Non-Global Shared Recursive Network for Image Super-Resolution

Bing Zheng and Lihong Ma*
School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China

*E-mail: eelhma@scut.edu.cn

Abstract. In Super-Resolution(SR) reconstruction, some recursive networks use shared parameters to keep lightweight. But these methods lack flexibility while use shared parameters to process different feature maps. We propose a non-global shared recursive network that consists of recursive units and channel attention modules. Firstly, we propose multi-scale channel attention module to learn the inter-channel interdependence and spatial information within the channel. The multi-scale channel attention module is composed of a multi-scale pooling layer, fully connected layers and activation functions. In order to flexibly process different feature maps in different layers, the attention module parameters are not shared, but only a small number of parameters are used. Secondly, residual dense recursive unit is used to accelerate convergence and feature reuse. In order to further reduce parameters, bottleneck layer is added to the dense connection. Based on the above mechanism, we propose a lightweight and efficient network structure. Experiments show that the proposed method has better performance than other lightweight networks in the standard dataset.

1. Introduction
SR reconstruction is a computer vision task where the aim is to reconstruct high-resolution (HR) images from low-resolution (LR) images. We focus on single-image super-resolution reconstruction (SISR).

In SR, SRCNN[1] firstly used a three-layer convolutional neural network for nonlinear mapping. With the improvement of the network model, more and more accurate SR network models have been proposed, such as EDSR[2], but the parameters used exceed 40,000k. These models are too large to be applied to the actual scene. Some models use recursive networks to reduce parameters. DRCN[3] uses recursive network that shared parameters to reduce redundancy parameters, DRRN[4] adds residual connection in recursive to get more accurate results. These models effectively reduce the number of model parameters and show good performance compared to standard CNN. However, there are two drawbacks: (1) The recursive unit equally handles the feature map of different layers. These CNN-based methods equally handle channel features that lack flexibility in handling different types of information, such as low frequency and high frequency information. To increase flexibility, RCAN [5] uses the channel attention (CA) module to learn the interdependences between feature channels and rescale each channel feature. This channel attention module makes the network to focus on more useful information and enhance discernment learning. However, the channel attention mechanism used by RCAN does not make full use of the spatial information of the feature. It uses global pooling when learning the interdependence between feature channels and loses spatial information. (2) The recursive
unit use simple network structure. Such a structure does not make full use of the previously extracted features, and the network is inefficient. DenseNet show superior performance in the classification task. DenseNet feed the feature map of each layer into all subsequent layers in a dense block, so that the features of all layers are concatenated. The structure brings the advantages of mitigating gradient disappearance, enhancing feature propagation, supporting feature reuse, and reducing the number of parameters for the entire network.

Therefore, we propose: (1) Using multi-scale channel attention module in the recursive network, so that the recursive network can adaptively process the features of each channel. (2) Residual dense block is used as the basic unit of recursive network and add bottleneck layer in dense connections to reduce the number of parameters. (3) A non-global shared recursive network that is lightweight and efficient.

2. Related Work

2.1. Deep CNN for SR
Dong et al.[1] firstly proposed SRCNN, using only three-layer convolutional neural networks, corresponding to the main stages of the traditional SR method. Kim et al. [6] proposed VDSR, using a deeper network to obtain a larger receptive field, and adding global residual learning. Mao et al. [7] proposed RED composed of a symmetric convolution-deconvolution layer. The convolutional layer is used to capture the abstract content of the image, and the deconvolution layer is used to magnify the feature size and restore the image details. Lai et al. [8] proposed the Laplacian Pyramid Super Resolution Network (LapSRN) to reconstructs the HR image progressively. Tai et al. [9] proposed a persistent memory network (MemNet) to solve the long-term dependence problem in the previous CNN architecture. The SRDenseNet [10] and RDN [11] utilize the dense block structure of the DenseNet, which brings the advantages of mitigating gradient disappearance, enhancing feature propagation, supporting feature reuse, and reducing the number of parameters. Bee et al. [2] proposed EDSR to use the residual learning method to remove the batch normalization from the original ResNet, because the structure of original ResNet is not suitable for SR task.

2.2. Lightweight Network
There are some methods that try to build lightweight and accurate neural networks. DRCN reduces redundant parameters by using recursive networks while increasing receptive fields. DRRN improves DRCN by combining recursive networks and residual networks. CARN [12] proposes efficient residual blocks, which improve efficiency through group convolution and 1x1 convolution. CARN also uses recursive network to reduce model parameters.

3. The proposed method
In order to solve the problem of insufficient flexibility in recursive networks and improve the efficiency of recursive unit, we propose a non-global shared recursive network with multi-scale channel attention mechanism.

3.1. Network Architecture

![Figure 1. The architecture of our proposed method](image-url)
The network architecture is shown in Figure 1. $I_{LR}$ and $I_{SR}$ represent the input image and output image. The overall network consists three parts: shallow feature extraction network, non-global shared recursive network and upscale network. The shallow feature extraction network is used to receive low-resolution images and extract shallow features. It consists of a convolution layer. $H_{SFN}(\cdot)$ is used to represent the shallow feature extraction function, then the shallow feature $F_0$ is:

$$F_0 = H_{SFN}(I_{LR})$$  \hspace{1cm} (1)

The next step is the non-global shared recursive network, which is the core module for extracting advanced features. The non-global shared recursive network is shown in Figure 2, the orange RDB (residual dense block) indicates that these parameters of orange module are shared, while other modules do not share parameters. $H_{RDB}(\cdot)$ is denote the RDB module and $H_{MCA_n}(\cdot)$ denote the MCA (multi-scale channel attention) module, and the output $F_n$ of recursive unit is:

$$F_n = H_{MCA_n}(H_{RDB}(F_{n-1}))$$  \hspace{1cm} (2)

$$= H_{MCA_n}(H_{RDB}(...H_{MCA_1}(H_{RDB}(F_0))...))$$  \hspace{1cm} (3)

After extracting the advanced features by non-global shared recursive network, global feature fusion and global residual learning are performed, and finally, a high-resolution image can be obtained by upscale network.

3.2. Non-global Shared Recursive Network

![Figure 2. Recursive network of DRCN, DRRN and ours](image)

The non-global shared recursive network is shown in Figure 2, and the parameters of orange module are shared. The basic recursive unit of DRCN is the simple convolutional layer. The Recursive unit of DRRN is residual block (RB), residual connection makes the network faster to converge. The last one is our non-global shared recursive network. There are two differences from the other networks. The first difference is that a multi-scale channel attention (MCA) module is added after each recursive unit. The MCA exploit the interdependencies among feature channels and spatial information, then focus on more useful features and improve the reconstruction result. The MCA does not share the parameters, but only a small number of parameters are used; The second difference is that we use residual connection and dense connection in the basic recursive unit to form the residual dense block RDB. We use the bottleneck layer that consist of a 1x1 convolution layer. It reduces the number of output channels and further reduce overall parameters.
3.3. Multi-scale Channel Attention

The multi-scale channel attention module is shown in Figure 3. It makes the network focus on more informative features by exploiting the interdependencies among feature channels. Different from the channel attention module used in [5], we use multi-scale pooling to make full use of spatial information to improve the discriminating ability. The pooled result \( P_d \) can be formulated as

\[
P_d = [H_{pooling,1}(F_{d,RDB}), H_{pooling,2}(F_{d,RDB}), H_{pooling,3}(F_{d,RDB})]
\]  

(4)

Next, our network should learn the interdependence between channels from the pooling results. Same as [6], we use the fully connected layer and the ReLU layer to learn the relationship between channels and use the Sigmoid function to convert to the corresponding weights. The weight \( S_d \) can be formulated as

\[
S_d = f(W_{d,2} \delta(W_{d,1} P_d))
\]  

(5)

Through the multi-scale channel attention module, the output \( F_{d,RDB} \) of the RDB can be adaptively adjusted, and the final output \( F_d \) is

\[
F_d = S_d \cdot F_{d,RDB}
\]  

(6)

**Figure 3.** The architecture of multi-scale channel attention module

4. Experiments

4.1. Datasets

We use the DIV2K dataset for training our non-global shared recursive network and implement our models by PyTorch. For testing, we use the standard test dataset, including Set5, Set14, B100 and Urban100.

4.2. Experiments results

To demonstrate the effect of our non-global shared recursive network, we conducted the following experiment: (1) Effectiveness of non-global shared recursive network with multi-scale channel attention module and Visual display; (2) Effectiveness of residual dense block; (3) Comparison with other lightweight networks.

4.2.1. Non-global shared recursive network with multi-scale channel attention. In order to directly demonstrate the effect of our proposed method, we visually display the results of the intermediate features of the network. The effect is shown in Figure 4. The upper left image of the figure is the input LR image, the other is the output feature map of RDB module, and the weights obtained by the multi-scale channel attention module are shown below. As can be seen that the more information the feature map contains, the greater weight it gets. Some feature maps contain little useful information, so the multi-scale channel attention module regards it as a redundant result and produce a low weight. It can
be seen from the visualization results that the multi-scale channel attention module can distinguish whether the feature map contains useful information and adjust it.

To illustrate the effectiveness of our non-global shared recursive network, we evaluated the network without channel attention and with different channel attention, as shown in Table 1.

![LR image] 0.015 0.990

0 1 0.475

**Figure 4.** Visual display of feature maps and weights (under the corresponding feature map), the more information the feature map contains, the greater weight it gets.

**Table 1.** The performance of different attention module.

|                        | No channel attention | Single-scale channel attention | Multi-scale channel attention |
|------------------------|----------------------|-------------------------------|------------------------------|
| Parameters             | 615k                 | 627k                          | 682k                         |
| PSNR                   | 33.47                | 33.48                         | 33.55                        |

4.2.2. **Residual dense block.** We remove residual connection or/dense connection from our recursive unit and follow the principle that the total number of parameters is constant. Table 2 shows the PSNR on Set14 dataset of each model. As can be seen, both the residual connection and the dense connection can improve the performance of the model.

**Table 2.** The performance of different recursive unit (Set14 x2) with the same number of parameters.

|                        | ours                  | No residual connection | No dense connection | No residual and dense connection |
|------------------------|-----------------------|------------------------|---------------------|----------------------------------|
| PSNR                   | 33.55                 | 33.42                  | 33.41               | 33.23                            |

4.2.3. **Comparison with other lightweight network.** We compare our non-global shared recursive network with other lightweight on number of parameters and two metrics: PSNR and SSIM. The result is shown in Table 3.

5. **Conclusions**  
We propose non-global shared recursive network for accurate and efficient image SR. We use multi-scale channel attention module after recursive unit to adaptive rescale features by considering interdependencies among channels and spatial information. Furthermore, to improve ability of
recursive unit, we use residual dense block as recursive unit. Our experiments show that ours outperform other lightweight networks.

| scale  | parameters | Set5 PSNR | Set5 SSIM | Set14 PSNR | Set14 SSIM | B100 PSNR | B100 SSIM | Urban100 PSNR | Urban100 SSIM |
|--------|------------|-----------|-----------|------------|------------|-----------|-----------|--------------|--------------|
| DRCN   | 1774K      | 37.63     | 0.9588    | 33.04      | 0.9118     | 31.85     | 0.8942    | 30.75        | 0.9133       |
| DRRN   | 297K       | 37.74     | 0.9591    | 33.23      | 0.9136     | 32.05     | 0.8973    | 31.23        | 0.9188       |
| LapSRN | 2          | 813K      | 37.52     | 0.9590    | 33.08      | 0.9130    | 31.80     | 0.8950       | 30.41        |
| MemNet | 667K       | 37.78     | 0.9597    | 33.28      | 0.9142     | 32.08     | 0.8978    | 31.31        | 0.9195       |
| Ours   | 682K       | 37.98     | 0.9594    | 33.55      | 0.9165     | 32.12     | 0.8979    | 31.99        | 0.9254       |

| scale  | parameters | Set5 PSNR | Set5 SSIM | Set14 PSNR | Set14 SSIM | B100 PSNR | B100 SSIM | Urban100 PSNR | Urban100 SSIM |
|--------|------------|-----------|-----------|------------|------------|-----------|-----------|--------------|--------------|
| DRCN   | 1774K      | 33.82     | 0.9226    | 29.76      | 0.8311     | 28.80     | 0.7963    | 27.15        | 0.8276       |
| DRRN   | 297K       | 34.03     | 0.9244    | 29.96      | 0.8349     | 28.95     | 0.8004    | 27.53        | 0.8378       |
| LapSRN | 3          | 813K      | 33.81     | 0.9220    | 29.79      | 0.8325    | 28.82     | 0.7980       | 27.07        |
| MemNet | 667K       | 34.09     | 0.9248    | 30.00      | 0.8350     | 28.96     | 0.8001    | 27.56        | 0.8376       |
| Ours   | 703K       | 34.43     | 0.9261    | 30.33      | 0.8403     | 29.07     | 0.8032    | 28.09        | 0.8490       |

| scale  | parameters | Set5 PSNR | Set5 SSIM | Set14 PSNR | Set14 SSIM | B100 PSNR | B100 SSIM | Urban100 PSNR | Urban100 SSIM |
|--------|------------|-----------|-----------|------------|------------|-----------|-----------|--------------|--------------|
| DRCN   | 1774K      | 31.53     | 0.8854    | 28.02      | 0.7670     | 27.23     | 0.7233    | 25.14        | 0.7510       |
| DRRN   | 297K       | 31.68     | 0.8888    | 28.21      | 0.7720     | 27.38     | 0.7284    | 25.44        | 0.7638       |
| LapSRN | 4          | 813K      | 31.54     | 0.8850    | 28.19      | 0.7720    | 27.32     | 0.7280       | 25.21        |
| MemNet | 677K       | 31.74     | 0.8893    | 28.26      | 0.7723     | 27.40     | 0.7281    | 25.50        | 0.7630       |
| Ours   | 698K       | 32.23     | 0.8940    | 28.61      | 0.7810     | 27.56     | 0.7344    | 25.98        | 0.7808       |

**Acknowledgment**

This work is supported in part by the National Natural Science Founding of China (61471173), and The Natural Science Foundation of Guangdong Province (2017A30311028 and 2019A1515011949).

**References**

[1] Dong C., Loy C.C., He K and Tang X 2014 Image super-resolution using deep convolutional networks *IEEE Trans Pattern Anal Mach Intell.* 38(2), 295-307.

[2] Lim B, Son S and Kim H 2017 Enhanced deep residual networks for single image super-resolution *Computer Vision and Pattern Recognition* pp 450–462.

[3] Kim J, Lee J K and Lee K M 2016 Deeply-Recursive Convolutional Network for Image Super-Resolution *Computer Vision and Pattern Recognition* pp1637-1645.

[4] Tai Y, Yang J and Liu X 2017 Image Super-Resolution via Deep Recursive Residual Network *Computer Vision and Pattern Recognition* pp 2790-2798.

[5] Zhang Y, Li K and Li K 2018 Image Super-Resolution Using Very Deep Residual Channel Attention Networks *European Conf. on Computer Vision* pp 1–16.

[6] Kim J, Kwon J and Mu K 2016 Accurate image super-resolution using very deep convolutional networks *Computer Vision and Pattern Recognition* pp 1646–1654.

[7] Mao X J, Shen C and Yang Y B 2016 Image Restoration Using Convolutional Auto-encoders with Symmetric Skip Connections *Neural Information Processing Systems* pp 2802-2810.

[8] Lai W S, Huang J B and Ahuja N 2017 Fast and Accurate Image Super-Resolution with Deep Laplacian Pyramid Networks *Computer Vision and Pattern Recognition* pp 624-632.

[9] Tai Y, Yang J and Liu X 2017 MemNet: A Persistent Memory Network for Image Restoration *Int. Conf. on Computer Vision* pp 4549–4557.

[10] Tongn T, Li G, Liu X and Gao Q 2017 Image super-resolution using dense skip connections *Int. Conf. on Computer Vision* pp 4809–4817.

[11] Zhang Y, Tian Y and Fu Y 2018 Residual dense network for image super-resolution *Computer Vision and Pattern Recognition* pp 2472-2481.

[12] Ahn N, Kang B and Sohn K A 20118 Fast, Accurate, and Lightweight Super-Resolution with Cascading Residual Network *European Conf. on Computer Vision* pp 256-272.