Image Super-Resolution Using Dense Skip Connections

Tong Tong, Gen Li, Xiejie Liu, Qinquan Gao
Imperial Vision Technology
Fuzhou, China
{ttraveltong,ligen,liu.xiejie,gqinquan}@imperial-vision.com

Abstract

Recent studies have shown that the performance of single-image super-resolution methods can be significantly boosted by using deep convolutional neural networks. In this study, we present a novel single-image super-resolution method by introducing dense skip connections in a very deep network. In the proposed network, the feature maps of each layer are propagated into all subsequent layers, providing an effective way to combine the low-level features and high-level features to boost the reconstruction performance. In addition, the dense skip connections in the network enable short paths to be built directly from the output to each layer, alleviating the vanishing-gradient problem of very deep networks. Moreover, deconvolution layers are integrated into the network to learn the upsampling filters and to speedup the reconstruction process. Further, the proposed method substantially reduces the number of parameters, enhancing the computational efficiency. We evaluate the proposed method using images from four benchmark datasets and set a new state of the art.

1. Introduction

The recovery of a high resolution (HR) image from a low resolution (LR) version is a highly ill-posed problem since the mapping from LR to HR space can have multiple solutions. When the upscaling factor is large, it becomes very challenging to recover the high-frequency details in image super-resolution (SR). Many SR techniques assume that the high-frequency information is redundant and can be accurately predicted from the low-frequency data. Therefore, it is important to collect useful contextual information in large regions from LR images so that sufficient knowledge can be captured for recovering the high-frequency details in HR images.

Recent works [11, 12] have successfully used very deep convolutional neural networks (CNN) to perform single image super-resolution (SISR), and significant improvements over shallow CNN structures [2] have been observed. One benefit from using deeper networks is that larger receptive field takes more contextual information from LR images to predict data in HR images. However, it is challenging to effectively train a very deep CNN due to the vanishing-gradient problem. One good solution to this problem is the use of skip connections, which create short paths from top layers to bottom layers. This helps the flow of information and gradient through the network, making it easy to train. In addition, in previous works [2, 11], only high-level features at top layers were used in the reconstruction of HR images. The features at low levels can potentially provide additional information to reconstruct the high-frequency details in HR images. Image SR may benefit from the collective knowledge of features at different levels. Moreover, previous studies [8, 7] have shown that redundant feature maps are learnt in different layers of deep networks. The reuse of feature maps from bottom layers is helpful for reducing feature redundancy, thus learning more compact CNN models.

In this work, we propose a novel super-resolution method termed SRDenseNet in which the dense connected convolutional networks were employed. The introduction of dense connections improves the flow of information through the network, alleviating the gradient vanishing problem. In addition, it allows the reuse of feature maps from preceding layers, avoiding the re-learning of redundant features. Different from previous works, we utilized the dense skip connections to combine the low-level features and high-level features in order to provide rich information for the SR reconstruction. Further, deconvolution layers were integrated to recover the image details and to speedup the reconstruction process. The proposed method has been evaluated on four publicly available benchmark datasets and outperforms the current state-of-the-art approaches.

2. Related work

2.1. Single image super-resolution

Many SISR methods have been developed in computer vision community. A detailed review of these methods can

be found in [26]. Among them, interpolation methods are easy to implement and widely adopted. However, these linear models have very limited representation power and often generate blurry high resolution outputs. Sparsity-based techniques [28, 24] have recently developed to enhance linear models with rich image priors. These techniques assume that any natural image patch can be sparsely represented by a dictionary of atoms. The dictionary can be formed by a database of patches or learnt from the database [27]. Such dictionary-based methods [25] have achieved comparable state-of-the-art results. One drawback of these methods is that it is generally computationally expensive to find the solution of the sparse coding coefficients .

In addition to sparsity-based methods, other sophisticated learning techniques have been developed to model the mapping from LR to HR space, including neighbor embedding [4], random forest [20] and convolutional neural network [2]. Among them, the CNN-based approaches [11, 12] have recently set state of the art for SISR. A network with three layers was first developed in [2] to learn an end-to-end mapping for SR. Subsequently, a deep network with 20 layers was proposed in [11] to improve the reconstruction accuracy of CNN. The residuals between the HR images and the interpolated LR images were used in [11] to speedup the converging speed in training and also to improve the reconstruction performance. Instead of using interpolation for upscaling as in [2, 11], recent studies [3, 21] have demonstrated that the SR performance can be further improved both in terms of accuracy and speed by learning the upscaling filters. The upscaling operation can be effectively learnt by using deconvolution layers [3] or sub-pixel convolution layers [21]. In our work, we employ the very deep network and also integrate the deconvolution layers to further boost the reconstruction performance.

2.2. Skip connections

As CNNs become increasingly deep, the problem of vanishing gradient hampers the training of networks. Many recent approaches have been proposed to address this problem. ResNets [6] and Highway Networks [22] use bypassing path between layers to effectively train networks with more than 100 layers. Stochastic depth [8] randomly drops layers to improve the training of deep residual networks, which demonstrates a great amount of redundancy in deep residual networks. FractalNets [14] combines several parallel networks with different depths and many short paths are created in the networks. DenseNets [7] links all layers in the networks and tries to fully explore the advantages of skip connections. All these networks share a key idea: it is essential to build many skip connections between layers to effectively train a very deep network.

A skip connection was used in [12] to link the input data and the final reconstruction layer in SR. State-of-the-art SR results were achieved in [12]. However, only a single skip connection was adopted in [12], which may not fully explore the advantages of skip connections. Many symmetric skip connections were introduced in an encoding-decoding network [17] for image restoration tasks. However, the improvement of the SR performance over the DRCN method [12] that used a single skip connection is marginal. An effective way of using a reasonable amount of skip connections in very deep CNNs may potentially improve the SR reconstruction performance.

2.3. Contribution

Skip connections can alleviate the vanishing-gradient problem and enhance the feature propagation in deep networks. In this work, we introduce dense skip connections in a deep network for SISR. Our main contributions are:

- We demonstrate that the deep CNN framework with the denseNet as basic blocks can achieve good reconstruction performance and that the fusion of features at different levels through dense skip connections can further boost the reconstruction performance for SISR.
- New state-of-the-art results have been achieved on four benchmark datasets with a upscaling factor of 4 and visual improvements can be easily noticed in the SR results. The proposed framework not only achieves impressive results but also can be implemented very fast.

The proposed network structure is introduced in Section 3, followed by the experimental results and visual comparsions with state-of-the-art results in Section 4. A further discussion is provided in Section 5 and the paper concludes in Section 6.

3. Method

The proposed network aims to learn an end-to-end mapping function F between the LR image I_L and the HR image I_H. As shown in Figure 1, SRDenseNet can be decomposed into several parts: the convolution layer for learning low-level feature, the blocks of DenseNet for learning high-level features, the deconvolution layers for learning upscaling filters and the reconstruction layer for generating the HR output. Each convolution or deconvolution layer is followed by a ReLu layer for nonlinear mapping except the reconstruction layer. The ReLu activation function is applied element-wise. Let X_{i-1} be the input, the output of i^{th} convolution or deconvolution layer is expressed as:

$$X_i = max(0, w_i * X_{i-1} + b_i) \quad (1)$$

where W_i and B_i are the weights and biases in the layer, and $*$ denotes either convolution or deconvolution operation for the convenience of formulation. Let Θ denote all

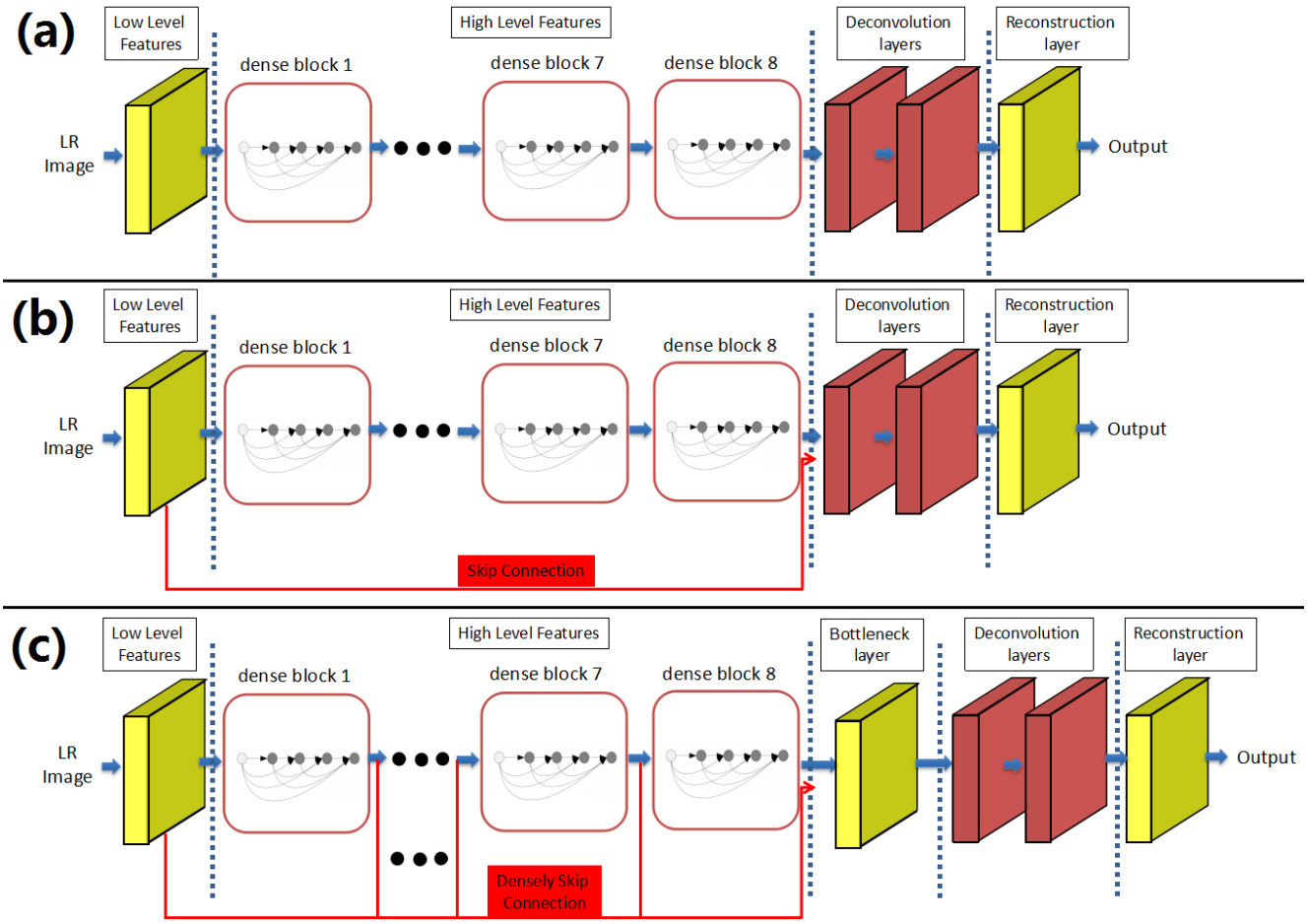

Figure 1. Different structures of the proposed networks. (a) SRDenseNet_H: only the high-level feature maps are used as input for reconstructing the HR images. (b) SRDenseNet_HL: the low-level and the high-level features are combined as input for reconstructing the HR images. (c) SRDenseNet_All: all levels of features are combined via skip connections as input for reconstructing the HR images.

the weights and biases in the network $\Theta = \{W_i, B_i\}, i = 1, ..., m$. Given a set of training image pairs $\{I_L^k, I_H^k\}$, we minimize the following Mean Squared Error (MSE):

$$l(\Theta) = \frac{1}{N} \sum_{k=1}^{N} \left\| F(I_L^k, \Theta) - I_H^k \right\|_2^2 \qquad (2)$$

Adam [13] is used to find the optimum weights and biases in the above equation. In the following, we will describe the details of the proposed network structures.

3.1. DenseNet blocks

After applying a convolution layer to the input LR images for learning low-level features, a set of DenseNet blocks are adopted for learning the high-level features. The DenseNet structure was first proposed in [7]. Different from ResNets as proposed in [6], the feature maps are concatenated in DenseNet rather than directly summed. Consequently, the i^{th} layer receives the feature maps of all preceding layers as input:

$$X_i = max(0, w_i * [X_1, X_2, ..., X_{i-1}] + b_i) \qquad (3)$$

where $[X_1, X_2, ..., X_{i-1}]$ represents the concatenation of the feature maps generated in the preceding convolution layers $1, 2, ..., i-1$. In the structure of DenseNet, short paths are created between a layer and every other layer. This strengthens the flow of information through deep networks, thus alleviating the vanishing-gradient problem. In addition, DenseNet can substantially reduce the number of parameters through feature reuse, thus requiring less memory and computation to achieve high performance [7]. Here, we employ the DenseNet structure as a building block in our network. The structure of each denseNet block can be seen in Figure 2. Specifically, there are 8 convolution layers in one DenseNet block in our work. If each convolution layer produce k feature maps as output, the total number of feature maps generated by one DenseNet block is $k * 8$,

where k is refered to as *growth rate*. The *growth rate* k regulates how much new information each layer contributes to the final reconstruction. To prevent the network from growing too wide, the *growth rate* k is set to 16 in this study. This results in a total number of 128 feature maps from one DenseNet block.

3.2. Deconvolution layers

In previous SR methods such as SRCNN [2] and VDSR [11], bicubic interpolation is used to upscale LR images to the HR space. After that, the SR process including the computationally expensive convolution is carried out in the HR space. This increases the computational complexity for SR. In addition, interpolation approaches do not bring new information for solving the SR problem. Therefore, recent works [3, 17] have employed deconvolution layers to learn the upscaling filters, which can also recover the image details. The deconvolution layer can be considered as an inverse operation of a convolution layer. It can learn diverse upscaling kernels that work jointly for predicting the HR images. There are two advantages in using the deconvolution layers for upscaling. First, it accelerates the SR reconstruction process. After the deconvolution layers are added at the end of networks, the whole computational process is performed in the LR space. If the upscaling factor is r, it will reduce the computational cost by a factor of r^2. In addition, a large amount of contextual information from the LR images is used to infer the high frequency details. Using the same depth, the receptive field of the network with deconvolution layers at the end is about r^2 times larger than that of the network using interpolation at the beginning. In our work, two successive deconvolution layers with small 3×3 kernels and 256 feature maps are trained for upscaling.

3.3. Combination of feature maps

As shown in Figure 1, three different types of network structures were studied and compared in our work. As in previous methods [2, 11], only the feature maps at the top layer are used as input for reconstructing the HR output. We denote this structure as SRDenseNet_H which is shown in Figure 1 (a). Further, a skip connection is introduced in the network as shown in Figure 1 (b) to concatenate the low-level and high-level features, which we term SRDenseNet_HL. The concatenated feature maps are then used as input for deconvolution layers. In addition, we use dense skip connections to combine the feature maps produced at all convolution layers for SR reconstruction, and denote this method as SRDenseNet_All. A comparison between the SR results using different network structures will be performed in the experimental section.

3.4. Bottleneck and Reconstruction layers

In the proposed SRDenseNet_All as shown in Figure 1 (c), all feature maps in the network are concatenated, yielding many inputs for the subsequent deconvolution layers. If the large number of feature maps are directly fed into deconvolution layers, it will significantly increase the computational cost and the model size. Thus, it is necessary to reduce the number of input feature maps in order to keep model compactness and to improve the computational efficiency. It has been demonstrated in previous studies [23] that a convolution layer with 1×1 kernel can be used as a bottleneck layer to reduce the number of input feature maps. To improve the model compactness and computational efficiency, we employ a bottleneck layer to reduce the number of feature maps before feeding them to the deconvolution layers. The number of feature maps is reduced to 256 using 1×1 bottleneck layer. After that, the deconvolution layers transform the 256 feature maps from the LR space to the HR space. Finally, the feature maps in the HR space are used to generate HR images via a reconstruction layer. The reconstruction layer is a convolution layer with 3×3 kernel and one channel of output.

4. Experiments

In this section, we evaluated the performance of the proposed method on four benchmark datasets. A description of the datasets is first provided, followed by the introduction of the implementation details. The benefit of using different levels of features is then introduced. After that, comparisons with state-of-the-art results are presented.

4.1. Datasets and metrics

During the evaluation, we used publicly available benchmark datasets for training and testing. Specifically, 50,000 images were randomly selected from ImageNet for the training. During testing, the dataset **Set5** [1] and **Set14** [29] are often used for SR benchmark. The **B100** from the Berkeley segmentation dataset [18] consisting of 100 natural images were used for testing. In addition, the proposed method was also evaluated using the **Urban100** dataset [9] which includes 100 challenging images. All experiments were performed using a scale factor of $4\times$ between LR and HR images. The peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) index were used as metrics for evaluation. Since SR was performed in the luminance channel in YCbCr colour space, the PSNR and SSIM were calculated on the Y-channel of images.

4.2. Implementation details

Non-overlapping sub-images with a size of 100×100 were cropped in the HR space. The LR images were obtained by downsampling the HR images using bicubic ker-

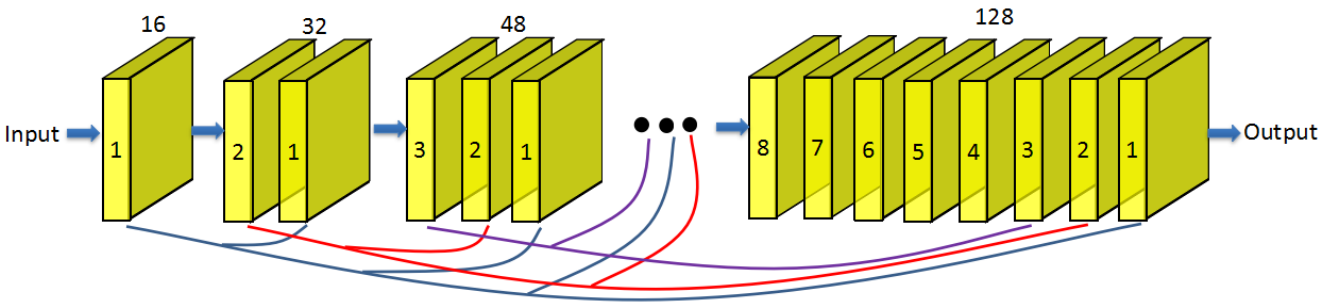

Figure 2. The structure of one DenseNet block. Each block consists of 8 convolution layers. The growth rate is set to 16 and the output of each block has 128 feature maps.

nel with a scale factor of $4\times$. As suggested by previous studies, each image has been transformed into YCbCr space and only the Y-channel was used for training. In all networks, 8 DenseNet blocks were used, resulting in 64 convolution layers. Within each block, a growth rate of 16 was set. This generated an output of 128 feature maps from each block. The filter size was set to 3×3 in all weight layers. The weights were initialized using the method proposed in [5] and the biases were initialized to zero. The rectified linear units (ReLu) was used as the activation function. All the networks were optimized using Adam [13]. The learning rate was initially set to 0.0001 and decreased by a factor of 10 after 30 epoches. A mini-batch size of 32 was set during the training. The training process stopped after no improvements of the loss was observed after 60 epoches. A NVIDIA Titan X GPU was used for training and testing.

4.3. Benefit of feature combination

The reconstruction performance using the three types of network structures as shown in Figure 1 were compared. Table 1 shows the obtained PSNR and SSIM values on four datasets. As expected, SRDenseNet_HL achieved better results than SRDenseNet_H after adding a skip connection. This indicates that the combination of low-level features and high-level feature can improve the SR reconstruction performance. A further improvement was observed by concatenating all levels of features. This suggests that there are complementary information among different levels of feature maps for SR. The improvements by combining different levels of features can also be seen in Figure 4.

4.4. Comparison with state-of-the-art methods

We compared the results using the proposed method and those using other SISR methods, including bicubic, Aplus [24], SRCNN [2], VDSR [11] and DRCN [12]. The implementations of these methods have been released online and thus can be carried out on the same datasets for fair comparisons. For SRCNN, the best 9-5-5 image model was used for comparison in this section. As for the Aplus method [24], it did not predict image boundaries. To enable a fair comparison, the borders of HR images were cropped so that all the results had the same region. The public code in [9] was used for calculating the evaluation metrics. Table 2 shows the average PSNR and SSIM values on four benchmark datasets. In terms of PSNR, the proposed method achieves an improvement of 0.2dB-0.8dB over state-of-the-art results on different datasets. On average, an increase of about 1.0 dB using the proposed method was achieved over SRCNN [2] with 3-layer CNN and an increase of about 0.5 dB over VDSR [11] with 20-layer CNN. It should be mentioned that the most significant improvement is obtained on the very challenging dataset Urban100.

Visual comparisons using different methods are given in Figures 3 and 5. In Figure 3, only the proposed method can well reconstruct the lines and the contours while other methods generate blurry results. In addition, severe distortions are found in some reconstructed results using existing methods (i.e. middle panel in Figure 5) whereas our method can reconstruct the texture pattern and avoid the distortions.

5. Discussion and future work

When the proposed SRDenseNet_All is unfolded, the longest chain has 69 weight layers and 68 activation layers. The SR task can benefit from using this very deep network in two aspects: (a) since the size of the receptive field is proportional to the depth, a large amount of contextual information in LR images can be utilized to infer the high frequency information in HR images; (b) due to the use of many ReLu layers, high nonlinearity can be exploited in very deep networks to model the complex mapping functions between LR image and HR images. One challenging problem in very deep network is the vanishing-gradient problem. In this work, we utilized the DenseNet structure as building blocks to alleviate this problem. DenseNets allow layers to use feature maps from their preceding layers. This provides an effective way to reuse feature maps that are already learnt and forces the current layer to learn complementary information, thus avoiding the learning of redundant features. In addition, each layer has a short path

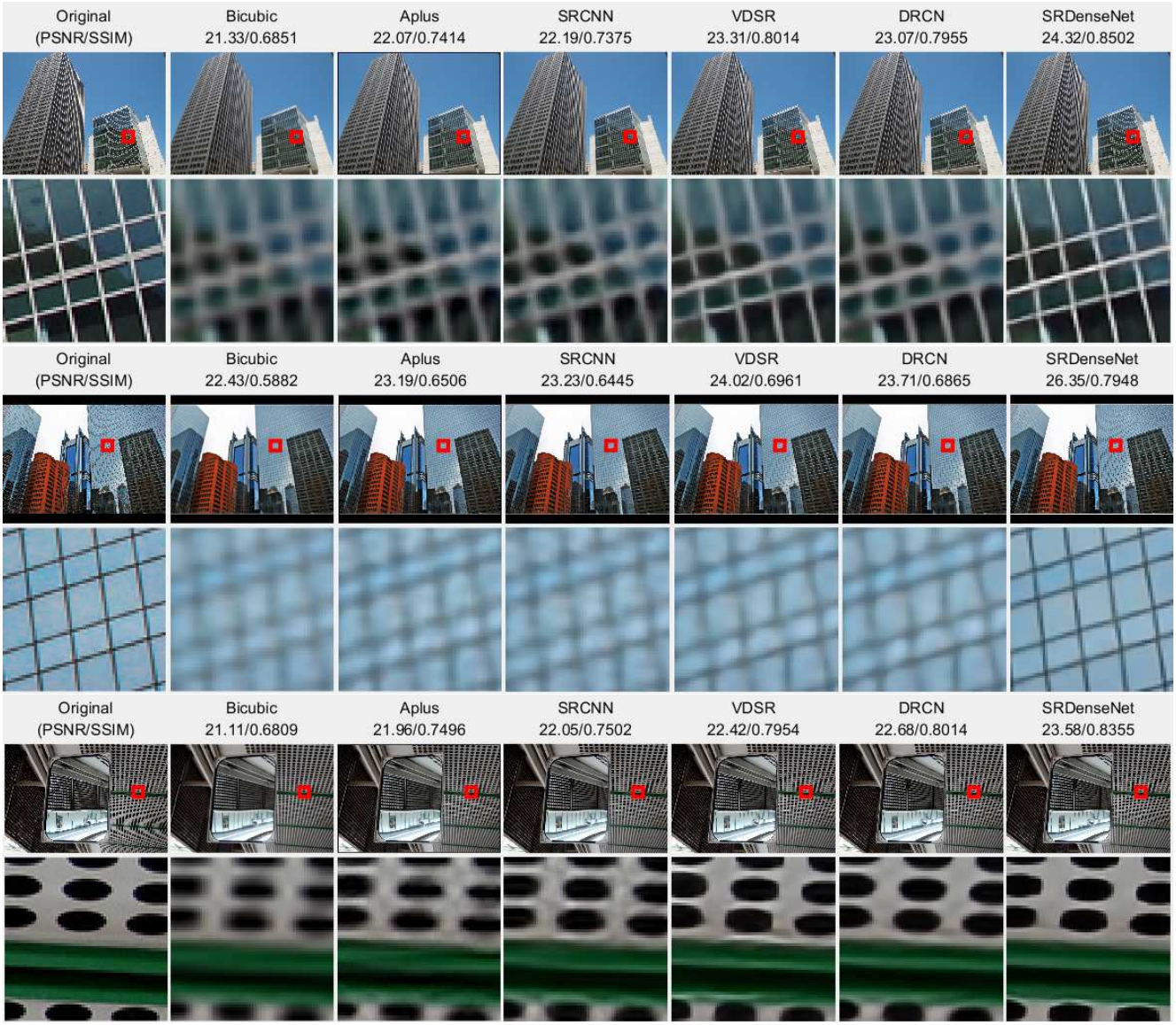

Figure 3. Super-resolution results for "img096" (top figure),"img099" (middle figure) and "img004" (bottom figure) from **Urban100** with an upscaling factor of 4. PSNR and SSIM values are shown on the top of each sub-figure.

Dataset	SRDenseNet_H	SRDenseNet_HL	SRDenseNet_All
Urban100	25.69/0.7700	25.86/0.7761	26.05/0.7819
Set5	31.66/0.8882	31.80/0.8907	32.02/0.8934
Set14	28.34/0.7744	28.40/0.7765	28.50/0.7782
B100	27.42/0.7300	27.47/0.7318	27.53/0.7337

Table 1. Comparison of results in terms of PSNR/SSIM on four benchmark data using three different network structures.

to the loss in the proposed network, leading to an implicit deep supervision [16]. This can help the training of very deep networks and improve the reconstruction performance in SR [12].

Several techniques were proposed to improve the accuracy and to speedup the SR process, contributing to the novelty of the proposed framework. In order to improve the reconstruction accuracy, three techniques were proposed and integrated. (a) First, the DenseNet was used as a basic block in our network. This is the first work that uses denseNet for SR. One benefit of using the DenseNet Block is to avoid the gradient vanishing problem, allowing us to train very deep

Dataset	Bicubic	Aplus [24]	SRCNN [2]	VDSR [11]	DRCN [12]	SRDenseNet_All
Urban100	23.14/0.6577	24.32/0.7183	24.52/0.7221	25.18/0.7524	25.14/0.7510	**26.05/0.7819**
Set5	28.42/0.8104	30.28/0.8603	30.48/0.8628	31.35/0.8838	31.53/0.8854	**32.02/0.8934**
Set14	26.00/0.7027	27.32/0.7491	27.49/0.7503	28.01/0.7674	28.02/0.7670	**28.50/0.7782**
B100	25.96/0.6675	26.82/0.7087	26.90/0.7101	27.29/0.7251	27.23/0.7233	**27.53/0.7337**
All	24.73/0.6685	25.79/0.7191	25.93/0.7216	26.47/0.7439	26.42/0.7424	**27.02/0.7622**

Table 2. Comparison of SR results in terms of PSNR/SSIM using different methods. All means the combination of four datasets including 219 testing images.

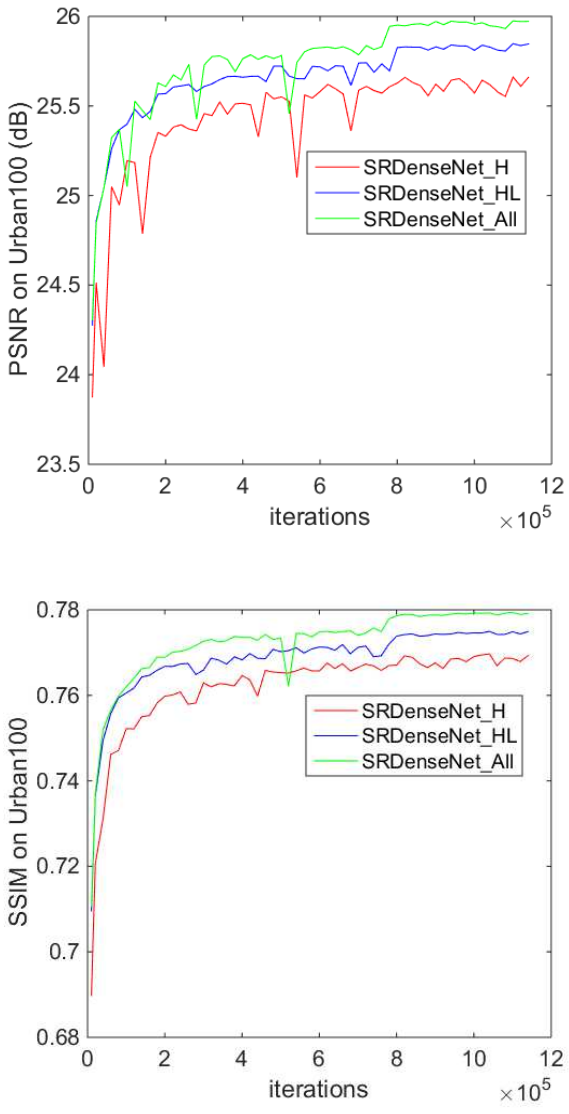

Figure 4. Comparison of PSNR and SSIM values on the Urban100 dataset using three different network structures.

CNN. (b) Second, low level features and high level features were fused via skip connections. The fusion of different levels of features can provide rich information for reconstructing the high-frequency information in high resolution images. We have demonstrated that the fusion process significantly improved the accuracy as shown in Figure 4, indicating complementary information among different levels of features.. (c) The use of successive deconvolutional layers also boosts the reconstruction performance. The deconvolutional layers can learn the up-scaling filters, thus avoiding the use of the bicubic interpolation as adopted in previous algorithms such as VDSR and DRCN. Therefore, the use of the DenseNet blocks is just one part of the contributions in the proposed framework for improving the accuracy.

In addition, the proposed framework not only achieves impressive results but also can be implemented very fast. This was resulted from three aspects: (a) The adoption of the 1*1 convolutional layer significantly reduced the parameters of the network; (b) The use of deconvolutional layers transferred the convolution process from high-resolution space to low-resolution space, thus substantially reducing the computational complexity. (c) A small growth rate of 16 was set in the DenseNet blocks. This means that only 16 new feature maps are required to learn for each convolutional layers. Although the growth rate is low, the total number of features is still large due to the fusion of different levels of features as mentioned above. This enables feature reuse and provides rich information for reconstructing the high-resolution images. In the end, we have achieved an average speed of $36.8ms$ for super-resolving one single image from the **B100** dataset on a Titan X GPU, reaching a real-time SR with a scaling factor of $4\times$.

In this work, only the MSE loss is used for guiding the training of networks. The use of MSE loss can lead to results with high PSNR values. However, high PSNR values do not necessarily represent visually pleasing results. Recently, perceptual loss was proposed in [10] for SR to replace the low-level pixel-wise loss. Further, adversarial loss using a generative adversarial network (GAN) was added to the loss function in [15, 19] and photo-realistic SR images can be generated. Although the generated high frequency details in SRGAN [15] may be 'fake' texture patterns, it yields visually pleasing high-resolution images. Note that the proposed method can provide a very good generator net-

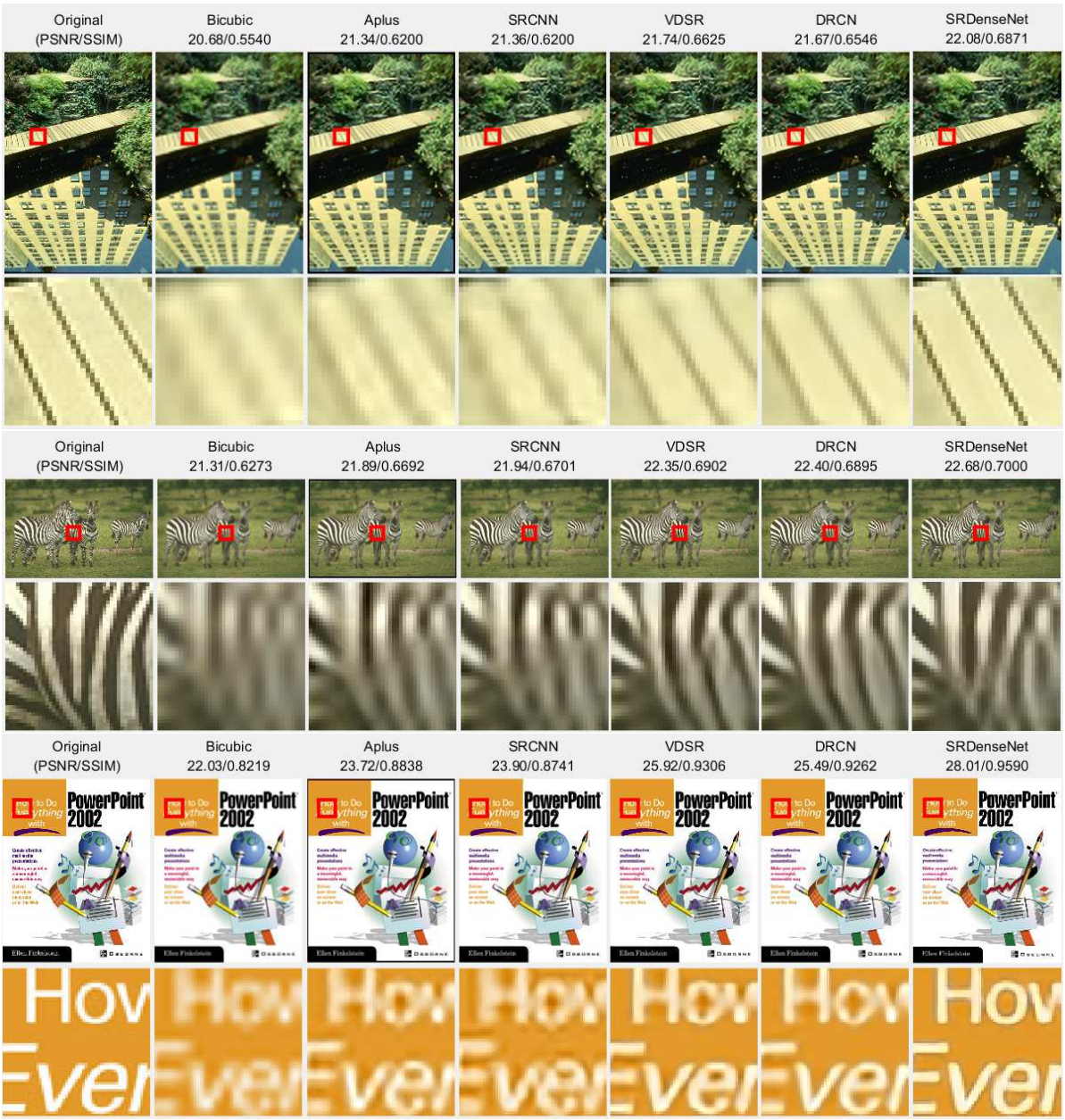

Figure 5. Super-resolution results for "148026" from **B100** (top figure),"253027" from **B100** (middle figure) and "ppt3" from **Set14** (bottom figure) with an upscaling factor of 4. PSNR and SSIM values are shown on the top of each sub-figure. Severe distortions are found in the results of "253027" from **B100** using other methods while the proposed method can accurately reconstruct the original pattern.

work initialized for GAN. It would be very interesting to investigate the integration of perceptual loss in the proposed framework in order to improve the visualized quality of the reconstructed images in future.

6. Conclusion

In this paper, we have presented a novel network that employs dense skip connections for SR. The proposed approach outperforms state-of-the-art methods by a considerable margin on four benchmark datasets in terms of PSNR and SSIM. Noticeable improvement can visually be found in the reconstruction results. In addition, we have demonstrated that the combination of features at different level is helpful for improving SR performance. Future work will focus on the integration of perceptual loss in the proposed network to reconstruct photo-realistic HR images.

References

[1] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. A. Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *British Machine Vision Conference*, 2012.

[2] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2016.

[3] C. Dong, C. C. Loy, and X. Tang. Accelerating the super-resolution convolutional neural network. In *European Conference on Computer Vision*, pages 391–407. Springer, 2016.

[4] X. Gao, K. Zhang, D. Tao, and X. Li. Image super-resolution with sparse neighbor embedding. *IEEE Transactions on Image Processing*, 21(7):3194–3205, 2012.

[5] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *International Conference on Computer Vision*, pages 1026–1034, 2015.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[7] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten. Densely connected convolutional networks. *arXiv preprint arXiv:1608.06993*, 2016.

[8] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger. Deep networks with stochastic depth. In *European Conference on Computer Vision*, pages 646–661. Springer, 2016.

[9] J.-B. Huang, A. Singh, and N. Ahuja. Single image super-resolution from transformed self-exemplars. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2015.

[10] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016.

[11] J. Kim, J. Kwon Lee, and K. Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1646–1654, 2016.

[12] J. Kim, J. Kwon Lee, and K. Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1637–1645, 2016.

[13] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[14] G. Larsson, M. Maire, and G. Shakhnarovich. Fractalnet: Ultra-deep neural networks without residuals. *International Conference on Learning Representations*, 2017. In press.

[15] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. *arXiv preprint arXiv:1609.04802*, 2016.

[16] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu. Deeply-supervised nets. In *International Conference on Artificial Intelligence and Statistics*, pages 562–570, 2015.

[17] X.-J. Mao, C. Shen, and Y.-B. Yang. Image restoration using convolutional auto-encoders with symmetric skip connections. In *Proceedings of the Neural Information Processing Systems*, 2016.

[18] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International Conference on Computer Vision*, volume 2, pages 416–423. IEEE, 2001.

[19] M. S. Sajjadi, B. Schölkopf, and M. Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. *arXiv preprint arXiv:1612.07919*, 2016.

[20] J. Salvador and E. Pérez-Pellitero. Naive bayes super-resolution forest. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 325–333, 2015.

[21] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1874–1883, 2016.

[22] R. K. Srivastava, K. Greff, and J. Schmidhuber. Training very deep networks. In *Advances in neural information processing systems*, pages 2377–2385, 2015.

[23] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[24] R. Timofte, V. De Smet, and L. Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *Asian Conference on Computer Vision*, pages 111–126. Springer, 2014.

[25] R. Timofte, R. Rothe, and L. Van Gool. Seven ways to improve example-based single image super resolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1865–1873, 2016.

[26] C.-Y. Yang, C. Ma, and M.-H. Yang. Single-image super-resolution: A benchmark. In *European Conference on Computer Vision*, pages 372–386. Springer, 2014.

[27] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang. Coupled dictionary training for image super-resolution. *IEEE Transactions on Image Processing*, 21(8):3467–3478, 2012.

[28] J. Yang, J. Wright, T. Huang, and Y. Ma. Image super-resolution as sparse representation of raw image patches. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[29] R. Zeyde, M. Elad, and M. Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010.