Machine Vision and Applications (2021) 32:3
https://doi.org/10.1007/s00138-020-01135-9

ORIGINAL PAPER

RSRGAN: computationally efficient real-world single image super-resolution using generative adversarial network

Vishal Chudasama[1] · **Kishor Upla**[1]

Received: 24 March 2020 / Revised: 19 July 2020 / Accepted: 29 September 2020

Abstract
Recently, convolutional neural network has been employed to obtain better performance in single image super-resolution task. Most of these models are trained and evaluated on synthetic datasets in which low-resolution images are synthesized with known bicubic degradation and hence they perform poorly on real-world images. However, by stacking more convolution layers, the super-resolution (SR) performance can be improved. But, such idea increases the number of training parameters and it offers a heavy computational burden on resources which makes them unsuitable for real-world applications. To solve this problem, we propose a computationally efficient real-world image SR network referred as RSRN. The RSRN model is optimized using pixel-wise L_1 loss function which produces overly-smooth blurry images. Hence, to recover the perceptual quality of SR image, a real-world image SR using generative adversarial network called RSRGAN is proposed. Generative adversarial network has an ability to generate perceptual plausible solutions. Several experiments have been conducted to validate the effectiveness of the proposed RSRGAN model, and it shows that the proposed RSRGAN generates SR samples with more high-frequency details and better perception quality than that of recently proposed SRGAN and SRFeat$_{IF}$ models, while it sets comparable performance with the ESRGAN model with significant less number of training parameters.

Keywords Real-world image super-resolution · Generative adversarial network · Perceptual index · Learned perceptual image patch similarity

1 Introduction

Single image super-resolution (SISR) refers to reconstruct the high-resolution (HR) image from its corresponding low-resolution (LR) counterpart. It can be applied in many fields such as medical imaging, surveillance and satellite imaging. The real-world LR image can be viewed as,

$$I_{LR} = (I_{HR} * w_b) \downarrow_r + \eta, \tag{1}$$

where, $*$ and $\downarrow_r$ denote the convolution function and downscaling operation with scale factor r, respectively. Here, w_b represents the unknown blur kernel and η represents the noise. In Eq. (1), one can notice that there are three different low-level vision tasks named as image deblurring, image denoising and image super-resolution take place for SISR in real-world scenarios. Hence, it is a very complicated task to reconstruct HR image from real-world LR image.

Recently, convolution neural networks (CNNs) obtain improvements in SISR by designing new CNN architectures and loss functions [12,13,17,27–29,33,45,46]. These methods are trained and evaluated on the simulated datasets in which the downsampled LR observations are obtained by applying known degradation such as bicubic. Hence, they obtain better results for this degradation. However, the SISR methods trained on such simulated datasets generalizes poorly for real-world LR images where the degradation is blind [8]. Additionally, the literature shows that the SR performance of the CNN-based methods can be improved by stacking more convolutional layers in the architecture; however, this offers a large number of training parameters which makes the model unsuitable to real-world applications. Hence, we propose a computationally efficient SR model for real-world images called RSRN [11] for upscaling factor $\times 4$. Furthermore, the CNN-based models rely on minimizing pixel-wise loss functions such as MSE, L_1 and Charbonnier, which result in blurry solutions. Hence, they are unable to

✉ Kishor Upla
kishorupla@gmail.com

[1] Electronics Engineering Department, Sardar Vallabhbhai National Institute of Technology (SVNIT), Surat, India

Published online: 08 October 2020

Fig. 1 The comparison of SR results of the proposed RSRN [11] and RSRGAN models (i.e., by training with synthetic and real dataset) of a single image of RealSR testing dataset for upscaling factor (×4)

generate high-frequency details in the SR results. The generative adversarial network (GAN) has an ability to restore the high-frequency information from LR observations. Here, we extend the proposed CNN-based model RSRN to GAN-based method which we referred as RSRGAN for upscaling factor ×4.

In Fig. 1, we display the SR results of the real-world image obtained using the proposed RSRN method [11] trained on a simulated dataset (i.e., RSRN-Synthetic). In this result, one can observe that there are more degradations observed in the SR result. This happens due to the unknown degradations of real-world images. In order to learn different degradations of real-world images, we also train the proposed method, RSRN on the RealSR training dataset [8] (i.e., RSRN-Real) which consists of real-world HR-LR image pairs of the same scene produced by adjusting the focal length of a digital camera. The SR result of the proposed method RSRN [11] trained using RealSR dataset is also displayed in Fig. 1 where one can observe that the SR result has low degradations and also it is close to ground truth image. The corresponding quantitative measurements [i.e., peak signal-to-noise-ratio (PSNR), structural similarity (SSIM), learned perceptual image patch similarity (LPIPS), perceptual index (PI) and root mean square error (RMSE)] are also mentioned at the bottom of all SR results. Here, one can also observe that the RSRN-Real model obtains better quantitative measurements than that of the RSRN-Synthetic model.

In spite of achieving higher PSNR and SSIM values, the SR results obtained using RSRN-Synthetic and RSRN-Real models look blurry which is due to the use of pixel-wise loss function in the training. However, higher PSNR does not reflect the perceptual quality of the SR images. Hence, to reconstruct the super-resolved image with high-frequency texture details at large upscaling factor, a generative adversarial network (GAN) has been used which has an ability to generate visually appealable solutions [16,34]. In GAN, the discriminator network forces the generator network to produce perceptually visible solutions. Hence, in this work, we propose an SR network using GAN for real-world images called RSRGAN for upscaling factor ×4. In the proposed RSRGAN, we employ the combination of VGG-based perceptual loss [22] and adversarial loss [16] functions as the total loss function in addition with a pixel-wise loss function. The comparison of the SR images obtained using the proposed RSRGAN trained on synthetic and RealSR datasets (i.e., RSRGAN-Synthetic and RSRGAN-Real) along with the ground truth is depicted in Fig. 1. Here, the SR result of RSRGAN-Synthetic has more degradation which proves that the SR method trained with synthetic dataset do not perform well on the real-world images. Additionally, one can also notice that the SR image obtained using the proposed RSRGAN-Real is more close to the ground truth image with better perception measurements (i.e., LPIPS, PI and RMSE) than that of other models. Although the PSNR and SSIM for the proposed RSRGAN-Real are lower than that of obtained using the proposed RSRN-Real approach, it results the SR images with more perceptual fidelity.

The key contributions of the proposed method are as follows:

- We propose a computationally efficient SR network using GAN for real-world images called RSRGAN to super-resolve the real-world LR observations with more perceptual details for upscaling factor ×4. This approach is the extension of our proposed RSRN model [11] (i.e., generator network of the proposed RSRGAN model).
- In RSRN, a densely connected parallel residual block is proposed which helps to learn more rich features in LR observations. The proposed RSRN sets new state-of-the-

art results in the real-world LR images with significant reduction in the number of the trainable parameters.
– For fair comparison, we re-train the recently proposed GAN-based models, i.e., SRGAN [27], SRFeat$_{IF}$ [33] and ESRGAN [39] with RealSR training dataset. The proposed RSRGAN model outperforms to those existing GAN-based state-of-the-art SISR methods in real-world SR application for upscaling factor $\times 4$.

In comparison with RSRN [11], the following extensions are carried out in this work.

– The ablation study of the RSRN model has been conducted with many experiments on the RSRN model. This ablation study proves the effectiveness of the structure design.
– In order to super-resolve the real-world LR images with high-frequency texture details, a computationally efficient GAN-based model called RSRGAN is proposed in this manuscript for upscaling factor $\times 4$.
– In addition to the ablation study of RSRN model, we have further carried out various experiments for validating the importance of different loss functions in the RSRGAN model.
– Different experiments have been carried out in order to validate the effectiveness of the proposed RSRGAN model. The quantitative and qualitative analysis of the proposed RSRGAN model is also discussed in brief in this manuscript.

The rest of the paper is organized as follows. We discuss the review of different CNN- and GAN-based SISR methods in Sect. 2. In Sect. 3, the architecture design of the proposed RSGAN method is discussed. The result analysis of the experiments is presented in Sect. 4. Finally, in Sect. 5, the conclusion of the work is presented.

2 Related work

Recently, CNN-based SISR methods obtain remarkable performance than that of traditional SISR methods. This is due to the recent advancement in graphical processing units (GPUs) and the availability of larger datasets. A detailed review of those methods has been investigated and evaluated in [2,18,40,42]. An SRCNN was the first SISR method using deep learning, and same was proposed by Dong et al. [13]. After this, many methods have been reported in order to improve the SR performance such as VDSR [24], DRCN [25], SRResNet [27], EDSR [29], SRFeat$_M$ [33], MSRN [28], DBPN [17], RDN [46], RCAN [45], SAN [12]. A brief details about those methods along with the proposed RSRN method [11] are summarized in terms of their training data, loss function to train the network, upsampling strategy, framework of their implementation, the number of training parameters required to train their networks and PSNR/SSIM value obtained on Set5 benchmark dataset in Table 1.

Here, one can notice that the recently proposed EDSR [29], RDN [46], RCAN [45] and SAN [12] methods obtain better PSNR and SSIM measures among all other methods but they obtain their performance with a cost of large number of training parameters. Hence, these methods have heavy computational complexity which makes them unsuitable for real-world applications. However, the proposed RSRN model [11] offers less number of training parameters (i.e., approximately 60–80% less training parameters than EDSR, RDN, RCAN and SAN models). One can also see in Table 1 that the proposed RSRN [11] obtains similar performance

Table 1 The summary of different CNN-based SR algorithms for upscaling factor $\times 4$

Models	Training data	Loss	Up-sampling	Framework	Number of parameters	Set5 ($\times 4$) PSNR/SSIM
SRCNN [13]	ImageNet subset	L_2	Pre	Caffe	57K	30.48/0.8628
VDSR [24]	G200 + Yang91	L_2	Pre	Caffe	665K	31.35/0.8838
DRCN [25]	Yang91	L_2	Pre	Caffe	1775K	31.53/0.8854
SRResNet [27]	ImageNet subset	L_2	post	Theano	1500K	32.05/0.8910
MSRN [28]	DIV2K	L_1	Post	PyTorch	6078K	32.26/0.8960
SRFeat-M [33]	DIV2K	L_2	Post	Tensorflow	6196K	32.29/0.8957
RSRN [11]	**DIV2K + Flickr2k**	$\boldsymbol{L_1}$	**Post**	**PyTorch**	**5370K**	**32.40/0.8976**
DBPN [17]	DIV2K + Flickr2k + ImageNet subset	L_2	Post	Caffe	10,426k	32.42/0.8975
EDSR [29]	DIV2K	L_1	Post	Torch	43M	32.46/0.8968
RDN [46]	DIV2K	L_1	Post	Torch	21.9M	32.47/0.8990
RCAN [45]	DIV2K	L_1	Post	PyTorch	16M	32.63/0.9002
SAN [12]	DIV2K	L_1	Post	PyTorch	15.6M	32.64/0.9003

Here, the details of the proposed RSRN method [11] are mentioned in bold font texts

with a recently proposed DBPN method [17] on Set5 dataset with approximately 50% less number of training parameters. This comparison is further explained in detail in the result analysis Sect. 4.2. All those CNN-based models rely on minimizing pixel-wise loss functions such as MSE, L_1 and Charbonnier, which results in blurry images as the minimization regresses to an average of all possible solutions. Hence, they are prone to restore high-frequency details. However, GAN has an ability to restore the high-frequency information from LR observations.

SISR using GAN (SRGAN) was proposed by Ledig et al. [27], and it was the first GAN-based SR model that is capable of inferring perceptually visible images for a large upscaling factor. After that, many GAN-based SISR methods have been proposed in the literature [33,35,39]. Sajjadi et al. [35] propose a SISR method, called EnhanceNet in which automated texture synthesis in combination with perceptual and adversarial losses are used to produce realistic texture details. However, both of the above methods fail to restore the high-frequency details which have been lost during the downsampling process and they generate arbitrary high-frequency artifacts in their SR results. To remedy the above problem, Park et al. [33] propose a GAN-based SISR method called SRFeat$_{IF}$ which overcome the above limitation by adopting feature discriminator to regress to the real distribution of features and generate more plausible high-frequency details. Recently, authors in [39] enhance the performance of SRGAN [27] (called as ESRGAN) by introducing a Residual-in-Residual Dense Block (RDDB) and relativistic GAN [23] approach in generator and discriminator network, respectively.

In Table 2, we summarize all the recent GAN-based SISR methods along with the proposed approaches in terms of their training datasets, number of parameters, loss functions to train the network and PI/RMSE values obtained on BSD100 benchmark dataset. The details of the proposed method are mentioned in bold font text in this table. Here, a pair of distortion and perception measures (i.e., PI and RMSE) can be used combinedly to evaluate the perceptual quality of SR image [5]. In the PIRM2018 challenge [6], organizers use this idea to validate the perception quality of SR results in which lower PI value with lesser RMSE value indicates SR observations with better perception quality. Hence, we use the same to evaluate the SR performance in this work. From Table 2, one can notice that the proposed RSRGAN model outperforms to the state-of-the-art SR methods such as EnhanceNet [35] and SRGAN [27] methods. However, RSRGAN sets comparable SR performance than that of recently proposed SRFeat$_{IF}$ [33] and ESRGAN [39] methods with a significant less number of training parameters.

All the above SISR methods are trained on simulated training datasets where the LR images are synthesized by a simple and uniform bicubic degradation function. Such SISR methods may exhibit poor performance on real-world LR images where the degradation functions are unknown. Recently, Cai et al. [7] reviewed the third NTIRE challenge on SISR (restoration of rich details in an LR image) which was aimed at the real-world SISR problem with an unknown degradation function. Cai et al. [8] have released a new benchmark dataset called RealSR which consists of HR-LR images of the same scenes captured by adjusting the focal length of a digital camera. They proposed a network named Laplacian pyramid-based kernel prediction network (LP-KPN) to recover the real-world HR images [8]. After this, several models have been reported to solve the SISR for real-world images [9,15,26,41]. Moreover, Cheng et al. [9] propose an encoder–decoder-based residual network for the real SR approach. They employ coarse to fine method which gradually restores lost information and reduces the noise effects. Kwak et al. [26] introduce a fractal residual network to super-resolve the real-world LR image by using autoencoder-based loss function. They also propose an inverse pixel shuffle at the beginning of the network architecture which helps them to reduce the number of training parameters. For high fidelity recovery of image details, Du et al. [15] propose an orientation-aware deep neural network (OA-DNN) which consists of several orientation attention modules (OAMs). Here, in each OAM, three well-designed convolution layers are used to extract orientation-aware features in different directions. Xu and Li [41] have introduced a spatial color attention-based network called SCAN for real SR. In SCAN, the spatial color attention module is designed to jointly exploit the spatial and spectral dependency within color images.

3 Proposed method: RSRGAN

The proposed RSRGAN consists of generator and discriminator networks. The generator network tries to fool the discriminator network by generating perceptually plausible SR images, while the discriminator network discriminates the generated SR image as a fake image. In this section, we first discuss the architecture design of the generator network of the proposed RSRGAN (i.e., RSRN [11]) and then the discriminator network of RSRGAN is discussed. In the last, we discuss the different loss functions which have been employed to train the proposed RSRGAN model.

3.1 Generator network (G): RSRN [11]

The overall network architecture of the proposed RSRN model [11] to super-resolve the real-world images for upscaling factor $\times 4$ is depicted in Fig. 2. Here, an LR image (I_{LR}) is applied as input to the generator network which generates the corresponding SR image as output. First, a convolution layer

Table 2 The summary of different GAN-based SR algorithms for upscaling factor (×4)

Models	Training datasets	Number of parameters	Losses	PI/RMSE
EnhanceNet [35]	ImageNet	0.852M	VGG54 + Standard GAN + Texture loss	2.9069/16.7386
SRGAN [27]	ImageNet	1.549M	VGG54 + Standard GAN	2.3513/16.3332
SRFeat$_{IF}$ [33]	DIV2K	6.196M	VGG54 + Standard GAN + Feature GAN	2.5188/15.5763
ESRGAN [39]	DIV2K + DF2K	16.697M	VGG54 + RaGAN	2.4868/16.3729
RSRGAN	**DIV2K + Flickr2K**	**5.370M**	**VGG54 + RaGAN + L_1**	**2.4911/16.1776**

Here, the details of the proposed model are mentioned in bold text

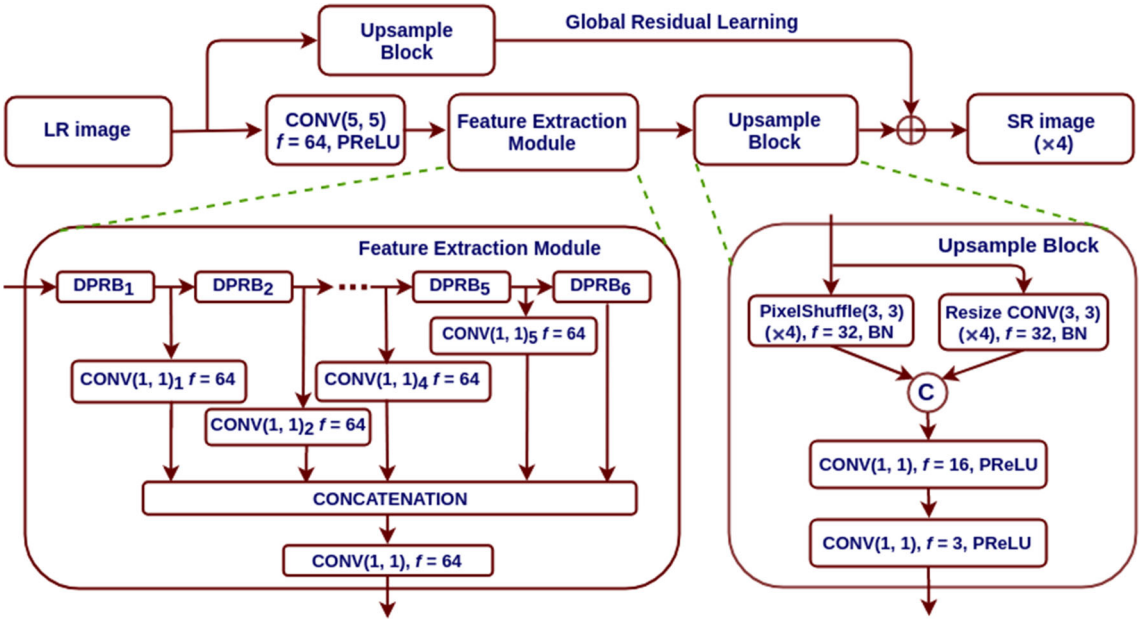


Fig. 2 The network architecture of the generator network (G) called RSRN [11] of the proposed RSRGAN model. Here, f indicates the number of feature maps

with 64 number of kernels is used to extract the low-level features from the LR input image as

$$O_{conv} = F_{conv}(I_{LR}), \tag{2}$$

where F_{conv} is the function of convolution layer and the O_{conv} is the extracted low-level feature maps that are passed through the feature extraction module. The feature extraction module is used to learn complex rich features from the low-level feature information, and it can be mathematically represented as,

$$O_{FEM} = F_{FEM}(O_{conv}). \tag{3}$$

Here, F_{FEM} denotes function of the feature extraction module and the O_{FEM} indicates the output feature maps obtained from the feature extraction module. The architecture design of feature extraction module is also illustrated in Fig. 2 which consists of six residual blocks.

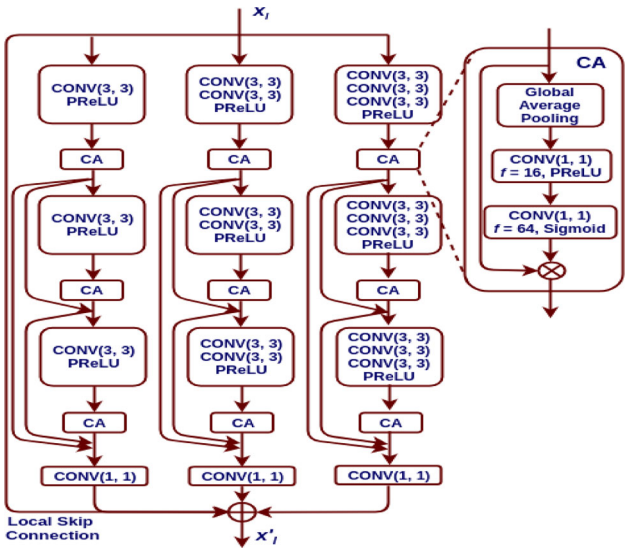


Fig. 3 The network design of densely connected parallel residual block (DBPR)

Here, we propose a novel residual block called densely connected parallel residual block (DPRB) inspired from the Inception [10] and the DenseNet [21] modules. Figure 3 shows the network design of the proposed DPRB which helps the network to extract the high-frequency abstract level features. The DPRB consists of three densely connected convolutional blocks in three parallel stacks. Each convolution block consists of several convolution layers, a parametric ReLU [19] activation function and one channel attention (CA) module [45]. Here, the CA module is employed to rescale the channel-wise features adaptively by considering interdependencies between channels. In DPRB, the local skip connection is used which helps the network to reduce the vanishing and exploding gradient problems [20].

The output of all DPRBs is concatenated via one convolution layer except the last DPRB. The concatenated feature maps are then passed through one convolution layer which convert the concatenated feature maps into the desired number of feature maps (see Fig. 2). Now, the output of the feature extraction module is passed through the upsample block which upsamples the feature maps to the desired level which is represented as,

$$O_{UP} = F_{UP}(O_{FEM}). \tag{4}$$

Here, F_{UP} defines the function of upsample block. Here, new upsampling approach is proposed which is also illustrated in Fig. 2. This approach consists of two upsampling modules; pixel-shuffle [36] and resize convolution [32]. Both upscaling modules are connected parallelly where the output feature maps of both upsampling modules are concatenated and then passed through two convolution layers.

Inspired from VDSR model [24], we also adopt the global residual learning (GRL) where the long skip connection is used to connect the input image and output residual image. Such GRL approach helps the network to stabilize the training process and reduces the color shifts in the output image. Instead of using the bicubic interpolation layer in GRL, we use the proposed upsample block with an upscaling factor $\times 4$ in GRL network. Finally, the output SR image (I_{SR}) is generated for upscaled factor $\times 4$ as,

$$I_{SR} = O_{UP} + F_{GRL}(I_{LR}), \tag{5}$$

where F_{GRL} represents the function of the proposed GRL network.

3.2 Discriminator network

The super-resolved image (i.e., I_{SR}) is passed through the discriminator network which discriminates it from real ones. The discriminator network is depicted in Table 3 which consists of eight convolution layers with kernel filters increased by a factor of 2 from 64 to 512 followed by two fully connected layers and one sigmoid layer. The proposed discriminator network follows the architecture guidelines suggested by Radford et al. [34]. In order to maintain the size of image at the output of each convolutional layer, strided convolutions are used in the proposed method whenever the number of features are doubled. The discriminator network takes HR and SR images as inputs and discriminates them by giving probability value between 0 to 1.

Table 3 The architecture design of discriminator network of the proposed RSRGAN model

Input size	Layer	k, s, f	Output size
192, 192, 3	Input Image	–	–
192, 192, 3	Conv, LReLU	3, 1, 64	192, 192, 64
192, 192, 64	Conv, BN, LReLU	3, 2, 64	96, 96, 64
96, 96, 64	Conv, BN, LReLU	3, 1, 128	96, 96, 128
96, 96, 128	Conv, BN, LReLU	3, 2, 128	48, 48, 128
48, 48, 128	Conv, BN,LReLU	3, 1, 256	48, 48, 256
48, 48, 256	Conv,BN, LReLU	3, 2, 256	24, 24, 256
24, 24, 256	Conv, BN, LReLU	3, 1, 512	24, 24, 512
24, 24, 512	Conv, BN, LReLU	3, 2, 512	12, 12, 512
12, 12, 512	Flatten	–	73,728
73,728	FC-1, LReLU	–	1024
1024	FC-2, Sigmoid	–	1

Here, k, s, f denotes the kernel size, stride value and number of feature maps, respectively

3.3 Loss functions

In this subsection, we describe the different loss functions which are used to train the proposed RSRGAN model. We train the proposed RSRN [11], i.e., the generator network of RSRGAN, using pixel-wise L_1 loss function in order to achieve better PSNR value. However, we formulate the loss function (L_{SR}) to train our GAN-based model, i.e., RSRGAN by weighted combination of L_1, perceptual (i.e., L_{VGG19}) and adversarial (i.e., $L_{adversarial}$) losses as,

$$L_{SR} = \lambda_1 \cdot L_1 + \lambda_2 \cdot L_{VGG19} + \lambda_3 \cdot L_{adversarial}. \tag{6}$$

Here, the λ_1, λ_2 and λ_3 are trade-off weighting constants of different loss functions. The pixel-wise L_1 loss function in image space is defined as [3],

$$L_1 = \frac{1}{r^2 wh} \sum_{x=1}^{rw} \sum_{y=1}^{rh} \| I_{HR(x,y)} - G(I_{LR})_{(x,y)} \|_1, \tag{7}$$

where, $G(I_{LR})$ is an SR observation produced from the generator network G. In the above equation, r is the downsampling factor while w and h indicate the width and height of the

LR image, respectively. Similar to MSE-based loss function, higher PSNR can be obtained by minimizing the L_1 loss function.

As mentioned earlier, the higher PSNR values are obtained by using the L_1 loss function in the training; however, such solution does not reflect the better quality of the SR image in appearance. Johnson et al. [22] and Dosovitskiy and Brox [14] propose a perceptual loss function to improve the visualizing quality where the loss function is computed in the feature space instead of in image space directly. Hence, the perceptual loss function [Eq. (8)] is used as content loss in combination with the adversarial loss to train proposed networks. In perceptual loss function, both I_{SR} and I_{HR} images are first mapped into the feature space by a feature map function (ϕ) obtained from a pre-trained VGG-19 network [37]. Then, the VGG loss is calculated between the feature maps of I_{SR} and I_{HR} as,

$$L_{VGG19(i,j)} = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} \|\phi_{i,j}(I_{HR(x,y)}) - \phi_{i,j}(G(I_{LR})_{(x,y)})\|_1, \tag{8}$$

where, $\phi_{i,j}$ indicates the feature map of j^{th} convolution layer before ith max-pooling layer within the VGG-19 network. Also, $W_{i,j}$ and $H_{i,j}$ relates the respective feature map dimensions within the VGG-19 network.

Recently, adversarial training is proven to be very effective in order to generate the visually appealable SR images [16]. In the adversarial training, the generator network is trained to learn the mapping between LR to HR image space. Here, we employ relativistic discriminator as suggested by Alexia et al. [23]. Different from the standard discriminator as used in SRGAN [27], which estimates the probability that the input image is real and natural, a relativistic discriminator tries to predict the probability that a real image I_{HR} is relatively more realistic than a fake one I_{SR}.

Specifically, the standard discriminator is replaced with the relativistic average discriminator [23]. The standard discriminator in SRGAN can be expressed as $D(x) = \sigma(C(x))$, where σ is the sigmoid function and $C(x)$ is the non-transformed discriminator output. Then, the relativistic average discriminator is formulated as $D_{Ra}(I_{HR}, I_{SR}) = \sigma(C(I_{HR}) - \mathbb{E}_{I_{SR}}[C(I_{SR})])$, where $\mathbb{E}_{I_{SR}}[\cdot]$ represents the operation of taking average for all fake data (super-resolved) in the mini-batch. The discriminator loss is then defined as,

$$L^D_{Ra} = -\mathbb{E}_{I_{HR}}[log(D_{Ra}(I_{HR}, I_{SR}))] - \mathbb{E}_{I_{SR}}[log(1 - D_{Ra}(I_{SR}, I_{HR}))]. \tag{9}$$

The generator loss is in a symmetrical form as,

$$L^G_{Ra} = -\mathbb{E}_{I_{HR}}[log(1 - D_{Ra}(I_{HR}, I_{SR}))] - \mathbb{E}_{I_{SR}}[log(D_{Ra}(I_{SR}, I_{HR}))]. \tag{10}$$

Here, it is observed that the adversarial loss for the generator contains both I_{HR} as well as I_{SR}. Therefore, the generator benefits from the gradients from both generated data and real data in adversarial training, while in standard GAN only generated part takes this effect.

The adversarial loss is a combination of generator loss (L^G_{Ra}) and discriminator loss (L^D_{Ra}) as,

$$L_{adversarial} = L^G_{Ra} + L^D_{Ra}. \tag{11}$$

Such simultaneous learning of generator and discriminator networks leads to adversarial behavior which tends to the min–max game where the generator is trained to fool discriminator by producing SR images highly similar to HR images, while discriminator is trained to distinguish SR images from HR images. By doing this process, the generator can learn to produce solutions that are highly similar to real images and thus, it becomes difficult to classify by discriminator. This encourages perceptually plausible solutions residing in the manifold of natural images.

4 Experimental analysis

Here, we discuss the implementation of the proposed model and analyze the experimental results for SISR on the upscaling factor ×4. The performance of the proposed method has been tested on the four synthetic testing benchmark datasets: Set5 [4], Set14 [43], BSD100 [31] and Urban100 [38] and on one real-world RealSR [8] testing dataset. Each dataset has different characteristics. The Set5, Set14 and BSD100 consist of natural scenes, while Urban100 contains urban scenes with details in different frequency bands. However, the RealSR testing dataset [8] consists of 100 real-world HR-LR image pairs of the same scene produced by adjusting the focal length of a digital camera.

We compare the SR performance of the proposed RSRN model [11] with only those methods which have less than 7M training parameters (except DBPN whose network has 10M number of training parameters). Such methods are SRCNN [13],[1] VDSR [24],[2] DRCN [25],[3] SRResNet [27],[4]

[1] https://github.com/jbhuang0604/SelfExSR.

[2] http://cv.snu.ac.kr/research/VDSR/.

[3] http://cv.snu.ac.kr/research/DRCN/.

[4] https://twitter.app.box.com/s/lcue6vlrd01ljkdtdkhmfvk7vtjhetog.

SRFeat$_M$ [33],[5] MSRN [28][6] and DBPN [17][7], and the SR results of these methods are obtained from the online available materials. In order to observe its effect on real-world images, we reproduce the SR results of SRResNet [27] and SRFeat$_M$ [33] by training on RealSR dataset. The SR performance of the proposed RSRGAN model is also compared with that of the existing GAN-based state-of-the-art SISR methods, i.e., EnhanceNet-PAT [35],[8] SRGAN [27], SRFeat$_{IF}$ [33] and ESRGAN [39].[9] We re-train the SRGAN, SRFeat$_{IF}$ and ESRGAN using the RealSR dataset to observe the effectiveness of the proposed RSRGAN model on real-world images. Additionally, we have also compare the SR performance of the proposed RSRGAN model along with that of recent real-world SISR model called LP-KPN [8][10] and the SR results this method have been obtained from the online available materials.

The PSNR and SSIM are used as metrics for quantitative comparison which are calculated after removing the boundary pixels of Y-channel images in YCbCr color space [27,33,39]. To perform the perceptual comparison of GAN-based SR methods, we employ distortion–perception pair [i.e., perceptual index (PI)-RMSE] strategy as suggested by Blau et al. [5]. We use this PI-RMSE pair as evaluation metrics to measure the perception quality, and the same is calculated by removing four border pixels of SR image. This strategy was also employed in PIRM 2018 challenge [6] to validate the perception quality of SR image. Here, lower the PI value with less RMSE measure indicates more perception quality in SR results. Additionally, we also use learned perceptual image patch similarity (LPIPS) metric to validate the SR results obtained using GAN-based SR methods which is introduced by Zhang et al. [44]. This metric is used to assess the perceptual similarity between two images. The lower value of LPIPS indicates a better perceptual quality of the SR image.

4.1 Training details and hyper-parameter settings

In the training, we are using two datasets: DF2K and RealSR. DF2K dataset consists of 2650 HR-LR pair images of Flickr2k [1] and 800 HR-LR pair images of DIV2K [1] datasets. To super-resolve the real-world images, we use the RealSR [8] dataset to train the proposed models, i.e., RSRN [11] and RSRGAN. The RealSR training dataset consists of 400 real-world HR-LR pair images captured using two different cameras, i.e., Canon and Nikon. The data samples in these datasets are augmented before the training process with flipping and random rotation up to 90°. The batch-size is set to 16 for both the models, i.e., RSRN [11] and RSRGAN.

We train the proposed RSRN model [11] on two different datasets for different applications. In the case of synthetic downsampled observations, the proposed model RSRN [11] is trained on the DF2K dataset up to 10^6 number of iterations. While for real-world super-resolution problem, we train RSRN model [11] on RealSR training dataset up to 5×10^4 number of iterations. In both cases, the proposed model RSRN [11] is trained using a L_1 loss function and optimized it on Adam optimizer with an initial learning rate of 10^{-4}. The learning rate is decayed by a factor of 2 at every 2×10^5 number of iterations.

We then employ the trained PSNR-oriented RSRN model as an initialization for the generator network of the proposed RSRGAN model. Similar to the training strategy of proposed RSRN [11] model, the proposed RSRGAN is also trained on two different datasets for two different applications using the combined loss function mentioned in Eq. (6) in Sect. 3.3. The weighting constants λ_1, λ_2 and λ_3 of Eq. (6) are set to value of 10^{-2}, 1 and 5×10^{-3}, respectively. For the synthetic SISR approach, it is trained on the DF2K dataset while in the case of a real-world scenario, it is trained on the RealSR dataset. In synthetic and real-world cases, the proposed RSRGAN model is trained up to 5×10^5 and 2×10^5 number of iterations, respectively. The learning rate is set to 10^{-4} for both the cases which is halved at every 1×10^5 number of iterations.

4.2 Result analysis

In this subsection, we discuss the quantitative and qualitative SR performance of the proposed RSRN [11] and RSRGAN models along with the existing CNN and GAN-based state-of-the-art SISR methods. The SR performance of RSRN [11] and RSRGAN models is analyzed and discussed for synthetic as well as real-world LR observations.

4.2.1 Ablation study of the RSRN model

In Table 4, we display the comparison of the proposed RSRN model [11] with different scenarios in terms of PSNR and SSIM measures for upscaling factor of 4. Here, we compare the SR results obtained using the proposed RSRN model [11] which is trained up to 5×10^5 number of iterations using different loss functions such as L_2 and Charbonnier. One can observe that the proposed RSRN model [11] trained using proposed L_1 loss function performs better than that of using L_2 and Charbonnier loss functions. To validate the proposed upsampling strategy, the proposed RSRN model [11] is also trained using SubPixelConv and ResizeConv upsampling

[5] https://github.com/HyeongseokSon1/SRFeat.

[6] https://github.com/MIVRC/MSRN-PyTorch.

[7] https://github.com/alterzero/DBPN-Pytorch.

[8] http://webdav.tuebingen.mpg.de/pixel/enhancenet/.

[9] https://github.com/xinntao/BasicSR.

[10] https://github.com/csjcai/RealSR.

Table 4 The comparison of proposed methods on testing datasets with different scenarios in terms of PSNR and SSIM values

RSRN	Set5		Set14		BSD100		Urban100	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Effect of loss function								
L_2	32.3019	0.8957	28.7011	0.7838	27.6336	0.7380	*26.2912*	0.7904
Charbonnier	32.2332	0.8951	28.6243	0.7822	27.5679	0.7360	26.1396	0.7884
Effect of network design								
SubPixelConv	32.2845	0.8953	28.6891	0.7834	**27.6375**	0.7379	26.2631	0.7899
ResizeConv	30.4306	0.8711	27.5794	0.7623	26.9685	0.7206	24.8249	0.7532
without CA	32.1828	0.8949	28.6562	0.7825	27.5972	0.7368	26.0911	0.7861
without GRL	*32.3325*	*0.8966*	**28.7076**	*0.7842*	27.6307	*0.7382*	26.2762	*0.7927*
Proposed	**32.3344**	**0.8968**	*28.7072*	**0.7843**	*27.6341*	**0.7388**	**26.2920**	**0.7931**

Here, the highest and second-highest values are mentioned with bold and italic fonts, respectively

Table 5 The quantitative comparison in terms of PSNR and SSIM measures for the upscaling factor ×4

Methods	Number of parameters	PSNR/SSIM			
		Set5	Set14	BSD100	Urban100
SRCNN [13]	57k	30.48/0.7503	27.49/0.7503	26.90/0.7101	24.52/0.7221
VDSR [24]	665k	31.35/0.8838	28.02/0.7678	27.29/0.7252	25.18/0.7525
DRCN [25]	1775k	31.53/0.8854	28.03/0.7673	27.24/0.7233	25.14/0.7511
SRResNet [27]	1549k	32.05/0.8910	28.53/0.7804	27.57/0.7354	26.07/0.7839
MSRN [28]	6166k	32.26/0.8960	28.63/0.7836	27.61/0.7380	26.22/0.7911
SRFeat$_M$ [33]	6078k	32.29/0.8957	*28.72*/0.7834	*27.65*/0.7371	26.26/0.7891
DBPN [17]	10,426k	**32.42**/*0.8975*	**28.75/0.7858**	**27.67**/*0.7389*	*26.38/0.7930*
RSRN [11]	5370k	*32.40*/**0.8976**	**28.75**/*0.7853*	*27.65*/**0.7396**	**26.39/0.7959**

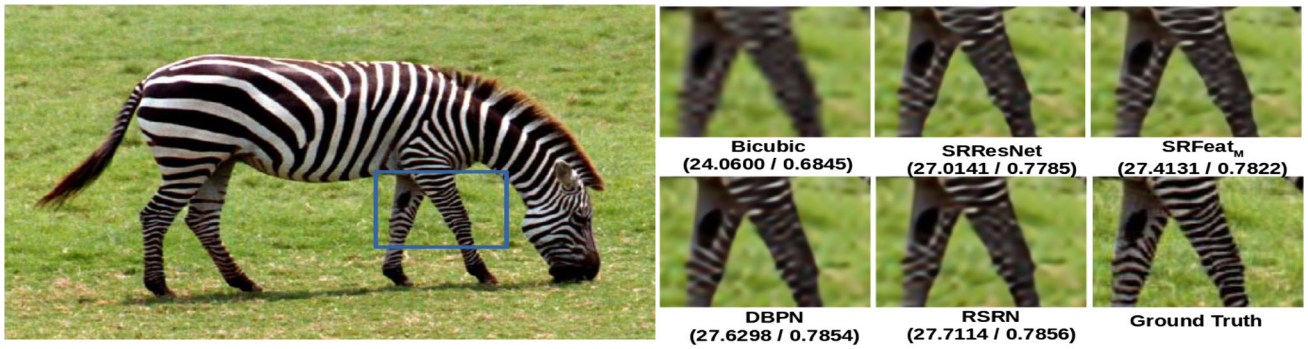
Here, the highest and second-highest values are mentioned with bold and italic fonts, respectively

strategies. From Table 4, one can found that the SR performance of the RSRN [11] with proposed upsampling strategy obtains better PSNR and SSIM values than that of SubPixel and ResizeConv upsampling strategies. In order to understand the effect of GRL strategy and use of channel attention (CA) module, we re-train the proposed RSRN model [11] without the GRL approach and without the CA module. The corresponding SR results are also mentioned in Table 4. One can observe here that the proposed method with the CA module has better PSNR and SSIM measures which justifies the effectiveness of the usage of the CA module in the proposed residual block. In case of No-GRL approach, the performance results of RSRN [11] without GRL approach have better PSNR measures for Set14 dataset while for remaining datasets, the proposed RSRN [11] with GRL approach has high PSNR and SSIM values than that of without GRL approach.

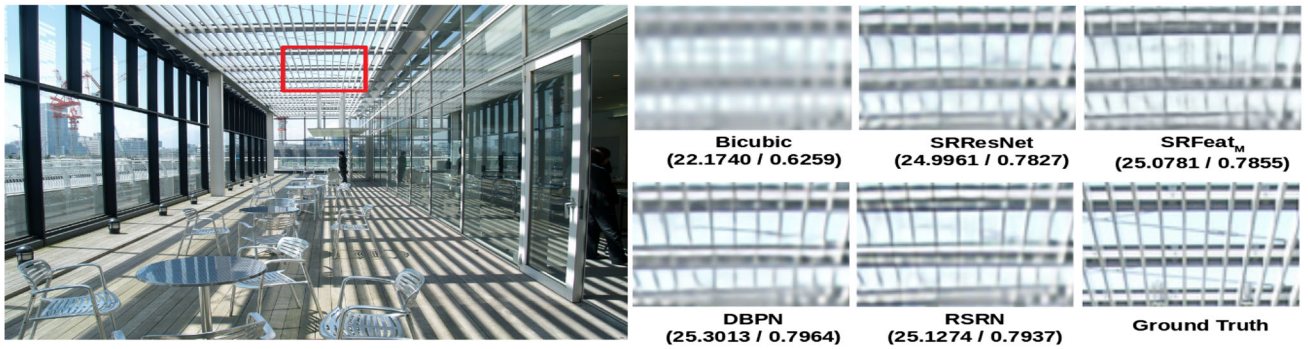
4.2.2 SR performance of the RSRN model

Table 5 shows the quantitative comparison in terms of PSNR and SSIM values obtained using the proposed RSRN [11] and other existing SR methods. For more intuitive comparison, the corresponding number of training parameters of respective SISR methods is also mentioned in Table 5. Here, the highest values are indicated with bold font and second-highest values are represented in italic font. In Table 5, we avoid comparing our SR results with that of EDSR [29], RDN [46], RCAN [45] and SAN [12] methods because these methods (i.e., [12,29,45,46]) require more than 15M number of training parameters in order to obtain better SR performance. From Table 5, one can notice that the proposed RSRN model [11] outperforms to the recently proposed MSRN [28], SRFeat$_M$ [33] and DBPN [17] methods. The proposed method RSRN [11] obtains the highest PSNR and SSIM measures than the other state-of-the-art methods except for PSNR measures of Set5 and BSD100 dataset and SSIM measures of Set14 dataset in which it obtains second-highest measures. However, our proposed RSRN method [11] sets this performance with approximately 50% less number of trainable parameters than DBPN [17] method.

In addition to quantitative comparison, the qualitative results obtained using the proposed and other existing state-of-the-art methods (i.e., SRResNet [27], SRFeat$_M$ [33] and DBPN [17]) are depicted in Fig. 4 for a single image of Set14, and Urban100 dataset for upscaling factor 4. Here, we eval-



(a) Set14 dataset



(b) Urban100 dataset

Fig. 4 The SR results obtained using the proposed RSRN method [11] along with the other existing state-of-the-art SISR methods on Set14 and Urban100 dataset for upscaling factor ×4

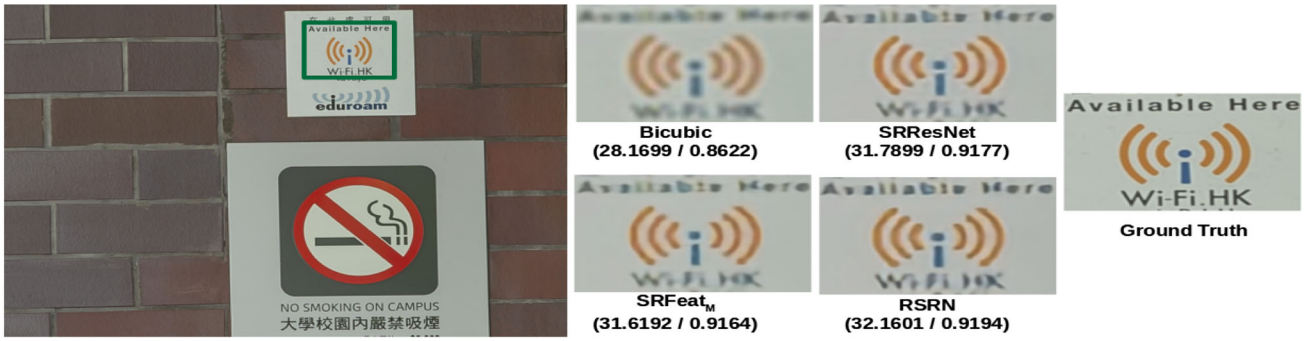
uate the SR performance for all testing benchmark datasets; however, due to the page constraint, we are displaying the SR results of two images only. The corresponding PSNR and SSIM of that SR image are also mentioned at the bottom of each SR result. Here, one can observe that the SR result of the RSRN model [11] preserves better high-frequency details than that of other methods with better PSNR and SSIM values.

We evaluate the proposed method RSRN [11] quantitatively as well as qualitatively on real-world image dataset, i.e., RealSR testing dataset [8]. As mentioned earlier, this dataset consists of 100 testing images which are different from the training data of the RealSR training dataset. Table 6 shows the quantitative comparison in terms of error measurements (i.e., PSNR and SSIM) as well as perception measurements (i.e., PI, RMSE and LPIPS) of the proposed method RSRN [11] along with two state-of-the-art methods (i.e., SRResNet [27] and SRFeat$_M$ [33]). We select these two methods as they are the most recent methods which perform better on the synthetic dataset than other existing methods. For better comparison, both methods are further trained on the RealSR dataset and the corresponding results are mentioned in Table 6. The number of training parameters used in those methods is also mentioned in the same table. Here, the highest values are depicted with bold font text. From Table 6, one can observe that the proposed method RSRN [11] outperforms in terms of error as well as perception measurements than that of the recently proposed SRFeat$_M$ method with less

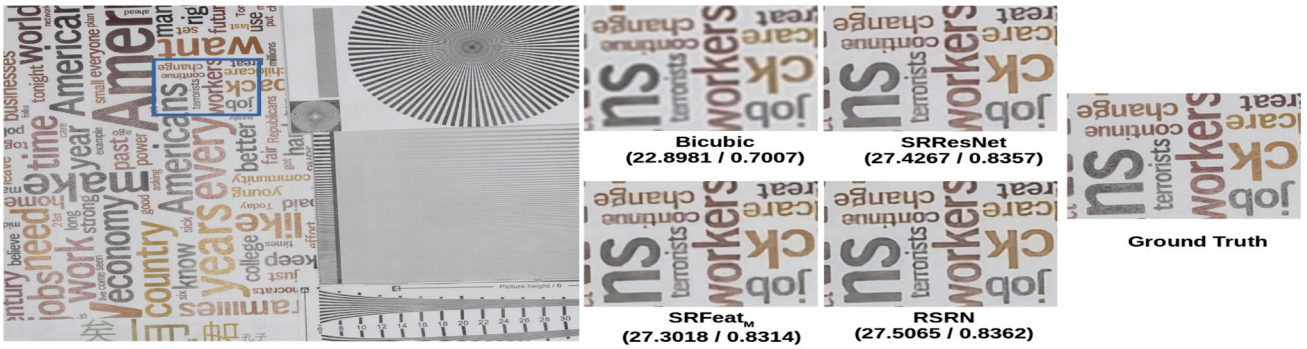
Table 6 The quantitative comparison in terms of error measurements (i.e., PSNR and SSIM) as well as perception measurements (i.e., PI, RMSE and LPIPS) on RealSR testing dataset for upscaling factor ×4

Metrics	Bicubic	SRResNet [27]	SRFeat$_M$ [33]	RSRN$_{synthetic}$	RSRN [11]
# of parameters	–	1549k	6166k	5370k	5370k
PSNR↑	27.2334	28.9408	28.9362	27.6506	**29.1555**
SSIM↑	0.7631	0.8161	0.8160	0.7803	**0.8197**
PI↓	8.1003	7.1298	7.2104	7.2838	**7.0253**
RMSE↓	12.2073	10.2102	10.1675	11.6373	**9.9299**
LPIPS↓	0.476	0.309	0.294	0.442	**0.293**

Here, the highest value is indicated with bold font text



(a) Canon camera Image



(b) Nikon camera Image

Fig. 5 The qualitative comparison of SISR methods on the RealSR testing dataset for upscaling factor ×4. The corresponding PSNR and SSIM values are also mentioned at the bottom of all SR results

number of training parameters. To understand the effectiveness of the RealSR training dataset, we also compare the SR performance of the proposed RSRN model [11] trained on the synthetic dataset (i.e., RSRN$_{synthetic}$). Here, one can observe that the proposed RSRN model [11] trained on the RealSR dataset obtains better error as well as perception measurements on real-world images than that of RSRN$_{synthetic}$ (i.e., RSRN trained synthetic dataset).

In addition to the quantitative comparison, the qualitative comparison is also carried out to see the visual improvement in the SR results of the proposed RSRN model [11]. Figure 5 shows the qualitative comparison on two images of the RealSR testing dataset which are captured by Canon and Nikon cameras. The corresponding PSNR and SSIM measures are also mentioned at the bottom of the SR results. From Fig. 5, one can observe that the proposed method RSRN [11] exhibits better texture details along with better PSNR and SSIM measures than that of SRResNet [27] and SRFeat$_M$ models. However, the proposed RSRN [11] obtains this performance with significant less number of training parameters than that of SRFeat$_M$ model.

4.2.3 Ablation study of the RSRGAN model

As mentioned earlier, the proposed RSRGAN model is trained using L_1 and VGG-based perceptual losses along with relativistic GAN-based adversarial loss function in order to improve the perceptual quality of the SR results with weighting constant of 0.01, 1 and 0.005, respectively. In the ablation study experiments, the proposed model has been trained using different scenarios such as without L_1 loss, without perceptual loss, with different GAN-based adversarial losses as well as by using different weighting constants to RaGAN adversarial loss functions. Then after, those trained models are evaluated on RealSR testing dataset for upscaling factor ×4 and compared in terms of perception measures (i.e., LPIPS and pair of PI-RMSE measures). In Table 7, the comparison of these experiments is depicted. Here, a lower value of LPIPS indicates better perceptual quality in the SR images and as discussed earlier, lower value of PI along with low RMSE value is required to have better perceptual quality of SR results.

From Table 7, one can observe that the RSRGAN with proposed loss function have better perceptual measures than that of without L_1 and without perceptual loss functions. This shows the effectiveness of L_1 and perceptual loss functions in the proposed model. In order to understand the effec-

Table 7 The effect of different loss functions along with different weighting constants in the proposed RSRGAN model in terms of LPIPS and pair of PI-RMSE measures on RealSR testing dataset for upscaling factor ×4

RSRGAN+	LPIPS↓	PI↓/RMSE↓
Effect of different loss functions		
No L_1 loss	0.213	4.7923/12.8056
No Perceptual loss	0.268	4.8263/14.2769
SGAN loss [16]	0.211	4.9893/12.1707
LSGAN loss [30]	0.217	5.7193/11.1103
RaGAN loss [23] (proposed)	0.212	4.7312/12.1776
Effect of different weightage of relativistic GAN loss		
weightage-0.05	0.237	4.5333/13.8890
weightage-0.005 (proposed)	0.212	4.7312/12.1776
weightage-0.0005	0.210	5.2295/12.1742
weightage-0.00005	0.222	5.8235/10.9842

tiveness of the relativistic GAN (RaGAN)-based adversarial loss function in the proposed model, we have conducted two additional experiments. In the first experiment, the proposed RSRGAN model is trained using standard GAN (SGAN) [16] adversarial loss function and in the another experiment least square GAN (LSGAN) [30] adversarial loss function has been employed in the training of RSRGAN model. The perceptual measures obtained from these two experiments are also mentioned in the same Table 7. It can be observed from this table that the proposed model with RaGAN loss function outperforms to the proposed model with LSGAN loss function; however, in comparison with that of SGAN loss function, the LPIPS value of the proposed model is slightly less than that of proposed model with SGAN loss function. However, the RSRGAN with RaGAN loss function (proposed approach) has better pair of PI-RMSE measure than that of model with SGAN loss function. Due to such improvement in the proposed method using RaGAN loss function, we use the same in the proposed approach.

Also, we have conducted few experiments to understand the effectiveness of weighting constants. The quantitative results of these experiments are also depicted in the same Table 7. In the first experiment, we kept larger weighting constant (i.e., weightage-0.05); here, the proposed model obtains less PI value than that of proposed method while it has high RMSE value for the same case. Additionally, this experiment also achieves high LPIPS score and hence, it leads to perceptually less effective results than that of model with proposed weighting constant. For the comparison of the RSRGAN with smaller weighting constant, two additional experiments with smaller weighting constants (i.e., weightage-0.0005 and weightage-0.00005) have been conducted in which the proposed weighting constant (weightage-0.05) obtains better quantitative measurements than that of model with smaller weighting constants (see Table 7). This proves the effectiveness of the weighting constant of 0.005 in the proposed RSRGAN model.

4.2.4 SR Performance of the RSRGAN model

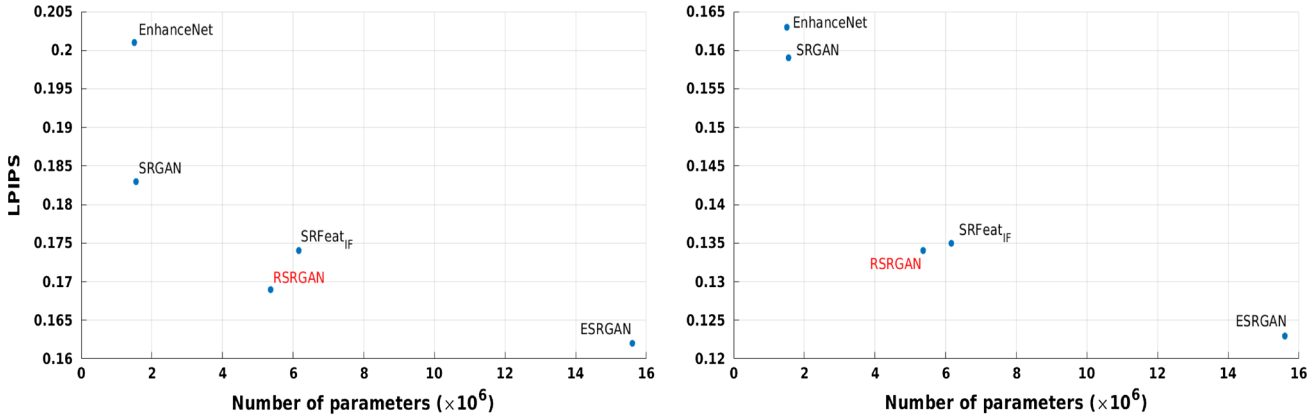
Here, we discuss the quantitative and qualitative comparison obtained using the proposed RSRGAN and other state-of-the-art GAN-based SISR methods. The performance comparisons are based on synthetic and real-world data.

Comparison on synthetic data In order to measure the perceptual quality of the SR images obtained using the proposed RSRGAN, we use a new perceptual metric called LPIPS as suggested by Zhang et al. [44] which is also proven to be effective to measure the perception quality of images. Lower the LPIPS value indicates the higher perception quality of an image. Here, we use this metric to measure the perceptual quality of super-resolved images obtained from different GAN-based models along with the proposed RSRGAN model. In Fig. 6, the graph of LPIPS values versus the number of training parameters are depicted for BSD100 and Urban100 testing datasets for upscaling factor ×4. The proposed RSRGAN model obtains comparable performance in terms of LPIPS with that of recently proposed SRFeat$_{IF}$ and ESRGAN models; however, it sets this performance with significant less number of training parameters.

Comparison on real-world data We further evaluate the quantitative and qualitative evaluations of the proposed method RSRGAN on real-world image dataset. Here, first we compare the SR performance of the proposed RSRGAN model with recent real-world SR model called LP-KPN [8]; then after, the proposed RSRGAN model is compared with the GAN-based state-of-the-art methods.

The SR comparison with LP-KPN method is depicted in Fig. 7 and in Table 8 for RealSR testing dataset for upscaling factor ×4. Here, the qualitative comparison is displayed between bicubic, LP-KPN [8] and proposed RSRGAN models along with ground-truth HR images. Additionally, it is worth to mention that the testing is performed on all the images of dataset; however, for the comparison purpose, here we show the SR results obtained using three real-world images acquired using Canon and Nikon cameras from RealSR testing dataset. For better visualization, the zoomed-in patches of all results are also displayed along with the complete images. In addition to visual comparison, the values of different perceptual metrics (i.e., LPIPS and pair of PI-RMSE) are also mentioned at the bottom of each SR results.

From visual inspection (i.e., see zoomed-in patches in Fig. 7b, f, j), one can observe that the SR results of LP-KPN model have blurry appearance; however, the SR results of proposed RSRGAN have better preservation of texture



(a) BSD100 dataset **(b)** Urban100 dataset

Fig. 6 The comparison of GAN-based SISR method in terms of LPIPS versus number of training parameters for the BSD100 and Urban100 testing benchmark datasets

Table 8 The quantitative comparison in terms of LPIPS and pair of PI-RMSE measures on RealSR testing dataset for upscaling factor ×4

Model	LPIPS↓	PI↓/RMSE↓
Bicubic	0.476	8.1003/12.2073
LP-KPN	0.276	7.0196/10.6047
RSRGAN	0.212	4.7312/12.1776

details with high-frequency information and they are also close to ground-truth HR images. One can also observed from the quantitative measures mentioned at the bottom of each SR result that the proposed RSRGAN model have quite better LPIPS measures than that of LP-KPN model in all three testing images. Also, the PI measure is significantly lower in all testing images of RSRGAN model than that of the LP-KPN model. Such minimal values of LPIPS score also proves better perceptual performance of the RSRGAN model than the LP-KPN model. Additionally, from Table 8, it can be observed that the proposed RSRGAN model has better LPIPS and pair of PI-RMSE measures than that of LP-KPN model on complete RealSR testing dataset.

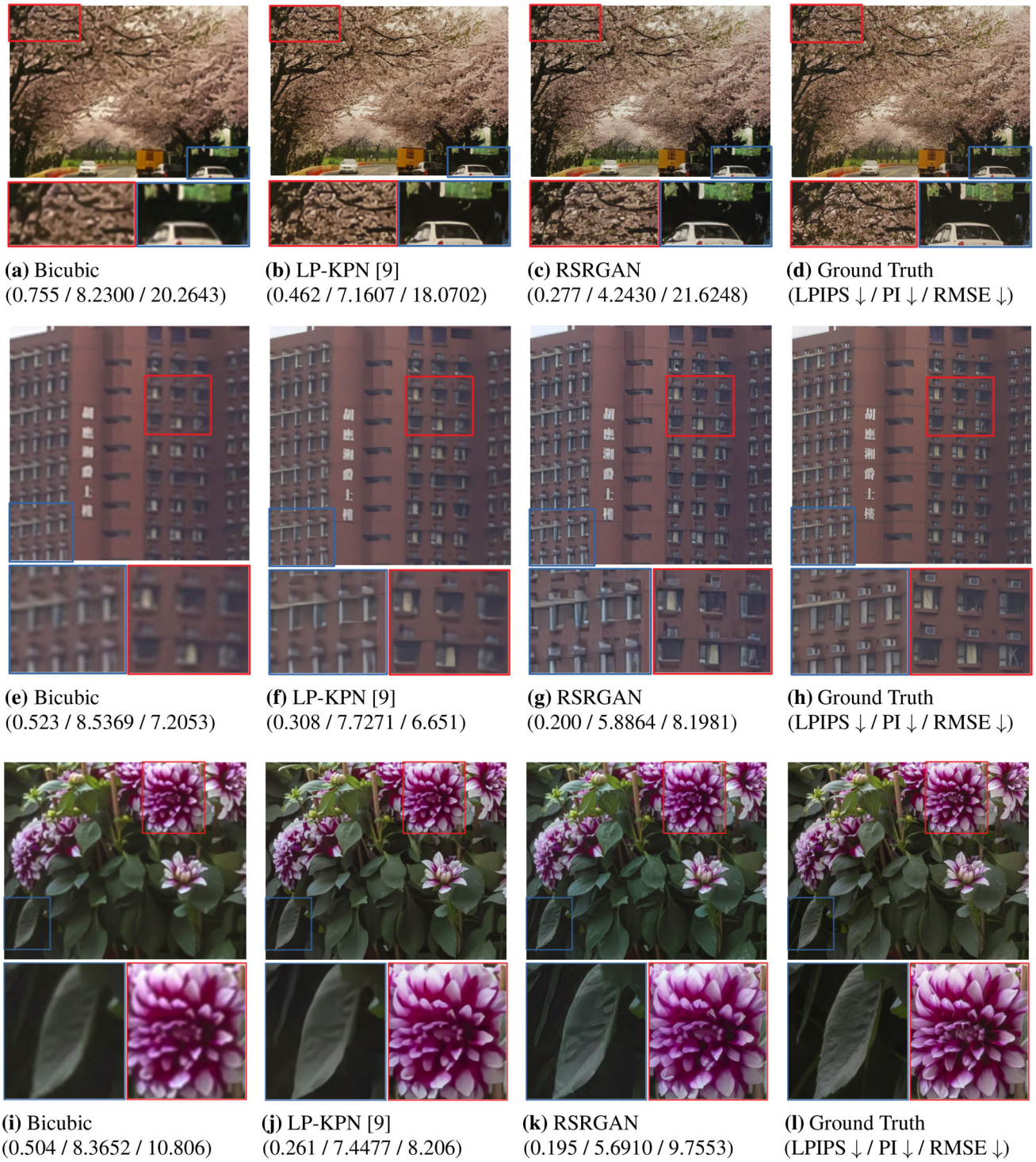
Furthermore, for the comparison with the GAN-based other existing methods, Table 9 shows the quantitative comparison of RealSR testing dataset in terms of error measurements (i.e., PSNR and SSIM) for upscaling factor (×4). For better comparison, the corresponding number of training parameters is also depicted in Table 9. For better comparison, we reproduce the SR results of SRGAN [27], SRFeat$_{IF}$ [33] and ESRGAN [39] models by training them using RealSR training dataset and the corresponding results are mentioned in Table 9. Here, the highest values are written in bold font text and the second-highest values are mentioned with italic font text. From this table, one can observe that the proposed RSRGAN obtains the highest PSNR and second-highest SSIM values on the RealSR testing dataset. However, the proposed RSRGAN offers significantly less number of training parameters than that of SRFeat$_{IF}$ and ESRGAN models.

To measure the perceptual quality of the real-world SR images obtained using the proposed RSRGAN model, here we estimate the pair of distortion–perception [i.e., perceptual index (PI) and root mean square (RMSE)] measures [6] and the same is depicted for RealSR testing dataset in Fig. 8a. A lower PI with low RMSE value indicates better perceptual quality. The proposed RSRGAN outperforms the SRGAN and SRFeat$_{IF}$ models while it obtains slightly high PI value with less RMSE value than that of the ESRGAN model. However, the proposed RSRGAN model obtains this performance with approximately 60% less number of training parameters with that of ESRGAN model. The GAN-based models are also compared in terms of LPIPS measures with respect to the number of parameters which is depicted in Fig. 8b. Here, lower LPIPS measure indicates better perception quality. From Fig. 8b, one can observe that the proposed

Table 9 The quantitative comparison in terms of PSNR and SSIM measures on the RealSR testing dataset for upscaling factor ×4

Metrics	SRGAN [27]	SRFeat$_{IF}$ [33]	ESRGAN [39]	RSRGAN
# of parameters	1549k	6166k	15,600k	5370k
PSNR ↑	24.2284	26.3857	*27.1275*	**27.4052**
SSIM ↑	0.7057	**0.7930**	0.7585	*0.7621*

Here, the highest values are written in bold font text and the second-highest values are mentioned with italic font text



(a) Bicubic (0.755 / 8.2300 / 20.2643) **(b)** LP-KPN [9] (0.462 / 7.1607 / 18.0702) **(c)** RSRGAN (0.277 / 4.2430 / 21.6248) **(d)** Ground Truth (LPIPS ↓ / PI ↓ / RMSE ↓)

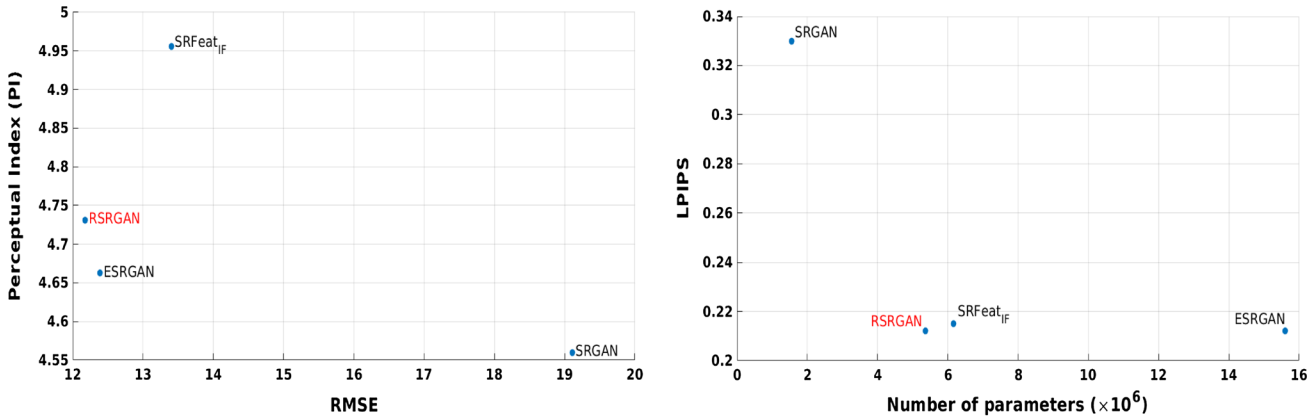
(e) Bicubic (0.523 / 8.5369 / 7.2053) **(f)** LP-KPN [9] (0.308 / 7.7271 / 6.651) **(g)** RSRGAN (0.200 / 5.8864 / 8.1981) **(h)** Ground Truth (LPIPS ↓ / PI ↓ / RMSE ↓)

(i) Bicubic (0.504 / 8.3652 / 10.806) **(j)** LP-KPN [9] (0.261 / 7.4477 / 8.206) **(k)** RSRGAN (0.195 / 5.6910 / 9.7553) **(l)** Ground Truth (LPIPS ↓ / PI ↓ / RMSE ↓)

Fig. 7 The visual and quantitative comparison obtained using the different LR images of RealSR dataset for upscaling factor ×4. The SR results obtained using the **a, e, i** bicubic, **b, f, j** LP-KPN model and **c, g, k** proposed method. **d, h, l** Ground-truth HR images. The corresponding LPIPS and pair of PI-RMSE values are mentioned at the bottom of all SR results

RSRGAN obtains lower LPIPS values on the RealSR testing dataset than that of other GAN-based models.

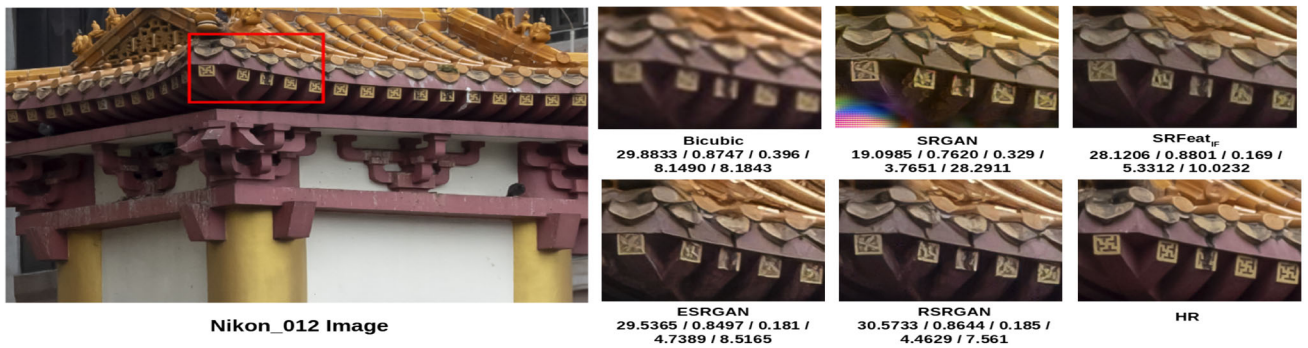
In addition to the quantitative comparison, Fig. 9 shows the qualitative comparison of the proposed and other existing GAN-based models on the RealSR testing dataset for



(a) PI vs. RMSE **(b)** LPIPS vs. Parameters

Fig. 8 The comparison of GAN-based SISR method for RealSR testing dataset for upscaling factor $\times 4$: **a** PI versus RMSE and **b** LPIPS versus number of training parameters

(a) Canon camera Image



(b) Nikon camera Image

Fig. 9 The qualitative comparison of SISR methods on RealSR testing dataset for upscaling factor $\times 4$. The corresponding PSNR, SSIM, LPIPS, PI and RMSE values are also mentioned at the bottom of all SR results

the upscaling factor $\times 4$. Here, it is worth to mention that the testing of the proposed RSRGAN method is performed on all the images of dataset; however, for the comparison purpose, here we show the results obtained using two real-world images captured using Canon and Nikon cameras of RealSR testing dataset. We compare the performance of our proposed RSGAN model with that of three GAN-based state-of-the-art SR methods such as SRGAN [27], SRFeat$_{IF}$ [33] and ESRGAN [39]. In order to see the preservation of high-frequency details in SR results, the zoomed-in regions of selected patches of all the SR results are displayed along with their results. In addition to that, the error measures, i.e., PSNR and SSIM, as well as perception measures, i.e., LPIPS, PI and RMSE of that SR images, obtained using different SR techniques are also mentioned at the bottom of each SR results. From Fig. 9, one can observe that the proposed method RSR-

GAN exhibits more perceptual details with better quantitative performance than that of SRGAN [27] and SRFeat$_{IF}$ [33] models, while it sets comparable performance with that of the ESRGAN [39] model. It can be seen from Fig. 9a that the proposed RSRGAN model has with slightly less PSNR and SSIM values than that of ESRGAN model [39] while it obtains better perceptual (i.e., PI and RMSE) as well as LPIPS measures than that of ESRGAN model. In case of Fig. 9b, the SR result of the proposed RSRGAN is very close to ground truth HR image and obtains better texture details than that of the SRGAN [27], SRFeat$_{IF}$ [33] and ESRGAN [39] models (see zoomed-in patches of SR results in Fig. 9b). Here, the proposed RSRGAN model also obtains better quantitative measures than that of other models (i.e., SRGAN, SRFeat$_{IF}$ and ESRGAN). However, the proposed RSRGAN sets this performance with significant less number of training parameters than that of SRFeat$_{IF}$ and ESRGAN models.

5 Conclusion

In this paper, we propose a GAN-based SISR method for real-world images. The literature on SISR shows that the CNN-based methods are trained using a synthetic training dataset in which the LR observations are generated by downsampling the HR images with known degradation function. Hence, they do not perform well on real-world images in which the degradation function is unknown. However, the SR performance can be improved by stacking more convolution layers. But such idea increases the number of training parameters and hence they are computationally inefficient for real-world applications. To tackle this problem, we propose a computationally efficient RSRN model [11] for real-world SR application for upscaling factor $\times 4$. In RSRN, a densely connected parallel residual block (DPRB) is introduced which helps the network to extract more complex features of LR observations. To observe the effectiveness of the RSRN model [11] on synthetic as well as real-world data, it is trained on synthetic and RealSR training datasets separately. The proposed RSRN model [11] trained on synthetic dataset outperforms the existing CNN-based SISR methods with significant less number of training parameters and sets new state-of-the-art SR results for upscaling factor $\times 4$. For the real-world SR approach, the recently proposed SRResNet [27] and SRFeat$_{M}$ [33] models have been re-trained using RealSR training dataset for better comparison. From experimental analysis, it is proven that the proposed RSRN model [11] trained using the RealSR dataset outperforms to these methods in terms of quantitative as well as qualitative measurements. In spite of obtaining better PSNR and SSIM values, the RSRN model fails to preserve the high-frequency details and hence it produces the overly-smooth blurry samples.

In order to obtain high-frequency details, the SR model using GAN for real-world image called RSRGAN is proposed for the upscaling factor $\times 4$. The proposed RSRGAN is trained using a weighted combination of VGG-based perceptual, adversarial and L_1 loss functions. Similar to the RSRN model [11], the proposed RSRGAN model is also trained using synthetic and RealSR training datasets, separately. For better comparison, SR results of the recently proposed state-of-the-art SISR methods, i.e., SRGAN [27], SRFeat$_{IF}$ [33] and ESRGAN [39] have been reproduced by training them on RealSR dataset. We compare these results with the results of the proposed RSRGAN model in terms of error measurements (i.e., PSNR and SSIM) as well as perception measurements (i.e., PI, RMSE and LPIPS) and conclude that the proposed RSRGAN outperforms SRGAN [27] and SRFeat$_{IF}$ [33] while it obtains comparable performance with ESRGAN [39] with significant less number of training parameters. We have also compared the SR performance of the proposed RSRGAN model with that of recent real-world SR model called LP-KPN [8]. From experimental analysis, we have found that the proposed RSRGAN generates SR samples with more high-frequency details and better perception measures than that of LP-KPN model.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: dataset and study. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 126–135 (2017)
2. Anwar, S., Khan, S., Barnes, N.: A deep journey into super-resolution: a survey (2019). arXiv preprint arXiv:1904.07523
3. Barron, J.T.: A more general robust loss function (2017). arXiv preprint arXiv:1701.03077
4. Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.L.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In: BMVC, pp. 135.1–135.10 (2012)
5. Blau, Y., Michaeli, T.: The perception-distortion tradeoff. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6228–6237 (2018)
6. Blau, Y., Mechrez, R., Timofte, R., Michaeli, T., Zelnik-Manor, L.: The 2018 PIRM challenge on perceptual image super-resolution. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 334–355 (2018)
7. Cai, J., Gu, S., Timofte, R., Zhang, L.: Ntire 2019 challenge on real image super-resolution: methods and results. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 2211–2223 (2019)

8. Cai, J., Zeng, H., Yong, H., Cao, Z., Zhang, L.: Toward real-world single image super-resolution: a new benchmark and a new model. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3086–3095 (2019)
9. Cheng, G., Matsune, A., Li, Q., Zhu, L., Zang, H., Zhan, S.: Encoder-decoder residual network for real super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 2169–2178 (2019)
10. Chollet, F.: Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1251–1258 (2017)
11. Chudasama, V., Prajapati, K., Upla, K.: Computationally efficient super-resolution approach for real-world images. In: 7th National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG). Springer, Singapore (2019) (accepted for publication)
12. Dai, T., Cai, J., Zhang, Y., Xia, S.T., Zhang, L.: Second-order attention network for single image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 11065–11074 (2019)
13. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE Trans. Pattern Anal. Mach. Intell. **38**(2), 295–307 (2016)
14. Dosovitskiy, A., Brox, T.: Generating images with perceptual similarity metrics based on deep networks. In: Advances in Neural Information Processing Systems, pp. 658–666 (2016)
15. Du, C., Zewei, H., Anshun, S., Jiangxin, Y., Yanlong, C., Yanpeng, C., Siliang, T., Ying Yang, M.: Orientation-aware deep neural network for real image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1944–1953 (2019)
16. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2672–2680 (2014)
17. Haris, M., Shakhnarovich, G., Ukita, N.: Deep back-projection networks for super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1664–1673 (2018)
18. Hayat, K.: Super-resolution via deep learning (2017). arXiv:1706.09077
19. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1026–1034 (2015)
20. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
21. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2261–2269 (2017)
22. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European Conference on Computer Vision, pp. 694–711. Springer, Berlin (2016)
23. Jolicoeur-Martineau, A.: The relativistic discriminator: a key element missing from standard GAN (2018). arXiv preprint arXiv:1807.00734
24. Kim, J., Kwon Lee, J., Mu Lee, K.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1646–1654 (2016)
25. Kim, J., Kwon Lee, J., Mu Lee, K.: Deeply-recursive convolutional network for image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1637–1645 (2016)
26. Kwak, J., Son, D.: Fractal residual network and solutions for real super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 2114–2121 (2019)
27. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., 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, pp. 4681–4690 (2017)
28. Li, J., Fang, F., Mei, K., Zhang, G.: Multi-scale residual network for image super-resolution. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 517–532 (2018)
29. Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 136–144 (2017)
30. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Paul Smolley, S.: Least squares generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2794–2802 (2017)
31. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of Eighth IEEE International Conference on Computer Vision, 2001. ICCV 2001, vol. 2, pp. 416–423. IEEE (2001)
32. Odena, A., Dumoulin, V., Olah, C.: Deconvolution and checkerboard artifacts. Distill **1**(10), e3 (2016)
33. Park, S.J., Son, H., Cho, S., Hong, K.S., Lee, S.: Srfeat: single image super-resolution with feature discrimination. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 439–455 (2018)
34. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks (2015). arXiv preprint arXiv:1511.06434
35. Sajjadi, M.S., Schölkopf, B., Hirsch, M.: Enhancenet: single image super-resolution through automated texture synthesis. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 4501–4510. IEEE (2017)
36. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: 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, pp. 1874–1883 (2016)
37. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2014). arXiv preprint arXiv:1409.1556
38. Single image super-resolution from transformed self-exemplars (cvpr 2015). https://github.com/jbhuang0604/SelfExSR
39. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., Change Loy, C.: Esrgan: enhanced super-resolution generative adversarial networks. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 63–79 (2018)
40. Wang, Z., Chen, J., Hoi, S.C.: Deep learning for image super-resolution: a survey (2019). arXiv preprint arXiv:1902.06068
41. Xu, X., Li, X.: Scan: spatial color attention networks for real single image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 2024–2032 (2019)
42. Yang, W., Zhang, X., Tian, Y., Wang, W., Xue, J.H., Liao, Q.: Deep learning for single image super-resolution: a brief review. In: IEEE Transactions on Multimedia, pp. 3106–3121 (2019)
43. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: International Conference on Curves and Surfaces, pp. 711–730. Springer, Berlin (2010)
44. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric.

In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 586–595 (2018)
45. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 286–301 (2018)
46. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2472–2481 (2018)

Vishal Chudasama is pursuing a Ph.D. at Sardar Vallabhbhai National Institute of Technology (SVNIT), Surat, India. His research interests include various deep learning-based image processing operations, high-level computer vision tasks such as object/face detection and recognition, and other related tasks.

Kishor Upla is an Assistant Professor in Sardar Vallabhbhai National Institute of Technology (SVNIT), Surat, India. He is currently a post-doctoral fellow at NTNU, Gjfvik, Norway. He received his Ph.D. degree from Dhirubhai Ambani Institute of Information and Communication Technology (DA-IICT), Gandhinagar, India. His areas of interest include signal and image processing, multispectral, low-resolution face recognition, and hyperspectral image analysis.