Multimedia Tools and Applications
https://doi.org/10.1007/s11042-019-08218-4

Super-resolution using lightweight detailnet network

Somayeh Barzegar[1] • Arash Sharifi[1] • Mohammad Manthouri[2]

Received: 16 February 2019 / Revised: 30 August 2019 / Accepted: 12 September 2019
Published online: 19 October 2019

Abstract

Image Super-Resolution is a complex method capable of converting a low-resolution image to a high-resolution image. Regarding many challenges of the Super-Resolution problem and a wide variety of its applications in image processing and man's interpretations, it is very crucial to find an operational method. Development of deep learning methods, especially Convolutional Neural Networks has increased the power of image enhancement methods including image Super-Resolution. The goal of this research is to propose a fast speed image Super-Resolution method using Convolutional Neural Networks. The proposed DetailNet network has a small structure to prevent the problems of training very deep networks. Super-Resolution is fast by this method due to its small simple network structure. The DetailNet network is designed to add details to an input image. According to the results, DetailNet has a significant ability to produce image details. This network has a general function. Therefore, the low-resolution image size can be increased first using any method, then DetailNet can enhance the image quality and add more details to the input image. The Proposed method is applied to natural color images which achieved acceptable results on benchmark datasets.

Keywords Super-resolution · Convolutional neural networks · Residual blocks · Natural images

1 Introduction

Super-Resolution is a process that converts a low-resolution image into a high-resolution one by increasing the image size and enhancing its quality. Super-Resolution is an Ill-Posed problem because it is possible to have multiple high-resolution images for a low-resolution image. Super-Resolution becomes more challenging when the size of the upscale factor is

✉ Arash Sharifi
a.sharifi@srbiau.ac.ir

[1] Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran

[2] Department of Electrical and Electronic Engineering, Shahed University, Tehran, Iran

increased. Accordingly, the model has to estimate lots of details for a low-resolution image due to the lack of essential information for efficiently generating the high-resolution image.

Super-Resolution can be used in various areas such as image enhancement, supervisory applications, disease detection, remote control, the initial step in image processing and image generation. Regarding the broad applications of Super-Resolution, providing an operational and accurate method is crucial.

According to many studies, simple mathematical methods such as interpolation are not successful enough in the Super-Resolution problem and the estimation of image details. The Super-Resolution problem requires techniques that can accurately add details to an image and generate a high-resolution image based on inadequate information of a low-resolution image. Super-Resolution methods can be divided into three categories which are interpolation-based, reconstruction-based, and learning-based methods [10]. Learning-based methods are widely used for Image Super-Resolution. Learning-based Super-Resolution methods learn a mapping from low-resolution image patch to a high-resolution image patch using a training dataset [18]. There are different kinds of learning-based methods such as sparse coding based and deep learning based methods [26]. Deep learning methods are powerful in data processing and learning complex interactions between input and output of the system. Hence, designing a machine learning system based on convolutional neural networks can efficiently enhance the resolution of an image.

Many Super-Resolution methods utilize very deep convolutional neural networks based on two main reasons. First, to generate high-quality images and second, to increase the efficiency of the Super-Resolution method in larger upscale factors. Increasing the number of layers and trainable parameters of a deep convolutional neural network can lead to some problems in the training of these networks. Using very deep convolutional neural networks will increase the computational complexity and the training and execution time. Although very deep convolutional neural networks have a significant capability of image Super-Resolution, it's not desirable to use these models for real-time applications. For example, using these complicated models as a pre-processing step for recognition systems will impose lots of computational and time-consuming loads.

Considering these possible problems, the aim of this research is to provide a fast speed Super-Resolution model. Therefore, the main focus of this research is to design and propose a small structure network capable of fast speed Super-Resolution and generating high-quality images.

The rest contents of the paper are as follows: In the second section, related work on image Super-Resolution is presented. The proposed Super-Resolution method is introduced in the third section. Section 4 contains experimental results and related findings. Section 5 includes an overall conclusion related to the proposed method. Finally, suggestions to develop the proposed method is discussed in section 6.

2 Related work

Many Super-Resolution models are based on deep learning methods. Implementing and using interpolation-based methods are simple which leads to its extensive applications. However, these kinds of linear methods are not able to signify complex dependencies between input and output. In practice, these methods cannot successfully estimate edge details and make the output of Super-Resolution blurry [2].

In [23] Anchored Neighborhood Regression (ANR) method is proposed for image Super-Resolution, which is an example-based method. The main focus of this method is to increase the Super-Resolution speed. ANR uses a combination of sparse learned dictionaries and neighbor embedding methods. In this method, neighbor embedding of a low-resolution image patch is anchored to the nearest atom in the dictionary and the related embedding matrix is pre-computed. The nearest neighbor is computed using the correlation of dictionary atoms instead of Euclidean distance.

The A+ Super-Resolution method in [24] is an improved version of ANR and an example-based method. A+ uses full training instead of learning the regressors on the dictionary. The neighborhoods in this approach are based on the nearest Euclidean distance of the low-resolution patches of training.

The proposed method (SRCNN) in [3] is a deep convolutional neural network that learns a mapping between low and high-resolution images. In this research, the Super-Resolution model is designed based on sparse coding in which all layers are optimized together. The proposed three-layer CNN provides a good restoration quality and has a fast speed which is a favorable feature for online usage. In this research, increasing the number of filters or their size improves the results, but increasing the number of layers does not improve the performance. First, the low-resolution image is upscaled by bicubic interpolation then the resulted image is fed to SRCNN.

In [27] a Super-Resolution model is proposed by the combination of deep learning methods and sparse coding. Sparse coding model (CSN) is based on LISTA [6] algorithm and designed as a neural network. CSCN is a cascade of several CSN networks. In fact, CSCN is a deep network in which the output of each CSN is interpolated by bicubic, then is applied to the next CSN. Using a cascade of CSNs allows the multi-scale Super-Resolution, and also provides more robustness to artifacts. In this method, domain knowledge in the form of sparse coding compresses the model size and makes the training faster and more efficient.

In [12] a Deeply-Recursive Convolutional Network (DRCN) is proposed for image Super-Resolution, which has 16 recursion layers. The recursion depth in this method is implemented without adding new parameters for additional convolutions. The recursive-supervision and skip connections are employed to improve the network performance and reduce the complexity of training using the standard gradient descent. The model has three sub-networks. The intermediate sub-network has recursive layers, and the rest of the layers are similar to the MLP[1] with one hidden layer. The output is a weighted average of the intermediate estimations and the interpolated input image.

In [13] a very deep convolutional network (VDSR) is proposed which is inspired by VGG [20] network. This method improves accuracy by increasing the depth to twenty layers and using residual learning between high-resolution and an interpolated low-resolution image. For the efficient extraction of contextual information over large regions of the image, a cascade of small filters is employed frequently in the structure of a deep network. The proposed network is trained on several upscale factors which makes it a multi-scale Super-Resolution model. The loss layer in this network has three inputs which are ground truth, low-resolution input, and the predicted residual image. Residual image is the difference between the ground truth and input image. The loss function is the Euclidean distance between the reconstructed image (sum of input and network output) and the ground truth image.

[1] Multi-Layer Perceptron

In [15] the Non-Convex Joint Bilateral Up-sampling (NCJBU) method is provided for guided depth map up-sampling to handle the edge inconsistency, especially for larger up-sampling factors. The results indicate that NCJBU can smooth the noise on depth maps, and effectively preserve fine details and sharp depth edges in larger up-sampling factors such as ×8 factor.

In [8] the DBPN network has iterative up-sampling and down-sampling layers and uses an error feedback mechanism for projection errors in each step. Each of up-sampling and down-sampling stages represents various kinds of image degradation and high-resolution components which are mutually connected. DBPN feeds the prediction error to each depth to revise sampling results. The super-resolved image is generated using the accumulation of self-correcting features of each up-sampling step. The method projects various kinds of the high-resolution image to the low-resolution feature space by down-sampling layers. DBPN provides good results on larger scales.

SRResNet is proposed in [14] which uses many ResNet blocks to provide the best results of PSNR and SSIM[2] on benchmark datasets. The proposed deep residual network is capable of restoring real image texture from very down-sampled images. The model can upscale natural images by ×4 factor.

Using MSE[3] as a loss function for Super-Resolution networks is very common, and training network to reduce it leads to an increase in the amount of PSNR. Since MSE and PSNR are defined based on pixel-to-pixel differences, they have limitations on providing related perceptual differences such as high details of the image context [7, 25]. Accordingly, some recent methods replace MSE with other types of functions to improve the quality of the generated images. Several examples of these methods are presented below.

SRGAN is proposed in [14] which is based on GAN[4] and optimized for a new perceptual loss. The perceptual loss is defined using high-level feature maps of the VGG network combined with a discriminator. The discriminator strengthens responses that make it hard to perceptually distinguish them from ground truth images. The adversarial network consists of a discriminator which is alternatively optimized with a generator network for solving the adversarial min-max problem. The perceptual loss function is the weighted average of a content loss and an adversarial loss. The content loss is a VGG loss in the form of the Euclidean distance between ground truth and feature representations of a reconstructed image. The adversarial loss is defined based on the discriminator probabilities on all training samples.

The study of [19] proposed a new combination of automatic texture synthesize with a perceptual loss to generate realistic textures. According to the results, utilizing perceptual losses will shift loss from the image space to a higher-level feature space derived from an object recognition system such as the VGG network. Hence, that leads to generating sharp results with a lower PSNR. In this research, four loss functions are defined including the pixel-based loss in image space (MSE), perceptual loss in feature space, texture matching and adversarial loss. Based on the results, the combination of perceptual, texture matching and adversarial losses generates images that are perceptually more real, but provides less PSNR value compared to the network with MSE.

The method presented in [21] is based on the works of [4, 5] with the adaptation to the Super-Resolution problem. The model encodes the feature correlation of an image in

[2] Structural Similarity Index
[3] Mean Square Error
[4] Generative Adversarial Network

a VGG network through the Gram matrix. The proposed model uses deep learning based on texture synthesis. Regarding the results, using the rich feature space can improve the transfer and reconstruction of high-frequency details of the image, as well as simplify the quality of results in a broad variety of textures and natural images. The goal of texture synthesis is to reduce perceptual differences by generating an output image based on an input texture. By some adjustments, this method can transmit valid texture details for a wide variety of textures, even if they are structural and regular. Although, it generates painterly artifacts for general natural scenes and produces inappropriate details for smooth areas.

3 Proposed method

The DetailNet network is inspired by ResNet [9] architecture and the idea of residual learning. The ResNet network using the residual blocks has been provided significant results in the image classification. Moreover, several Super-Resolution methods utilized the residual blocks in their architectures. Considering these two reasons, the architecture of DetailNet network also uses the residual blocks.

Employing residual connections is useful in many computer vision applications such as Super-Resolution and deep regression tracking methods. For instance, in [16] exploiting multi-level semantics across convolutional networks is efficiently performed by residual connections for regression tracking problem. The structure of the residual block is appropriate to use in the architecture of Super-Resolution networks for two reasons. First, applying the residual blocks allows the use of many layers, and second there are skip connections in its structure. Passing through many layers may lead to damaging some of the image details, therefore adding the input image or images from the earlier layers can preserve some of these details. Additionally, increasing network's depth and layers may slightly enhance the results and may lead to training problems such as vanishing gradient. Also, the results of several Super-Resolution methods indicates that using skip connections from the initial layers has a positive impact on the final image quality. Thus, the ResNet idea is used to prevent possible vanishing gradient problem and reduce the number of parameters while maintaining the network quality.

The proposed network uses four residual blocks in which the kernel size of convolutional layers varies in different blocks. The first two blocks of DetailNet are different from the original residual blocks of ResNet in kernel size. The architecture of a residual block of DetailNet is shown in Fig. 1. This structure is similar to the blocks of ResNet architecture, with the exception that DetailNet uses the sigmoid activation function instead of the ReLU. Evaluation of different kinds of activation functions for residual blocks indicates that sigmoid function is the best choice. Using sigmoid function provides better results with more sharpness and better color details. Also, a sigmoid function is a bounded function that has a smooth transition between 0 and 1 values which makes it a suitable activation function for the proposed method.

The order of the different layers of the DetailNet Network with their characteristics is reported in Table 1. B1 to B4 in Table 1, indicate four residual blocks and the related details of each block, including the convolution kernel size is shown on the right column of the table. The number of all feature maps is 32 except for layer 31 that is 3. The zero padding and stride size of all convolutional layers is defined in such a way that the size of the input and output of

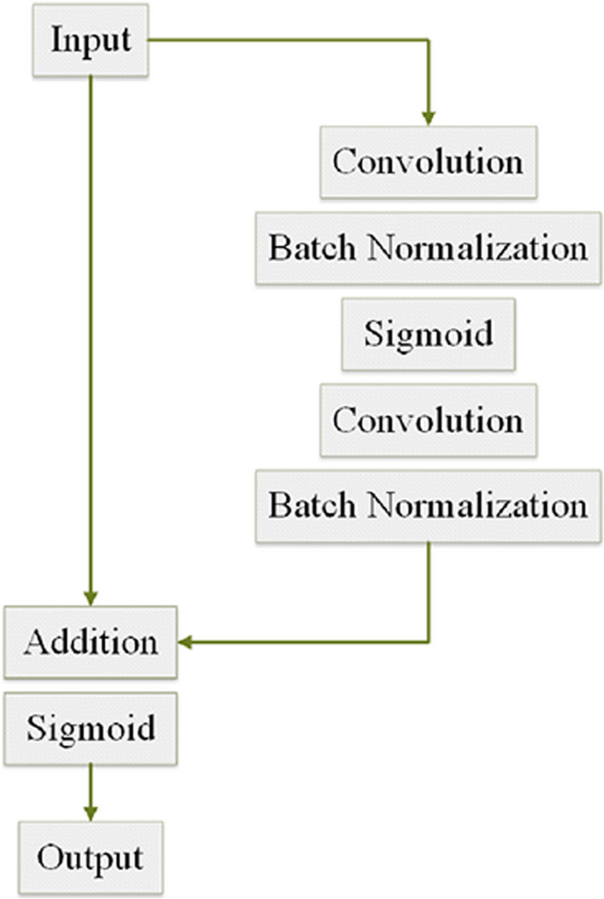

Fig. 1 Architecture of residual blocks used in DetailNet

each layer remains constant. Before and after four residual blocks, convolution 1 × 1 has been defined to adjust the number of feature maps. In this network, two loss functions are employed to train the DetailNet.

Table 1 Layers of DetailNet network

Layer Number	Layer	Details of Residual Blocks
1	Input image	
2	Conv 1 × 1	
3–9	B1	Conv 7 × 7 - Conv 5 × 5
10–16	B2	Conv 5 × 5 - Conv 3 × 3
17–23	B3	Conv 3 × 3 - Conv 3 × 3
24–30	B4	Conv 3 × 3 - Conv 3 × 3
31	Conv 1 × 1	
32	Sigmoid	
32	Sigmoid Cross Entropy Loss	
33	Euclidean Loss	

Defining the size of the kernels in the residual blocks, as reported in Table 1, is because of two reasons. First, in the first layers there is more potential for gaining more important information required for estimating the final image. Then, since there is a dependency among pixels, more neighbor pixels can participate in estimating each pixel. Thus, the kernel size of the first two blocks has been defined greater than last blocks while the kernel size of B3 and B4 is the same as residual blocks of ResNet.

In this method, first the bicubic interpolation upscales a low-resolution image, then the resulted image enters to DetailNet. The DetailNet takes an upscaled image as input and adds details to it. In fact, this network does not increase the image size and only learns to generate image details. So, the size of the input and output images of the network is identical. The proposed Super-Resolution method is shown in Fig. 2.

Super-Resolution using the DetailNet is designed to generate images with ample details by a small network structure, unlike those methods which use a large number of residual blocks such as [14] with 7 or [19] with 16 blocks. The main reason for designing DetailNet is to provide a model capable of adding details to an upscaled image which requires more details to have better quality.

The DetailNet depends on the upscale factor, but not on the image size. It has to be trained separately on each upscale factor, but can be applied to any image size. The structure of DetailNet is designed in a way that can add ample details to an upscaled image. In fact, this network has a general function. It is possible to perform the first part, which increases the image size, by any method then use the DetailNet to add more details to the image and enhance its quality.

4 Experimental results

In this section, the results of Super-Resolution using DetailNet network are reported on three color image datasets. Both quantitative and qualitative criteria are considered for comparison. Therefore, PSNR results of the images and running time are reported as quantitative criteria, and some of the images are provided for qualitative comparison.

4.1 Dataset

A set of natural images has been used to evaluate the quality of the proposed method. The combination of two datasets including a set of 91 images [28] and 200 images of the BSD200

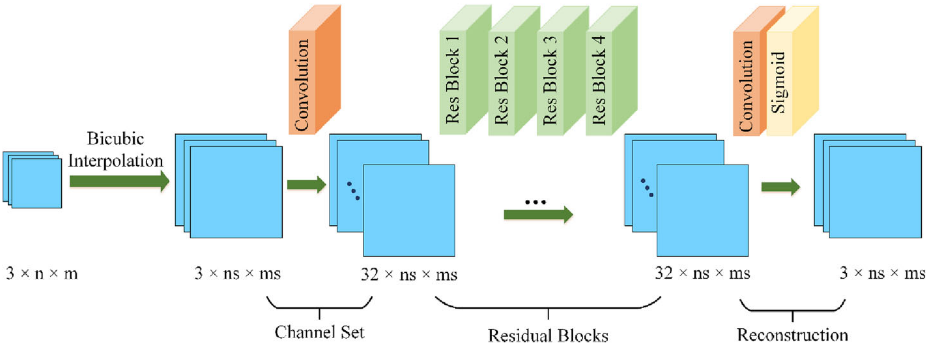

Fig. 2 Super-Resolution using DetailNet. (s) indicates the size of upscale factor

dataset [17] forms the train and test sets of the training phase. The 291 images are increased to 2328 images using data augmentation techniques applied in [22]. The original images are rotated by three degrees 90, 180 and 270, then flipped horizontally, which leads to constructing additional seven versions of each image. Each image is divided into 64*64 patches with a stride size of 44. Eighty percent of all image patches are considered as the train set and the remaining patches regarded as the test set. Thus, the train set contains 89,331 images and the test set includes 22,333 images. Details of the train and the test sets of the training phase are reported in Table 2.

Three benchmark datasets are employed to evaluate the proposed method, which are SET5 [1] with 5 images, SET14 [29] with 14 images and BSD100 [17] with 100 images.

4.2 Implementation details

The DetailNet network is trained by Caffe [11] deep learning framework. To make the training process faster, Cuda 8.0.6 and cuDNN v5.0 are employed. The models are trained and tested on a computer system with NVIDIA GeForce GTX 1070 GPU. Applying the trained models to test images and measuring the quantitative criteria are performed by Matlab 2017a software.

The size of input and target images in the training of DetailNet network are both 64*64. After training, DetailNet can be applied to images with any arbitrary size. The training of DetailNet is performed by the SGD[5] optimization method with a constant learning rate of 0.0001 and Weight-Decay of 0.0005 in one million iterations.

It should be noted that one of the images of the SET14, called Bridge, is a grayscale image. In the proposed method, the trained network on grayscale images should be applied to grayscale images. The only difference between the trained network on grayscale images and the trained network on color images is just the number of channels in the input and final output of the networks.

The evaluation of using different kinds of activation functions for the residual blocks of DetailNet indicates that using sigmoid function is better than ReLU. Using a sigmoid function provides better PSNR results. Accordingly, the Super-Resolution results of DetailNet with one residual block on ×2 upscale factor are provided with two different activation functions for the residual block. The results are evaluated on the images of SET5 dataset that can be seen in Table 3.

The appropriate number of residual blocks in the final design of the network structure is a tradeoff between the highest PSNR value and high execution speed. Accordingly, the results of the Super-Resolution using DetailNet network with one to four residual blocks are examined on the SET5 dataset on three upscale factors. Since the results on ×2 upscale factor are similar to other upscale factors in this evaluation, the results of Super-Resolution with ×2 upscale factor are presented. The results are shown in Table 4 and Fig. 3.

According to Table 4, increasing the number of residual blocks will increase the PSNR and the running time. As the number of blocks increases, the running time grows slightly, but the PSNR increases with a higher proportion. Considering the results of Fig. 3, DetailNet with three blocks and DetailNet with four blocks have not a significant difference in running time. But, regarding better PSNR results, number four is considered as the optimum number of residual blocks. In this study, the number of blocks is not defined greater than four, because the

[5] Stochastic Gradient Descent

Table 2 Details of the train and the test sets used in the training phase

Dataset	Number of			Patch size
	images after augmentation	train set	test set	
91 images BSD200	2328	89,331	22,333	64×64

purpose of this study is to find the best possible structure with a minimum number of parameters and high execution speed.

The kernel size of convolutional layers in the first two residual blocks of DetailNet has been defined greater than those in the last two blocks. Also, the kernel size of the convolutional layers in the first block is greater than those in the second block. Two different orders for the residual blocks are considered to evaluate the effectiveness of using bigger convolutional kernel size in the earlier residual blocks. Accordingly, the results of the Super-Resolution using DetailNet network with two different orders for the first two residual blocks are examined on the SET5 dataset on upscale factor ×2. The results are presented in Table 5.

The results of Table 5 indicate that using bigger convolutional kernel size in the earlier residual blocks provides better Super-Resolution results. Applying conv 7×7 in the first block provides better PSNR results than using conv 5×5 for the first block. Using bigger convolutional kernel size in the first block enables the model to employ more neighbor pixels of the earlier feature maps for the estimation, which can provide better results.

4.3 Super-resolution results

This section contains the results of the proposed method on natural images. The results have been evaluated through the value of PSNR, running time and visual quality of the images. The PSNR of images in three test sets in upscale factors ×2, ×3 and ×4 are reported in Table 6. Then, the resulted images of different methods are presented for qualitative evaluation. All the reported results are on the color images.

The quality of the DetailNet results is compared with the bicubic interpolation, SRCNN [3], ANR [23], A + [24], CSCN [27], VDSR [13] and DBPN [8] methods, all of which have been implemented based on the codes published by the authors of the papers. Due to the re-implementation of methods and the recalculation of the PSNR on identical images, the PSNR results may differ from the values presented in the main papers. The evaluation of the results is

Table 3 The PSNR of SET5 images with different activation functions for the residual block of 1-block DetailNet

Criteria	PSNR	
Activation Function	Sigmoid	ReLU
Baby	38.15	28.16
Butterfly	32.60	27.09
Bird	39.72	29.48
Head	35.37	26.18
Woman	34.93	24.55
Average	**36.15**	**27.09**

Table 4 The PSNR and running time of SET5 images with different number of residual blocks

Criteria	Time				PSNR			
Number of blocks	1	2	3	4	1	2	3	4
baby	0.04	0.08	0.11	0.14	38.15	38.15	38.30	38.32
bird	0.02	0.03	0.04	0.06	39.72	40.31	40.65	40.62
butterfly	0.02	0.02	0.04	0.05	32.60	33.14	33.51	33.62
head	0.02	0.03	0.04	0.06	35.37	35.56	35.54	35.45
woman	0.02	0.03	0.04	0.06	34.93	35.25	35.47	35.47
Average	**0.02**	**0.04**	**0.05**	**0.07**	**36.15**	**36.48**	**36.69**	**36.70**

performed with the focus on the three upscale factors ×2, ×3, and ×4, as they are common in all Super-Resolution methods.

The calculation of PSNR is performed similarly to other articles to have an accurate comparison between them. Some methods need to separate a margin from the images and then calculate the PSNR value to have a similar condition with those methods that have problems in producing the image border. The proposed method does not have any problem to produce the image border. But, according to other methods, it separates a margin in size of the upscale factor around the image and calculates the PSNR from the remaining center of the image. Also, the reported PSNR values in color images are calculated on the Y channel of the YCbCr color space to have a fair comparison with other methods.

The analysis of the results in Table 6 shows that after DBPN and VDSR, the DetailNet network has better PSNR results. According to the results, the difference between the DetailNet and VDSR reduces when the upscale factor increases.

The running time of different methods in two upscale factors of ×2 and ×4 on the images of the SET5 is calculated in a similar condition on a Core i7-7700HQ CPU 2.80 GHz. The running time of all methods is calculated using CPU to have a fair comparison since the codes of all methods are not available for running on GPU. Moreover, the running time of VDSR and DetailNet is calculated using GPU to signify the strength of DetailNet in fast speed Super-Resolution. VDSR is chosen for this comparison because it has the least running time difference with DetailNet on CPU. The related results are shown in Table 7 and Fig. 4.

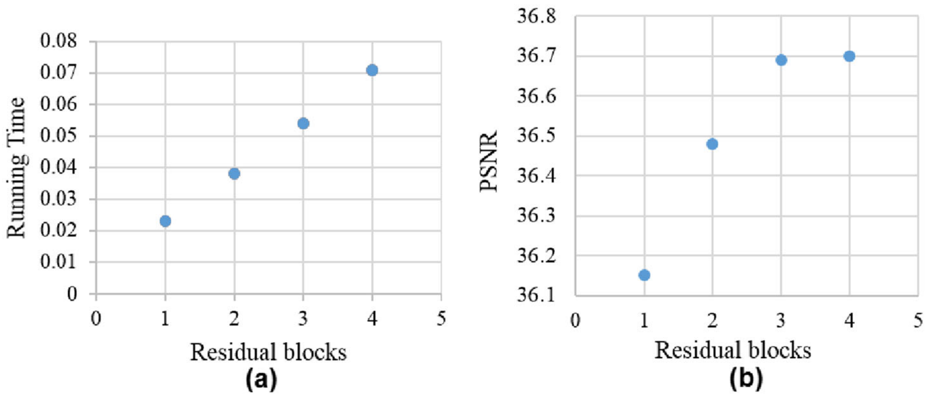

Fig. 3 The effects of increasing the number of residual blocks on the PSNR value and running time. (a) running time vs. number of residual blocks; (b) PSNR vs. number of residual blocks

Table 5 Super-Resolution using DetailNet network with two different orders for the first two residual blocks

Criteria	PSNR	
The order of the first two residual blocks	B1-B2(proposed DetailNet)	B2-B1
Baby	38.32	38.27
Butterfly	40.62	39.75
Bird	33.62	33.63
Head	35.45	35.52
Woman	35.47	35.44
Average	**36.70**	**36.52**

Based on the results of Table 7, DetailNet is the fastest method among DBPN, VDSR and SRCNN methods. Similar to SRCNN and VDSR, the running time in DetailNet does not change with the increasing of the upscale factor. In other methods, increasing of upscale factor reduces the running time. On CPU the speed of DetailNet is slightly better than VDSR, but on GPU DetailNet is significantly faster than VDSR.

The results of Fig. 4 indicate that DetailNet provides better PSNR results than four methods of SRCNN, ANR, A+, and CSCN. By increasing the PSNR value in the methods, the running time also increases. So, the DBPN method with the highest PSNR value has the highest running time with a considerable difference among methods. The quality of the five convolutional methods is compared in Fig. 5 based on their total number of trainable parameters. Approximately, DetailNet with 147,904 parameters has 4.5 times fewer parameters than VDSR and 40 times fewer parameters than DBPN.

To evaluate the qualitative results, three images from the SET5 and SET14 datasets are presented in three factors of ×2, ×3, and ×4 in Figs. 6, 7 and 8. In each figure, a complete image is presented, followed by a small part of that image to better represent the difference between the methods.

From left to right: images of first two rows are input low-resolution image, Super-Resolution using bicubic interpolation, ANR, A+ and SRCNN. Images of the last two rows are Super-Resolution using CSCN, VDSR, DBPN, DetaiNet and ground truth image. Corresponding PSNR of the methods and the size of the input and ground truth images are shown below of the images.

Table 6 The PSNR results on three test sets

Dataset	Scale	PSNR							
		Bicubic	ANR	A+	SRCNN	CSCN	VDSR	DBPN	DetailNet
Set5	**2**	33.65	35.84	36.54	36.66	36.59	37.56	**38.09**	36.70
	3	30.39	31.96	32.64	32.75	32.63	**33.67**	–	32.91
	4	28.42	29.72	30.32	30.48	30.41	31.35	**32.47**	30.67
Set14	**2**	30.24	31.80	32.32	32.42	32.34	33.02	**33.92**	32.48
	3	27.55	28.66	29.15	29.28	29.16	**29.77**	–	29.36
	4	26.00	26.88	27.33	27.49	27.39	28.00	**28.75**	27.62
BSD100	**2**	29.56	30.42	30.75	31.36	29.53	31.92	**32.27**	31.47
	3	27.21	27.85	28.14	28.41	26.77	**28.84**	–	28.51
	4	25.96	26.45	26.71	26.90	25.64	27.29	**27.72**	27.01

Table 7 Running time of different methods in ×2 and ×4 upscale factors on SET5 images

Image	Scale	Time(s)								
		Evaluated on CPU							Evaluated on GPU	
		SRCNN	ANR	A+	CSCN	VDSR	DBPN	DetailNet	VDSR	DetailNet
baby	**2**	9.54	1.00	1.10	1.40	4.47	62.13	4.70	0.36	0.14
	4	9.53	0.50	0.50	1.64	4.45	20.55	4.69	0.36	0.14
bird	**2**	1.84	0.30	0.30	0.45	1.53	10.66	1.46	0.32	0.06
	4	1.86	0.10	0.20	0.58	1.52	6.16	1.46	0.32	0.06
butterfly	**2**	1.58	0.20	0.20	0.37	1.35	8.33	1.20	0.33	0.05
	4	1.60	0.10	0.10	0.50	1.34	4.99	1.20	0.32	0.05
head	**2**	1.77	0.30	0.30	0.44	1.46	9.69	1.38	0.32	0.06
	4	1.79	0.10	0.10	0.56	1.45	5.77	1.38	0.32	0.05
woman	**2**	1.78	0.30	0.30	0.45	1.46	9.70	1.38	0.33	0.06
	4	1.76	0.10	0.10	0.54	1.46	5.83	1.38	0.33	0.05
average	**2**	**3.30**	**0.42**	**0.44**	**0.62**	**2.05**	**20.10**	**2.02**	**0.33**	**0.07**
	4	**3.31**	**0.18**	**0.25**	**0.77**	**2.04**	**8.66**	**2.02**	**0.33**	**0.07**

From left to right: images of first two rows are input low-resolution image, Super-Resolution using bicubic interpolation, ANR, A+ and SRCNN. Images of the last tow rows are Super-Resolution using CSCN, VDSR, DetaiNet and ground truth image. Corresponding PSNR of the methods and the size of the input and ground truth images are shown below of the images.

From left to right: images of first two rows are input low-resolution image, Super-Resolution using bicubic interpolation, ANR, A+ and SRCNN. Images of the last tow rows are Super-Resolution using CSCN, VDSR, DBPN, DetaiNet and ground truth image. Corresponding PSNR of the methods and the size of the input and ground truth images are shown below of the images.

The results of Figs. 6, 7 and 8 show that bicubic interpolation provides very blurry images. The ANR and A+ also produce blurry images and provide weak performance in generating the image details. The SRCNN method provides better results than CSCN, but the details of SRCNN differ from the details of real images. The results of the DetailNet

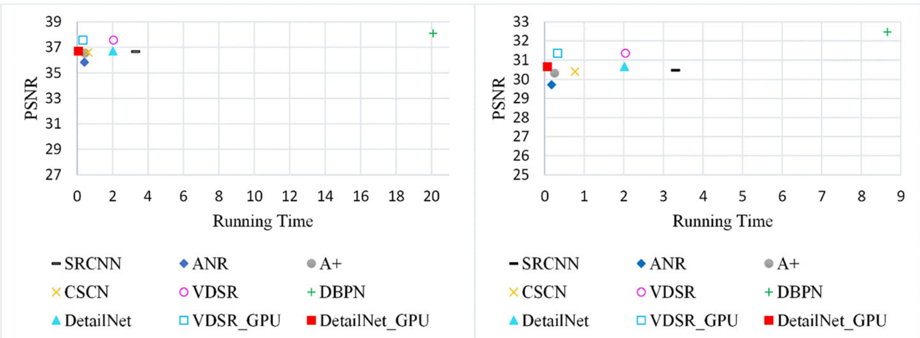

Fig. 4 The average PSNR to the average of running time on the SET5 dataset. (a) the results of factor ×2; (b) the results of factor ×4

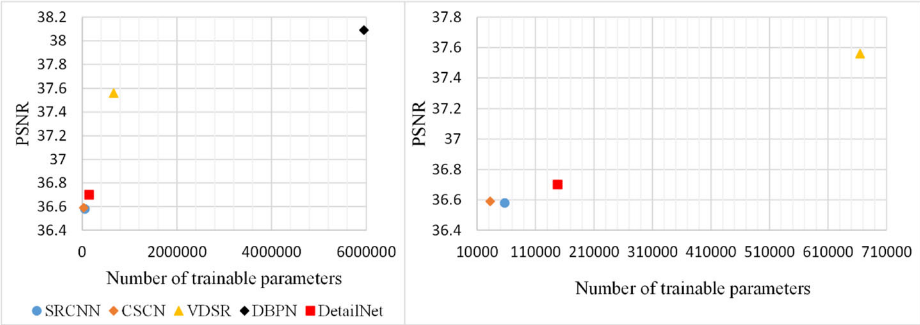

Fig. 5 The average PSNR to the total number of trainable parameters of five methods. (a) all methods are present; (b) DBPN is removed for better observation

method are apparently close to the results of both VDSR and DBPN methods. Therefore, it is hard to distinguish among them.

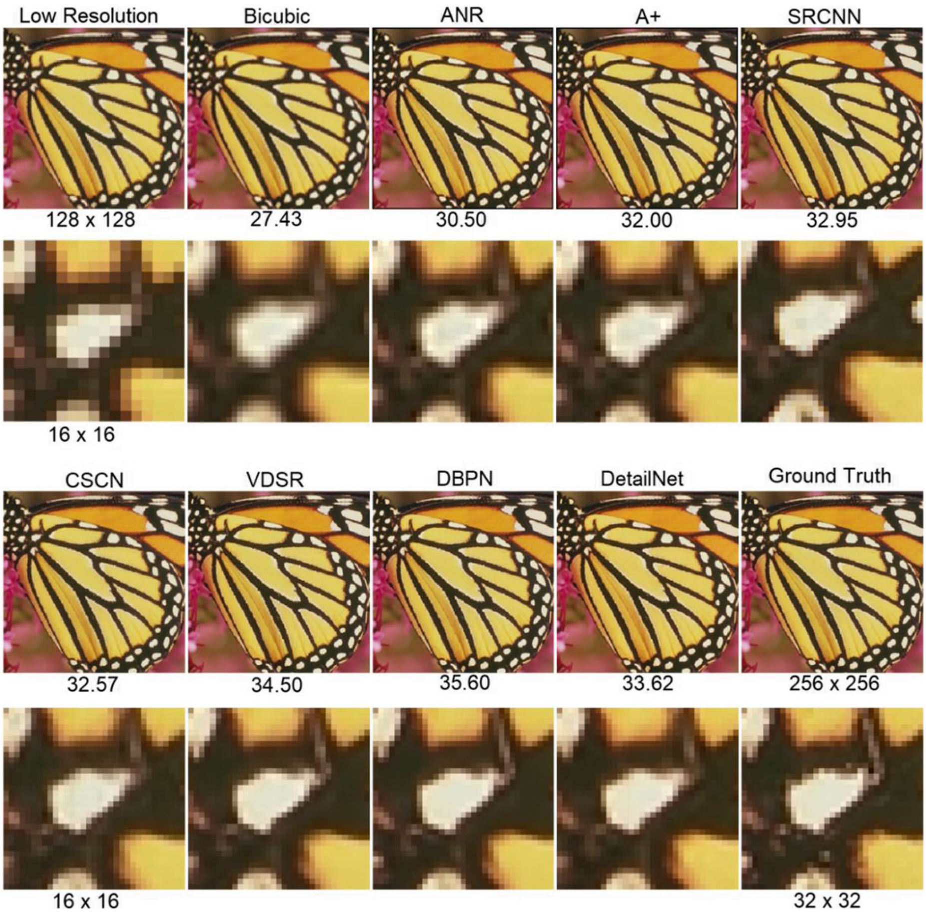

Fig. 6 Super-Resolution results on ×2 upscale factor

Fig. 7 Super-Resolution results on ×3 upscale factor

5 Conclusion and future work

There are many challenges in solving the Super-Resolution problem. In this research, providing a small structure network has been considered to prevent the problems of training a deep network and to have a low execution time. So, the goal of this research is to provide a fast speed Super-Resolution model that could produce high-quality images.

The DetailNet method provides greater PSNR results than some methods, while lower than some others. The results indicate that despite the lower PSNR results of the DetailNet than those of VDSR and DBPN methods, the images are not significantly different in their appearance. Regarding the visual quality, the images generated by these two methods are very close to the results of DetailNet. The proposed method provides sufficient and precise details and can produce even better details in comparison with those methods with higher PSNR results. Similar to previous studies, the evaluation of the results of this study also shows that using the PSNR, which is based on pixel differences,

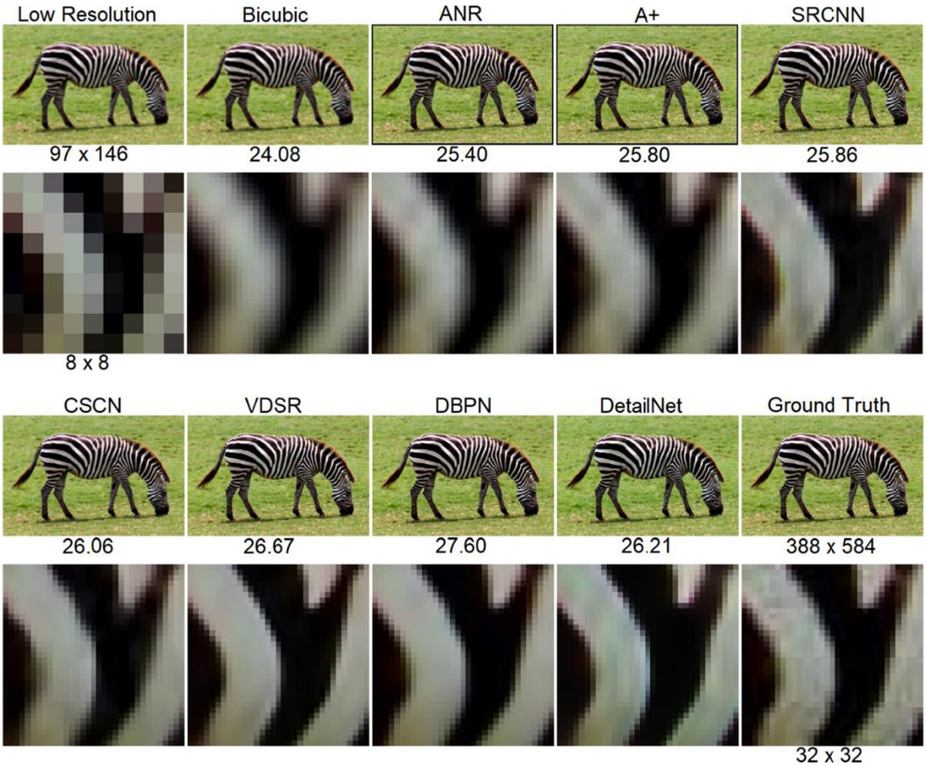

Fig. 8 Super-Resolution results on ×4 upscale factor

is not sufficient to evaluate the quality of Super-Resolution models. Therefore, besides evaluating the results through quantitative criteria, the visual quality of the results should be considered too.

The DetailNet network has a strong capability to produce image details and due to its relatively small structure does not face the problems of training a deep convolutional neural network. The proposed method is a fast speed model capable of producing high-quality images in such a way that its visual quality has an insignificant difference with methods providing higher PSNR. Also, the DetailNet network performs Super-Resolution independent of the input image size.

The DetailNet network has a general function. The proposed network performs the Super-Resolution process by adding details to the upscaled image. In this way, the first part of the work, which is increasing the image size, can be performed using any method, then the resulted image can be fed to the DetailNet to enhance its quality and add more texture details to it.

This method is a single-scale Super-Resolution, which needs to be trained separately on each upscale factor. The next goal to improve the power of the DetailNet network is to change the network structure to have a multi-scale functionality.

The DetailNet network can produce good quality Super-Resolution results for various colorful and grayscale images. Regarding the design of the DetailNet structure, the trained model improves the image details without changing the size of the input image, which increases the quality of the final image. Developing this structure as a general method can be very useful to improve the quality of images. The next step in improving the DetailNet performance is to train the network to perform Super-Resolution as well as other types of image enhancement methods such as deblurring, noise removal, and edge enhancement. In fact, changing the network to perform multiple functionalities will be considered in the future.

Developing the loss function of the DetailNet network will be investigated in the future to make the network more powerful. The goal is to focus on improving the quality of the images instead of improving them based on quantitative measures. The results of this research and other papers suggest that the use of loss functions such as MSE that are optimized to improve the quantitative measurements is not sufficient for the Super-Resolution problem. Accordingly, determining the proper function with the ability to restore the perceptual details of an image can be very useful in improving the quality of the final image.

Compliance with ethical standards

Conflict of interest There is no conflict of interest in this work.

References

1. Bevilacqua M, Roumy A, Guillemot C, Alberi-Morel ML (2012) Low-Complexity Single-Image Super-Resolution based on Nonnegative Neighbor Embedding. In: British Machine Vision Conference. BMVA Press, pp 135.1–135.10
2. Dahl R, Norouzi M, Shlens J (2017) Pixel Recursive Super Resolution. arXiv
3. Dong C, Loy CC, He K, Tang X (2014) Learning a deep convolutional network for image super-resolution. In: ECCV. Springer, pp 184–199
4. Gatys LA, Echer AS, Bethge M (2015) A neural algorithm of artistic style. arXiv Prepr arXiv150806576
5. Gatys LA, Echer AS, Bethge M (2015) Texture synthesis using convolutional neural networks. Advances in neural information processing systems. pp 262–270
6. Gregor K, LeCun Y (2010) Learning fast approximations of sparse coding. In: Proceedings of the 27th International Conference on International Conference on Machine Learning. Omnipress, pp 399–406
7. Gupta P, Srivastava P, Bhardwaj S, Bhateja V (2011) A modified psnr metric based on hvs for quality assessment of color images. In: IEEE International Conference on Communication and Industrial Application (ICCIA). IEEE, pp 1–4
8. Haris M, Shakhnarovich G, Ukita N (2018) Deep backprojection networks for super-resolution. In Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE, pp 1664–1673
9. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: In Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE, pp 770–778
10. Huang D, Liu H (2015) A short survey of image super resolution algorithms. J Comput Sci Technol Updates 2(2):19–29
11. Jia Y, Shelhamer E, Donahue J et al (2014) Caffe: Convolutional architecture for fast feature embedding. In: Proc. 22nd ACM Int. Conf. Multimedia. ACM pp 675–678
12. Kim J, Lee JK, Lee KM (2016) Deeply-recursive convolutional network for image super-resolution. IEEE Conf Comput Vis Pattern Recognit IEEE, pp 1637–1645
13. Kim J, Lee JK, Lee KM (2016) Accurate image super resolution using very deep convolutional networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, pp 1646–1654
14. Ledig C, Theis L, Huszar F, et al (2016) Photo-realistic single image super-resolution using a generative adversarial network. arXiv:160904802
15. Lu X, Guo Y, Liu N, Wan L, Fang T (2018) Non-convex joint bilateral guided depth upsampling. Multimed Tools Appl Springer 77(12):15521–15544
16. Lu X, Ma C, Ni B, Yang X, Reid I, Yang MH (2018) Deep regression tracking with shrinkage loss. In: Proceedings of the European Conference on Computer Vision (ECCV). Springer, pp 353–369
17. Martin D, Fowlkes C, Tal D, Malik J (2001) A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: ICCV IEEE pp 416–423
18. Nasrollahi K, Moeslund TB (2014) Super-resolution: a comprehensive survey. Machine vision and applications. Springer Berlin Heidelberg 25(6):1423–1468

19. Sajjadi MSM, Scholkopf B, Hirsch M (2017) Enhancenet: Single image super-resolution through automated texture synthesis. In: IEEE International Conference on Computer Vision (ICCV). IEEE, Venice, pp 4501–4510
20. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556
21. Sun L, Hays J (2017) Super-resolution Using Constrained Deep Texture Synthesis. CoRR, abs/170107604
22. Tai Y, Yang J, Liu X (2017) Image super-resolution via deep recursive residual network. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE pp 3147–3155
23. Timofte R, De Smet V, Van Gool L (2013) Anchored neighborhood regression for fast example-based super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision. IEEE, pp. 1920–1927
24. Timofte R, De Smet V, Van Gool L (2014) A+: Adjusted Anchored Neighborhood Regression for Fast Super-Resolution. In: Asian Conference on Computer Vision (ACCV). Springer, pp 1–5
25. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: From error visibility to structural similarity. In: IEEE Transactions on Image Processing. IEEE, pp 600–612
26. Wang L, Huang Z, Gong Y, Pan C (2017) Ensemble based deep networks for image super-resolution. Pattern Recogn 68:191–198
27. Wang Z, Liu D, Yang J, et al (2015) Deep Networks for Image Super-Resolution with Sparse Prior. In: Proceedings of the IEEE International Conference on Computer Vision. IEEE, pp 370–378
28. Yang J, Wright J, Huang TS, Ma Y (2010) Image super resolution via sparse representation. IEEE Trans Image Process IEEE 19:2861–2873
29. Zeyde R, Elad M, Protter M (2012) On Single Image Scale-Up using Sparse-Representations. In: International Conference on Curves and Surfaces. Springer, pp 711–730

Somayeh Barzegar received her B.S. degree in Computer Software Engineering from Golestan University in 2014, and her M.S. degree in Artificial Intelligence from Islamic Azad University, Science and Research Branch in 2018. Her main research interests include deep learning methods, computer vision, and machine learning.

Arash Sharifi received the B.S. degree in computer hardware engineering from IAU South Tehran Branch, M.S. degree and Ph.D. degree in artificial intelligence from IAU science and research branch, in 2007 and 2012 respectively. He is currently head of computer engineering department of SRBIAU. His current research interests include image processing, machine learning and deep learning.

Mohammad Manthouri received his B.Sc. in control engineering from Shiraz University, Shiraz, Iran, in 2005. He received the M.Sc. and Ph.D degrees in control engineering from K. N. Toosi University of Technology, Tehran, Iran, in 2008 and 2015, respectively. He is currently an assistant professor at the Electrical and Electronic Engineering Department, Shahed University, Tehran, Iran. His research interests include, fuzzy control, adaptive control systems, neural networks, deep learning and pattern recognition.