 2017.

[8] Y. Tai et al., "MemNet: A Persistent Memory Network for Image Restoration", Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4539-4547.

[9] E. Agustsson and R. Timofte, "NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017, pp. 1122-1131.

[10] W. Shi et al., "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1874-1883.

[11] B. Lim et al., "Enhanced Deep Residual Networks for Single Image Super-Resolution", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017, pp. 1132-1140.

[12] Y. Zhang et al., "Learning Deep CNN Denoiser Prior for Image Restoration", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3929-3938.

[13] Y. Zhang et al., "FFDNet: Toward a Fast and Flexible Solution for CNN-Based Image Denoising", IEEE Transactions on Image Processing, vol. 27, no. 9, pp. 4608-4622, Sept. 2018.

[14] K. Zhang et al., "Learning Deep CNN Denoiser Prior for Image Restoration", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3929-3938.

[15] Y. Chen and T. Pock, "Trainable Nonlinear Reaction Diffusion: A Flexible Framework for Fast and Effective Image Restoration", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1256-1272, 1 June 2017.

[16] X. Mao, C. Shen and Y. Yang, "Image Restoration Using Very Deep Convolutional Encoder-Decoder Networks with Symmetric Skip Connections", Advances in Neural Information Processing Systems 29 (NIPS 2016), 2016, pp. 2802-2810.

[17] J. Kim, J. K. Lee and K. M. Lee, "Deeply-Recursive Convolutional Network for Image Super-Resolution", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1637-1645.

[18] C. Dong, C. C. Loy, K. He and X. Tang, "Image Super-Resolution Using Deep Convolutional Networks", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 2, pp. 295-307, 1 Feb. 2016.

[19] J. Kim, J. K. Lee and K. M. Lee, "Deeply-Recursive Convolutional Network for Image Super-Resolution", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1637-1645.

[20] W. Shi et al., "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1874-1883.

[21] C. Ledig et al., "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 105-114.

[22] A. Radford, L. Metz and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks", arXiv:1511.06434 [cs.LG], 2015.

[23] I. Goodfellow et al., "Generative Adversarial Nets", Advances in Neural Information Processing Systems 27 (NIPS 2014), 2014, pp. 2672-2680.

[24] L. A. Gatys, A. S. Ecker and M. Bethge, "Image Style Transfer Using Convolutional Neural Networks", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2414-2423.

[25] D. Ulyanov et al., "Instance Normalization: The Missing Ingredient for Fast Stylization", arXiv:1607.08022 [cs.CV], 2016.

[26] K. He et al., "Deep Residual Learning for Image Recognition", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778.

[27] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift", Proceedings of the 32nd International Conference on Machine Learning, PMLR 37:448-456, 2015.

[28] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning", arXiv:1603.07285 [stat.ML], 2016.

[29] A. Odena, V. Dumoulin and C. Olah, "Deconvolution and checkerboard artifacts", Distill, 2016. http://distill.pub/2016/deconv-checkerboard.

[30] D. Pathak et al., "Context Encoders: Feature Learning by Inpainting", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2536-2544.

[31] C. Yang et al., "High-Resolution Image Inpainting using Multi-Scale Neural Patch Synthesis", The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.

[32] R. Yeh et al., "Semantic Image Inpainting with Perceptual and Contextual Losses", arXiv:1607.07539 [cs.CV], 2016.

[33] J. Johnson, A. Alahi and L. Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution", European Conference on Computer Vision (ECCV), 2016, pp. 694-711.

[34] A. Dosovitskiy and T. Brox, "Generating Images with Perceptual Similarity Metrics based on Deep Networks", Advances in Neural Information Processing Systems 29 (NIPS 2016), 2016, pp. 658-666.

[35] L. A. Gatys, A. S. Ecker, M. Bethge, A. Hertzmann and E. Shechtman, "Controlling Perceptual Factors in Neural Style Transfer", arXiv:1611.07865 [cs.CV], 2016.

[36] T. Kim, M. Cha, H. Kim, J. K. Lee and J. Kim, "Learning to Discover Cross-Domain Relations with Generative Adversarial Networks", arXiv:1703.05192 [cs.CV], 2017.

[37] Y. Li and N. Snavely, "MegaDepth: Learning Single-View Depth Prediction from Internet Photos", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2041-2050.

[38] Y. Li, S. Liu, J. Yang and M.-H. Yang, "Generative Face Completion", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3911-3919.

[39] Y. Li, M.-Y. Liu, X. Li, M.-H. Yang and J. Kautz, "A Closed-form Solution to Photorealistic Image Stylization", arXiv:1802.06474 [cs.CV], 2018.

[40] X. Wang et al., "High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8798-8807.