2021 International Conference on Culture-oriented Science & Technology (ICCST) | 978-1-6654-4254-1/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICCST53801.2021.00104

Lightweight Image Super-Resolution via Dual Feature Aggregation Network

Shang Li[1,2], Guixuan Zhang[3], Zhengxiong Luo[1,2], Jie Liu[4], Zhi Zeng[2], Shuwu Zhang[1,2]
[1] School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS)
[2] Institute of Automation, Chinese Academy of Sciences (CASIA)
[3] Key Laboratory of Digital Rights Services (KLDRS)
[4] Beijing Engineering Research Center of Digital Content Technology (BJDCRC)
{lishang2018, guixuan.zhang, jie.liu, zhi.zeng, shuwu.zhang}@ia.ac.cn zhengxiong.luo@cripac.ia.ac.cn

***Abstract*—With the power of deep learning, super-resolution (SR) methods enjoy a dramatic boost of performance. However, they usually have a large model size and high computational complexity, which hinders the application in devices with limited memory and computing power. Some lightweight SR methods solve this issue by directly designing shallower architectures, but it will affect SR performance. In this paper, we propose the dual feature aggregation strategy (DFA). It enhances the feature utilization via feature reuse, which largely improves the representation ability while only introducing marginal computational cost. Thus, a smaller model could achieve better cost-effectiveness with DFA. Specifically, DFA consists of local and global feature aggregation modules (LAM and GAM). They work together to further fuse hierarchical features adaptively along the channel and spatial dimensions. Extensive experiments suggest that the proposed network performs favorably against the state-of-the-art SR methods in terms of visual quality, memory footprint, and computational complexity.**

***Keywords*—Lightweight, Super-Resolution, Dual Feature Aggregation**

I. Introduction

Single image super-resolution (SISR) aims to reconstruct a visually natural high-resolution (HR) image from its low-resolution (LR) counterpart, which is an inherently ill-posed inverse problem. Due to the essential role in video processing, surveillance system, and so on, super-resolution is still an active research area.

Recently, deep learning-based image super-resolution (SR) methods [1], [2], [3], [4] have largely outperformed conventional methods such as Bicubic interpolation. After the proposal of residual learning [5] which simplifies the optimization of deep CNNs, SR networks tend to become even deeper and larger. As Figure 1 shows, these state-of-the-art models focus on improving visual quality measured by PSNR but are memory-consuming and computationally demanding. However, for devices with limited memory and battery capacity, cost-effective methods are preferred, which encourages the design of lightweight SR models. To reduce the number of parameters, some approaches adopt a recursive manner or parameter sharing scheme, such as DRRN [6] and MemNet [7]. However, to compensate for the performance drop, these methods have to increase the network width or depth, thus resulting in high computational complexity (*i.e.*FLOPs). Some other methods directly design shallower network architectures,

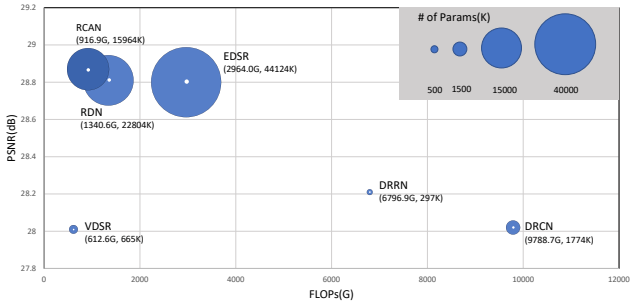

Fig. 1. Trade-off between performance $vs.$ number of operations and parameters on Set14 $\times 4$ dataset. The x-axis and the y-axis denote the FLOPs and PSNR, and the size of the circle represents the number of parameters. The FLOPs is computed by assuming that the resolution of HR image is 720p.

which reduce parameters and calculations simultaneously. For example, CARN [8] and IDN [9] are such compact models less than 40 layers. But their representation ability is restricted by the shallow architecture.

Towards these drawbacks, we propose the dual feature aggregation network (DFAN) that can strike a better trade-off between SR performance and computational cost. The key component of DFAN is the dual feature aggregation strategy (DFA). It could largely improve feature utilization via feature reuse. Specifically, DFA consists of two modules: local and global feature aggregation modules (LAM and GAM). LAM uses an efficient connection method and one convolutional layer to adaptively fuse hierarchical features along the channel dimension. Then GAM further fuses the local aggregated features along the spatial dimension in an iterative manner. This coarse to fine strategy fully leverages all hierarchical features, which enables the lightweight model to achieve better SR performance. In this paper, we also design an efficient convolutional block (ECB) as the basic building block of DFAN. It comprises group convolutional layers with channel shuffle operation. Although ECB is compact, DFAN can still achieve competitive results with the help of DFA.

In summary, our main contributions are as follows:

- We propose the dual feature aggregation network (DFAN), which can achieve better SR performance with limited computational cost. It is more practical in real

978-1-6654-4254-1/21/$31.00 ©2021 IEEE
DOI 10.1109/ICCST53801.2021.00104

applications.

- We propose the dual feature aggregation strategy (DFA) consisting of local and global feature aggregation modules (LAM and GAM). It could make full use of all hierarchical features through feature reuse, which enhances the feature utilization while introducing only marginal computation cost. With DFA, the lightweight SR model can achieve better cost-effectiveness.
- We show through extensive experiments that our model can achieve competitive results with only relatively fewer parameters and calculations.

II. Related Work

Most existing SR methods have hundreds of convolutional layers, such as RDN [10] and SAN [11]. Although promising SR performance has been achieved, these methods are computationally expensive for real application. Thus, more and more lightweight SR methods are proposed. some methods introduce recursive learning or weight sharing schemes to reduce parameters, such as DRRN [6] and MemNet [7]. However, they need to increase the computational complexity to compensate for performance drop. Another idea is to build relatively shallower models, which can cut down the model size and calculations at the same time. CARN [8], IDN [9], and IMDN [12] are all lightweight networks that have less than 40 layers. But the shallow architecture could restrict their representation ability to some extent.

To improve the representation ability in a computationally economical way, deep feature aggregation has gradually attracted more attention. ResNets [5], DenseNets [13], and FPNs [14] are the dominant architectures to fuse hierarchical features for visual tasks. Recently, Yu *et al.*[15] proposed an iterative aggregation method and a hierarchical aggregation method, which can further improve the performance of the aforementioned dominant methods. Inspired by [15], we propose the dual feature aggregation network (DFAN), which can improve the representation ability for SISR.

III. Proposed Method

A. Network Architecture

As depicted in Figure 2 (a), DFAN mainly consists of four parts: shallow feature extraction layer (SFL), stacked local feature aggregation modules (LAMs), the global feature aggregation module (GAM), and the upsample module.

SFL contains only one convolutional layer. It extracts shallow features $\mathbf{F}_0$ from the LR image $\mathbf{I}_{LR}$. Then $\mathbf{F}_0$ is input into the stacked local feature aggregation modules (LAMs) for global residual learning. There are M stacked LAMs in DFAN, and the local aggregated feature from the m^{th} LAM can be formulated as

$$\begin{aligned}\mathbf{F}_m &= f_{LAM}^m(\mathbf{F}_{m-1}) \\ &= f_{LAM}^m(f_{LAM}^{m-1}(...(f_{LAM}^1(\mathbf{F}_0)))),\end{aligned} \tag{1}$$

where f_{LAM}^m refers to the operation of the m^{th} LAM, and $\mathbf{F}_m$ is the local aggregated feature from the m^{th} LAM. As shown in Figure 2, each LAM is composed of a series of efficient convolutional blocks (ECBs), therefore f_{LAM}^m can be viewed as a composite function.

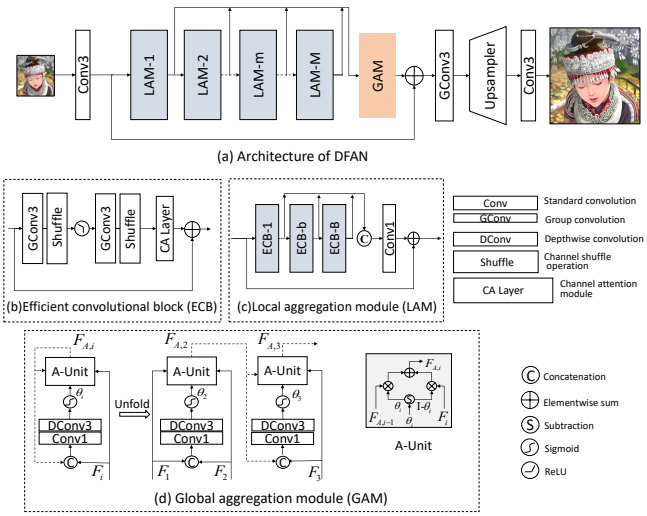

Fig. 2. Framework of DFAN and its submodules.

After that, GAM fully leverages local aggregated features from LAMs in an iterative way, which can be expressed as

$$\mathbf{F}_A = f_{GAM}(\mathbf{F}_1, \mathbf{F}_2, ..., \mathbf{F}_G), \tag{2}$$

where $\mathbf{F}_A$ is the global aggregated feature. f_{GAM} denotes the operation of GAM. Then the global long skip connection adds $\mathbf{F}_0$ to $\mathbf{F}_A$, obtaining the final aggregated feature $\mathbf{F}$. The global skip connection can better propagate information and gradients, thus stabilizing the training of DFAN.

Finally, we use an upscale module proposed in [16] to restore the final SR image $\mathbf{I}_{SR}$. That is,

$$\mathbf{I}_{SR} = f_{conv}(f_{\uparrow}(f_{gconv}(\mathbf{F}))) = f_{DFAN}(\mathbf{I}_{LR}), \tag{3}$$

where f_{gconv} denotes the group convolution, f_{conv} indicates the standard convolution, and $f_{\uparrow}$ is the upscale module.

B. Local Feature Aggregation

Considering that features of different channels contain different weighted information and frequency distribution is uneven on different spatial locations, our dual aggregation feature aggregation (DFA) strategy uses local and global feature aggregation modules (LAM and GAM) to adaptively suppress or emphasize information along the channel and spatial dimensions. In this subsection, we first explain the local feature aggregation.

Efficient convolutional block. As depicted in Figure 2 (c), there are B efficient convolutional blocks (ECBs) in each LAM. ECB is a residual learning module consisting of two group convolutional layers with channel shuffle operation [17], and a channel attention module [4]. It is a basic module with lightweight parameters and computational complexity. LAM

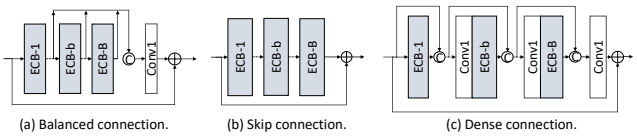

Fig. 3. Comparisons of three different connection methods.

fuses hierarchical features from ECBs by exploring the inter-channel relationship. The local aggregated feature $\mathbf{F}_m$ from the m^{th} LAM can be obtained by

$$\mathbf{F}_m = f_{conv}([\mathbf{F}_m^1, \mathbf{F}_m^2 ..., \mathbf{F}_m^B]) + \mathbf{F}_{m-1}, \tag{4}$$

where $[\mathbf{F}_m^1, \mathbf{F}_m^2 ..., \mathbf{F}_m^B]$ represents the concatenation of local features from ECBs in the m^{th} LAM.

Balanced connection. The connection method in LAM is what we call *balanced connection*. As shown in Figure 3, we compare our balanced connection with two commonly used connection methods in SR, *i.e.*skip connection and dense connection. The analysis is as follows:

1) Difference to skip connection. As shown in Figure 3 (b), for each LAM, if we only use skip connection which makes the element-wise sum of the hierarchical feature maps, all hierarchical features will contribute equally to the final aggregated feature. It may be inflexible since different features contain information of different importance. Our balanced connection can simply solve this issue by a 1×1 convolutional kernel. This convolutional kernel assigns specific learned weights to each pixel of local features, thus adaptively aggregating them along channel dimension.

2) Difference to dense connection. As shown in Figure 3 (c), the dense connection makes each ECB and all preceding ECBs concatenated and compressed as inputs to all subsequent ECBs, which requires more 1×1 convolutional kernels and harms the overall efficiency. However, our balanced connection directly connects each ECB for feature aggregation, which not only fully uses local features but also greatly reduces the parameter number and computation operations.

C. Global Feature Aggregation

The global feature aggregation module (GAM) fuses the local aggregated features of different frequencies with a spatial attention mechanism and an iterative scheme. Specifically, GAM aggregates features along the spatial dimension, orthogonal to channel direction, which could be complementary to LAM. Moreover, referring to abundant local aggregated features, the iterative fusion scheme could progressively refine the global aggregated feature and improve representation capability.

In Figure 2 (d), $\mathbf{F}_{A,i}$ represents the global aggregated feature in the i^{th} iteration, and $\mathbf{F}_i$ represents the output of the i^{th} LAM. The iterative fusion of GAM can be formulated as:

$$\mathbf{F}_{A,i} = \begin{cases} \mathbf{F}_1, & i = 1 \\ f_G(\ [\mathbf{F}_{A,i-1},\ \mathbf{F}_i]\ ), & i > 1. \end{cases} \tag{5}$$

where $\mathbf{F}_{A,i}$ is initialized with $\mathbf{F}_1$, which is the output of the first LAM. [,] denotes the feature concatenation. f_G represents the global aggregation of GAM.

Specifically, the main parts of GAM are spatial attention generation and iterative feature aggregation in A-Unit. Firstly, the spatial attention θ_i is generated by

$$\theta_i = \sigma(f_{dconv}(f_{conv}([\mathbf{F}_{A,i-1},\ \mathbf{F}_i]))), \tag{6}$$

where f_{conv} denotes a 1×1 convolutional kernel that reduces the channel number of $[\mathbf{F}_{A,i}, \mathbf{F}_i]$ by half. f_{dconv} denotes a 3×3 depthwise convolutional kernel to extract spatial information. Depthwise convolution applies a single filter to each input channel, which is more efficient than common convolution in terms of memory and computation. σ is the *Sigmoid* activation function constraining the spatial attention to $(0, 1)$. The spatial attention θ_i is the same size as $\mathbf{F}_i$ and $\mathbf{F}_{A,i-1}$. Second, as shown in Figure 2(d), the feature fusion in A-Unit can be formulated as

$$\mathbf{F}_{A,i} = \theta_i \otimes \mathbf{F}_{A,i-1} + (\mathbf{I} - \theta_i) \otimes \mathbf{F}_i, \tag{7}$$

where $\otimes$ denotes Hadamard product. $\mathbf{I}$ is the tensor with all elements being 1. After the G^{th} iteration, we obtain the final global aggregated feature $\mathbf{F}_A$.

IV. Experiments

A. Experimental Setup

Datasets and metrics. We use the training set of DIV2K [21] to train all of our models. For testing, we use five standard benchmark datasets: Set5, Set14, BSD100, Urban100, and Manga109. The visual quality of SR results is evaluated with PSNR and SSIM on the Y channel (*i.e.*luminance) of transformed YCbCr space. We also represent the number of parameters and multiply-adds (FLOPs) to evaluate memory footprint and computation complexity respectively.

Degradation models. We use the bicubic degradation model (denote as **BI**) to simulate LR images, which simulate LR images on scale $\times 2$, $\times 3$, $\times 4$.

Training details. The size of LR patches is 48×48. During training, we randomly rotate input images by $90°$, $180°$, or $270°$ and flip them horizontally or vertically. The batch size is 32. We use $L1$ loss as the loss function. We use Adam as the optimizer. The initial learning rate is $2e-4$, decayed by half every 200 epochs. We train our model for 1000 epochs.

B. Study on Dual Feature Aggregation

In this section, we experimentally investigate the effectiveness of the dual feature aggregation strategy (DFA). *LAM0_GAM0* is the baseline network by removing balanced connections in LAM and GAM from DFAN, *i.e.*without DFA. *LAM1_GAM0* is built by removing GAM from DFAN. *LAM1_GAM1* (*i.e.*DFAN) has both LAM and GAM. As shown in Table II, when only LAM is added, PSNR is improved by about $0.05dB$. When both LAM and GAM are added, the performance is improved by a large margin (PSNR: $+\mathbf{0.15}dB$, SSIM: $+\mathbf{0.0035}$ on Set14 for $\times 4$ SR). It indicates that

TABLE I
QUANTITATIVE RESULTS WITH BI DEGRADATION MODEL. RED: BEST.

Method	Scale	Params (K)	FLOPs (G)	Set5		Set14		BSD100		Urban100		Manga109	
				PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	×2	-	-	33.66	0.9299	30.24	0.8688	29.56	0.8431	26.88	0.8403	30.80	0.9339
SRCNN [1]	×2	57	52.7	36.66	0.9542	32.42	0.9063	31.36	0.8879	29.50	0.8946	35.60	0.9663
FSRCNN [18]	×2	12	6.0	37.00	0.9558	32.63	0.9088	31.53	0.8920	29.88	0.9020	36.67	0.9710
VDSR [2]	×2	665	612.6	37.53	0.9587	33.03	0.9124	31.90	0.8960	30.76	0.9140	37.22	0.9750
DRCN [19]	×2	1774	9788.7	37.63	0.9588	33.04	0.9118	31.85	0.8942	30.75	0.9133	37.55	0.9732
LapSRN [20]	×2	813	29.9	37.52	0.9590	33.08	0.9130	31.80	0.8950	30.41	0.9100	37.27	0.9740
DRRN [6]	×2	297	6796.9	37.74	0.9591	33.23	0.9136	32.05	0.8973	31.23	0.9188	37.88	0.9749
MemNet [7]	×2	677	2262.4	37.78	0.9597	33.28	0.9142	32.08	0.8978	31.31	0.9195	37.72	0.9740
IDN [9]	×2	677	168.4	37.83	0.9600	33.30	0.9148	32.08	0.8985	31.27	0.9196	38.01	0.9749
IMDN [12]	×2	694	158.8	38.00	0.9605	33.63	0.9177	32.19	0.8996	32.17	0.9283	38.88	0.9774
CARN [8]	×2	1592	222.8	37.76	0.9590	33.52	0.9166	32.09	0.8978	31.51	0.9312	38.36	0.9765
DFAN	×2	877	201.4	38.11	0.9609	33.74	0.9189	32.21	0.9002	32.36	0.9305	38.93	0.9774
Bicubic	×3	-	-	30.39	0.8682	27.55	0.7742	27.21	0.7385	24.46	0.7349	26.95	0.8556
SRCNN [1]	×3	57	52.7	32.75	0.9090	29.28	0.8209	28.41	0.7863	26.24	0.7989	30.48	0.9117
FSRCNN [18]	×3	12	5.0	33.16	0.9140	29.43	0.8242	28.53	0.7910	26.43	0.8080	31.10	0.9210
VDSR [2]	×3	665	612.6	33.66	0.9213	29.77	0.8314	28.82	0.7976	27.14	0.8279	32.01	0.9340
DRCN [19]	×3	1774	9788.7	33.82	0.9226	29.76	0.8311	28.80	0.7963	27.15	0.8276	32.24	0.9343
DRRN [6]	×3	297	6796.9	34.03	0.9244	29.96	0.8349	28.95	0.8004	27.53	0.8378	32.71	0.9379
MemNet [7]	×3	677	2262.4	34.09	0.9248	30.00	0.8350	38.96	0.8001	27.56	0.8376	32.51	0.9369
IDN [9]	×3	677	84.4	34.11	0.9253	29.99	0.8354	28.95	0.8031	27.42	0.8359	32.71	0.9381
IMDN [12]	×3	703	71.5	34.36	0.9270	30.32	0.8417	29.09	0.8046	28.17	0.8519	33.61	0.9445
CARN [8]	×3	1592	118.8	34.29	0.9255	30.29	0.8407	29.06	0.8034	27.38	0.8404	33.50	0.9440
DFAN	×3	900	92.6	34.49	0.9280	30.37	0.8426	29.13	0.8059	28.31	0.8551	33.78	0.9458
Bicubic	×4	-	-	28.42	0.8104	26.00	0.7027	25.96	0.6675	23.14	0.6577	24.89	0.7866
SRCNN [1]	×4	57	52.7	30.48	0.8628	27.49	0.7503	26.90	0.7101	24.52	0.7221	27.58	0.8555
FSRCNN [18]	×4	12	4.6	30.71	0.8657	27.59	0.7535	26.98	0.7150	24.62	0.7280	27.90	0.8610
VDSR [2]	×4	665	612.6	31.35	0.8838	28.01	0.7674	27.29	0.7251	25.18	0.7524	28.83	0.8870
DRCN [19]	×4	1774	9788.7	31.53	0.8854	28.02	0.7670	27.23	0.7233	25.14	0.7510	28.93	0.8854
LapSRN [20]	×4	813	149.4	31.54	0.8850	28.19	0.7720	27.32	0.7280	25.44	0.7638	29.09	0.8900
DRRN [6]	×4	297	6796.9	31.68	0.8888	28.21	0.7720	27.38	0.7284	25.44	0.7638	29.45	0.8946
MemNet [7]	×4	677	2262.4	31.74	0.8893	28.26	0.7723	27.40	0.7281	25.50	0.7630	29.42	0.8942
IDN [9]	×4	677	54.9	31.82	0.8903	28.25	0.7730	27.41	0.7297	25.41	0.7632	29.41	0.8942
IMDN [12]	×4	715	40.9	32.21	0.8948	28.58	0.7811	27.56	0.7353	26.04	0.7838	30.45	0.9075
CARN [8]	×4	1592	90.9	32.13	0.8937	28.60	0.7806	27.58	0.7349	26.07	0.7837	30.47	0.9084
DFAN	×4	896	55.8	32.30	0.8964	28.65	0.7832	27.60	0.7373	26.22	0.7898	30.59	0.9097

the DFA can efficiently utilize hierarchical features through feature reusing.

C. Results with Bicubic Degradation Model

We compare DFAN with other state-of-the-art methods: SRCNN [1], FSRCNN [18], VDSR [2], DRCN [19], DRRN [6], MemNet [7], CARN [8], IDN [9] and IMDN [12].

Quantitative Results. We evaluate the average PSNR and SSIM on five benchmark datasets. In particular, we also calculate the parameter amount and FLOPs (*i.e.*multiply-adds) of these models by assuming the HR image size to be 720p (1280×720). In Table I, the proposed DFAN performs favorably against these methods on all benchmark datasets for $2\times$, $3\times$ and $4\times$ SR respectively. CARN [8] used to be a strong baseline for lightweight SR models, but our DFAN outperforms it by a large margin (PSNR:$+0.17dB$, SSIM:+0.0024 on Set5) with 42% fewer parameters and 37% fewer FLOPs on scale $\times 4$. It indicates our method can achieve a better trade-off between computational cost and effectiveness. Therefore, feature aggregation has a promising prospect in the research of lightweight image SR.

Visual Results. In Figure 4, we show visual comparisons on scale $\times 4$. Our method restores the letter "*g*" in "*ppt3*" more clearly, while most other methods encounter artifacts or edge distortion. For "*img030*" in Urban100, most methods don't reconstruct the contour of the window well, but our method can reconstruct these edges better.

TABLE II
STUDY ON DFA. PSNR FOR $4\times$ SR. "LA" DENOTES LOCAL FEATURE AGGREGATION, AND 'GA' DENOTES GLOBAL FEATURE AGGREGATION.

Method	LA	GA	Set5	Set14	BSD100	Urban100
LAM0_GAM0	-	-	32.11	28.50	27.52	25.96
LAM1_GAM0	✓	-	32.16	28.57	27.57	26.11
LAM1_GAM1	✓	✓	**32.29**	**28.65**	**27.60**	**26.22**

V. CONCLUSIONS

We propose the dual feature aggregation network (DFAN) that can strike a better trade-off between SR performance and computational cost. The proposed dual feature aggregation strategy (DFA) makes local and global feature aggregations adaptively. Through feature reuse, it could improve feature utilization and representation ability. Benefitting from DFA, our network achieves competitive performances with fewer

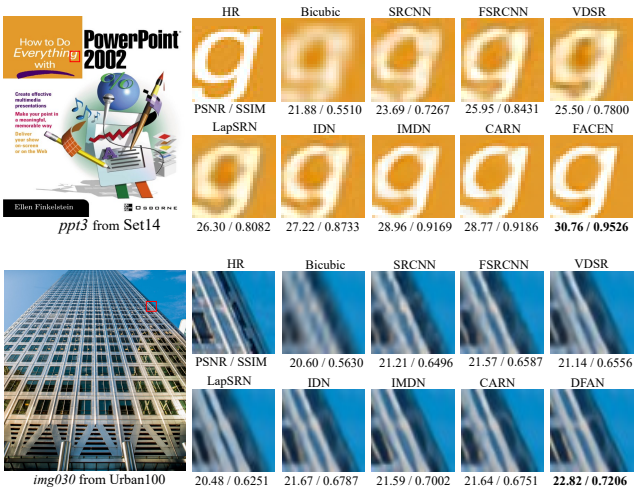

Fig. 4. Visual results for $4\times$ SR with BI model.

parameters and lower computational complexity, which is more practical for real applications.

ACKNOWLEDGMENT

This work was supported by the National Key R&D Program of China (2019YFB1406200). It was also the research achievement of the Key Laboratory of Digital Rights Services.

REFERENCES

[1] 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*, vol. 38, no. 2, pp. 295–307, 2015.

[2] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1646–1654.

[3] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 136–144.

[4] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 286–301.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[6] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3147–3155.

[7] Y. Tai, J. Yang, X. Liu, and C. Xu, "Memnet: A persistent memory network for image restoration," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 4539–4547.

[8] N. Ahn, B. Kang, and K.-A. Sohn, "Fast, accurate, and lightweight super-resolution with cascading residual network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 252–268.

[9] Z. Hui, X. Wang, and X. Gao, "Fast and accurate single image super-resolution via information distillation network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 723–731.

[10] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2472–2481.

[11] T. Dai, J. Cai, Y.-B. Zhang, S. Xia, and L. Zhang, "Second-order attention network for single image super-resolution," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11 057–11 066, 2019.

[12] Z. Hui, X. Gao, Y. Yang, and X. Wang, "Lightweight image super-resolution with information multi-distillation network," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 2024–2032.

[13] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[14] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[15] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2403–2412.

[16] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1874–1883.

[17] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6848–6856.

[18] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *European conference on computer vision*. Springer, 2016, pp. 391–407.

[19] J. Kim, J. Kwon Lee, and K. Mu Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1637–1645.

[20] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep laplacian pyramid networks for fast and accurate super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 624–632.

[21] E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 126–135.