




## Article

# URNet: A U-Shaped Residual Network for Lightweight Image Super-Resolution

Yuntao Wang <sup>1</sup>, Lin Zhao <sup>2</sup>, Liman Liu <sup>1,\*</sup>, Huaifei Hu <sup>1</sup> and Wenbing Tao <sup>2</sup>

<sup>1</sup> School of Biomedical Engineering, South-Central University for Nationalities, Wuhan 430074, China; ytao-wang@scuec.edu.cn (Y.W.); huaifeihu@mail.scuec.edu.cn (H.H.)

<sup>2</sup> National Key Laboratory of Science and Technology on Multi-Spectral Information Processing, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China; linzhao@hust.edu.cn (L.Z.); wenbingtao@hust.edu.cn (W.T.)

\* Correspondence: limanliu@mail.scuec.edu.cn

**Abstract:** It is extremely important and necessary for low computing power or portable devices to design more lightweight algorithms for image super-resolution (SR). Recently, most SR methods have achieved outstanding performance by sacrificing computational cost and memory storage, or vice versa. To address this problem, we introduce a lightweight U-shaped residual network (URNet) for fast and accurate image SR. Specifically, we propose a more effective feature distillation pyramid residual group (FDPRG) to extract features from low-resolution images. The FDPRG can effectively reuse the learned features with dense shortcuts and capture multi-scale information with a cascaded feature pyramid block. Based on the U-shaped structure, we utilize a step-by-step fusion strategy to improve the performance of feature fusion of different blocks. This strategy is different from the general SR methods which only use a single *Concat* operation to fuse the features of all basic blocks. Moreover, a lightweight asymmetric residual non-local block is proposed to model the global context information and further improve the performance of SR. Finally, a high-frequency loss function is designed to alleviate smoothing image details caused by pixel-wise loss. Simultaneously, the proposed modules and high-frequency loss function can be easily plugged into multiple mature architectures to improve the performance of SR. Extensive experiments on multiple natural image datasets and remote sensing image datasets show the URNet achieves a better trade-off between image SR performance and model complexity against other state-of-the-art SR methods.



**Citation:** Wang, Y.; Zhao, L.; Liu, L.; Hu, H.; Tao, W. URNet: A U-Shaped Residual Network for Lightweight Image Super-Resolution. *Remote Sens.* **2021**, *13*, 3848. <https://doi.org/10.3390/rs13193848>

Academic Editors: Wanshou Jiang, San Jiang and Xiongwu Xiao

Received: 9 July 2021

Accepted: 21 September 2021

Published: 26 September 2021

**Keywords:** single image super-resolution; lightweight image super-resolution; U-shaped residual network; dense shortcut; effective feature distillation; high-frequency loss

## 1. Introduction

Single image super-resolution (SISR) aims to reconstruct a high-resolution (HR) image from its low-resolution (LR) image. It has a wide range of applications in real scenes, such as medical imaging [1–3], video surveillance [4], remote sensing [5–7], high-definition display and imaging [8], super-resolution mapping [9], hyper-spectral images [10,11], iris recognition [12], and sign and number plate reading [13]. In general, this problem is inherently ill-posed because many HR images can be downsampled to an identical LR image. To address this problem, numerous super-resolution (SR) methods are proposed, including early traditional methods [14–17] and recent learning-based methods [18–20]. Traditional methods include interpolation-based methods and regularization-based methods. Early interpolation methods such as bicubic interpolation are based on sampling theory but often produce blurry results with aliasing artifacts in natural images. Therefore, some regularization-based algorithms use machine learning to improve the performance of SR, mainly including projection onto convex sets (POCS) methods and maximum a posteriori (MAP) methods. Patti and Altunbasak [15] consider a scheme to utilize a constraint to represent the prior belief about the structure of the recovered high-resolution image.



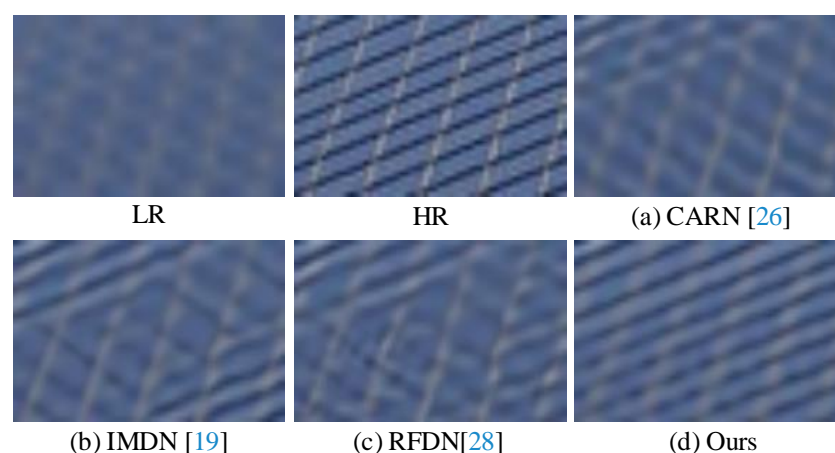
**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

The POCS method assumes that each LR image imposes prior knowledge on the final solution. Later work by Hardie et al. [17] uses the L2 norm of a Laplacian-style filter over the super-resolution image to regularize their MAP reconstruction.

Recently, a great number of convolutional neural network-based methods have been proposed to address the image SR problem. As a pioneering work, Dong et al. [21,22] propose a three-layer network (SRCNN) to learn the mapping function from an LR image to an HR image. Some methods focus mainly on designing a deeper or wider model to further improve the performance of SR, e.g., VDSR [23], DRCN [24], EDSR [25], and RCAN [18]. Although these methods achieve satisfactory results, the increase in model size and computational complexity limits their applications in the real world.

To reduce the computational burden or memory consumption, CARN-M [26] proposes a cascading network architecture for mobile devices, but the performance of this method significantly drops. IDN [27] aggregates current information with partially retained local short-path information by an information distillation network. IMDN [19] designs an information multi-distillation block to further improve the performance of IDN. RFDN [28] proposes a more lightweight and flexible residual feature distillation network. However, these methods are not lightweight enough and the performance of image SR can still be further improved. To build a faster and more lightweight SR model, we first propose a lightweight feature distillation pyramid residual group (FDPRG). Based on the enhanced residual feature distillation block (E-RFDB) of E-RFDN [28], the FDPRG is designed by introducing a dense shortcut (DS) connection and a cascaded feature pyramid block (CFPB). Thus, the FDPRG can effectively reuse the learned feature with DS and capture multi-scale information with CFPB. Furthermore, we propose a lightweight asymmetric residual non-local block (ANRB) to capture the global context information and further improve the SISR performance. The ANRB is modified from ANB [29] by redesigning the convolution layers and adding a residual shortcut connection. It can not only capture non-local contextual information but also become a lightweight block benefitting from residual learning. Combined with the FDPRG, ANRB, and E-RFDB, we build a more powerful lightweight U-shaped residual network (URNet) for fast and accurate image SR by using a step-by-step fusion strategy.

In the image SR field, L1 loss (i.e., mean absolute error) and L2 loss (i.e., mean square error) are usually used to measure the pixel-wise difference between the super-resolved image and its ground truth. However, using only pixel-wise loss will often cause the results to lack high-frequency details and be perceptually unsatisfying with over-smooth textures, as depicted in Figure 1. Subsequently, content loss [30], texture loss [8], adversarial loss [31], and cycle consistency loss [32] are proposed to address this problem. In particular, the content loss transfers the learned knowledge of hierarchical image features from a classification network to the SR network. For the texture loss, it is still empirical to determine the patch size to match textures. For the adversarial loss and cycle consistency loss, the training process of generative adversarial nets (GANs) is still difficult and unstable. In this work, we propose a simple but effective high-frequency loss to alleviate the problem of over-smoothed super-resolved images. Specifically, we first extract the detailed information from the ground truth by using an edge detection algorithm (e.g., Canny). Our model also predicts a response map of detail texture. The mean square error between the response map and detail information is taken as our high-frequency loss, which makes our network pay more attention to detailed textures.



**Figure 1.** Visual results for  $\times 3$  SR on “img074” from Urban100. Our method obtains better visual quality than other SR methods.

The main contributions of this work can be summarized as follows:

- (1) We propose a lightweight feature distillation pyramid residual group to better capture the multi-scale information and reconstruct the high-frequency detailed information of the image.
- (2) We propose a lightweight asymmetric residual non-local block to capture the global contextual information and further improve the performance of SISR.
- (3) We design a simple but effective high-frequency loss function to alleviate the problem of over-smoothed super-resolved images. Extensive experiments on multi-benchmark datasets demonstrate the superiority and effectiveness of our method in SISR tasks. It is worth mentioning that our designed modules and loss function can be combined with the numerous advancements in the image SR methods presented in the literature.

## 2. Related Work

In previous works, methods of image SR can be roughly divided into two categories: traditional methods [17,33,34] and deep learning-based methods [18,19,35,36]. Due to the limitation of space, we only briefly review the works related to deep learning networks for single image super-resolution, attention mechanism, and perceptual optimization.

### 2.1. Single Image Super-Resolution

The SRCNN [22] is one of the first pioneering works of directly applying deep learning to image SR. The SRCNN uses three convolution layers to map LR images to HR images. Inspired by this pioneering work, VDSR [23] and DRCN [24] stack more than 16 convolution layers based on residual learning to further improve the performance. To further unleash the power of the deep convolutional networks, EDSR [25] integrates the modified residual blocks into the SR framework to form a very deep and wide network. MemNet [37] and RDN [38] stack dense blocks to form a deep model and utilize all the hierarchical features from all the convolutional layers. SRFBN [39] proposes a feedback mechanism to generate effective high-level feature representations. EBRN [40] handles the texture SR with an incremental recovery process. Although these methods achieve significant performance, they are costly in memory consumption and computational complexity, limiting their applications in resource-constrained devices.

Recently, some fast and lightweight SISR architectures have been introduced to tackle image SR. These methods can be approximately divided into three categories: the knowledge distillation-based methods [19,27,28], the neural architecture search-based methods [41,42], and the model design-based methods [26,43]. Knowledge distillation aims to transfer the knowledge from a teacher network to a student network. IDN [27] proposes an information distillation network for better exploiting hierarchical features by separation

processing of the current feature maps. Based on IDN, an information multi-distillation network (IMDN) [19] is proposed by constructing cascaded information multi-distillation blocks. RFDN [28] uses multiple feature distillation connections to learn more discriminative feature representations. FALSr [41] and MoreMNAS [42] apply neural architecture search to image SR. The performance of these methods is limited because of limitations in strategy. In addition, CARN [26] proposes a cascading mechanism based on a residual network to boost performance. LatticeNet [43] proposes a lattice block in which two butterfly structures are applied to combine two residual blocks. These works indicate that the lightweight SR networks can maintain a good trade-off between performance and model complexity.

## 2.2. Attention Mechanism

The attention mechanism is an important technique which has been widely used in various vision tasks (e.g., classification, object detection, and image segmentation). SENet [44] models channel-wise relationships to enhance the representational ability of the network. Non-Local [45] captures long-range dependencies by computing the response at a pixel position as a weighted sum of the features at all positions of an image. In the image SR domain, RCAN [18] and NLRN [46] improve the performance by considering attention mechanisms in the channel or the spatial dimension. SAN [35] proposes a second-order attention mechanism to enhance feature expression and correlation learning. CS-NL [47] proposes a cross-scale non-local attention module by exploring cross-scale feature correlations. HAN [48] models the holistic interdependencies among layers, channels, and positions. Due to the effectiveness of attention models, we also embed the attention mechanism into our framework to refine the high-level feature representations.

## 2.3. Perceptual Optimization

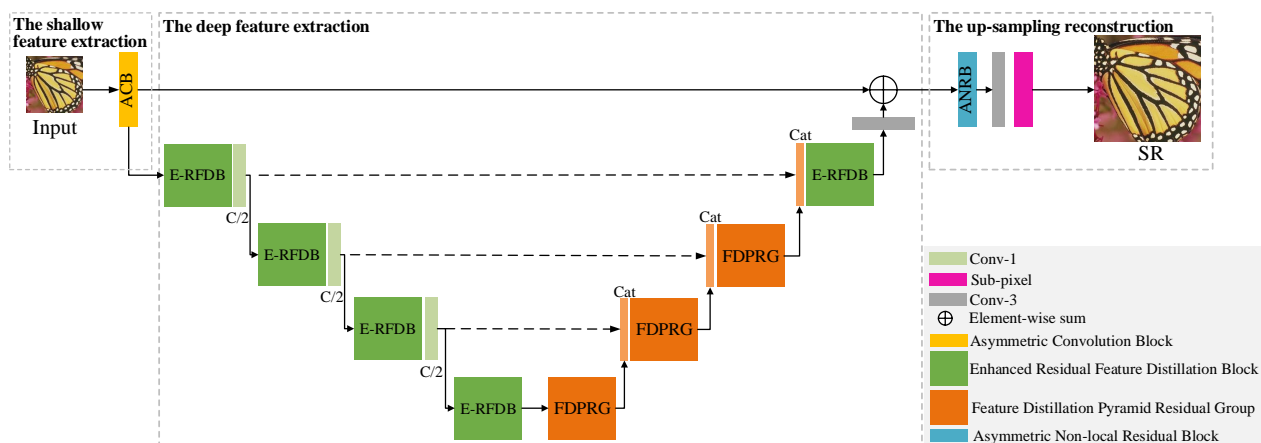
In the image SR field, the objective functions used to optimize models mostly contain a loss term with the pixel-wise distance between the prediction image and the ground truth image. However, researchers discovered that using this function alone leads to blurry and over-smoothed super-resolved images. Therefore, a variety of loss functions are proposed to guide the model optimization. Content loss [30] is introduced into SR to optimize the feature reconstruction error. EnhanceNet [8] uses a texture loss to produce visually more satisfactory results. MSDEPC [49] introduces an edge feature loss by using the phase congruency edge map to learn high-frequency image details. SRGAN [31] uses an adversarial loss to favor outputs residing on the manifold of natural images. CinCGAN [32] uses a cycle consistency loss to avoid the mode collapse issue of GAN and help minimize the distribution divergence.

## 3. U-Shaped Residual Network

In this section, we first describe the overall structure of our proposed network. Then, we elaborate on the feature distillation pyramid residual group and the asymmetric non-local residual block, respectively. Finally, we introduce the loss function of our network, including reconstruction loss and the proposed high-frequency loss.

### 3.1. Network Structure

As shown in Figure 2, our proposed U-shaped residual network (URNNet) consists of three parts: the shallow feature extraction, the deep feature extraction, and the final image reconstruction.



**Figure 2.** The architecture of the proposed U-shaped residual network (URN).

**Shallow Feature Extraction.** Almost all previous works only used a  $3 \times 3$  standard convolution as the first layer in their network to extract the shallow features from the input image. However, the extracted features are single scale and not rich enough. The importance of richer shallow features is ignored in subsequent deep learning methods. Inspired by the asymmetric convolution block (ACB) [50] for image classification, we adapt the ACB to SR domain to extract richer shallow features from the LR image. Specifically,  $3 \times 3$ ,  $1 \times 3$ , and  $3 \times 1$  convolution kernels are used to extract features from the input image in parallel. Then, the extracted features are fused by using an element-wise addition operation to generate richer shallow features. Compared with the standard convolution, the ACB can enrich the feature space and significantly improve the performance of SR with the addition of a few parameters and calculations.

**Deep Feature Extraction.** We use a U-shaped structure to extract deep features. In the downward flow of the U-shaped framework, we use the enhanced residual feature distillation block (E-RFDB) of E-RFDN [28] to extract features because the E-RFDN has shown its excellent performance in the super-resolution challenge of AIM 2020. In the early stage of deep feature extraction, there is no need for complex modules to extract features. Therefore, we only stack  $N$  E-RFDBs in the downward flow. The number of channels of the extracted feature map is halved by using a  $1 \times 1$  convolution for each E-RFDB (except the last one).

Similarly, the upward flow of the U-shaped framework is composed of  $N$  basic blocks including  $N - 1$  feature pyramid residual groups (FDPRG, see Section 3.2) and an E-RFDB. Based on the U-shaped structure, we utilize a step-by-step fusion strategy to fuse the features by using a *Concat* and FDPRG in the downward flow and upward flow. Specifically, the output features of each module in the downward flow are fused into the modules in the upward part in a back-to-front manner. This strategy transfers the information from a low level to a high level and allows the network to fuse the features of different receptive fields, resulting in effectively improving the performance of SR. The number of channels of the feature map increases with the use of the *Concat* operation. Especially for the last *Concat*, using the FDPRG will greatly increase the model complexity. Therefore, only one E-RFDB is used to extract features in the last upward flow.

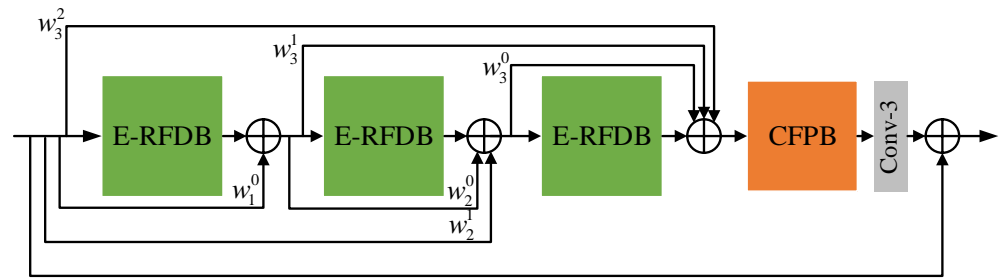
**Image Reconstruction.** After the deep feature extraction stage, a simple  $3 \times 3$  convolution is used to smooth the learned features. Then, the smoothed features are further fused with the shallow features (extracted by ACB) by an element-wise addition operation. In addition, the regression value of each pixel is closely related to the global context information in the image SR task. Therefore, we propose a lightweight asymmetric residual non-local block (ANRB, described in Section 3.3) to model the global context information and further refine the learned features. Finally, a learnable  $3 \times 3$  convolution and a non-parametric sub-pixel [51] operation are used to reconstruct the HR image. Similar to [19,25,28], L1 loss is used to optimize our network. In particular, we propose a



high-frequency loss function (see Section 3.4) to make our network pay more attention to learning high-frequency information.

### 3.2. Feature Distillation Pyramid Residual Group

In the upward flow of the U-shaped structure, we propose a more effective feature distillation pyramid residual group (FDPRG) to extract the deep features. As shown in Figure 3, the FDPRG consists of two main parts: a dense shortcut (DS) part based on three E-RFDBs and a cascaded feature pyramid block (CFPB). After the CFPB, a  $3 \times 3$  convolution is used to refine the learned features.

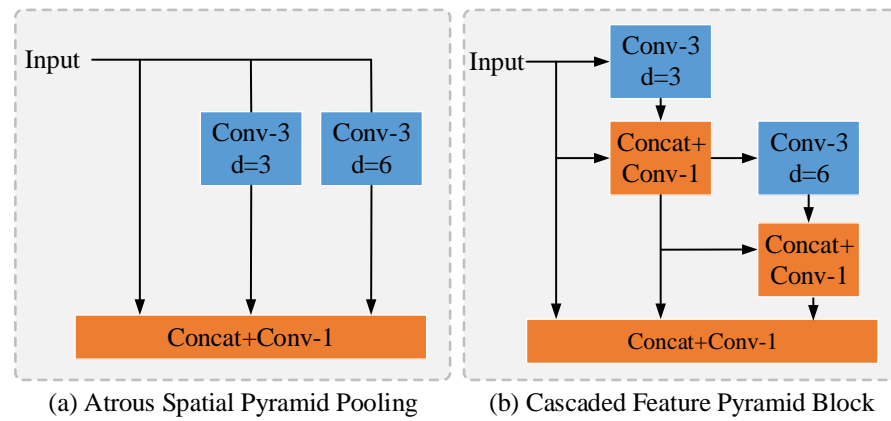


**Figure 3.** The feature distillation pyramid residual group (FDPRG).  $W_i^j$  is a learnable parameter.

**Dense Shortcut.** Residual shortcut (RS) connection is an important technique in various vision tasks. Benefitting from the RS, many SR methods have greatly improved the performance of image SR. RFDN also uses the RS between each RFDB. Although the RS can transfer the information from the input layer of the RFDB to the output layer of the RFDB, it lacks flexibility and simply adds the features of two layers. Later, we consider introducing a dense concatenation [52] to reuse the information of all previous layers. However, this dense connection is extremely GPU memory intensive. Inspired by the dense shortcut (DS) [53] for image classification, we adapt the DS to our SR model by removing the normalization in DS, because the DS has the efficiency of RS and the performance of the dense connection. As shown in Figure 3, the DS is used to connect the  $M$  E-RFDBs in a learnable manner for better feature extraction. In addition, the algorithm proves through experiments that the addition of DS reduces the memory and calculations, while slightly improving performance.

**Cascaded Feature Pyramid Block.** For the image SR task, the low-frequency information (e.g., simple texture) for an LR input image does not need to be reconstructed by a complex network, which allows more information in the low-level feature map. High-frequency information (e.g., edges or corners) needs to be reconstructed by a deeper network, so that the deep feature maps contain more high-frequency information. Hence, different scale features have different contributions to image SR reconstruction. Most previous methods do not utilize multi-scale information, which limits the improvement of image SR performance. Atrous spatial pyramid pooling (ASPP) [54] is an effective multi-scale feature extraction module, which adopts a parallel branch structure of convolutions with different dilation rates to extract multi-scale features, as shown in Figure 4a. However, the ASPP structure is more dependent on the setting of dilation rate parameters and each branch of ASPP is independent of the other.

Different from the ASPP, we propose a more effective multi-scale cascaded feature pyramid block (CFPB) to learn the different scale information, as shown in Figure 4b. The CFPB is designed by cascading multi-different scale convolution layers in a parallel manner. Then, the features of the different branches are fused by a *Concat* operation. The CFPB uses the idea of convolution cascading so that the next layer multi-scale features can be superimposed on the basis of the receptive field of the previous layer. Even if the dilation rate is small, it can still represent a larger receptive field. Additionally, in each parallel branch, the multi-scale features are no longer independent, which makes it easy for our network to learn multi-scale high-frequency information.



**Figure 4.** CFPB (b) is an improvement of ASPP (a).

### 3.3. Asymmetric Non-Local Residual Block

The non-local mechanism [45] is an attention model, which can effectively capture the long-range dependencies by modeling the connection relationship between a pixel position and all positions. In the image SR task, it is image-to-image learning. Most existing works only focus on learning detailed information while ignoring the long-range feature-wise similarities in natural images, which may produce incorrect textures globally. For the image “img092” (see Figure 8), other SR methods have learned the details of the texture (dark lines in the picture), but the direction of these lines is completely wrong in the global scope. The global texture learned by the proposed URNet after adding the non-local module is consistent with the GT image.

However, the classic Non-Local module has expensive calculation and memory consumption. It cannot be directly applied to the lightweight SR network. Inspired by the asymmetric non-local block (ANB) [29] for semantic segmentation, we propose a more lightweight asymmetric non-local residual block (ANRB, shown in Figure 5) for fast and lightweight image SR. Specifically, let  $X \in R^{C \times H \times W}$  represent a feature map, where  $C$  and  $H \times W$  are the numbers of channels and spatial size of  $X$ . We use three  $1 \times 1$  convolutions to compress multi-channel features  $X$  into single-channel features  $X_\phi$ ,  $X_\theta$ ,  $X_\gamma$ , respectively. Afterwards, similar to the ANB, we use the pyramid pool sampling algorithm [55] to sample only  $S$  ( $S \ll N = H \times W$ ) representative feature points from the Key and Value branches. We perform four average pooling operations to obtain four feature maps with sizes of  $1 \times 1$ ,  $3 \times 3$ ,  $6 \times 6$ ,  $8 \times 8$ , respectively. Subsequently, we flatten and expand the four maps, then stitch them together to obtain a sampled feature map with a length of 110. Then, the non-local attention can be calculated as follows:

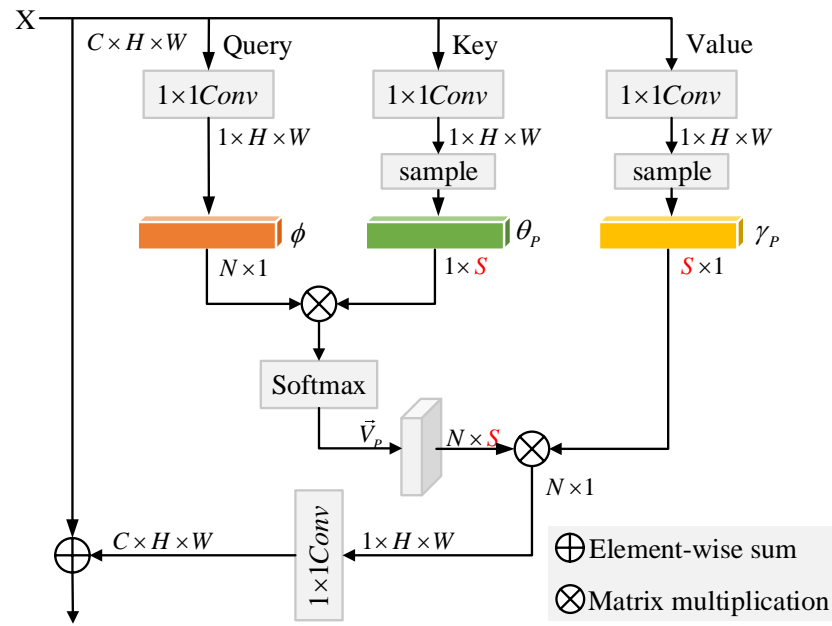
$$X_\phi = f_\phi(X), \quad X_\theta = f_\theta(X), \quad X_\gamma = f_\gamma(X), \quad (1)$$

$$\theta_P = P_\phi(X_\phi), \quad \gamma_P = P_\gamma(X_\gamma), \quad (2)$$

$$Y = Softmax(X_\phi^T \otimes \theta_P) \otimes \gamma_P, \quad (3)$$

where  $f_\phi$ ,  $f_\theta$ , and  $f_\gamma$  are  $1 \times 1$  convolutions.  $P_\phi$  and  $P_\gamma$  represent the pyramid pooling sampling for generating the sampled features  $\theta_P$  and  $\gamma_P$ .  $\otimes$  is matrix multiplication and  $Y$  is a feature map containing contextual information.

The last step of the attention mechanism generally uses dot multiplication to multiply the generated attention weight feature map  $Y$  with the original feature map to achieve the function of attention. However, the value of a large number of elements in  $Y$ , a matrix of  $1 \times H \times W$ , is close to zero due to the *Softmax* operation and the characteristics of the *Softmax* function itself:  $\sum_i^H \sum_j^M (Softmax(y_{ij})) = 1$ . If we directly use the operation of the dot multiplication for attention weighting, it will inevitably cause the value of the element in the weighted feature map to be too small, making the gradient disappear, which makes the gradient impossible to iterate.



**Figure 5.** The asymmetric non-local residual block (ANRB).

In order to solve the above problems, we use the addition operation to generate the final attention weighted feature map  $X_{weighted} = H_{1 \times 1}(Y) + X$ , allowing the network to converge more easily, where  $H_{1 \times 1}(\cdot)$  is a  $1 \times 1$  convolution operation to convert the single-channel feature map  $Y$  into a  $C$ -channel feature map for the subsequent element-wise sum. Benefitting from the channel compression and the sampling operation, the ANRB is a lightweight non-local block. The ANRB is used to capture global context information for fast and accurate image SR.

### 3.4. Loss Function

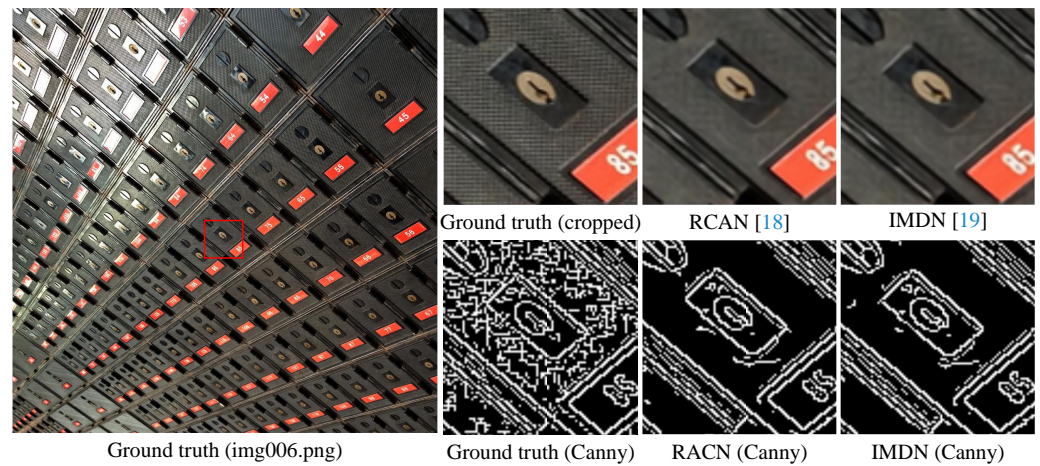
In the SR domain, L1 loss (i.e., mean absolute error) and L2 loss (e.g., mean squared error) are the most frequently used loss functions for the image SR task. Similar to [18,19,25,51], we adopt L1 loss as the main reconstruction loss function to measure the differences between the SR images and the ground truth. Specifically, the L1 loss is defined as

$$\mathcal{L}_1 = \frac{1}{N} \sum_{i=1}^N \|I_{HR}^i - I_{SR}^i\|_1, \quad (4)$$

where  $I_{SR}^i, I_{HR}^i$  denote the  $i$ -th SR image generated by URNet and the corresponding  $i$ -th HR image used as ground truth.  $N$  is the total number of training samples.

For the image SR task, only using L1 loss or L2 loss will cause the super-resolved images to lack high-frequency details, presenting unsatisfying results with over-smooth textures. As depicted in Figure 6, comparing the natural image and the SR images generated by SR methods (e.g., RCAN [18] and IMDN [19]), we can see the reconstructed image is over-smooth in detailed texture areas. By applying edge detection algorithms to natural images and SR images, the difference is more obvious.





**Figure 6.** Ground truth/SR images and their edge images extracted by Canny operator.

Therefore, we propose a simple but effective high-frequency loss to alleviate this problem. Specifically, we first use the edge detection algorithm to extract the detailed texture maps of the HR and the SR images. Then, we adopt mean absolute error to measure the detailed differences between the SR image and the HR image. This process can be formulated as follows:

$$\mathcal{L}_{hf} = \frac{1}{N} \sum_{i=1}^N \|H_c(I_{HR}^i) - H_c(I_{SR}^i)\|_1, \quad (5)$$

where  $H_c$  denotes edge detection algorithm. In this work, we use Canny to extract detailed information from the SR images and the ground truth, respectively. Therefore, the training objective of our network is  $\mathcal{L} = \alpha \mathcal{L}_{hf} + \beta \mathcal{L}_1$ , where  $\alpha$  and  $\beta$  are weights and used to adjust these two loss functions.

## 4. Experiments

### 4.1. Datasets and Metrics

DIV2K [56] is a high-quality image dataset, which contains 1000 DIVerse 2 K resolution RGB images including various scenes, such as animals, plants, and landscapes. The HR DIV2K is divided into 800 training images, 100 validation images, and 100 testing images. Similar to [19,27,28], we train all models with the DIV2K training images, and the corresponding LR images are generated by bicubic down-sampling the HR image with  $\times 2$ ,  $\times 3$ ,  $\times 4$  scale, respectively. To better evaluate the performance and generalization of our proposed URNet, we report the performance on four standard benchmark datasets including Set5 [57], Set14 [58], B100 [59], and Urban100 [16]. Following the previous works [19,26,28], the peak signal-to-noise ratio (PSNR) [60] and structural similarity index (SSIM) [61] are used to quantitatively evaluate our model on the Y channel in the YCbCr space converted from RGB space. PSNR is used to measure the differences between corresponding pixels of the super-resolved image and ground truth. SSIM is used to measure the structural similarity (e.g., luminance, contrast, and structures) between images.

### 4.2. Implementation Details

In order to clearly see the improvement effect of our method relative to RFDN, our model parameters and calculations are set as almost or less than RFDN's counterparts to exceed the performance of RFDN. The deeper or wider the convolutional network is, the better the performance is. Based on this, we tend to use as many modules as possible in the two flow branches. The number of channels, determining the width of the network, should not be too small. Therefore, we set  $N = 4$ , and the minimum number of channels to 8. Considering the complexity of the model, we use the most basic structure in [53], that is, setting  $M = 3$ . Then, considering the three-channel halving operations of the downward

flow and the three *Concat* operations of the upward flow, we set the basic channel number of our URNet to 64. Specifically, for the four E-RFDBs in the downward flow (from top to bottom), the number of input channels is 64, 32, 16, and 8, respectively, while the number of input channels in the four modules in the upward flow (from bottom to top) is just the opposite.

Following the EDSR [25], the training data are augmented with random horizontal flips and 90 rotations. In the training phase, we randomly extract 32 LR RGB patches with the size of  $64 \times 64$  from all the LR images in every batch. Our model is optimized by Adam with  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ , and  $\epsilon = 10^{-8}$ . The batch size is set to 32. The learning rate is initialized as  $5 \times 10^{-4}$  and halved for every  $2 \times 10^5$  iterations for 1000 epochs. Each epoch has 1000 iterations of back-propagation. Similar to the IMDN [19], the hyper-parameter of Leaky ReLU is set as 0.05. The weight parameters of the loss function are set as  $\alpha = 0.25$  and  $\beta = 1.0$ , respectively. The proposed method is implemented with PyTorch on a single GTX 1080Ti GPU.

#### 4.3. Ablation Studies

To better validate the effectiveness of different blocks in our network, we conduct a series of ablation experiments on DIV2K. We first utilize the step-by-step fusion strategy to design a baseline model (denoted as URNet-B) based on the E-RFDB. Then, we gradually add different modules to the URNet-B. Detailed ablation experiment results are presented in Table 1. After adding the ACB into the URNet-B, the PSNR increases to 35.56 dB. Adding the DS and CFPB, we can see that the performance of image SR has increased from 35.56 dB to 35.59 dB. After adding all the blocks into the URNet-B, the PSNR increases to 35.62 dB. This is mainly because our model can consistently accumulate the hierarchical features to form more representative features and it is well focused on spatial context information. These results demonstrate the effectiveness of our ACB, FDPRG (including DS and CFPB), and ANRB.

**Table 1.** Ablation experiment results of different blocks on DIV2K val. **Bold** indicates the best performance ( $\times 2$  SR).

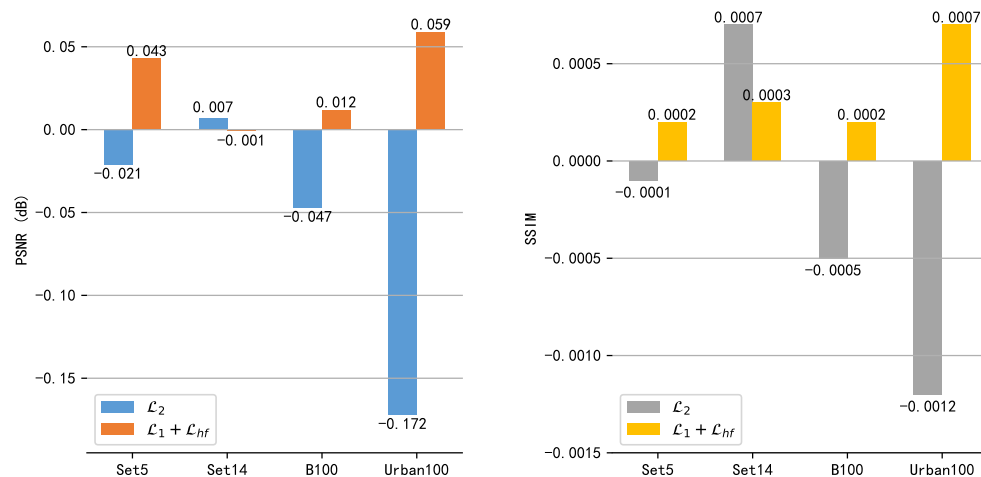
URNet-B	✓	✓	✓	✓	✓
ACB		✓	✓	✓	✓
FDPRG/DS			✓	✓	✓
FDPRG/CFPB				✓	✓
ANRB					✓
PSNR (dB)	35.54	35.56	35.58	35.59	<b>35.62</b>

Afterwards, we conduct ablation experiments on the four benchmark datasets on  $\times 2$  scale SR to validate the effectiveness of our proposed high-frequency loss  $\mathcal{L}_{hf}$  against other loss functions widely used in the field of SR (see Section 2.3). For the adversarial loss and the cyclic consistency loss, these two loss functions are suitable for the GAN, but not for our proposed URNet. Therefore, we only report the comparison results with the other five loss functions (see Table 2). For the content loss (denoted as  $\mathcal{L}_c$ ) and the texture loss (denoted as  $\mathcal{L}_t$ ), we use the same configuration with SRResNet [31] and EnhanceNet [8], respectively. We observe a trend that using content loss or texture loss yields worse performance. In practice, these two loss functions are used in combination with the adversarial loss in the GAN of SR.

**Table 2.** Performance of different loss functions. Best results are **bolded** ( $\times 2$  SR).

		Set5	Set14	B100	Urban100
$\mathcal{L}_1$	PSNR	38.020	33.685	32.228	32.356
	SSIM	0.9606	0.9184	0.9003	0.9303
$\mathcal{L}_2$	PSNR	37.999	<b>33.692</b>	32.181	32.184
	SSIM	0.9605	<b>0.9191</b>	0.8998	0.9291
$\mathcal{L}_c$	PSNR	35.823	31.776	30.283	30.145
	SSIM	0.9350	0.8763	0.8439	0.8822
$\mathcal{L}_t$	PSNR	35.267	31.230	29.870	29.587
	SSIM	0.9328	0.8747	0.8518	0.8900
$\mathcal{L}_1 + \mathcal{L}_{hf}$	PSNR	<b>38.063</b>	33.684	<b>32.240</b>	<b>32.415</b>
	SSIM	<b>0.9608</b>	0.9187	<b>0.9005</b>	<b>0.9310</b>

As shown in Figure 7, we visualize the performance difference for the other three loss functions (including  $\mathcal{L}_1$ ,  $\mathcal{L}_2$ , and  $\mathcal{L}_1 + \mathcal{L}_{hf}$ ). Compared with  $\mathcal{L}_1$  and  $\mathcal{L}_1 + \mathcal{L}_{hf}$ , the performance of  $\mathcal{L}_2$  on the four datasets is generally lower, especially on Urban100 with richer texture details. This is because the  $\mathcal{L}_2$  loss uses the square of the pixel value error, so high-value differences are more important than low-value differences, resulting in too smooth results (in the case of minimum error values). Therefore, the  $\mathcal{L}_1$  loss function is more widely used than the  $\mathcal{L}_2$  loss in the image super-resolution [25,62]. After adding the high-frequency loss  $\mathcal{L}_{hf}$  to the total loss function, the performance of image SR achieves significant improvement on both Set5 and Urban100. Compared with only using  $\mathcal{L}_1$  loss, our high-frequency loss also achieves comparable PSNR and SSIM scores on the Set14 and B100 datasets. Our high-frequency loss performs especially well on Urban100 because the dataset has richer structured texture information. The high-frequency loss makes our network more focused on the texture structure of images.



**Figure 7.** Comparison results of the performance difference between the three loss functions. We take PSNR/SSIM scores of  $\mathcal{L}_1$  as a baseline and the PSNR/SSIM scores of  $\mathcal{L}_2$  and the proposed  $\mathcal{L}_1 + \mathcal{L}_{hf}$  are subtracted from it, respectively.

In order to further gain a clearer insight on the improvements of the step-by-step fusion strategy based on the U-shaped structure, we conduct experiments to compare this strategy and the general *Concat* operation to fuse the features of all blocks. Specially, we train the URNet-B and E-RFDN from scratch with the same experiment configurations to validate the effectiveness of this fusion strategy, because these two models are built based on the E-RFDB and using different fusion strategies. The experiment results are presented in Table 3.

We can see that the URNet-B not only achieves significant performance improvements on the four benchmark datasets, especially in Urban100 (PSNR: **+0.11 dB**), but also has fewer parameters (URNet-B: **567.6 K** vs. E-RFDN: 663.9 K) and calculations (FLOPs: **35.9 G** vs. 41.3 G). These results demonstrate that the step-by-step fusion strategy can not only reduce model complexity but also effectively preserve the hierarchical information to facilitate subsequent feature extraction.

**Table 3.** The comparison of different fusion strategies (the step-by-step and *Concat* the features of all blocks). URNet-B achieves the best PSNR (dB) scores on the four benchmark datasets ( $\times 2$  SR).

Method	Set5	Set14	B100	Urban100	Params	FLOPs
E-RFDN [28]	37.99	<b>33.56</b>	32.19	32.16	663.9 K	41.3 G
URNet-B	<b>38.03</b>	<b>33.56</b>	<b>32.20</b>	<b>32.27</b>	<b>567.6 K</b>	<b>35.9 G</b>

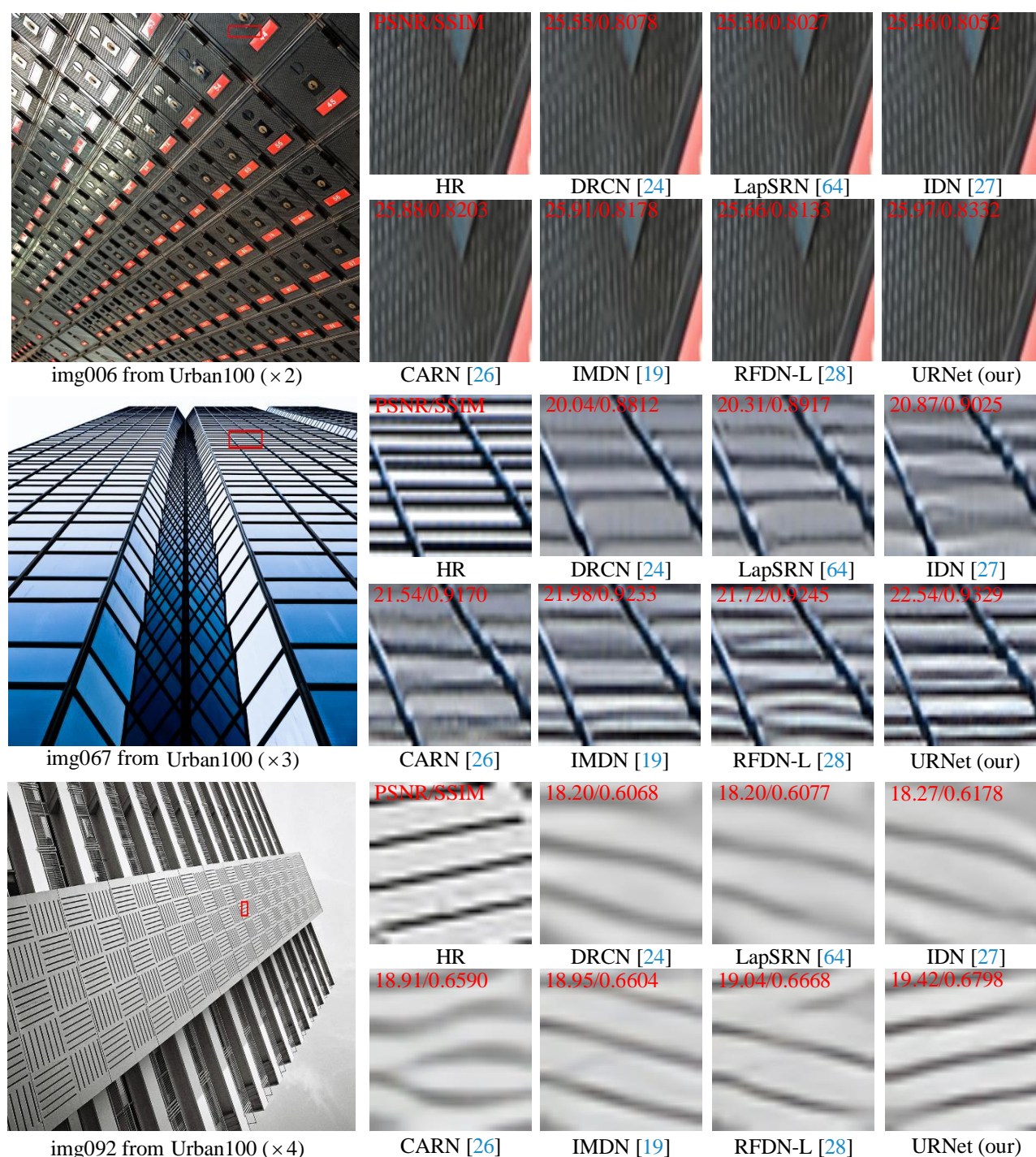
#### 4.4. Comparison with State-of-the-Art Methods

In this section, numerous experiments are described on the four public SR benchmark datasets mentioned above. We extensively compare our proposed method with various state-of-the-art lightweight SISR methods, including Bicubic, SRCNN [21], FSRCNN [63], VDSR [23], DRCN [24], LapSRN [64], DRRN [65], MemNet [37], IDN [27], SRMDNF [66], CARN [26], IMDN [19], and RFDN-L [28]. Similar to [18,25], we also introduce a self-ensemble strategy to improve our URNet and denote the self-ensembled one as URNet+.

**Quantitative Results by PSNR/SSIM.** Table 4 presents quantitative comparisons for  $\times 2$ ,  $\times 3$ , and  $\times 4$  SR. For a clearer and fairer comparison, we re-train the RFDN-L [28] by using the same experimental configurations as in their paper. We test the IMDN [19] (using the official pre-trained models (<https://github.com/Zheng222/IMDN>, accessed on 15 September 2021)), RFDN-L, and our URNet with the same environment. The results of other methods come from their papers. Compared with all the aforementioned approaches, our URNet performs the best in almost all cases. For all scaling factors, the proposed method achieves obvious improvement in the Urban100 dataset. These results indicate that our algorithm could successfully reconstruct satisfactory results for images with rich and detailed structures.

**Qualitative Results.** The qualitative results are illustrated in Figure 8. For challenging details in images “img006”, “img067”, and “img092” of the Urban100 [16] dataset, we observe that most of the compared methods would suffer from blurring edges and noticeable artifacts. IMDN [19] and RFDN-L [28] can alleviate blurred edges and recover more details (e.g., “img006” and “img067”) but produce different degrees of the fake information (e.g., “img092”). In contrast, our URNet gains much better results in recovering sharper and more precise edges, more faithful to the ground truth. Especially for the image “img092” on the  $\times 4$  SR, the texture direction of the reconstructed edges from all compared methods is completely wrong. The URNet can make full use of the learned features and obtain clearer contours without serious artifacts. These comparisons indicate that the URNet can better recover more informative components in HR images and show satisfactory image SR results than other methods.





**Figure 8.** Visual qualitative comparisons of the state-of-the-art lightweight methods and our URNet on Urban100 dataset for  $\times 2$ ,  $\times 3$ , and  $\times 4$  SR. Zoom in for best view.

**Model Parameters.** For the lightweight image SR, the number of model parameters is a key factor to take into account. Table 4 depicts the comparison of image SR performance and model parameters on the four benchmark datasets with scale factor  $\times 2$ ,  $\times 3$ , and  $\times 4$ , respectively. To obtain a more comprehensive understanding of the model complexity, the comparisons of the model parameters and performance are visualized in Figure 9. We can see that the proposed URNet achieves a better trade-off between the performance of image SR and model complexity than other state-of-the-art lightweight models.



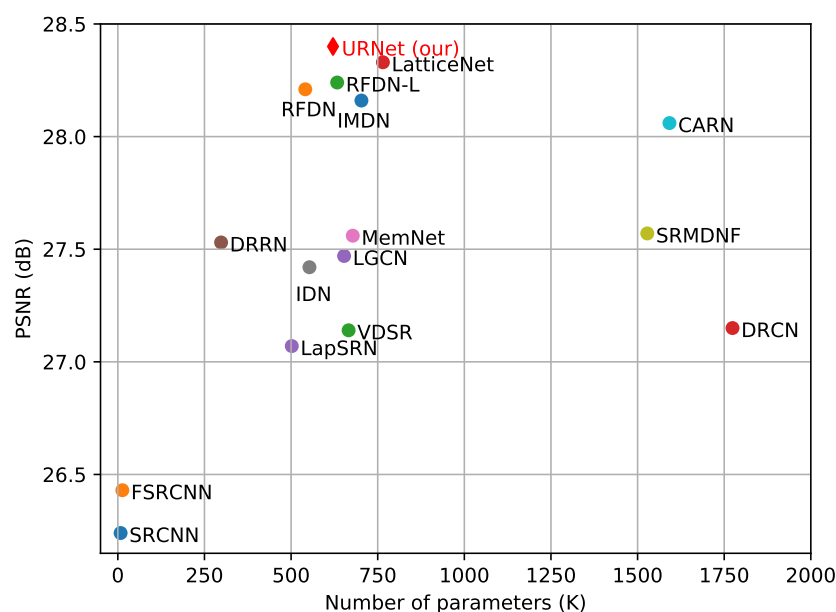
**Table 4.** The average performance of the state-of-the-art methods for scale factor  $\times 2$ ,  $\times 3$ , and  $\times 4$  on the four benchmark datasets Set5, Set14, B100, and Urban100. Best and second best results are **bolded** and underlined.

Method	Scale	Params	Set5 PSNR/SSIM	Set14 PSNR/SSIM	B100 PSNR/SSIM	Urban100 PSNR/SSIM
Bicubic	$\times 2$	-	33.66/0.9299	30.24/0.8688	29.56/0.8431	26.88/0.8403
SRCNN [21]		8 K	36.66/0.9542	32.45/0.9067	31.36/0.8879	29.50/0.8946
FSRCNN [63]		13 K	37.00/0.9558	32.63/0.9088	31.53/0.8920	29.88/0.9020
VDSR [23]		666 K	37.53/0.9587	33.03/0.9124	31.90/0.8960	30.76/0.9140
DRCN [24]		1774 K	37.63/0.9588	33.04/0.9118	31.85/0.8942	30.75/0.9133
LapSRN [64]		251 K	37.52/0.9591	32.99/0.9124	31.80/0.8952	30.41/0.9103
DRRN [65]		298 K	37.74/0.9591	33.23/0.9136	32.05/0.8973	31.23/0.9188
MemNet [37]		678 K	37.78/0.9597	33.28/0.9142	32.08/0.8978	31.31/0.9195
IDN [27]		553 K	37.83/0.9600	33.30/0.9148	32.08/0.8985	31.27/0.9196
SRMDNF [66]		1511 K	37.79/0.9601	33.32/0.9159	32.05/0.8985	31.33/0.9204
CARN [26]		1592 K	37.76/0.9590	33.52/0.9166	32.09/0.8978	31.92/0.9256
IMDN [19]		694 K	38.00/0.9605	33.63/0.9177	32.18/0.8996	32.17/0.9283
RFDN-L [28]		626 K	38.03/0.9606	33.65/0.9183	32.17/0.8996	32.16/0.9282
URNet (ours)		612 K	<u>38.06/0.9608</u>	<u>33.68/0.9187</u>	<u>32.24/0.9005</u>	<u>32.42/0.9310</u>
URNet+ (ours)		612 K	<b>38.14/0.9611</b>	<b>33.70/0.9190</b>	<b>32.29/0.9009</b>	<b>32.61/0.9325</b>
Bicubic	$\times 3$	-	30.39/0.8682	27.55/0.7742	27.21/0.7385	24.46/0.7349
SRCNN [21]		8 K	32.75/0.9090	29.30/0.8215	28.41/0.7863	26.24/0.7989
FSRCNN [63]		13 K	33.18/0.9140	29.37/0.8240	28.53/0.7910	26.43/0.8080
VDSR [23]		666 K	33.66/0.9213	29.77/0.8314	28.82/0.7976	27.14/0.8279
DRCN [24]		1774 K	33.82/0.9226	29.76/0.8311	28.80/0.7963	27.15/0.8276
LapSRN [64]		502 K	33.81/0.9220	29.79/0.8325	28.82/0.7980	27.07/0.8275
DRRN [65]		298 K	34.03/0.9244	29.96/0.8349	28.95/0.8004	27.53/0.8378
MemNet [37]		678 K	34.09/0.9248	30.00/0.8350	28.96/0.8001	27.56/0.8376
IDN [27]		553 K	34.11/0.9253	29.99/0.8354	28.95/0.8013	27.42/0.8359
SRMDNF [66]		1528 K	34.12/0.9254	30.04/0.8382	28.97/0.8025	27.57/0.8398
CARN [26]		1592 K	34.29/0.9255	30.29/0.8407	29.06/0.8034	28.06/0.8493
IMDN [19]		703 K	34.36/0.9270	30.32/0.8417	29.09/0.8047	28.16/0.8519
RFDN-L [28]		633 K	34.39/0.9271	30.35/0.8419	29.11/0.8054	28.24/0.8534
URNet (ours)		621 K	<u>34.51/0.9281</u>	<u>30.40/0.8433</u>	<u>29.14/0.8061</u>	<u>28.40/0.8574</u>
URNet+ (ours)		621 K	<b>34.60/0.9288</b>	<b>30.48/0.8444</b>	<b>29.19/0.8072</b>	<b>28.57/0.8599</b>
Bicubic	$\times 4$	-	28.42/0.8104	26.00/0.7027	25.96/0.6675	23.14/0.6577
SRCNN [21]		8 K	30.48/0.8626	27.50/0.7513	26.90/0.7101	24.52/0.7221
FSRCNN [63]		13 K	30.72/0.8660	27.61/0.7550	26.98/0.7150	24.62/0.7280
VDSR [23]		666 K	31.35/0.8838	28.01/0.7674	27.29/0.7251	25.18/0.7524
DRCN [24]		1774 K	31.53/0.8854	28.02/0.7670	27.23/0.7233	25.14/0.7510
LapSRN [64]		251 K	31.54/0.8852	28.09/0.7700	27.32/0.7275	25.21/0.7562
DRRN [65]		298 K	31.68/0.8888	28.21/0.7720	27.38/0.7284	25.44/0.7638
MemNet [37]		678 K	31.74/0.8893	28.26/0.7723	27.40/0.7281	25.50/0.7630
IDN [27]		553 K	31.82/0.8903	28.25/0.7730	27.41/0.7297	25.41/0.7632
SRMDNF [66]		1552 K	31.96/0.8925	28.35/0.7787	27.49/0.7337	25.68/0.7731
CARN [26]		1592 K	32.13/0.8937	28.60/0.7806	27.58/0.7349	26.07/0.7837
IMDN [19]		715 K	32.21/0.8948	28.58/0.7810	27.55/0.7353	26.04/0.7838
RFDN-L [28]		643 K	<u>32.23/0.8953</u>	28.59/0.7814	27.56/0.7362	26.14/0.7871
URNet (ours)		633 K	<u>32.20/0.8952</u>	<u>28.63/0.7826</u>	<u>27.60/0.7369</u>	<u>26.23/0.7905</u>
URNet+ (ours)		633 K	<b>32.35/0.8969</b>	<b>28.71/0.7840</b>	<b>27.66/0.7383</b>	<b>26.41/0.7945</b>

#### 4.5. Model Analysis

**Model Calculations.** It is not enough to measure the weight of the model only by the model parameters. Calculation consumption is also an important metric. In Table 5, we report the comparison of URNet and other state-of-the-art algorithms (e.g., CARN [26],

IMDN [19], and RFDN-L [28]) in terms of FLOPs (using a single image with the size  $256 \times 256$ ) and PSNR/SSIM (using the Set14 dataset with the  $\times 4$  scale factor). As we can see, our URNet achieves higher PSNR/SSIM than other methods while using fewer calculations. These results demonstrate that our method can balance the calculation costs and the performance of image reconstruction well.



**Figure 9.** PSNR vs. the number of parameters. The comparison is conducted on Urban100 with the  $\times 3$  scale factor.

**Table 5.** PSNR/SSIM vs. FLOPs on Set14 ( $\times 4$ ).

	CARN [26]	IMDN [19]	RFDN-L [28]	URNet (ours)
SSIM	0.7806	0.7810	0.7814	<b>0.7826</b>
PSNR	28.60	28.58	28.59	<b>28.63</b>
FLOPs (G)	103.58	46.60	41.54	<b>39.51</b>

**Lightweight Analyses.** We also choose two non-lightweight methods and one SOTA lightweight SISR method, i.e., EDSR [25], RCAN [18], and IMDN [19], for comparison. We use official codes (<https://github.com/cszn/KAIR>, accessed on 15 September 2021) (AIM 2020 efficient super-resolution challenge (<https://data.vision.ee.ethz.ch/cvl/aim20/>, accessed on 15 September 2021)) to test the running time of these methods in a feed-forward process on the B100 ( $\times 4$ ) dataset. The results are reported in Table 6. We can observe that both methods, EDSR and RCAN, outperform our URNet. This is a reasonable result since they have a deeper and wider network structure that contains large quantities of convolutional layers and parameters. Actually, the parameters of EDSR and RCAN are 40 M and 16 M, while that of ours is only 0.6 M. However, compared with other methods, URNet runs the fastest inference speed. Simultaneously, our URNet achieves dominant performance in terms of parameter usage and time consumption, compared to IMDN. These comparison results show that our method can obtain fast and accurate image SR.

**Table 6.** Comparison with non-lightweight and SOTA lightweight methods.

	Scale	EDSR [25]	RCAN [18]	IMDN [19]	URNet (ours)
Set5	2	38.11/0.9602	38.27/0.9614	38.00/0.9605	38.06/0.9608
	3	34.65/0.9280	34.74/0.9299	34.36/0.9270	34.51/0.9281
	4	32.46/0.8968	32.63/0.9002	32.21/0.8948	32.20/0.8952
Set14	2	33.92/0.9195	34.12/0.9216	33.63/0.9177	33.68/0.9187
	3	30.52/0.8462	30.65/0.8482	30.32/0.8417	30.40/0.8433
	4	28.80/0.7876	28.87/0.7889	28.58/0.7810	28.63/0.7826
B100	2	32.32/0.9013	32.41/0.9027	32.18/0.8996	32.24/0.9005
	3	29.25/0.8093	29.32/0.8111	29.09/0.8047	29.14/0.8061
	4	27.71/0.7420	27.77/0.7436	27.55/0.7353	27.60/0.7369
Urban100	2	32.93/0.9351	33.24/0.9384	32.17/0.9283	32.42/0.9310
	3	28.80/0.8653	29.09/0.8702	28.16/0.8519	28.40/0.8574
	4	26.64/0.8033	26.82/0.8087	26.04/0.7838	26.23/0.7905
Parameters (K)		43,090	15,592	715	633
FLOPs (G)		3293.9	1044.0	46.6	39.5
Running Time (Sec.)		0.2178	0.2596	0.0939	0.0310

#### 4.6. Remote Sensing Image Super-Resolution

To better evaluate the generalization of our method, we also conduct experiments on the remote sensing datasets. The natural image SR and remote sensing image SR belong to different image domains but the same task. Consequently, we can use the URNet trained on the natural image dataset (i.e., DIV2K) as a pre-trained model and fine-tune the model on the remote sensing dataset. By transferring the external knowledge from the natural image domain to the remote sensing domain, our proposed URNet achieves a better performance on the remote sensing image SR task.

Following most remote sensing image SR methods [67–71], we conduct experiments on the UC Merced [72] land-use dataset. The UC Merced dataset is one of the most popular image collections in the remote sensing community, which contains 21 classes of land-use scenes in total with 100 aerial images per class. These images have a high spatial resolution (0.3 m/pixel). We randomly select 840 images (40 images per class) from the UC Merced as the training set, and we randomly select 40 images from the training set as a validation set. Moreover, we construct a testing set named UCTest by randomly choosing 120 images from the remaining images of the UC Merced dataset. The LR-HR image pair acquisition operation and implementation details are the same as for experiments on the DIV2K dataset. The model is trained for 100 epochs with an initial learning rate of 0.0001 and the input patch size set to  $16 \times 16$ . Similarly, we also re-train RFDN-L [28] by using the same training strategies. MPSR [68] randomly selects 800 images from the UC Merced dataset as the training samples. For a fair and convincing comparison, we re-train the MPSR by using the same experimental configurations as in their paper and the same dataset as this paper.

The NWPU-RESISC45 [73] dataset is a public benchmark with spatial resolution varying from 30 m to 0.2 m per pixel. We also randomly select 180 images from the NWPU-RESISC45 dataset as a testing set (named RESISCTest) to validate the robustness of our model.

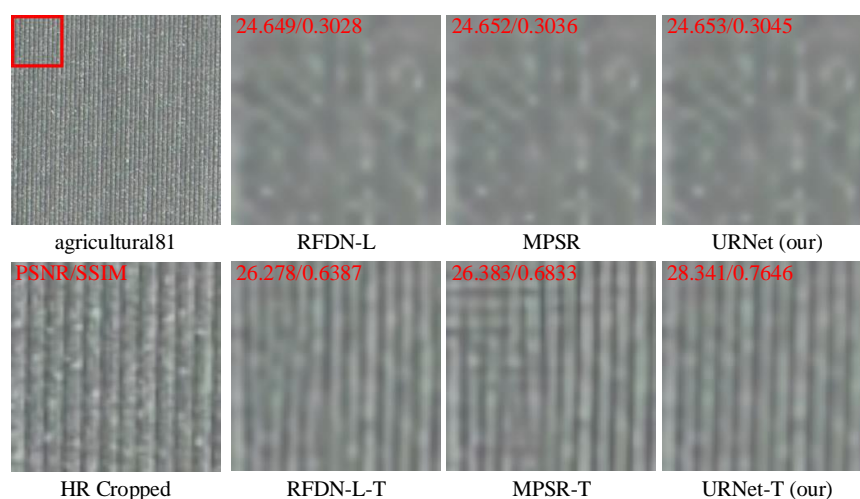
Table 7 shows the quantitative results of the state-of-the-art SR methods on remote sensing datasets UCTest and RESISCTest for scale factor  $\times 4$ . We can see that our proposed URNet and URNet-T (using the pre-trained model) achieve the highest PSNR and SSIM scores on these two datasets. The methods could gain better performance by using the strategy of the pre-trained model, which means that this strategy allows low-level feature information from DIV2K to be shared to another dataset, achieving better performance on super-resolving remote sensing images. The performance of MPSR is further improved on UCTest by using the same strategy but fails on RESISCTest because the MPSR-T is a

non-lightweight model (MPSR-T: 12.3 M vs. URNet-T: 633 K, and MPSR-T: 835.5 G vs. URNet-T: 39.5 G, in terms of parameters and FLOPs) and more likely to overfit on the training set.

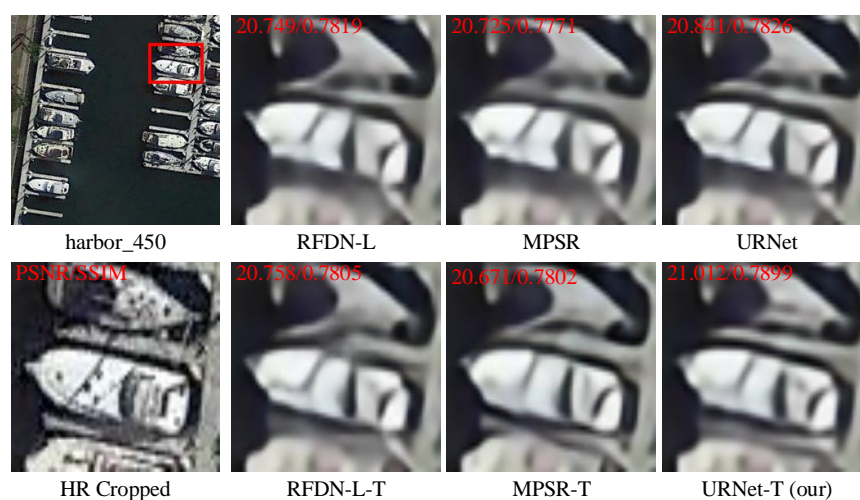
**Table 7.** The PSNR/SSIM of UCTest and RESISCTest with a scale factor of  $\times 4$ . (\*-T denotes using the pre-trained model.)

		RFDN-L [28]	MPSR [68]	URNet (ours)	RFDN-L-T	MPSR-T	URNet-T (ours)
UCTest	PSNR	29.03	29.09	<b>29.15</b>	29.37	29.34	<b>29.58</b>
	SSIM	0.7940	0.7953	<b>0.7968</b>	0.8047	0.8060	<b>0.8102</b>
RESISCTest	PSNR	29.06	29.09	<b>29.13</b>	29.09	29.01	<b>29.19</b>
	SSIM	0.7710	0.7718	<b>0.7730</b>	0.7721	0.7706	<b>0.7750</b>

To fully demonstrate the effectiveness of our method, we also show the  $\times 4$  SR visual results from UCTest’s “agricultural81” in Figure 10 and RESISCTest’s “harbor\_450” in Figure 11. We can see that our proposed URN-T shows significant improvements, reducing aliasing, blur artifacts, and better reconstructing high-fidelity image details.



**Figure 10.** Comparison of reconstructed HR images of “agricultural81” obtained from UCTest dataset with  $256 \times 256$  pixel images using different methods with a scale factor of  $\times 4$ .



**Figure 11.** Comparison of reconstructed HR images of “harbor\_450” obtained from RESISCTest dataset with  $256 \times 256$  pixel images using different methods with a scale factor of  $\times 4$ .

## 5. Conclusions

In this paper, we introduce a novel lightweight U-shaped residual network (URNet) for fast and accurate image SR. Specifically, we design an effective feature distillation pyramid residual group (FDPRG) to extract deep features from an LR image based on the E-RFDB. The FDPRG can effectively reuse the shallow features with dense shortcut connections and capture multi-scale information with a cascaded feature pyramid block. Based on the U-shaped structure, we utilize a step-by-step fusion strategy to fuse the features of different blocks and further refine the learned features. In addition, we introduce a lightweight asymmetric non-local residual block to capture the global context information and further improve the performance of image SR. In particular, to alleviate the problem of smoothing image details caused by pixel-wise loss, we design a simple but effective high-frequency loss to help optimize our model. Extensive experiments indicate the URNet achieves a better trade-off between image SR performance and model complexity against other state-of-the-art SR methods. In the future, our method will be applied to super-resolution images with fuzzy or even real degradation models. At the same time, we will also consider deep separable convolutions or other lightweight convolutions as an alternative to standard convolutions to further reduce the number of parameters and calculations.

**Author Contributions:** Y.W. and L.Z. have equal contribution to this work and are co-first authors. Conceptualization, Y.W. and L.Z.; methodology, Y.W. and L.Z.; software, Y.W.; validation, L.L., H.H., and W.T.; writing—original draft preparation, Y.W. and L.Z.; writing—review and editing, Y.W., L.Z., and L.L.; supervision, W.T.; project administration, L.L.; funding acquisition, L.L., H.H., and W.T. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported in part by the National Natural Science Foundation of China (Grant 61976227, 62176096, and 62076257) and in part by the Natural Science Foundation of Hubei Province under Grant 2019CFB622.

**Data Availability Statement:** Code is available at <https://github.com/ytao-wang/URNet>, accessed on 15 September 2021.

**Acknowledgments:** The authors are grateful to the Editor and reviewers for their constructive comments, which significantly improved this work.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Isaac, J.S.; Kulkarni, R. Super resolution techniques for medical image processing. In Proceedings of the 2015 International Conference on Technologies for Sustainable Development, Mumbai, India, 4–6 February 2015; pp. 1–6.
2. Liu, H.; Xu, J.; Wu, Y.; Guo, Q.; Ibragimov, B.; Xing, L. Learning deconvolutional deep neural network for high resolution medical image reconstruction. *Inf. Sci.* **2018**, *468*, 142–154. [\[CrossRef\]](#)
3. Yamashita, K.; Markov, K. Medical Image Enhancement Using Super Resolution Methods. In *International Conference on Computational Science*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 496–508.
4. Rasti, P.; Uiboupin, T.; Escalera, S.; Anbarjafari, G. Convolutional neural network super resolution for face recognition in surveillance monitoring. In *International Conference on Articulated Motion and Deformable Objects*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 175–184.
5. Xu, W.; Guangluan, X.; Wang, Y.; Sun, X.; Lin, D.; Yirong, W. High quality remote sensing image super-resolution using deep memory connected network. In Proceedings of the IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 8889–8892.
6. Ma, W.; Pan, Z.; Yuan, F.; Lei, B. Super-resolution of remote sensing images via a dense residual generative adversarial network. *Remote Sens.* **2019**, *11*, 2578. [\[CrossRef\]](#)
7. Gong, Y.; Liao, P.; Zhang, X.; Zhang, L.; Chen, G.; Zhu, K.; Tan, X.; Lv, Z. Enlighten-GAN for Super Resolution Reconstruction in Mid-Resolution Remote Sensing Images. *Remote Sens.* **2021**, *13*, 1104. [\[CrossRef\]](#)
8. Sajjadi, M.S.M.; Schölkopf, B.; Hirsch, M. EnhanceNet: Single Image Super-Resolution through Automated Texture Synthesis. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4501–4510.
9. Wang, P.; Wang, L.; Leung, H.; Zhang, G. Super-Resolution Mapping Based on Spatial-Spectral Correlation for Spectral Imagery. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 2256–2268. [\[CrossRef\]](#)
10. Wan, W.; Guo, W.; Huang, H.; Liu, J. Nonnegative and nonlocal sparse tensor factorization-based hyperspectral image super-resolution. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 8384–8394. [\[CrossRef\]](#)



11. Li, J.; Cui, R.; Li, B.; Song, R.; Li, Y.; Du, Q. Hyperspectral image super-resolution with 1D–2D attentional convolutional neural network. *Remote Sens.* **2019**, *11*, 2859. [\[CrossRef\]](#)
12. Nguyen, K.; Sridharan, S.; Denman, S.; Fookes, C. Feature-domain super-resolution framework for Gabor-based face and iris recognition. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 2642–2649.
13. Zhou, F.; Yang, W.; Liao, Q. A coarse-to-fine subpixel registration method to recover local perspective deformation in the application of image super-resolution. *IEEE Trans. Image Process.* **2011**, *21*, 53–66. [\[CrossRef\]](#) [\[PubMed\]](#)
14. Stark, H.; Oskoui, P. High-resolution image recovery from image-plane arrays, using convex projections. *JOSA A* **1989**, *6*, 1715–1726. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Patti, A.J.; Altunbasak, Y. Artifact reduction for set theoretic super resolution image reconstruction with edge adaptive constraints and higher-order interpolants. *IEEE Trans. Image Process.* **2001**, *10*, 179–186. [\[CrossRef\]](#)
16. Huang, J.B.; Singh, A.; Ahuja, N. Single Image Super-Resolution From Transformed Self-Exemplars. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5197–5206.
17. Hardie, R.C.; Barnard, K.J.; Armstrong, E.E. Joint MAP registration and high-resolution image estimation using a sequence of undersampled images. *IEEE Trans. Image Process.* **1997**, *6*, 1621–1633. [\[CrossRef\]](#)
18. Zhang, Y.; Li, K.; Li, K.; Wang, L.; Zhong, B.; Fu, Y. Image Super-Resolution Using Very Deep Residual Channel Attention Networks. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 294–310.
19. Hui, Z.; Gao, X.; Yang, Y.; Wang, X. Lightweight Image Super-Resolution with Information Multi-distillation Network. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 2024–2032.
20. Feng, X.; Zhang, W.; Su, X.; Xu, Z. Optical Remote Sensing Image Denoising and Super-Resolution Reconstructing Using Optimized Generative Network in Wavelet Transform Domain. *Remote Sens.* **2021**, *13*, 1858. [\[CrossRef\]](#)
21. Dong, C.; Loy, C.C.; He, K.; Tang, X. Learning a deep convolutional network for image super-resolution. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 184–199.
22. Dong, C.; Loy, C.C.; He, K.; Tang, X. Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 295–307. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Kim, J.; Kwon Lee, J.; Mu Lee, K. Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1646–1654.
24. Kim, J.; Lee, J.K.; Lee, K.M. Deeply-recursive convolutional network for image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016; pp. 1637–1645.
25. Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K.M. Enhanced Deep Residual Networks for Single Image Super-Resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 1132–1140.
26. Ahn, N.; Kang, B.; Sohn, K.A. Fast, Accurate, and Lightweight Super-Resolution with Cascading Residual Network. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 252–268.
27. Hui, Z.; Wang, X.; Gao, X. Fast and Accurate Single Image Super-Resolution via Information Distillation Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 723–731.
28. Liu, J.; Tang, J.; Wu, G. Residual Feature Distillation Network for Lightweight Image Super-Resolution. In Proceedings of the European Conference on Computer Vision AIM Workshops, Glasgow, UK, 23–28 August 2020.
29. Zhu, Z.; Xu, M.; Bai, S.; Huang, T.; Bai, X. Asymmetric non-local neural networks for semantic segmentation. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October 2019; pp. 593–602.
30. Justin, J.; Alexandre, A.; Li, F.-F. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In Proceedings of the European Conference on Computer Vision; Springer: Berlin, Germany, 2016; pp. 694–711.
31. Ledig, C.; Theis, L.; Huszar, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5892–5900.
32. Yuan, Y.; Liu, S.; Zhang, J.; Zhang, Y.; Dong, C.; Lin, L. Unsupervised Image Super-Resolution using Cycle-in-Cycle Generative Adversarial Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018; pp. 701–710.
33. Zhang, H.; Yang, Z.; Zhang, L.; Shen, H. Super-resolution reconstruction for multi-angle remote sensing images considering resolution differences. *Remote Sens.* **2014**, *6*, 637–657. [\[CrossRef\]](#)
34. Chantas, G.K.; Galatsanos, N.P.; Woods, N.A. Super-resolution based on fast registration and maximum a posteriori reconstruction. *IEEE Trans. Image Process.* **2007**, *16*, 1821–1830. [\[CrossRef\]](#) [\[PubMed\]](#)
35. Dai, T.; Cai, J.; Zhang, Y.; Xia, S.T.; Zhang, L. Second-order Attention Network for Single Image Super-Resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 11065–11074.
36. Feng, X.; Su, X.; Shen, J.; Jin, H. Single space object image denoising and super-resolution reconstructing using deep convolutional networks. *Remote Sens.* **2019**, *11*, 1910. [\[CrossRef\]](#)
37. Tai, Y.; Yang, J.; Liu, X.; Xu, C. MemNet: A Persistent Memory Network for Image Restoration. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4539–4547.

38. Zhang, Y.; Tian, Y.; Kong, Y.; Zhong, B.; Fu, Y. Residual Dense Network for Image Super-Resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2472–2481.
39. Li, Z.; Yang, J.; Liu, Z.; Yang, X.; Jeon, G.; Wu, W. Feedback Network for Image Super-Resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 3862–3871.
40. Qiu, Y.; Wang, R.; Tao, D.; Cheng, J. Embedded Block Residual Network: A Recursive Restoration Model for Single-Image Super-Resolution. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 4180–4189.
41. Chu, X.; Zhang, B.; Ma, H.; Xu, R.; Li, J.; Li, Q. Fast, accurate and lightweight super-resolution with neural architecture search. *arXiv* **2019**, arXiv:1901.07261.
42. Chu, X.; Zhang, B.; Xu, R.; Ma, H. Multi-objective reinforced evolution in mobile neural architecture search. *arXiv* **2019**, arXiv:1901.01074.
43. Luo, X.; Xie, Y.; Zhang, Y.; Qu, Y.; Li, C.; Fu, Y. LatticeNet: Towards Lightweight Image Super-resolution with Lattice Block. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
44. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7174.
45. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-Local Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
46. Liu, D.; Wen, B.; Fan, Y.; Loy, C.C.; Huang, T.S. Non-Local Recurrent Network for Image Restoration. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2018; pp. 1680–1689.
47. Mei, Y.; Fan, Y.; Zhou, Y.; Huang, L.; Huang, T.S.; Shi, H. Image Super-Resolution With Cross-Scale Non-Local Attention and Exhaustive Self-Exemplars Mining. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020.
48. Niu, B.; Wen, W.; Ren, W.; Zhang, X.; Yang, L.; Wang, S.; Zhang, K.; Cao, X.; Shen, H. Single Image Super-Resolution via a Holistic Attention Network. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 191–207.
49. Liu, H.; Fu, Z.; Han, J.; Shao, L.; Hou, S.; Chu, Y. Single image super-resolution using multi-scale deep encoder–decoder with phase congruency edge map guidance. *Inf. Sci.* **2019**, *473*, 44–58. [[CrossRef](#)]
50. Ding, X.; Guo, Y.; Ding, G.; Han, J. Acnet: Strengthening the kernel skeletons for powerful cnn via asymmetric convolution blocks. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 1911–1920.
51. Wang, Z.; Liu, D.; Yang, J.; Han, W.; Huang, T. Deep networks for image super-resolution with sparse prior. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 370–378.
52. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 22–25 July 2017; pp. 4700–4708.
53. Zhang, C.; Benz, P.; Argaw, D.M.; Lee, S.; Kim, J.; Rameau, F.; Bazin, J.C.; Kweon, I.S. Resnet or densenet? introducing dense shortcuts to resnet. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Waikoloa, Hawaii, US, 5–9 January 2021; pp. 3550–3559.
54. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [[CrossRef](#)]
55. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 22–25 July 2017; pp. 2881–2890.
56. Timofte, R.; Agustsson, E.; Van Gool, L.; Yang, M.H.; Zhang, L. Ntire 2017 challenge on single image super-resolution: Methods and results. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 114–125.
57. Bevilacqua, M.; Roumy, A.; Guillemot, C.; Alberi-Morel, M.L. Low-complexity single-image super-resolution based on non-negative neighbor embedding. In Proceedings of the 2012 British Machine Vision Conference, Surrey, UK, 3–7 September 2012.
58. Zeyde, R.; Elad, M.; Protter, M. On single image scale-up using sparse-representations. In *International Conference on Curves and Surfaces*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 711–730.
59. Arbelaez, P.; Maire, M.; Fowlkes, C.; Malik, J. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *33*, 898–916. [[CrossRef](#)]
60. Gao, X.; Lu, W.; Tao, D.; Li, X. Image quality assessment based on multiscale geometric analysis. *IEEE Trans. Image Process.* **2009**, *18*, 1409–1423.
61. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [[CrossRef](#)]
62. Zhao, H.; Gallo, O.; Frosio, I.; Kautz, J. Loss functions for image restoration with neural networks. *IEEE Trans. Comput. Imaging* **2016**, *3*, 47–57. [[CrossRef](#)]
63. Dong, C.; Loy, C.C.; Tang, X. Accelerating the super-resolution convolutional neural network. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 391–407.

64. Lai, W.S.; Huang, J.B.; Ahuja, N.; Yang, M.H. Deep laplacian pyramid networks for fast and accurate super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 624–632.
65. Tai, Y.; Yang, J.; Liu, X. Image super-resolution via deep recursive residual network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3147–3155.
66. Zhang, K.; Zuo, W.; Zhang, L. Learning a single convolutional super-resolution network for multiple degradations. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3262–3271.
67. Lei, S.; Shi, Z.; Zou, Z. Super-resolution for remote sensing images via local–global combined network. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1243–1247. [[CrossRef](#)]
68. Dong, X.; Xi, Z.; Sun, X.; Gao, L. Transferred multi-perception attention networks for remote sensing image super-resolution. *Remote Sens.* **2019**, *11*, 2857. [[CrossRef](#)]
69. Dong, X.; Sun, X.; Jia, X.; Xi, Z.; Gao, L.; Zhang, B. Remote sensing image super-resolution using novel dense-sampling networks. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 1618–1633. [[CrossRef](#)]
70. Ma, Y.; Lv, P.; Liu, H.; Sun, X.; Zhong, Y. Remote Sensing Image Super-Resolution Based on Dense Channel Attention Network. *Remote Sens.* **2021**, *13*, 2966. [[CrossRef](#)]
71. Dharejo, F.A.; Deeba, F.; Zhou, Y.; Das, B.; Jatoi, M.A.; Zawish, M.; Du, Y.; Wang, X. TWIST-GAN: Towards Wavelet Transform and Transferred GAN for Spatio-Temporal Single Image Super Resolution. *arXiv* **2021**, arXiv:2104.10268.
72. Yang, Y.; Newsam, S. Bag-of-visual-words and spatial extensions for land-use classification. In Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems, San Jose, CA, USA, 2–5 November 2010; pp. 270–279.
73. Cheng, G.; Han, J.; Lu, X. Remote sensing image scene classification: Benchmark and state of the art. *Proc. IEEE* **2017**, *105*, 1865–1883. [[CrossRef](#)]