Single Image Super-Resolution via Laplacian Information Distillation Network

Mengcheng Cheng[1], Zhan Shu[1], Jiapeng Hu[1], Ying Zhang[2], Zhuo Su[1,*]
[1]*School of Data and Computer Science, National Engineering Research Center of Digital Life, Sun Yat-sen University*
[2]*21CN Corporation Ltd Guangzhou, China*
[*] *Corresponding Author: suzhuo3@mail.sysu.edu.cn*

***Abstract*—Recently, deep convolutional neural networks (C-NNS) have been revealed significant progress on single image super-resolution (SISR). Nevertheless, as the depth and width of the networks increase, CNN-based super-resolution (SR) methods have been confronted with the challenges of computational complexity and memory consumption in practice. In order to solve the above issues, we combine the Laplacian Pyramid with the previous methods to propose a convolutional neural network, which is able to reconstruct the HR image from low resolution image step by step. Our Laplacian-Pyramid structure allows each layer to share common parameters with other layers as well as its inner structure; this kind of characteristic reduces the number of parameters dramatically while still extracts sufficient features at the same time. In experiment part, we compare our method with the state-of-art methods. The results demonstrate that the proposed method is superior to the previous methods, furthermore our x2 model also gains an ideal effect.**

***Keywords*-Laplacian pyramid, Single image super-resolution, Deep learning**

I. Introduction

SISR is a classical low-level image processing problem, which needs to reconstruct a high resolution (HR) image from a low resolution (LR) image. It is an ill-posed inverse problem without a unique solution. In fact, the same LR image can be obtained by downsmpling infinite numbers of different HR images. Since SR problem has been established, plentiful SISR methods have been proposed in the literature, such as interpolation-based methods, reconstruction-based methods as well as example-based methods. Since the former two kinds of methods typically suffer dramatically drop in restoration performance with larger upscaling factors, recent SR methods fall into the example-based methods which try to learn prior knowledge from LR and HR pairs.

Currently, as a result of the rapid development of deep convolutional neural networks (CNN), many CNN-based SR methods try to train a deeper network to gain better reconstruction performance. From VDSR [6] to DRRN [8], the depth of their networks become deeper and deeper, hence they have to apply the shared-parameter strategy to their models to remit the computational complexity carried by the increased parameters. However, deeper networks means more complex computation and lager memory consumption. It is inadvisable in practical application, especially in real time application.

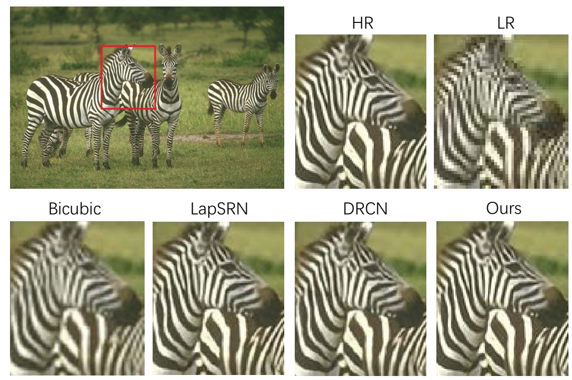


Figure 1. Under the data set of BSD100, The comparison among our method and other SR methods such as LapSRN, DRCN.

To solve the problems mentioned above, we propose a novel model called LapIDN, which combines Information Distillation Network and Laplacian Pyramid to upsample the LR image step by step. In our model, a feature extraction block (FBlock) first extracts features from the LR image. Then, multiple information distillation blocks (DBlocks) are stacked to progressively distill residual information. Finally, a reconstruction Block (RBlock) aggregates the obtained HR residual representations to generate the residual image. The comparison of our model' s performance with the state-of-the-art methods are illustrated in Figure 1, taking "253027" image from BSD100 as the example.

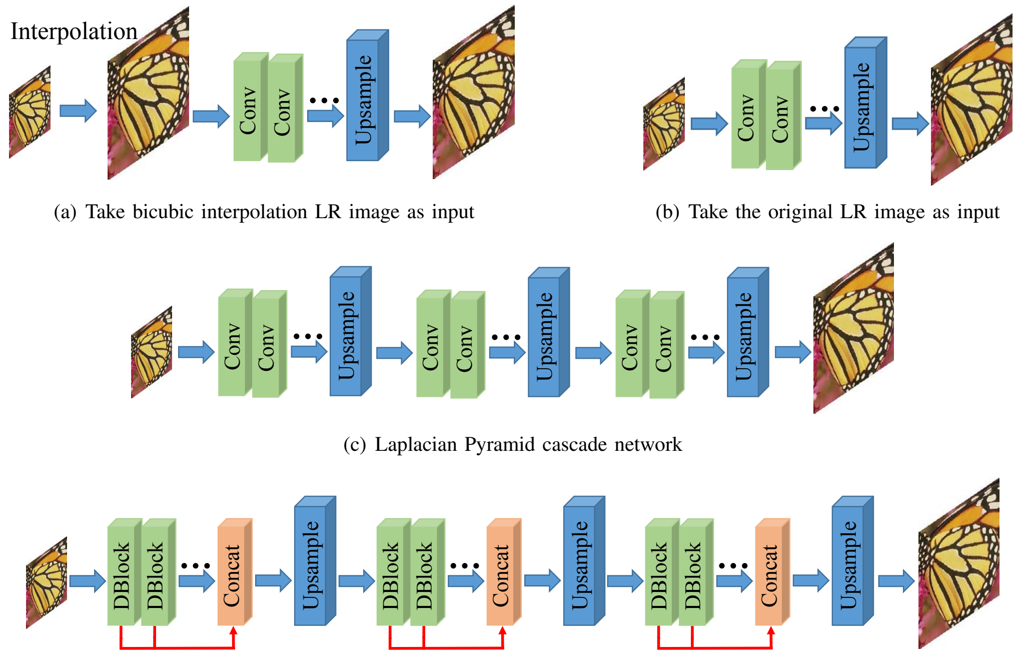
Our work provides the following contributions:

(1) We construct an end-to-end trainable architecture, putting the LR image into the network directly and constructing a HR image step by step. The proposed method allows us to mitigate the complexity of computation and improve the performance of reconstruction effectively at the same time.

(2) The proposed LapIDN model combines IDN with Laplacian Pyramid, comparing to the state-of-the-art methods, it keeps good performance of reconstruction, and boosts the speed of reconstruction dramatically in the meanwhile.

(3) The Laplacian Pyramid permits the parameters to be shared within layers as well as between layers, markedly decreasing the number of parameters, speeding the reconstruction of HR image.

978-1-5386-9497-8/18/$31.00 ©2018 IEEE
DOI 10.1109/ICDH.2018.00012



(a) Take bicubic interpolation LR image as input

(b) Take the original LR image as input

(c) Laplacian Pyramid cascade network

(d) Laplacian Information Distillation Network

Figure 2. The comparison of deep-learning-based super-resolution network structures. (a) take bicubic interpolation LR image as input (SRCNN [5], VDSR [6], DRNN [8] etc.); (b) take the original LR image as input (FSRCNN [9], ESPCN [10] etc.); (c) Laplacian Pyramid cascade network [11]; (d) LapIDN (ours) combed Laplacian Pyramid with IDN [12], where every layer is composed of several DBlocks connected by concat operation. The blue arrow represents the convolution layer.

II. RELATED WORK

The example-based methods exploit numerous samples to establish learning sets, and then construct the mapping relations between LR and HR; prior information is implicit in the mapping relations. Recently, with the development of deep learning technique, example-based methods are divided into tradition-based methods and deep-learning-based methods.

The tradition-based methods mainly exploit the mapping relations between LR images and HR images to indicate the reconstruction of HR images. In 2010, Yang et al. [1] combine sparse representation with dictionary learning to propose a novel approach to reconstruct HR images. In 2013, Timofte et al. [2] integrate neighborhood embedding with sparse representation to present anchored neighborhood regression to solve the SR problem effectively. In 2014, Timofte et al. [3] propose an adjusted anchored neighborhood regression based on their previous work to present a better performance. In 2016, Jiang et al. [4] propose "reconstruct the HR via locally regularized anchored neighborhood regression and nonlocal means" based on Timofte' s work.

Taking the LR image as input, two different kinds of input LR image represent two kinds of reconstruction approachs.

The first one is to take the bicubic-interpolation LR images as input. In 2014, Dong et al. [5] succeed to exploit convolutional neural networks to establish the mapping relations between LR images and HR images (SRCNN). Although there are only three layers in SRCNN, its performance is far superior to the tradition-based methods. In 2015, Kim et al. [6] presente VDSR with 20 layers of convolutional layers, which brings a significantly improvement in reconstruction. Kim et al. [7] come up with DRCN based on their previous work at the same year, dramatically decreasing the number of parameters while keeping the model' s performance. In 2017, Tai et al. [8] used a deeper networks (DRRN) to gain a better performance.

The second one is to take the original LR images as input. It reduces computational consumption a lot, because the mapping step is operated on low-dimensional LR features. In 2016, Dong et al. [9] propose an innovative model (FSRCNN) which takes original LR images as input rather than interpolated images, and get better performance than SRCRNN. Then Shi et al. [10] propose a real-time single image and video super-resolution model (ESPCN), which exploits the original LR images to learn the downsampled images of multiple HR images, then obtains the HR images through a sub-pixel convolution layer. In 2017, Lai et al. [11] present Deep Laplacian Pyramid Network (LAPSRN), adopting a cascaded amplification strategy, to upsample the LR images using deconvolution.

In 2018, Hui et al. [12] propose a model to exploit their

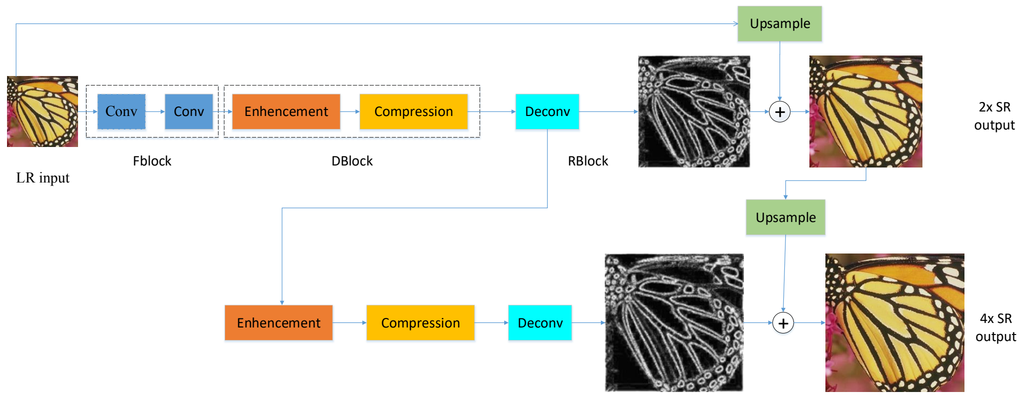


Figure 3. The information distillation network model of Laplacian structure.

designed distillate block to extract plenty of valid features in order to reconstruct the HR image (IDN). Zhang et al. [13] proposed Residual dense network, which makes more efficient using the hierarchical features of the original LR images through the convolutional layer. Zhang et al. [14]take some crucial degradation factors such as burring kernel and noise into consideration, which can handle SISR problems at higher magnifications effectively.Shoche et al. [15] propose "zero shot" which takes advantage of the repetitive information that appears inside a single image. Han et al. [16] propose a Dual-State Recurrent Networks to reconstruct HR images to make full use of recursive signals from LR images to HR images as well as HR images to LR images.

III. APPROACH AND MODEL ARCHITECTURE

In this section, we first introduce the prevalent deep-learning-based super-resolution networks in recent years, then we propose our novel model which combine Laplacian Pyramid and Information Distillation Network perfectly. Now we first introduce the three core modules of the proposed LapIDN model and the loss function.

A. Laplacian Pyramid model

Figure 2 illustrates the structures of super-resolution networks in recent years, where (a) takes bicubic interpolation images as input. This kind of input begins at the SRCNN model proposed by Dong et al, who think the consistence of the size of the input and output feature maps is advantageous to non-linear mapping; this also rises the complexity of calculation. Other representative methods are VDSR [6], DRRN [8] etc. Whereas the increment of calculation brought by the structure of (b), most of the model proposed latter take the original LR images as input directly to reconstruct the HR images. Compared with the network structure of (a), directly processing the LR images dramatically reduces the amount of calculation. The representative methods include FSRCNN [9], which uses a deconvolution layer to conduct the upsampled operation, and ESPCN [10], which uses a sub-pixel convolution layer to upsample the feature map. (c) represents the Laplacian Pyramid model (LapSRN) proposed by Lai et al. [11] in 2017. Whose structure is actually cascaded by the structure of multiple (b). This kind of cascaded amplification structure will give better performance at higher magnifications. (d) is the combination of Laplacian Pyramid and IDN named as LapIDN, where every layer of it has several DBlocks connected by concat operation.

To obtain the image residual, namely the interpolation of the original image of each layer of Laplacian pyramid and the result of Gaussian burring, we refer to the construction of the Laplacian pyramid, combined with IDN network, to propose our novel model LapIDN. In order to increase the resolution of the image, we add a deconvolution layer at the end of each pyramid level. Each layer of our designed pyramid network upsamples the image by a factor of two and our x4 model is based on the x2 model. Because of the similarity of each layer, the parameters are allowed to be shared among layers. Each Information Distillation Block is composed of 4 enhancement units and a compression unit, hence parameters can be shared between these Information Distillation Blocks, that is, parameters are allowed to be share within each layer. Layer-to-layer and intra-layer parameter sharing impressively reduces the number of parameters in our proposed model.

B. network structure

The proposed network, shown in Figure 3, contains three parts: a feature extraction module (FBlock), multiple stacked information distillation modules (DBlock) and a reconstruction module (RBlock). Here we make x and y the input and output of the IDN. As for FBlock, two convolution networks of 3x3 are used to extract the feature graph of the original LR image. The process is described as:

$$B_0 = f(x) \tag{1}$$

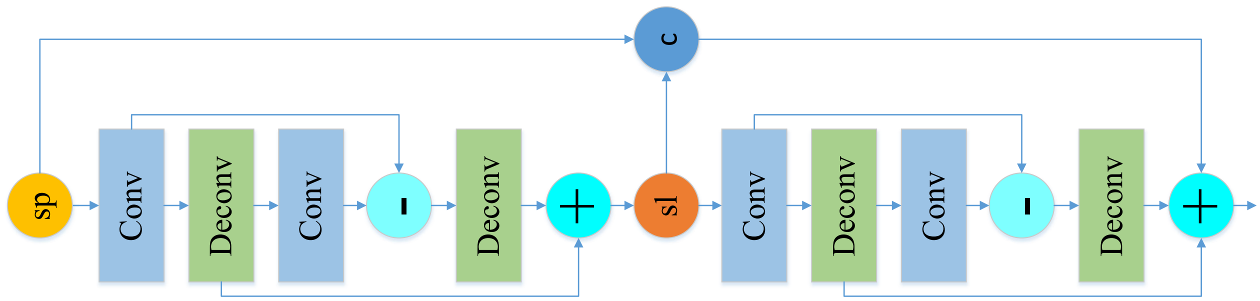


Figure 4. The structure diagram of Enhancement unit, where sp within the circle represents the split operation, sl represents the slice operation, c represents the concat operation, + represents the additive operation, and - represents the subtraction operation.

f refers to the feature extraction function, B_0 is the feature extracted, and x is the input later. The next section consists of multiple information distillation modules. Each module contains a stack of raised and compressed units. This process can be formulated as:

$$B_k = F_k(B_{k-1}), k = 1, \dots, n \tag{2}$$

Where F_k refers to the k-th DBlock function, B_k and B_{k-1} are the input and output of the k-th DBlock respectively. Finally, we use a transposition convolution without activation function as RBlock, so IDN can be expressed as:

$$y = R(F_n(B_{n-1})) + U(x) \tag{3}$$

Where, $R$and Urepresent RBlock and bicubic linear interpolation respectively.

The overall pseudo-code of our algorithm is shown in algorithm 1. Input is LR image $x$and LR image x_0 amplified by Bicubic, and output is HR image y.

C. Enhancement unit

As shown in Figure 4, we use the split layer as the input of the enhancement element. The first part adopts four consecutive convolutional layers and deconvolution layers, in which two convolutional layers are subtracted and two deconvolution layers are added. The output of the first part is obtained by slice layer and is the input to the second part. The second part is similar to the first part. After the output of split layer and slice layer pass Concat layer, add the output results of the later part to get the output of this unit. The convolution layer and the deconvolution layer mentioned above are both 3x3 convolution with a leaky rectified linear unit (LReLU)activation function. Assuming that the input for this module is B_{k-1}, we have

$$P_1^k = C_a(B_{k-1}) \tag{4}$$

B_{k-1} is the output of the previous unit and the input of the current unit. C_a denotes chain convolution operation, P_1^k it the output of the k-th unit mentioned above. The characteristic graph of dimension $\frac{D_3}{s}$ in P_1^k and the input

Algorithm 1 A single image super-resolution using Laplacian Information Distillation Network

Require: an original LR image $x \in R^{h \times w}$ and bicubic-magnified image $x_0 \in R^{rh \times rw}$, where h and w denotes the height and width of the LR image respectively, r is the upscale factor.
Ensure: the HR image y

//FBlock: feature extraction block, f represents the feature extraction function
$B_0 = f(x)$
//DBlocks: multiple stacked information distillation blocks
for $l = 1, 2, \dots, n$ **do**
  // Enhancement unit
  $P_1^k = Conv(split(B_{k-1}))$
  $P_2^k = Deconv(P_1^k)$
  $P_3^k = P_1^k \ominus Conv(P_2^k)$
  $P_4^k = P_2^k \oplus Deconv(P_3^k)$
  $R^k = Concat(Slice(P_4^k, 1/s), B_{k-1})$
  $P_5^k = Conv(Slice(P_4^k, 1 - 1/s))$
  $P_6^k = Deconv(P_5^k)$
  $P_7^k = P_5^k \ominus Conv(P_6^k)$
  $P_8^k = P_6^k \oplus Deconv(P_7^k)$
  $B_k = P_8^k \oplus R^k$
  //Compression unit
  $B_k = f_F^k(P^k)$
end for
//Rblock: reconstruction block
$y = Upsample(B_n) + x_0$

of the first convolution layer are connected on the channel dimension:

$$R^k = C(S(P_1^k, 1/s), B_{k-1}) \tag{5}$$

C and S denote connection concatenate operations and slice operations, respectively. The dimension of P_1^k is D_3, Therefore, $S(P_1^k, 1/s)$ denotes $\frac{D_3}{s}$ that dimensions features are fetched from P_1^k. In addition, $S(P_1^k, 1/s)$ concatenates features with B_{k-1} in channel dimension. the purpose is to combine the information before with current information. It can be regarded as locally retained short path information. We take the remaining locally retained short path information as the input of the following module, and mainly further extract the long path feature map:

$$P_2^k = C_b(S(P_1^k, 1 - 1/s)) \tag{6}$$

P_2^k and C_b are the stack convolution operations of the output and the following modules, respectively. Finally, the input information, the reserved local short path information and the minister path information were merged. Therefore, the enhancement unit can be formulated as:

$$\begin{aligned} P^k &= P_2^k + R^k \\ &= C_b(S(P_1^k, 1 - 1/s)) + C(S(P_1^k, 1/s), B_{k-1}) \end{aligned} \tag{7}$$

where P^k is the output of enhancement unit. local long-path features P^k and the combination of local short-path features and the untreated features R^k are utilized without exception by a compression unit.

D. Compression unit

We use the 1x1 convolution to implement the compression mechanism. Specifically, the output of the enhancement unit is sent to the 1x1 convolutional layer for dimensionality reduction or for subsequent network extraction of relevant information. Therefore, the compression unit can be formulated as:

$$B_k = f_F^k(P^k) = \alpha_F^k(W_F^k(P^k)) \tag{8}$$

f_F^k denotes 1 x1 convolution operation, α_F^k denotes the activation function, and W_F^k denotes weighting parameters.

E. Loss Function

We consider two loss functions to measure the predicted HR image $\hat{I}$ and the corresponding real image I. It's been proved that it is not a good choice to use MSE only for training. In the experiment, we first use MAE loss for training, and then use MSE loss for fine-tuning:

$$l_{MSE} = \frac{1}{N}\sum_{i=1}^{N}\|I_i - \hat{I}_i\|_2^2 \tag{9}$$

$$l_{MAE} = \frac{1}{N}\sum_{i=1}^{N}\|I_i - \hat{I}_i\|_1 \tag{10}$$

IV. EXPERIMENTS

In this section, we mainly introduces the source and acquisition of experimental data of this method, and the training process of specific parameters and network model. In addition, the methods proposed in this chapter are compared with state-of-the-arts, which are shown and compared respectively from objective data and subjective visual feelings.

A. Datasets processing

1) Training datasets: we use 91 images from Yang et al. and 200 images from Berkeley Segmentation Dataset (BSD) as the training data. During the training, we divided the image into image blocks for training. In order to make full use of the training data, we apply data augmentation in three ways: (1) Rotate the images with the degree of 90°、180°、270°. (2) Flip images horizontally. (3) Downscale the images with the factor of 0.9, 0.8, 0.7 and 0.6.

2) Testing datasets: The proposed method is evaluated on many widely used datasets including: Set5, Set14, BSD100, Urban100. The ground truth images are downscaled by bicubic interpolation to generate LR/HR image pairs for both training and testing datasets. We convert each RGB image into the YCbCr color space and only process the Y-channel, while other components are simply enlarged using bicubic interpolation.

B. Training process

We initially set the learning rate to 1e-4, and divided by 10 in the fine-tuning phase. When the weight is initialized, the bias value is set to 0, batch-size is 64, and the weight decay is set to 1e-4. All experiments were conducted using Caffe, MATLAB R2015b on on the machine with 4.0GHz Intel i7 CPU (32G RAM) and Nvidia GeForce GTX 1080 (Pascal) GPU (8G memory).

C. Model analysis

As shown in Figure 5, LapIDNs and LapIDN-Cs achieve better performance than other networks on dataset Set5 and factor 2x.

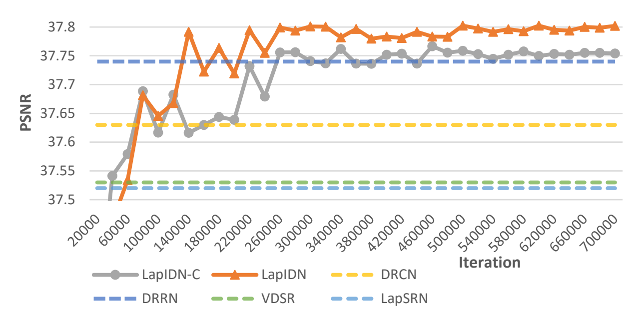


Figure 5. LapIDNs and LapIDN-Cs compare with other networks (DRCN, DRRN, VSDR, LapSRN) on Set5 dataset for scale factor 2×.

D. Comparisons with state-of-the-arts

We compared our proposed method with other SR methods, including bicubic, SRCNN, VDSR, DRCN, LapSRN, DRRN and IDN, etc. Table 1 shows the average peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) values of the four benchmark data sets.Compared to the latest results on these datasets, our approach has good performance. Figure 6 and 7 show visual comparisions. The

Table I
AVERAGE PSNR/SSIMS FOR SCALE 2X, 4X AND 8X. RED COLOR INDICATES THE BEST AND BLUE COLOR INDICATES THE SECOND BEST PERFORMANCE

		Set5		Set14		BSDS100		Urban100	
Algorithm	Scale	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	2	33.66	0.930	30.24	0.869	29.56	0.843	26.88	0.840
VDSR	2	37.53	0.958	32.97	0.913	31.90	0.896	30.77	0.914
DRCN	2	37.63	0.959	32.98	0.913	31.85	0.894	30.76	0.913
LapSRN	2	37.52	0.959	33.08	0.913	31.80	0.895	30.41	0.910
DRRN	2	37.74	0.959	33.23	0.913	32.05	0.897	31.23	0.919
IDN	2	37.75	0.959	33.10	0.913	32.02	0.898	31.13	0.918
LapIDN	2	37.80	0.960	33.11	0.914	32.03	0.898	31.19	0.919
Bicubic	4	28.42	0.810	26.00	0.703	25.96	0.668	23.14	0.658
VDSR	4	31.35	0.882	28.03	0.770	27.29	0.726	25.18	0.753
DRCN	4	31.53	0.884	28.04	0.770	27.24	0.724	25.14	0.752
LapSRN	4	31.54	0.885	28.19	0.772	27.32	0.728	25.21	0.756
DRRN	4	31.68	0.888	28.21	0.772	27.38	0.728	25.44	0.764
IDN	4	31.44	0.884	28.06	0.769	27.27	0.725	25.09	0.752
LapIDN	4	31.61	0.887	28.22	0.773	27.37	0.730	25.42	0.761
Bicubic	8	24.39	0.657	23.19	0.568	23.67	0.547	20.74	0.516
FSRCNN	8	25.41	0.682	23.93	0.592	24.21	0.567	21.32	0.537
VDSR	8	25.72	0.711	24.21	0.609	24.37	0.576	21.54	0.560
LapSRN	8	26.14	0.738	24.44	0.623	24.54	0.586	21.81	0.582
IDN	8	25.72	0.711	24.21	0.609	24.37	0.576	21.54	0.560
LapIDN	8	25.89	0.739	24.45	0.619	24.55	0.590	21.70	0.577

"108005" image has serious artifacts in the read box due to the loss of high frequency information, which can be seen from the result of bicubic interpolation. It can be seen that the "img097" image from the Urban 100 dataset with an upscaling factor 4 shows better performance than other methods.

V. CONCLUSION

In this paper, we propose a new convolutional neural network combining with the Laplacian pyramid structure, input the original LR image, and extract a large number of image features layer by layer through the network for high resolution image reconstruction. In the experiment, our method achieved good results in PSNR, SSIM. In addition, compared with the extremely deep network, our method takes much less time than them.

ACKNOWLEDGMENT

This work is supported by the Natural Science Foundation of Guangdong Province (2016A030310202), the Fundamental Research Funds for the Central Universities (Sun Yat-sen University, 16lgpy39), and the Science and Technology Planning Project of Guangzhou (201802020013).

REFERENCES

[1] T. Huang Y. Ma J. Yang, J. Wright, "Image super-resolution via sparse representation," IEEE transactions on image processing, vol. 19, no. 11, pp. 2861 – 2873, 2010

[2] L.V. Gool R. Timofte, V.D. Smet, "Anchored neighborhood regression for fast example-based super-resolution," in Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1920 – 1927

[3] L.V. Gool R. Timofte, V.D. Smet, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in Asian Conference on Computer Vision, 2014, pp. 111 – 126

[4] Junjun Jiang, Xiang Ma, Chen Chen, Tao Lu, Zhongyuan Wang, and Jiayi Ma, "Single image super-resolution via locally regularized anchored neighborhood regression and nonlocal means," IEEE Transactions on Multimedia, vol. 19, no. 1, pp. 15 – 26, 2016

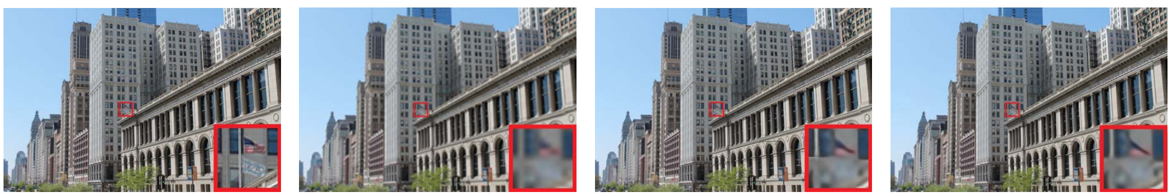
[5] K. He X. Tang C. Dong, C. Loy, "Image super-resolution using deep convolutional networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 2, pp. 295 – 307, 2016

[6] K. Lee J. Kim, J. Lee, "Accurate image super-resolution using very deep convolutional networks," computer vision and pattern recognition, pp.1646 – 1654, 2016

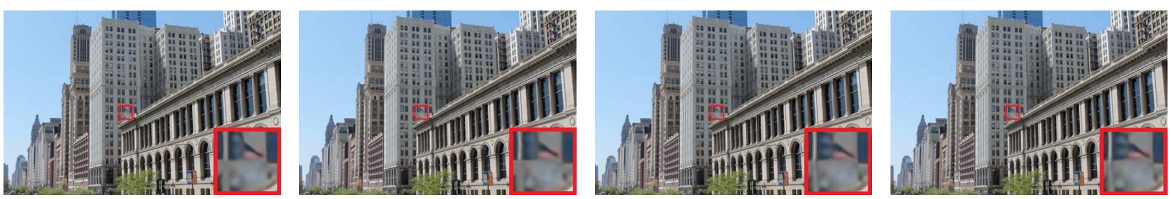
(a) Original (b) Bicubic (c) VDSR (d) DRCN

(e) LapSRN (f) DRRN (g) IDN (h) LapIDN

Figure 6. The "108005" image from the BSD100 dataset with an upscaling factor 2.



(a) Original (b) Bicubic (c) VDSR (d) DRCN



(e) LapSRN (f) DRRN (g) IDN (h) LapIDN

Figure 7. The "img097" image from the Urban100 dataset with an upscaling factor 4.

[7] K. Lee J. Kim, J. Lee, "Deeply-recursive convolutional network for image super-resolution," computer vision and pattern recognition, pp.1637 – 1645, 2016

[8] X. Liu Y. Tai, J. Yang, "Image super-resolution via deep recursive residual network," IEEE Conference on Computer Vision and Pattern Recognition, 2017

[9] K. He X. Tang C. Dong, C. Loy, "Accelerating the super-resolution convolutional neural network," in European Conference on Computer Vision, 2016, pp. 391 – 407

[10] F. Huszar J. Totz A. Aitken R. Bishop Z. Wang W. Shi, J. Caballero, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," computer vision and pattern recognition, pp. 1874 – 1883, 2016

[11] Lai W S, Huang J B, Ahuja N, et al. Deep Laplacian Pyramid Networks for Fast and Accurate Super-Resolution[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2017:5835-5843.

[12] Hui Z, Wang X, Gao X. Fast and Accurate Single Image Super-Resolution via Information Distillation Network[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 723-731.

[13] Zhang, Yulun, et al. "Residual dense network for image super-resolution." The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018.

[14] Zhang K, Zuo W, Zhang L. Learning a single convolutional super-resolution network for multiple degradations[C]//IEEE Conference on Computer Vision and Pattern Recognition. 2018,6.

[15] Shocher A, Cohen N, Irani M. "Zero-Shot" Super-Resolution using Deep Internal Learning[C]//Conference on Computer Vision and Pattern Recognition. 2018.

[16] Han W, Chang S, Liu D, et al. Image Super-Resolution via Dual-State Recurrent Networks[C]//Conference on Computer Vision and Pattern Recognition. 2018.