Image and Vision Computing 88 (2019) 9–18

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Image super resolution by dilated dense progressive network☆

Pourya Shamsolmoali, Masoumeh zareapoor, Junhao Zhang, Jie Yang *

School of Electronic, Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China

ARTICLE INFO

Article history:
Received 7 March 2019
Accepted 27 March 2019
Available online 25 April 2019

Keywords:
Image supper resolution
Dense network
Dilated convolution

ABSTRACT

Image super-resolution (SR) is an interesting topic in computer vision. However, it remains challenging to achieve high-resolution image from the corresponding low-resolution version due to inherent variability, high dimensionality, and small ground targets images. In this paper, a new model based on dilated convolutional neural network is proposed to improve the image resolution. Recently, deep learning methods have led to significant improvements and completely outpace other models. However, these methods have not fully exploited all the features of the original low-resolution image, because of complex imaging conditions and the degradation process. To address this issue, we proposed an effective model based on dilated dense network operations to accelerate deep networks for image SR, which support the exponential growth of the receptive field parallel by increasing the filter size. In particular, residual network and skip connections are used for deep recovery. The experimental evaluations on several datasets prove the efficiency and stability of the proposed model. The proposed model not only achieves state-of-the-art performance but also has more efficient computation.

1. Introduction

High-resolution (HR) images offer more information in comparison with low-resolution (LR) images. These details are essential in various applications, e.g. remote sensing, medical imaging, and video surveillance. Image super-resolution (SR) refers to algorithms that predict high-frequency information from one or more LR image(s) to produce more detailed information image [1]. Recently, SR has been used to deal with image resolution enhancement in several applications such as face recognition [2], remote sensing imaging [3], and video surveillance [4]. Especially, single image super-resolution (SISR) approaches have achieved impressive results by learning and mapping from LR to HR images by using up_sampling function in convolution neural networks (CNN).

Current SISR methods adopt one of the following approaches. The first approach, upscale the LR image with a one of an interpolation method, for example, bicubic at the first step and then use a learning model to deblur it [5–7]. The second approach uses upscaling after the learning process, generally using a sub-pixel convolution layer [8] or rearranged convolution layer to improve the HR result [9–11].

Dong et al. [15] termed as SRCNN. They used only three convolutional layers in their network structure. A couple of years later, they [19] extended their work by increasing the number of filters and the size of filters with a fixed depth of CNN. These two experiments proved that deeper models not only hard to train but even failed to improve the performance. Since then, many works have appeared that focusing on deep networks with a combination of other strategies to improve the resolution of the image. However, their observations convey that deep networks which stand-alone for image super-resolution tend to degrade the due to two reasons, firstly, when there are several convolution layers, there is a high probability to lose the image details, and as the details decreased the results do not own full satisfaction quality (vanishing gradient problem). The second problem refers to the computation cost and the optimization process that leads to lengthy training time. Practically, when any models contain more parameters, the network faces difficulty while training [12–14]. Based on these observations, in this paper, we address the problem of image super-resolution, by combining the dilated dense convolution network, with the residual network, and we show that if we adopt residual learning and skip connections in each layer of the network, it allows the proposed model to extract deep features. In addition, skip connections ease the gradient vanishing problems in the deep networks. In [16], they compared different networks with several depths for SR performance and indicated that wider and deeper networks have better performance because of high nonlinearities and wide receptive field. Additionally, it has been proved that the size of receptive filed has a more significant effect in SR performance as compared to the depth of the network. High receptive filed provides more contextual information which affects the reconstruction result. Hence, we propose a dilated Densenet for the SR. In this model,

☆ This paper has been recommended for acceptance by S. Todorovic.

* Corresponding author.
E-mail address: jieyang@sjtu.edu.cn (J. Yang).

https://doi.org/10.1016/j.imavis.2019.03.006
0262-8856/

the resolution of the network's output increases by changing a subset of internal down-sampling layers with dilation [17,18] Compared to standard convolution, dilated convolution provide the exponential increase of the receptive field with the same filter size. As a result, shallower networks with fewer parameters are able to achieve the same size receptive field like very deep networks. By using dilated convolution in Densenet and adopting proper skip connections, the proposed method can outperform state-of-the-art approaches.

The main contributions of this paper are as follows:

- We introduce a dilated dense convolution network for SR.
- We prove the receptive filed is one of the major factors in the SR task. We showed the networks with the same receptive field size but with different depths produce similar results.

The rest of the paper is organized as follow. In Section 2 we present some SR-related works. Section 3 describes the details of dilated convolution network. Section 4 shows the experimental results and the performance of the proposed model. Finally, Section 5 concludes the paper.

2. Related works

In this section, we present a brief description of the existing models for SR and background concepts, which are helpful for understanding the proposed model. CNN is successfully applied in a wide range of computer vision areas, such as classification, recognition, detection, and SR. Therefore, we have special emphases on the most recent prominent works based on deep learning in image super-resolution. Dong et al. [15,19] proposed the deep convolutional neural network for the image SR by using 2 to 4 CNN layers. They proved the deep models are not suitable for image SR, and they suggested using a CNN with a larger filter size is better than the deep layers. On the other hand, Kim et al. [20] proposed a very deep CNN called (VDSR) which contains 20 layers with some long connections. The proposed model outpaces SRCNN [15]. Deeply recursive convolutional network proposed by Kim et al. [21] proposed a deep recursive convolutional network that has 16, and 20 convolution layers with the share CNN weight, to minimize the number of parameters and the training problems of deep CNN. The other CNN based Image SR model called deep Residual Encoder-Decoder Networks (RED-CNN) [22]. RED is based on residual learning and contains symmetric convolution and deconvolution layers. In addition, the authors used a different range of connections to connect every two or three layers to increase the capacity of the network while training [23,24].

Romano et al. [25] proposed a shallow model with a fast learning technique. Their model called "Rapid and Accurate Image SR" which can classify the input image based on their patches. Mei et al. [26] used 3D convolution to exploit both the spatial context of neighboring pixels and spectral correlation of neighboring bands. Galliani et al. [27] and Zhu et al. [28] proposed a novel method for single hyperspectral image SR. Jiang et al. [3] proposed a progressive enhanced CNN model for remote sensing image SR. The authors proposed an evolution unit to obtain operational information from the base output. Later, the extracted information and the low-level feature maps are transferred to the enhanced deep CNN for the feature expression. Huang et al. [17] proposed an SR model based on dilated CNN.

Akhtar et al. [29] proposed an SR approach for hyperspectral images using non-parametric Bayesian sparse representation. The model first gathers probability distributions for the material spectra in the scene and their sizes. The distributions are then used to calculate sparse codes of the high-resolution image. Cui et al. [30] used a cascade of several combined local auto-encoders that progressively up-scales the low-resolution image. In each layer of the cascade, non-local match search is checked first; next, the input patches are fed into a combined local auto-encoder for further improvement. Next, Chen et al. [45] proposed a consecutive gradient constrained regression-based single image SR named "SGCRSR", which delivers an operational way to combine the conventional learning and reconstruction methods. Zhang et al. [46] developed a learning method based on sparse representation and an adaptive sample collection scheme to acquire mixed samples. They also proposed an adaptive mixed samples ridge regression model to efficiently learn from the matching information. Xiao et al. [47] developed a detail-enhancement and SR algorithm based on detail synthesis. The proposed algorithm recovers the facet or line phenomenon on edges and areas that have high texture. Wang et al. [31] introduced a deep joint SR model to exploit both external and self-similarities for reconstruction, in which a fixed de-noising convolutional auto-encoder first pertains on some big dataset, and later it is fine-tuned with multi-scale input data. Ledig et al. [32] also made a contribution to SR by having a generative adversarial network (GAN) to produce high-resolution images, and they presented promising output in textured images, additionally, the technique falls behind the state-of-the-art approaches in terms of the objective metrics.

A standard progressive reconstruction model is described by Lai et al. [33]. In this approach, the upscaling layer follows the principle of Laplacian pyramids, i.e. each level learns to predict a residual which should describe the difference between the upscale of the previous layer and the desired result. Meanwhile, the loss functions are calculated at a different scale, this provides a form of intermediate supervision. Lai et al. [34] upgraded their method with wider recursive architecture and multi-scale training. But, still, there is a considerable gap between the top-performing approaches in terms of PSNR.

3. Proposed methods

This section presents the details of the proposed model.

3.1. Convolutional neural networks

Currently, CNNs are one of the best neural network models that learn a hierarchy of complex features by sequential convolution, (max or average) pooling and non-linear activation function [35]. The first CNN designed for image recognition and classification. However, currently, CNNs are used in image SR, semantic segmentation and different computer vision tasks. A method follows a sliding-window approach where regions defined by the window are processed individually. This system has two main weaknesses: a decrease of resolution accuracy and low efficiency. A substitute methodology, known as fully CNNs (FCNs) [36], eases these boundaries by considering the network as a non-linear convolution layer which has an end-to-end training process. The main strength of FCNs compared to normal CNNs is that they can be used in any type and size of images. Besides, FCNs can escape redundant convolution operations, and make the computation efficient.

The networks explored in this paper are based on the DenseNet architecture, which has outstanding performance in various SR tasks [7,11,13]. This network consists of direct connections from each layer to all the subsequent layers. Fig. 1 shows the outline of DenseNet architecture. In this model, the lth layer obtains all the feature-maps of the earlier layers, $x_0, \ldots, x_{l-1}$, as input:

$$x_l = H_l([x_0, x_1, \ldots, x_{l-1}]), \tag{1}$$

where $[x_0, x_1, \ldots, x_{l-1}]$ denotes to the concatenation of the feature-maps generated in layers $0, \ldots, l-1$. These connections allow integrating high-level features and fine pixel-wise details simultaneously. SR requires certain information regarding the global context. Standard convolutions have difficulty integrating global context, even when pooling actions are sequentially added into the network. For example, in the standard DenseNet model, the receptive field spanned by the deepest layer is only 112×112 pixels. This means that the context of the

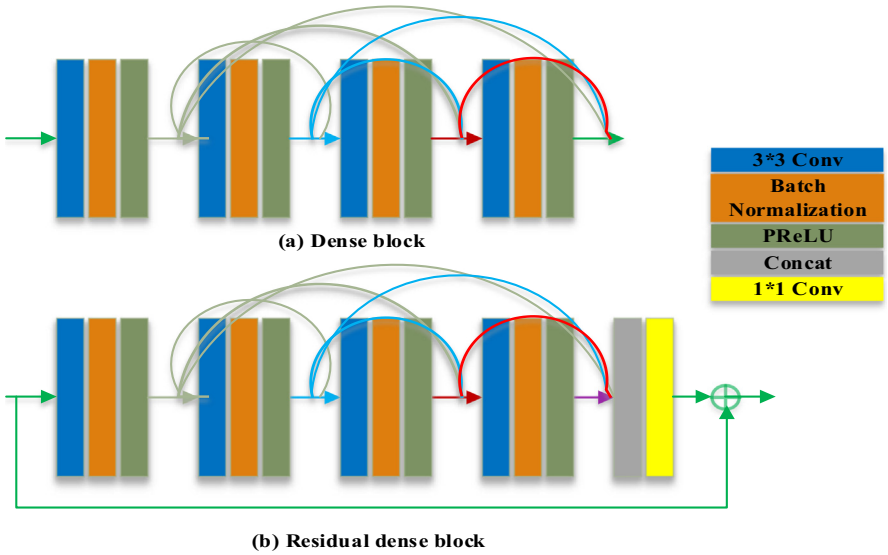

Fig. 1. The network architecture of standard DenseNet (top) and the proposed DenseNet (bottom).

whole image is not fully considered in the deep architecture to generate its final prediction. A direct solution for increasing the receptive field is to place additional pooling tasks in the network. Though, this approach usually decreases the performance since relevant information is lost in the new down-sampling processes.

3.2. Dilated convolutions

Firstly, in this section, we discuss the details of the dilated convolution network. Secondly, we illustrate the significance of the progressive and receptive field in CNN based image SR. To start the construction of the proposed model we followed the set of network architectures presented by Yu et al. [6]. Any of these architectures contains five sets of convolutional layers. The earliest layers in each set perform down-sampling by striding. If each set of layers denoted by G^l, for $l = 1, \ldots, 5$. Denote the *ith* layer in group l by G_i^l.

To simply illustrate the idea of the proposed model, each layer contains a single feature map: the extension to various feature maps is straightforward. Let f_i^l be the filter size associated with layer G_i^l. In the standard model, the result of G_i^l is:

$$\left(G_i^l * f_i^l\right)(p) = \sum_{a+b=p} G_i^l(a) f_i^l(b), \tag{2}$$

while the p domain is the feature map in G_i^l that is followed by a nonlinear function. A best approach to improve the resolution in the network

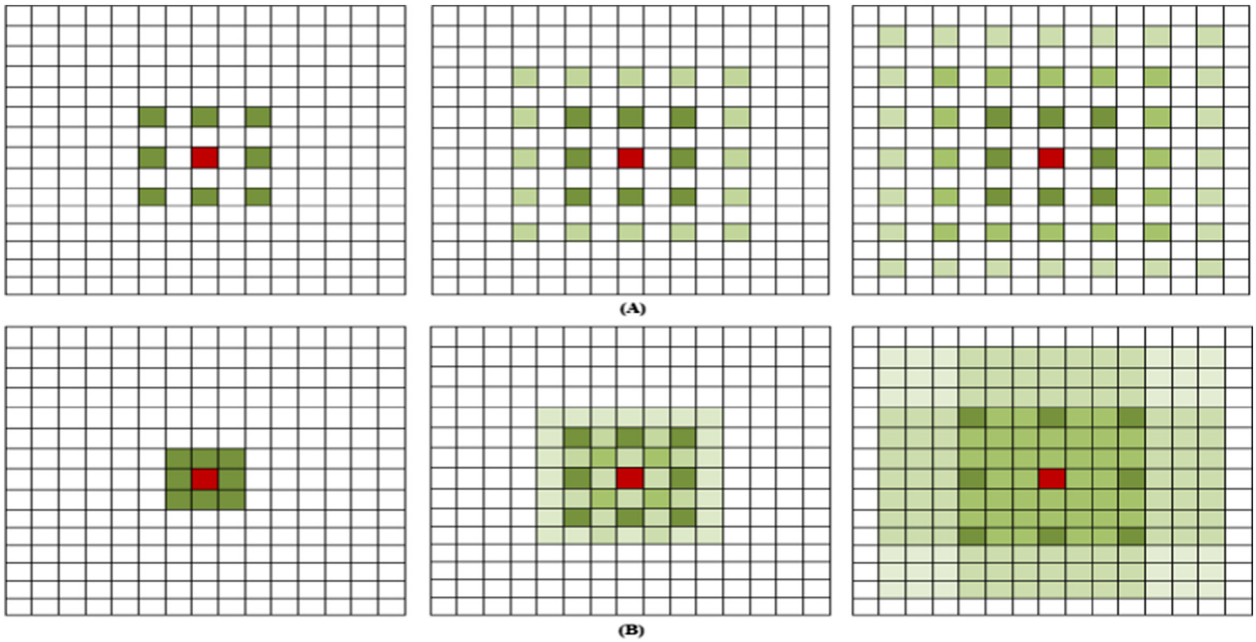

Fig. 2. The details of gridding artifacts. The processes between layers are both dilated convolutions with 3×3 kernel size. We mark their proper receptive fields in layer $i - 1$ and $i - 2$ using the same colors. Their receptive fields are fully separate sets of units. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

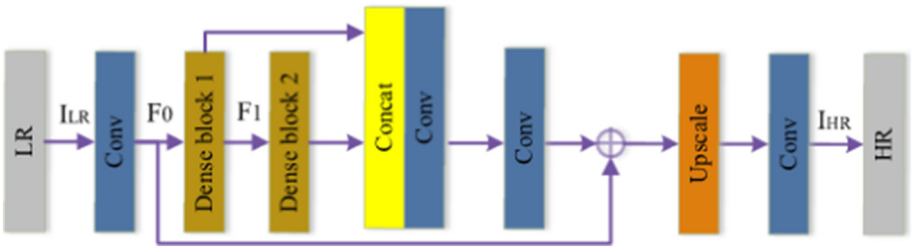

Fig. 3. The framework of the proposed model. Convolutional blocks contain progressive dilated rates. In each convolutional block, the dilated ratios of ×1, ×2, and ×4 are included.

would be to remove down-sampling (striding) from some of the interior layers. This significantly raises the resolution but has a negative effect which reduces the receptive field in subsequent layers. Therefore, eliminating striding such that the resolution of the output layer is improved by a factor of 2 meanwhile reduces the receptive field of the respective output unit by a 2 factor. This extremely decreases the amount of context that can be useful for the prediction. Contextual information has a key role in disambiguating local cues [17], such reduction in the receptive field is an undesirable price to pay for high resolution. This is the main reason for using dilated convolutions [18] for increasing the receptive field in the higher layers; the best solution is the subsampling layers removing. The main effect is the units in the corresponding dilated layers have an equal receptive field as corresponding units in the original model. This work is mainly focuses on the two final sets of convolutional layers: $G^4$and G^5. In the original DenseNet, the first layer in each set (G^4 and G^5) is strided: the convolution is weighed at the even rows and columns, which decreases the resolution result of these layers by 2 factor in all dimensions.

The idea is to add "holes" (i.e., zeros) among the pixels in convolutional kernels to increase the image resolution of intermediate feature maps, to enable dense feature extraction in DenseNet with an enlarged field of convolutional kernels the details presented in Fig. 2. Consider a convolutional kernel K^l in layer l which has a size of $k \times k$. The receptive field of K^l, also named as actual kernel size, can be defined as

$$RFRF_{K_l} = k + (k-1) \times (D_k - 1) \tag{3}$$

while D_k signifies the dilation rate of kernel K_l, stating the total number of zeros to be placed among pixels. Consider that, in original convolutions, D_k and stride are equal to *1*.

3.3. Architecture details

In this paper, we propose to use progressive dilated convolutions in DenseNet architecture. For evaluating the performance of the proposed model on image SR, several models are considered. First, we used the standard DenseNet implementation and an altered version which is serving as baselines (Section 3.3.1). Second, the first regular convolution layer of the individual block in the baseline model is changed by a dilated convolution (Section 3.3.2). Finally, in the proposed model, the whole normal convolution block in the baseline is changed by the proposed progressive dilated DenseNet block (Section 3.3.3). Additionally, the proposed method is compared with the state-of-the-art SR. Particularly; we investigated related deep convolutional neural networks that include DenseNet or dilated convolutions, including [7,11, 13,17].

3.3.1. DenseNet baselines

We used the standard version of DenseNet as described in [13] as a baseline network, which will be indicated as DenseNet-Original (Fig. 1, a). Moreover, we added a key modification on the standard version as another baseline and indicated it DenseNet-Baseline. To reuse the features and get a more compact representation of learned features, residual connections (Fig. 1b) is presented between the layers. The main aim of these connections is to reuse the information in the further blocks as the input without modification, thus encouraging the path through non-linearity to learn a residual representation of the input data [16]. Additionally, in the proposed model each convolution layer performs batch normalization to accelerate convergence in the parameter learning process and make the model more robust in testing. Also, the activation function in the proposed network sets to parametric rectifier linear units (PReLU) [38].

3.3.2. Dilated DenseNet

The proposed dilated model follows the standard architecture of Densenet [13] but introduces a context module at each block of dense connected layers. The context module holds a dilated convolution layers block to systematically combine the multi-scale contextual information. A natural problem when using dilated convolutions is gridding [39] (Fig. 2, top). As zeros are inserted among pixels in a dilated convolutional kernel, the receptive field applied by this kernel only that covers a range with some sort of checkerboard patterns, sampling few positions with non-zero values. These results significantly have effects on the learning process. If dilation rate D rises, this concern becomes even worse, as the convolution kernel due to sparseness not able to get any local information. To ease this issue, we use the approach proposed by [40], while dilated convolutions are replaced with normal convolutions and dilation rates are progressively enlarged. Hence, the dilation rate D in the convolutional blocks of this model are set to 1, 2 and 4, respectively from shallow to deep layers.

3.3.3. Progressive dilated DenseNet blocks

Rather than gradual layer by layer increasing of the dilation by factor D, we propose to increase it in each context module. As the features learned at respective block are able to get multi-scale level information. Consequently, at every block, the dilation rate D will be gradually set to 1, 2, and 4. Therefore, we do not allocate large D values which may span broader regions, while keeping the same network receptive field. Fig. 3 shows the overall architecture of the proposed model.

The proposed model consists of three parts: feature extraction net (FE), residual dense blocks (RDBs), and end with the up-sampling net (UP). The input and output of the proposed model represented as

Table 1
Network details of the proposed model progressive dilated model. The model can achieve 21 × 21 receptive field.

Layers	Kernel	Dilation	Receptive filed	Output channels
Conv	3 × 3	1	3 × 3	32
Dense block	3 × 3	2	5 × 5	32
Dense block	3 × 3	2	9 × 9	32
Conv	1 × 1	4	13 × 13	32
Conv	3 × 3	2	17 × 17	32
Conv	3 × 3	1	21 × 21	3

Table 2
Different network settings performance. In these networks, the scale factor is set to ×2 and all have 41 × 41 receptive fields.

	VDSR 12	VDSR 12 dilated	DenseNet-original	DenseNet-baseline	Proposed model
First part	Conv(3,64,1,D1)	Conv(3,64,1,D1)	Conv(3,32,1,D1)	Conv(3,32,1,D1)	Conv(3,32,1,D1)
Mid part	Conv(3,64,64,D1)- 8Conv (5,64,64,D1) -Conv (3,64,64,D1)	Conv(3,64,64,D1)- 8Conv (3,64,64,D2) -Conv (3,64,64,D1)	Conv(3,32,32,D1)- 3 dB(3,32,32,D1)- Conv(1,32,32,D1) -Conv(3,32,32,D1)	Conv(3,32,32,D1)- 3 dB(3,32,32,D1)- Conv(1,32,32,D1) -Conv(3,32,32,D1)	2 dB(3,32,32,D2)- Conv (1,32,32,D4) -Conv (3,32,32,D2)
Last part	Conv(3,1,64,D1)	Conv(3,1,64,D1)	Conv(3,32,32,D1)	Conv(3,32,32,D1)	Conv(3,32,32,D1)
PSNR (Set5)	37.42 dB	37.46 dB	37.45 dB	37.47 dB	37.49 dB
PSNR (Set14)	32.80 dB	32.87 dB	32.85 dB	32.87 dB	32.88 dB
PSNR (BSD100)	31.63 dB	31.72 dB	31.70 dB	31.73 dB	31.75 dB
PSNR(Urban100)	32.86 dB	32.91 dB	32.87 dB	32.90 dB	32.92 dB
Parameters	895,873	370,368	974,736	585,124	367,286
Speedup	1/23.6×	2.3×	1/04.3×	1/04.5×	2.1×

I_{LR} and I_{SR}. Two Convolution layers implemented to extract shallow features. The first Conv layer extracts feature F_{-1} from the LR input.

$$F_{-1} = H_{FE1}(I_{LR}), \tag{4}$$

while $H_{FE1}(\cdot)$ indicates the convolution process. F_{-1} is then used for additional feature extraction and residual learning. Moreover, we can have:

$$F_0 = H_{FE2}(F_{-1}), \tag{5}$$

where $H_{FE2}(\cdot)$ presents the convolution process of the second feature extraction layer to feed the input to the residual dense blocks. Assuming the numbers of residual dense blocks are D, the result F_d of the *dth* RDB can be achieved by:

$$F_d = H_{RDB,d}(F_{d-1}), \tag{6}$$

Eq. (6) can be written as:

$$F_d = H_{RDB,d}\left(H_{RDB,d-1}\left(\ldots\left(H_{RDB,1}(F_0)\right)\ldots\right)\right), \tag{7}$$

where the $H_{RDB,d}$ represents the *dth* RDB task. $H_{RDB,d}$ is the combined function for example convolution and activation function (rectified linear units (ReLU)) [37,38]. As F_d is extracted by the *dth* RDB fully operating convolution layers inside the block, we can consider F_d as the local feature. After extracting deep features by using the set of RDBs, we applied global residual learning (GRL) to reuse the intact features. After extracting all the features, we have an up-sampling layer (UPNet) that adopted from [16], the UPNet layer followed by a single Convolution layer. The final high-resolution result of the proposed network can be obtained by:

$$I_{SR} = H_{RDN}(I_{LR}), \tag{8}$$

where H_{RDN} signifies the function of proposed DenseNet.

The dilation rate D is applied at the end of the convolution layers. In the rest of the network, same color blocks corresponding to the same dilation rate. Table 1 presents the details of the proposed model.

In the proposed model, the size of all convolutional layers are set to 3 × 3 except the one that used for local and global feature concatenation, whose kernel size is 1 × 1. The convolutional layers that have 3 × 3 kernel size, zeros padding added to all sides of the input to hold the input size. Shallow feature extraction layers and local and global feature have $G_0 = 32$ filters. The other layers in each RDB have G filters and are used PReLU [38] as the activation. ESPCNN [4] has been used to upscale the resolution. The last Convolution layer has 3 output channels, as the outputs are color HR images. Nonetheless, the network has the ability to process gray images.

4. Experiments

In this section the details of datasets, implementations and experimental results presented.

4.1. Datasets

Training Dataset; To train the proposed model, 200 images from Berkeley Segmentation Dataset [41] have been used. Data augmentation technique is used to getting more training data. Moreover, to train such a deep network large number of training images required, therefore, DIV2K [5] also added to the training set, which has 800 high-resolution images. Scale augmentation which proposed by [11] is also adopted for the training process. Hence, the proposed model can handle multiple scales of SR tasks. For evaluation, the benchmark datasets Set5 [42], Set14 [43], BSD100 [41], Urban100 [44], and some real-world dataset are used. As it is generally applied in SISR, all evaluations are conducted on the luminance channel.

Table 3
Comparison of PSNR and SSIM on four test datasets for different SR methods. The highest results are bolded.

Dataset	Scale	Bicubic PSNR/SSIM	SRCNN [15] PSNR/SSIM	DRRN [23] PSNR/SSIM	RDN [11] PSNR/SSIM	DCNSR [17] PSNR/SSIM	Proposed model PSNR/SSIM
Set5	×2	33.66/0.9299	36.66/0.9542	37.74/0.9591	37.46/0.9585	38.30/0.9616	**38.32/0.9617**
	×3	30.39/0.8682	32.75/0.9090	34.03/0.9244	33.74/0.9219	34.78/0.9300	**34.81/0.9302**
	×4	28.42/0.8104	30.48/0.8628	31.68/0.8888	31.37/0.8831	32.61/0.9003	**32.63/0.9006**
Set14	×2	30.24/0.8688	32.45/0.9067	33.23/0.9136	32.91/0.9116	34.10/0.9218	**34.13/0.9221**
	×3	27.55/0.7742	29.30/0.8215	29.96/0.8349	29.76/0.8312	**30.67/0.8482**	**30.67**/0.8481
	×4	26.00/0.7027	27.50/0.7513	28.21/0.7721	27.99/0.7661	**28.92/0.7893**	28.91/0.7892
B100	×2	29.56/0.8431	31.36/0.8879	32.05/0.8973	31.81/0.8947	32.40/0.9022	**32.42/0.9024**
	×3	27.21/0.7385	28.41/0.7863	28.95/0.8004	28.80/0.7972	29.33/0.8105	**29.34/0.8108**
	×4	25.96/0.6675	26.90/0.7101	27.38/0.7284	27.26/0.7241	27.80/0.7434	**27.82/0.7435**
Urban100	×2	26.88/0.8403	29.50/0.8946	31.23/0.9188	31.25/0.8852	33.09/0.9368	**33.12/0.9370**
	×3	24.46/0.7349	26.24/0.7989	27.53/0.8378	27.98/0.7729	29.00/0.8683	**29.05/0.8685**
	×4	23.14/0.6577	24.52/0.7221	25.44/0.7638	26.57/0.6997	26.82/0.8069	**26.85/0.8071**

4.2. Implementation settings

4.2.1. Training setting

The details of the parameters setting and the network structure are illustrated in Table 2. In each training batch, we randomly as an input select 20 LR patches with a size of 32 × 32. Then apply augmentation on the patches by flipping vertically or horizontally and 45-degree rotation. 500 iterations of back-propagation create an epoch. To implement the proposed model we used Keras 2.1.2, the deep learning open-source library and used TensorFlow 1.3.0 GPU as the backend deep learning engine. Python 3.6 is used for all the implementations. All the implementations of the network are conducted on a workstation equipped with an Intel i7-6850K CPU with a 64 GB Ram and an NVIDIA GTX Geforce 1080 Ti GPU and the operating system is Ubuntu 16.04. The learning rate is initialized to 0.01 for all layers and divided by 10 every 20 epochs.

In Table 2, the architectures and performances of 5 networks are presented. All these networks more or less achieve similar PSNR result. It confirms our assumption that different networks with equal receptive filed size will generate similar results. While having the same depths, the networks that dilated operation implemented (2) and (5) have better performance, fewer parameters and faster speed than (1), (3) and (4). This is due to the downside of a large filter size. Bigger filter size can get a large receptive field, they also take more noises into the learning procedure. Furthermore, in 5 × 5 and 7 × 7 kernels because the values in these big kernels are highly correlated to the learning result is much redundant as compared to 3 × 3 filter size. It also has been observed, the speed of (2) and (5) are much higher than the rest of the network. To conclude, the receptive field is a key aspect in SR task and dilated convolution is one of the best methods to achieve a large receptive field.

Table 3 demonstrations quantitative comparisons for ×2, ×3, and 7×4 SR. The proposed model performance is higher than all other approaches. This shows the effect of our proposed model in comparison with simple dense blocks in DCNSR [17]. Once compared with the other models, our network also has the best average performance on the most datasets. Especially, for the scaling factor ×2, the proposed model has the best performance on all datasets. While increasing the scale factor (e.g., ×3), our model would not have a similar advantage over DCNSR [17] on Set14 dataset. There are generally two reasons for this case. First, DCNSR utilizes multi-scale inputs as VDSR do equally [20]. Second, DCNSR uses bigger input patch size (64 vs. 32) for training. Most of the images in Set14 hold similar structures; training with the larger input patch size helps the deep network to catch more information by using the large receptive field.

In Fig. 4, we show visual comparisons on scale ×3. For 3 images from google earth dataset, it is observed that most of the compared approaches would generate noticeable artifacts and blurred edges. On

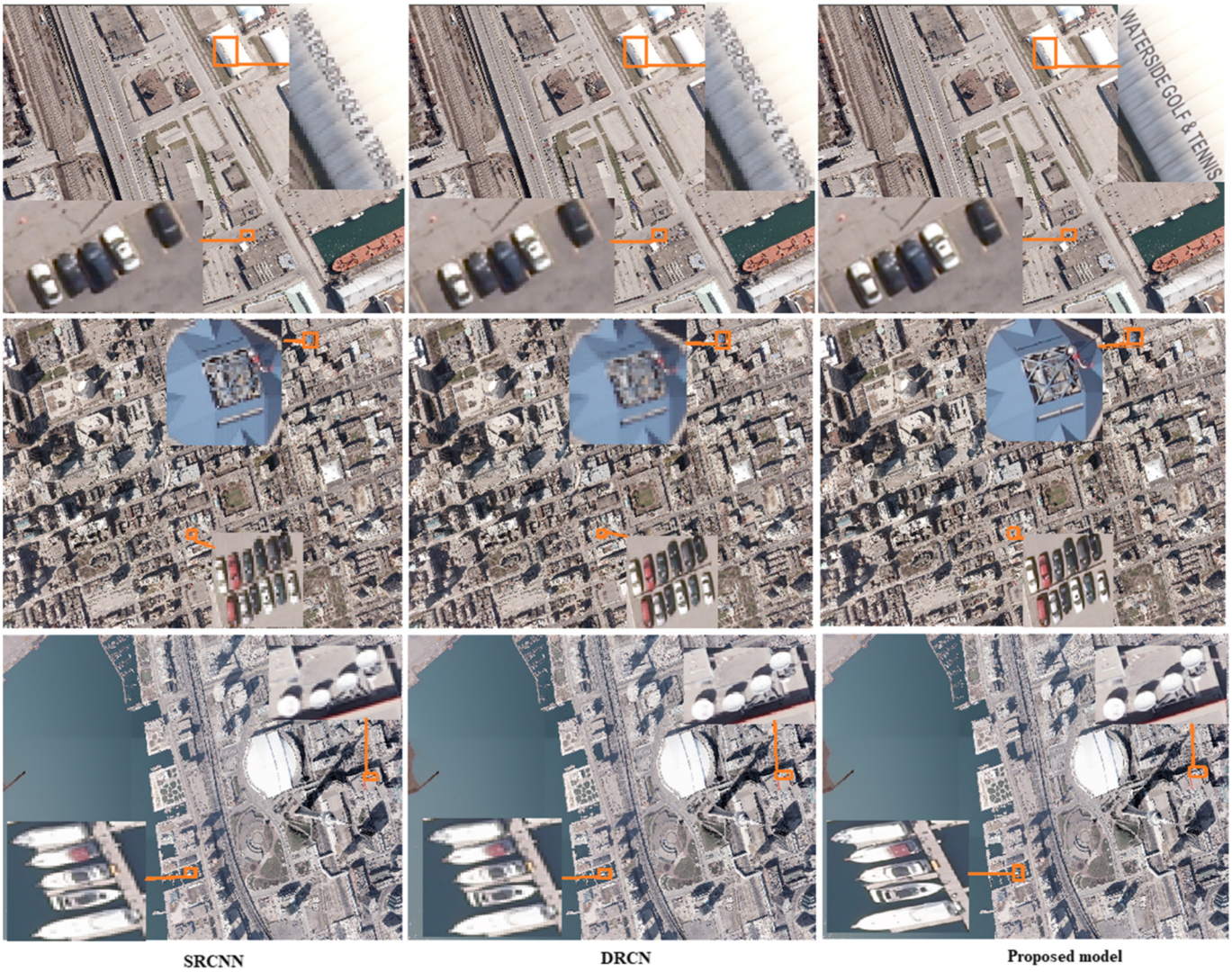

Fig. 4. Visual results with scaling factor ×3. The SR results are from google earth dataset.

the other hand, the proposed model can improve the resolution significantly and as Fig. 4 shows all the images have sharper and clearer edges. For the tiny text in the first row figures, none of the methods could recover it properly except the proposed model. This is mostly due to reusing the features through the dense blocks which tremendously improve the overall results.

In Fig. 5 we compared our proposed model with SRCNN [15], DRRN [23], RDN [11], and DCNSR [17]. Table 3 presents the average PSNR and SSIM results on the following 4 datasets Set5, Set14, B100, and Urban100 with scaling factor ×3. As the results show the proposed model has the best performs on the all datasets. The approaches using LR images as input to generate artifacts but unable to remove the blurring artifacts. However, our model eradicates the blurring artifacts and significantly recovers the sharp edges. This comparison shows that extracting ordered features from the LR input would ease the blurring artifacts. It also proves the efficiency of the proposed model. For the last image of Fig. 5, As the LR image is processed with some noise and missed details. It has been observed the other methods failed to recover the noised images [2,11,20]. However, our model not only efficiently handles the noisy image but also recovers more details. This comparison shows that our model has the ability to apply for the image denoising task as well.

Additionally, to more widely check the performance of our model we conduct SR experiments on three real-world images, with 300 × 300 and 300 × 244 pixels. In this experiment, as shown in Fig. 6, the performance of proposed model in three scale factor presented.

Fig. 7(a) presents the trade-offs between the performance (PSNR) and runtime on the Set5 dataset for 2× SR. The speed of the proposed model is faster than all the existing methods. Our dilated model

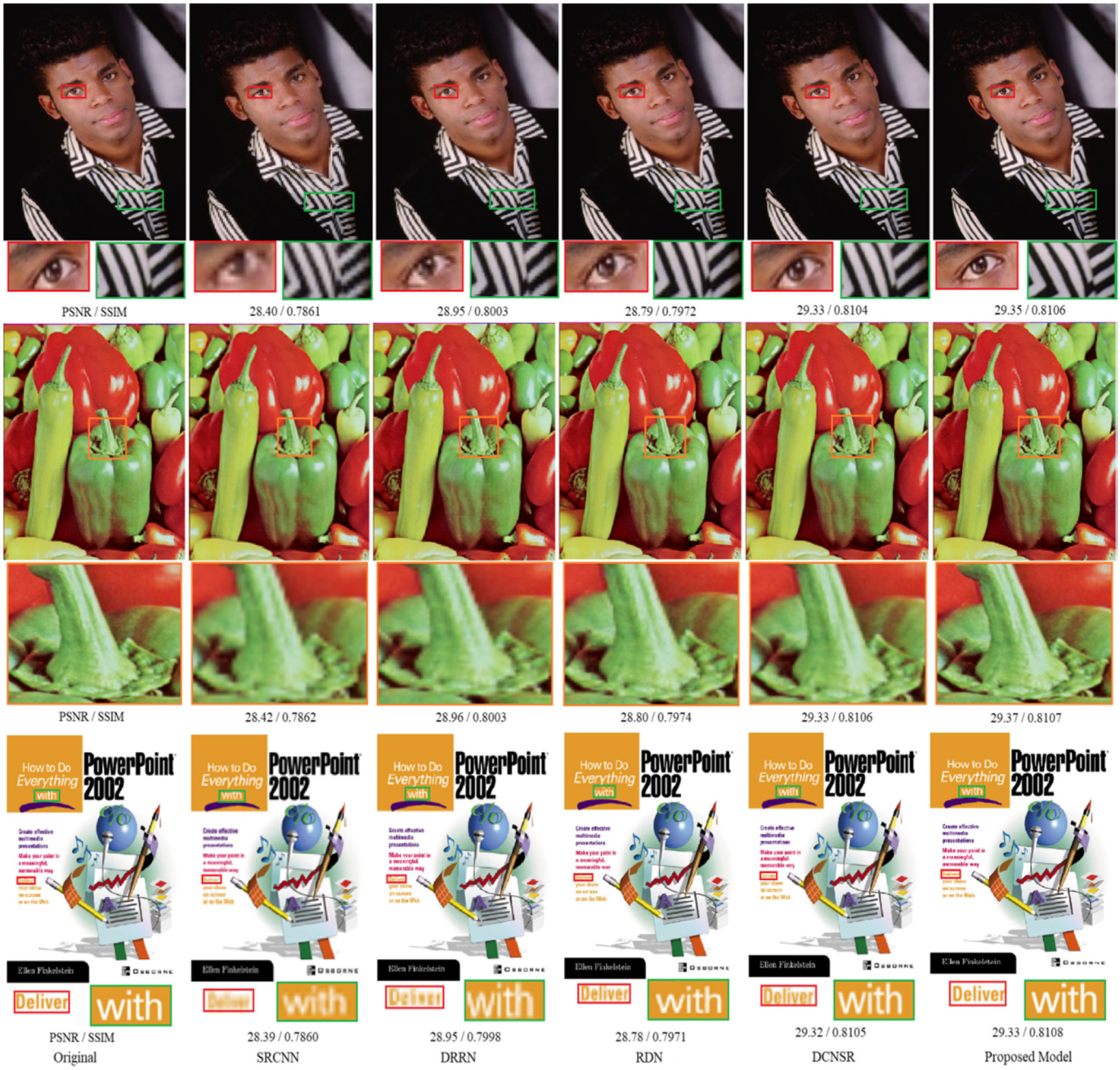

Fig. 5. Visual results with scaling factor ×3. The SR results are from B100 dataset.

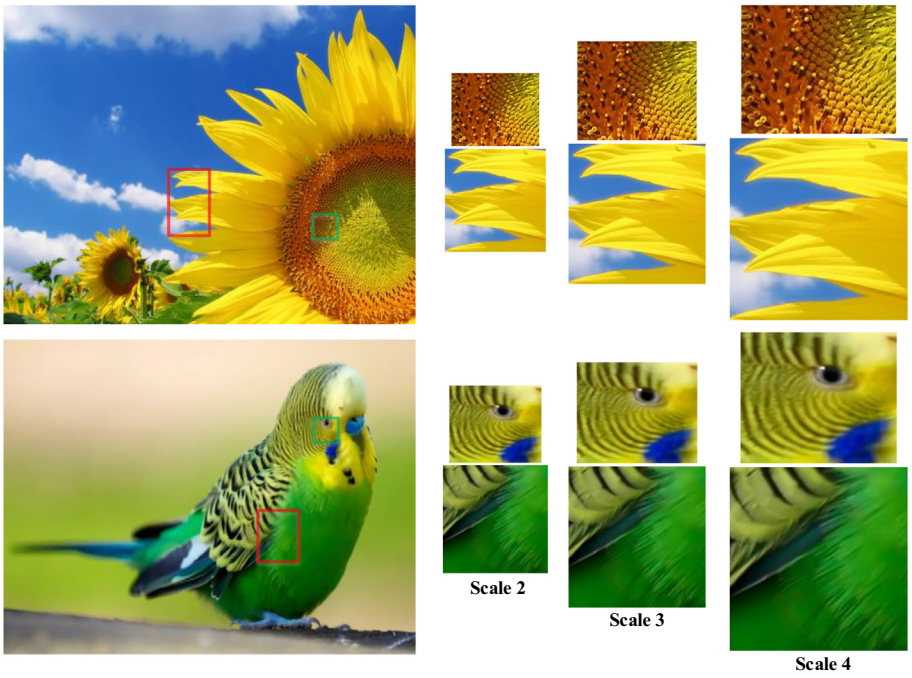

Fig. 6. Visual results with scaling factor ×2, ×3 and ×4. The SR results are from real-world images.

outperforms the existing approaches that used DenseNet and dilated technique, DCNSR [17] DRRN [23]. Additionally, we present the performance versus the number of parameters of CNN-based SR models in Fig. 7(b) By parameters sharing, the proposed dilated method has parameters about 64% less than the RDN [11], 56% less than the VDSR [20], 11% less than the DRRN [23], and 8% less than the DCNSR [17].

Furthermore, we have done a comparison between SR methods: SRCNN [15], DRRN [23], VDSR [20], RDN [11], DCNSR [17], and our proposed model with and without dilated structure. We use an LR image with 300 × 300 resolution, and implement 2×, 4× and 8× SR. Each method 15 times evaluated and the averaged runtime reported in Fig. 8. For the SRCNN and VDSR the runtime depends to the size of output images. However, the speed of our proposed model is mostly based on the input images size. As our model and DCNSR [17] used dilated convolution layers they have the best performance as compared to other approaches. The time complexity generally increases with respect to the wanted upsampling scales. Though, the processing time of our model still performs favorably better than other existing models.

5. Conclusions

We introduced the dilated convolution neural network to accelerate the speed of dense network for image SR. Initially; we proved the receptive field is a key factor in image SR. The networks with a similar receptive field but with different depths produce similar HR results. Secondly, we propose the dilated convolution network instead of standard convolution operation. Dilated convolution has a better performance for collecting a large receptive field. Based on our deep network, we designed three different networks settings and present the efficiency and effectiveness of the dilated convolution operation in both SR performance and speed. By having full access to the local and global features, the proposed model leads to a dense feature and deep supervision. We used same structure to handle several real-world

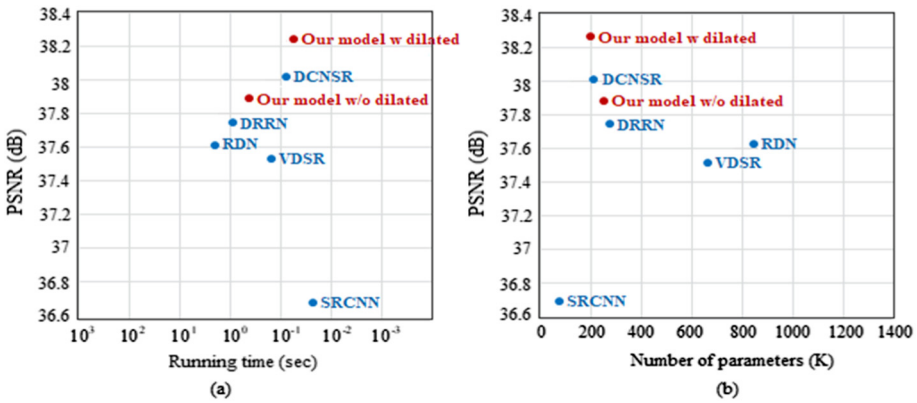

Fig. 7. The results are evaluated on the set5 dataset for 2× SR. (a) Performance versus runtime. (b) Performance versus a number of network parameters.

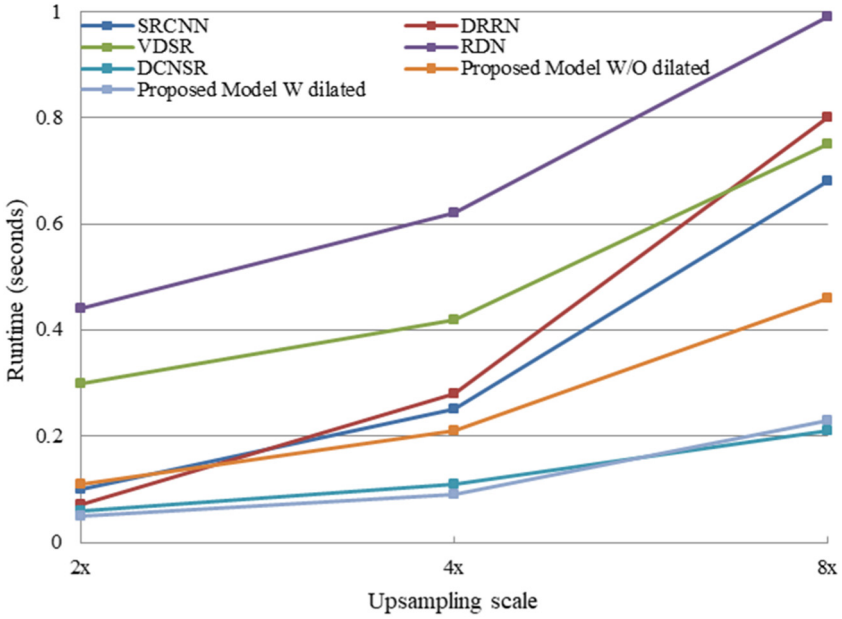

Fig. 8. The trade-off between runtime and upsampling scales. The size of input images is fixed to 300 × 300 and perform 2×, 4× and 8× SR with the SRCNN [15], DRRN [23], VDSR [20], RDN [11], DCNSR [17] and the proposed model.

datasets. Wide benchmark evaluations prove that our model can achieve superiority over state-of-the-art models.

Acknowledgment

This research is partly supported by NSFC, China (Nos: 61572315, 6151101179) and 973 Program, China (No. 2015CB856004).

References

[1] J. Jiang, X. Ma, C. Chen, T. Lu, Z. Wang, J. Ma, Single image superresolution via locally regularized anchored neighborhood regression and nonlocal means, IEEE Trans. Multimedia 19 (1) (2017) 15–26.
[2] C. Dong, C.C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks, IEEE Trans. Pattern Anal. Mach. Intell. 38 (2) (2016) 295–307.
[3] K. Jiang, Z. Wang, P. Yi, J. Jiang, A progressively enhanced network for video satellite imagery Superresolution, IEEE Signal Processing Lett. 25 (11) (2018) 1630–1634.
[4] W. Shi, J. Caballero, F. Huszar, J. Totz, A.P. Aitken, R. Bishop, D. Rueckert, Z. Wang, Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network, CVPR, 2016.
[5] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, L. Zhang, B. Lim, S. Son, H. Kim, S. Nah, K.M. Lee, Ntire 2017 challenge on single image super-resolution: methods and results, CVPRW, 2017.
[6] R. Timofte, R. Rothe, L. Van Gool, Seven ways to improve example-based single image super resolution, CVPR, 2016.
[7] H. Zhang, V.M. Patel, Density-aware single image deraining using a multi-stream dense network, CVPR, 2018.
[8] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi, Photo-realistic single image super-resolution using a generative adversarial network, CVPR, 2017.
[9] N. Akhtar, F. Shafait, A. Mian, Bayesian sparse representation for hyperspectral image super resolution, CVPR 2015, pp. 3631–3640.
[10] C. Dong, C.C. Loy, X. Tang, Accelerating the superresolution convolutional neural network, ECCV, 2016.
[11] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, Y. Fu, Residual dense network for image super-resolution, CVPR, 2018.
[12] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, Y. Fu, Image super-resolution using very deep residual channel attention networks, ECCV, 2018.
[13] T. Tong, G. Li, X. Liu, Q. Gao, Image super-resolution using dense skip connections, ICCV, 2017.
[14] W. Shi, J. Caballero, C. Ledig, X. Zhuang, W. Bai, K. Bhatia, A.M.S.M. de Marvao, T. Dawes, D. ORegan, D. Rueckert, Cardiac image super-resolution with global correspondence using multi-atlas patchmatch, MICCAI, 2013.
[15] C. Dong, C.C. Loy, K. He, X. Tang, Learning a deep convolutional network for image super-resolution, ECCV, 2014.
[16] B. Lim, S. Son, H. Kim, S. Nah, K.M. Lee, Enhanced deep residual networks for single image super-resolution, CVPRW, 2017.
[17] Z. Huang, L. Wang, G. Meng, C. Pan, Image super-resolution via deep dilated convolutional networks, ICIP, 2017.
[18] Fisher Yu, Vladlen Koltun, Multi-scale context aggregation by dilated convolutions, abs/1511.07122, ICLR, 2016.
[19] M. Zareapoor, M.E. Celebi, J. Yang, Diverse adversarial network for image super-resolution, Signal Processing: Image Communication 74 (2019) 191–200.
[20] J. Kim, J. Kwon Lee, K. Mu Lee, Accurate image superresolution using very deep convolutional networks, CVPR, 2016.
[21] J. Kim, J. Kwon Lee, K. Mu Lee, Deeply-recursive convolutional network for image super-resolution, CVPR, 2016.
[22] H. Chen, Y. Zhang, M.K. Kalra, F. Lin, Y. Chen, P. Liao, J. Zhou, G. Wang, Low-dose CT with a residual encoder-decoder convolutional neural network (RED-CNN), IEEE Trans. Med. Imaging 36 (12) (2017) 2524–2535.
[23] Y. Tai, J. Yang, X. Liu, Image super-resolution via deep recursive residual network, CVPR, 2017.
[24] H. Zhang, V.M. Patel, Densely connected pyramid dehazing network, CVPR, 2018.
[25] Y. Romano, J. Isidoro, P. Milanfar, RAISR: rapid and accurate image super resolution, IEEE Trans. Comput. Imaging 3 (1) (2017) 110–125.
[26] S. Mei, X. Yuan, J. Ji, Y. Zhang, S. Wan, Q. Du, Hyperspectral image spatial super-resolution via 3D full convolutional neural network, Remote Sens. 9 (11) (2017) 1139.
[27] S. Galliani, C. Lanaras, D. Marmanis, E. Baltsavias, K. Schindler, Learned Spectral Super-resolution arXiv PreprintarXiv:1703.09470, 2017.
[28] H. Zhu, X. Tang, J. Xie, W. Song, F. Mo, X. Gao, Spatio-temporal super-resolution reconstruction of remote-sensing images based on adaptive multi-scale detail enhancement, Sensors 18 (2) (2018) 498.
[29] N. Akhtar, A. Mian, Hyperspectral recovery from RGB images using Gaussian processes, IEEE Trans. Pattern Anal. Mach. Intell. 14 (8) (2018).
[30] P. Cui, S. Jin, L. Yu, F. Wang, W. Zhu, S. Yang, Cascading outbreak prediction in networks: A data-driven approach, Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2013, pp. 901–909.
[31] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.
[32] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi, Photo-realistic single image super-resolution using a generative adversarial network, CVPR 2017, pp. 4681–4690.
[33] W.-S. Lai, J.-B. Huang, N. Ahuja, M.-H. Yang, Deep laplacian pyramid networks for fast and accurate super resolution, CVPR, 2017.
[34] W.-S. Lai, J.-B. Huang, N. Ahuja, M.-H. Yang, Fast and accurate image super-resolution with deep Laplacian pyramid networks, CVPR, 2018.
[35] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, International 11 Conference on Medical Image Computing and Computer-assisted Intervention, Springer 2015, pp. 234–241.
[36] P.F. Christ, M.E.A. Elshaer, F. Ettlinger, S. Tatavarty, M. Bickel, P. Bilic, M. Rempfler, M. Armbruster, F. Hofmann, M. DAnastasi, Automatic liver and lesion segmentation in CT using cascaded fully convolutional neural networks and 3D conditional random fields, International Conference on Medical Image Computing and Computer-Assisted Intervention 2016, pp. 415–423.
[37] X. Glorot, A. Bordes, Y. Bengio, Deep sparse rectifier neural networks, AISTATS, 2011.
[38] V. Nair, G.E. Hinton, Rectified Linear Units Improve Restricted Boltzman Machines, ICML, 2010.
[39] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, G. Cottrell, Understanding Convolution for Semantic Segmentation, (arXiv preprint arXiv:1702.08502) 2017.

[40] E. Romera, J.M. Alvarez, L.M. Bergasa, R. Arroyo, Efficient convnet for real-time semantic segmentation, Intelligent Vehicles Symposium (IV) 2017, pp. 1789–1794.
[41] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, ICCV, 2001.
[42] M. Bevilacqua, A. Roumy, C. Guillemot, M.L. AlberiMorel, Low-complexity single-image super-resolution based on nonnegative neighbor embedding, BMVC, 2012.
[43] R. Zeyde, M. Elad, M. Protter, On single image scale-up using sparse-representations, Proc. 7th Int. Conf. Curves Surf, 2010.
[44] J.-B. Huang, A. Singh, N. Ahuja, Single image superresolution from transformed self-exemplars, CVPR, 2015.
[45] H. Chen, X. He, L. Qing, Q. Teng, C. Ren, SGCRSR: sequential gradient constrained regression for single image super-resolution, Signal Process. Image Commun. 66 (2018) 1–18.
[46] C. Zhang, W. Liu, J. Liu, C. Liu, C. Shi, Sparse representation and adaptive mixed samples regression for single image super-resolution, Signal Process. Image Commun. 67 (2018) 79–89.
[47] J. Xiao, E. Liu, L. Zhao, Y.F. Wang, W. Jiang, Detail enhancement of image super-resolution based on detail synthesis, Signal Process. Image Commun. 50 (2017) 21–33.