Large Kernel Frequency-enhanced Network for Efficient Single Image Super-Resolution

Jiadi Chen Chunjiang Duanmu Huanhuan Long
Zhejiang Normal University, Jinhua, China
zjnucjd@163.com duanmu@zjnu.cn huanhuanlong@zjnu.edu.cn
https://github.com/ThediidehT/LKFN

Abstract

In recent years, there has been significant progress in efficient and lightweight image super-resolution, due in part to the design of several powerful and lightweight attention mechanisms that enhance model representation ability. However, the attention maps of most methods are obtained directly from the spatial domain, limiting their upper bound due to the locality of spatial convolutions and limited receptive fields. In this paper, we shift focus to the frequency domain, since the natural global properties of the frequency domain can address this issue. To explore attention maps from the frequency domain perspective, we investigate and correct some misconceptions in existing frequency domain feature processing methods and propose a new frequency domain attention mechanism called frequency-enhanced pixel attention (FPA). Additionally, we use large kernel convolutions and partial convolutions to improve the ability to extract deep features while maintaining a lightweight design. On the basis of these improvements, we propose a large kernel frequency-enhanced network (LKFN) with smaller model size and higher computational efficiency. It can effectively capture long-range dependencies between pixels in a whole image and achieve state-of-the-art performance in existing efficient super-resolution methods.

1. Introduction

As a low-level computer vision task, single-image super-resolution (SISR) aims to reconstruct a high resolution (HR) image from its low resolution (LR) counterpart. Since SR-CNN [9] introduced deep learning to super-resolution for the first time, there has been a significant surge in the development of deep-learning-based SR models. By leveraging large amounts of data and powerful computing resources, deep learning has enabled researchers to develop increasingly sophisticated SR models that can generate high quality image from low-resolution inputs. Despite their impressive results, due to their high complexity and computational cost, traditional super-resolution networks are often difficult to use in practical applications. In this context, efficient super-resolution (ESR) networks with greatly reduced parameters and less computational complexity are gradually being introduced and developed.

Among these ESR methods, a class of methods based on information distillation paradigm have been verified effective, which consist of three parts: feature distillation, feature condensation and feature enhancement. For the feature distillation part, IMDN [18] first introduced a progressive refinement module to reduce computational cost and achieve multi-level feature map fusion by splitting channels. RFDN [26] further introduced shallow residual block (SRB) and applied channel compression to greatly reduce the number of model parameters. By rethinking the design of SRB, BSRN [25] introduced the blueprint separable convolution (BSConv) to replace the vanilla $3 \times 3$ convolution and the GELU [16] activation function was used instead of ReLU, which achieved remarkable results. LKDN [43] used the technique of reparameterization [6, 49] to fur-
ther improve the representation capability of BSConv with zero additional inference overhead. In the part of feature enhancement, all the aforementioned methods used some form of attention mechanism, from spatial attention, channel attention, pixel attention, to their combinations, such as IMDN, RFDN’s contrast-aware channel attention (CCA), BSRN’s enhanced spatial attention (ESA) [27] plus CCA, LKDN’s large kernel attention (LKA), and MDRN’s multi-level dispersion spatial attention (MDSA) plus enhanced contrast-aware channel attention (ECCA) [31]. These attention mechanisms have shown remarkable effects in preserving model accuracy while keeping the model lightweight.

In this paper, we further explore the potential of the feature distillation and feature enhancement parts as well as how to make them work better together to adapt to the super-resolution task. Keeping the information distillation framework unchanged, we propose a new spatial feature extraction block with a larger convolution kernel to replace BSConv, and a novel attention mechanism based on frequency domain image processing to realize feature enhancement. We refer to this new ESR method as the large kernel frequency-enhanced network (LKFN). Extensive experiments demonstrate that our LKFN better balances the accuracy and complexity of the model, and achieves the state-of-the-art performance among existing ESR methods (See Fig. 1). The contributions of this paper can be summarized as follows:

- We introduce larger kernel convolution to the basic feature extraction block, which provides a larger receptive field while maintaining lightweight.
- We propose a brand new attention mechanism that is completely based on frequency domain processing, which can truly achieve a global view of the whole image and is more flexible for different scales.
- The proposed LKFN achieves better super-resolution performance in a more concise and efficient manner.

2. Related Works

2.1. Exploration of Efficient Super-Resolution

Numerous approaches have been explored and achieved effective results in reducing the computational complexity of deep-learning-based super-resolution methods in various aspects. FSRCNN [10] proposed a network paradigm that places the upsampling step in the last stage, replacing the enormous computational cost incurred by SRCNN [9] which processes the upscaled input image from interpolation. The sub-pixel convolutional upsampling method proposed by ESPCN [37] has been widely adopted as an upsampling module due to its exceptional performance. DRCN [19] proposed a deep recursive convolutional network to increase the depth of the network, ensuring effectiveness while reducing the burden of too many parameters. CARN [2] used group convolutions and a cascading mechanism on residual networks to improve efficiency. ASSLN [50] proposed an aligned structured sparsity learning strategy, which successfully introduced the filter pruning technique in the SR models. DIPNet [46] integrated reparameterization, filter pruning, and knowledge distillation techniques, and won the championship in inference speed in the NTIRE 2023 efficient super-resolution challenge [24].

2.2. The Renaissance of Large Kernel Convolution

In the early days of CNN models, large convolution kernels were commonly used (e.g. AlexNet [21], SRCNN). This changed with the VGG model [38], which popularized the stacking of small convolution kernels ($3 \times 3$) and became the standard for CNN architecture design. However, following the emergence of transformer-based vision models [11, 29] that emphasized the importance of global receptive field, many researchers found that using larger convolution kernels in traditional CNNs [7, 8, 15, 28, 30] can achieve comparable or even better performance than transformer-based models by reducing network depth and improving feature extraction efficiency. This trend has also influenced the design of SR models. Nevertheless, simply using large convolution kernels will lead to higher computational costs. Inspired by large kernel attention (LKA) in VAN [15], convolution kernel decomposition technique have been widely adopted to address this issue, which decomposes a large convolution kernel into three parts equivalently: a depth-wise convolution, a depth-wise dilation convolution, and a $1 \times 1$ convolution. LKASR [13] imitates the framework structure of the Transformer [41] and achieves good performance by replacing the self-attention
module with LKA. VAPSR [52] uses a concise structure mainly composed of the LKA module with its attention channels amplified, it achieves excellent results with fewer parameters, proving the superiority of the LKA module. MAN [42] improves the LKA module by proposing a multi-scale large kernel attention (MLKA) that combines multiple scales in parallel and integrates it with a gated spatial attention. LKDN [43] applies the LKA to the effective information distillation framework and achieves SOTA performance. These methods all incorporate large kernel convolution into attention mechanisms to enhance feature representation. However, our proposed LKFN finds that using large kernel convolution directly in the feature extraction process can achieve significant improvement as well.

2.3. Frequency Domain Methods in CV

The Fourier transform has long been an essential tool in digital image processing. In deep learning-based vision tasks, a variety of works have attempted to incorporate it in their model design because according to the convolution theorem, point-wise update in the frequency domain globally affects all input features involved in Fourier transform. This property has a natural global attribute. FFC [5] replaces the convolution in CNNs with a local Fourier unit and performs convolutions in the frequency domain, which can complementarily address different scales. GFNet [35] proposes a global filter network that performs element-wise multiplication between frequency domain features and learnable global filters. SpectFormer [34] combines spectral and multi-headed self-attention in the original ViT [11] architecture to obtain a better representation ability. For image super-resolution, FNNSR [22] proposes a neural network design that operates entirely in the frequency domain. It takes a bicubic-upsampled image as input, transforms it into the frequency domain, and then performs element-wise multiplication using weight matrices of the same size, to achieve the effect of non-linear activation, they utilize the frequency-domain convolution. IFNNSR [45] improves FNNSR by dividing its weight matrices into four quadrants and sharing parameters, which reduces the number of parameters and improves the speed. These two methods are entirely based on the frequency domain and are too radical. Their actual performance is far behind all spatial domain methods and is only slightly better than the interpolation methods. Inspired by FFC [5], SwinFIR [48] integrates frequency-domain fast Fourier convolution with spatial domain convolution into a complementary dual-branch structure module, and embeds it into the SwinIR [29] framework, achieving impressive results. Other methods like ShuffleMixer [39] and SAFMN [40], different from directly adding frequency-domain processing modules in the model structure, they instead add frequency-domain constraints to the loss function. Our LKFN explores the combination of frequency-domain methods and attention mechanisms, proposes a frequency-enhanced pixel attention, and explains why some traditional frequency-domain operations are not suitable for super-resolution tasks.

3. Method

3.1. Rethinking Frequency-domain Operations

As we know, convolution operations in the spatial domain are equivalent to element-wise multiplication in the frequency domain. To enjoy the advantages of global view in the frequency domain, it is reasonable to use a learnable parameter matrix as a global filter in GFNet [35]. However, this method is not suitable for SR for two reasons. First, the learnable weight matrix size is fixed, that is, $C \times H \times W$, the same size as the input feature, which means it is only applicable to networks with fixed input size like image classification, object detection or semantic segmentation models while SR networks receive inputs of arbitrary resolution. Second, even if the input size can be fixed in some way, the
number of parameters in the weight matrix is large enough, greatly increasing the model size.

SwinFIR [48] explored directly replacing the vanilla convolution with the Fourier unit in the FFC [5]. However, the results did not improve the SR performance as expected, and decreased instead, which seems to contradict the theoretical advantages brought by the global view in the frequency domain. So they added spatial residual blocks to form a dual-branch structure, which improved the performance. However, it is difficult to determine how much of the effect is due to the frequency-domain processing in this design.

We studied the specific operation of the Fourier unit and found the problem. When the Fourier transform is applied to real numbers, each element of the frequency-domain feature map is a binary tuple consisting of real and imaginary parts. Since mainstream deep learning frameworks do not support direct operations on complex numbers, the Fourier unit uses a method of stacking real and imaginary parts in the channel direction and then using a $1 \times 1$ convolution to process across the doubled channels (the left part in Fig.4). Here lies the problem. This processing method causes data exchange between real and imaginary parts, which greatly destroys the phase angle, fundamentally disrupting the spatial structure and feature localization of the image. Super-resolved images obtained in this way have obvious louver-like artifacts, as shown in Fig.3. Considering that the FFC is designed for image classification, this method would cause a severe decrease in performance when directly applied to SR. So, we made an improvement, which is to isolate the data communication between the real and imaginary parts, see Fig.4. We use the same convolution to process the real and imaginary parts separately, which also avoids the increase in parameters caused by doubling the number of channels. The super-resolved images immediately returned to normal, and the artifacts disappeared.

### 3.2. Frequency-enhanced Pixel Attention

Through rethinking and improving the frequency-domain operations in SR, we can further explore the advantages brought by frequency-domain methods. Therefore, we combined frequency-domain processing with the attention mechanism and proposed the Frequency-enhanced Pixel Attention (FPA). Normally, attention maps are extracted from feature maps in the spatial domain. Due to the locality of the convolution operator, learning the correlation between pixel locations in the spatial domain can only cover a small range, which greatly reduces the effectiveness of attention mechanisms. Although using larger convolution kernels can alleviate this problem to some extent, it can not truly achieve the global attention like self-attention, while bringing larger computational costs and larger model sizes.

In our FPA, see Fig.5, we first use the fast Fourier transform $fft(\cdot)$ to convert the spatial domain feature map with shape $C \times H \times W$ into the frequency domain, obtaining a frequency-domain feature map of shape $C \times H \times \lfloor W/2 \rfloor + 1$. Since the Fourier transform of a 2D real signal is a Hermitian matrix which is conjugate symmetric, so half of the information is redundant. The frequency domain feature map is then processed by a three-layer $1 \times 1$ convolution, followed by two LeakyReLUs, and a residual connection is added with the initial frequency domain feature map. Then the pixel attention map is obtained by inverse fast Fourier transform $ifft(\cdot)$ back to the spatial domain, and multiplied by the initial input $F$. This process can be expressed as follows:

$$F_{attention} = ifft(fft(F) + f\epsilon(fft(F)))$$
$$F_{enhanced} = F_{attention} \otimes F$$

where $f\epsilon(\cdot)$ denotes the module of frequency-domain enhancement with the three-layer $1 \times 1$ convolution, $F_{attention}$ denotes the pixel attention map, $\otimes$ denotes element-wise product operation.

### 3.3. Large Kernel Frequency-enhanced Block

The specific architecture is shown in Fig.6. Inspired by LKDB in KDN [43], we design a large kernel frequency-enhanced block (LKFB), which incorporates our powerful FPA module. On the other hand, inspired by large kernel convolutions and PConv [4], we propose the Partial Large Kernel Block (PLKB) to replace the RBSB in LKDB. In order to extract more hierarchically rich feature maps and cope with the increased parameters and computation caused by the larger convolution kernel, we further use partial convolution to reduce the channels. The finer-grained feature maps obtained in this way, combined with the global attention brought by FPA, enable the proposed LKFB to achieve comparable or even better performance in a lightweight manner. For the input $F_{in}$, feature distillation is performed first, the process can be expressed as

$$F_{d1}, F_{r1} = D_1(F_{in}), PLKB_1(F_{in}),$$
$$F_{d2}, F_{r2} = D_2(F_{r1}), PLKB_2(F_{r1}),$$
$$F_{d3}, F_{r3} = D_3(F_{r2}), PLKB_3(F_{r2}),$$
$$F_{r4} = BSConv(F_{r3}).$$
$PLKB_i(F) = Conv_{1 \times 1}(Conv_{DW}(F_{split_1}), F_{split_2})$, \hspace{1cm} (4)

where $D_i$, $PLKB_i$ denotes the $i$th distillation ($1 \times 1$ conv) and $i$th refinement layer using the proposed PLKB, respectively. $F_{d_i}$, $F_{r_i}$ represents the $i$th distilled feature and $i$th refined feature, respectively. $BSConv$[25] is used as the last refinement layer. In the PLKBs, $Conv_{DW}$ denotes a $5 \times 5$ depth-wise convolution. $F_{split_1, 2}$ represent the split two parts of the input feature. $(\cdot, \cdot)$ means concatenating two parts in the channel dimension. Subsequently, the distilled features from the distillation layers and the final refinement output are concatenated and fused with a $1 \times 1$ convolution:

$F_{fused} = Conv_{1 \times 1}(Concat(F_{d_1}, F_{d_2}, F_{r_1}, F_{r_2}))$, \hspace{1cm} (5)

Next, the fused feature map undergoes image enhancement through FPA module, followed by a layer of $1 \times 1$ convolution, and finally normalized through Pixel Normalization [52]:

$F_{enhanced} = \text{PixelNorm}(Conv_{1 \times 1}(FPA(F_{fused})))$, \hspace{1cm} (6)

Finally, a residual connection within the block is connected with the input to enhance the learning ability of the deep model:

$F_{out} = F_{enhanced} + F_{in}$. \hspace{1cm} (7)

### 3.4. Network Architecture

Follow LKDN, our approach copy the original input image $I_{LR}$ $n$ times and stack them along the channel direction to obtain $I_{LR}^{n}$, then map it to the feature space through a $3 \times 3$ BSConv to obtain $F_0$:

$F_0 = BSConv(I_{LR}^{n})$, \hspace{1cm} (8)

Then we feed $F_0$ into $m$ stacks of LKFBs to extract deep features. The output of each module in the middle is stacked together and undergoes channel compression through a $1 \times 1$ convolution and then through a GELU activation layer and a $3 \times 3$ BSConv. After that, a skip connection is used to enhance global residual learning and fuse $F_0$ and $F_m$. This process can be formulated as:

$F_k = f_{LKFB}^k(\cdots f_{LKFB}^1(F_0)\cdots), 1 \leq k \leq m,
F_{fusion} = BSConv((\text{GELU}(\text{Concat}(F_1, \cdots, F_m)))$, \hspace{1cm} (9)

$F_{df} = F_{fusion} + F_0$,

where $f_{LKFB}^k(\cdot)$ denotes the $k$th LKFB, $m$ is the number of used LKFBs, $F_k$ and $F_{df}$ represent the output feature of the $k$th module and the final deep feature respectively. In the final image reconstruction stage, deep feature is transformed by a vanilla $3 \times 3$ convolution to a specific number of channels, and then the super-resolved image is obtained through pixel-shuffle operation [37]:

$I_{SR} = \text{PixelShuffle}(Conv_{3 \times 3}(F_{df}))$. \hspace{1cm} (10)

### 4. Experiments

#### 4.1. Datasets and Metrics

The training dataset consists of 800 images from DIV2K[1] and first 10K images from LLSR[23]. Our evaluation of the models is performed on commonly used benchmark datasets, including Set5[3], Set14[47], B100[32], Urban100[17], and Manga109[33]. The training data was augmented with random horizontal flips and 90-degree rotations. The evaluation metrics used are the av-

![Image 6. The architecture of Large Kernel Frequency-enhanced Block (LKFB).]
Table 1. Ablation study on frequency-enhanced pixel attention.

<table>
<thead>
<tr>
<th>Method</th>
<th>Params[k]</th>
<th>Set5</th>
<th>Set14</th>
<th>B100</th>
<th>Urban100</th>
<th>Manga109</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>259</td>
<td>37.84 / 0.9604</td>
<td>33.61 / 0.9178</td>
<td>32.14 / 0.8994</td>
<td>32.09 / 0.9280</td>
<td>38.28 / 0.9767</td>
</tr>
<tr>
<td>baseline+LKA</td>
<td>308</td>
<td>37.95 / 0.9605</td>
<td>33.75 / 0.9187</td>
<td>32.22 / 0.9003</td>
<td>32.41 / 0.9311</td>
<td>38.76 / 0.9775</td>
</tr>
<tr>
<td>baseline+MDSA+ECCA</td>
<td>449</td>
<td>37.99 / 0.9606</td>
<td>37.80 / 0.9193</td>
<td>32.22 / 0.9004</td>
<td>32.52 / 0.9316</td>
<td>38.74 / 0.9775</td>
</tr>
<tr>
<td>baseline+FPA (LKFN)</td>
<td>291</td>
<td>37.88 / 0.9603</td>
<td>33.78 / 0.9189</td>
<td>32.21 / 0.9003</td>
<td>32.49 / 0.9315</td>
<td>38.72 / 0.9772</td>
</tr>
</tbody>
</table>

4.2. Implementation Details

LKFN consists of 8 LKFBs with the feature channel number set to 56. The mini-batch size and input patch size for each LR input are set to 64 and 64 × 64, respectively. We train the model using the common $L_1$ loss function and the Adam optimizer [44] with default settings. The initial learning rate is set to $5 \times 10^{-3}$. The learning rate decay is following cosine annealing with $T_{max} = \text{total iterations}$, $\eta_{min} = 1 \times 10^{-7}$. The total number of iterations is 1000K.

A mini version of our LKFN, called LKFN-S, was designed for the NTIRE 2024 Efficient SR Challenge [36]. It consists of 8 LKFBs and the feature channel is set to 28. We set the dilation ratio of the $5 \times 5$ depth-wise convolution to 3 in the third PLKB in LKFBs. The training process includes 2 stages: (1) Training with a input patch size of $64 \times 64$ and a mini-batch size of 64 from scratch by minimizing the $L_1$ loss. The learning rate schedule is the same as the standard LKFN and the total number of iterations is 1000K. (2) Fine-tuning with a input patch size of $120 \times 120$ and a mini-batch size of 64 by minimizing the MSE loss. The learning rate is set to $2 \times 10^{-5}$ during this stage. The total number of iterations is 150K.

We implement all our models using PyTorch 2.0.1 and a NVIDIA GeForce RTX 4090 GPU.

4.3. Ablation Study

Effectiveness of the FPA module. To verify the effectiveness of our FPA module and compare it with other attention modules, we use LKFN without the FPA module as the baseline, and compared with the attention mechanisms of LKA and MDSA+ECCA in two SOTA models LKDN and MDRN respectively. The results are shown in Tab 1. Obviously, the performance on each benchmark of the baseline is far behind the models with attention modules. Except for Set5, the improvement brought by our FPA module is significant. We think the local features play a more important role in Set5. On the other 4 benchmarks, comparing with the MDSA+ESA method, we achieved comparable performance with only 65% parameters. With slightly fewer parameters, we exceed the performance of LKA method, demonstrating the benefits of the frequency-domain global view. The local attribution maps (LAMs) [14] and diffusion indices (DIs) [14] results are shown in Fig.8. The first three models are based on obtaining attention maps in the spatial domain, which rely more on the surrounding pixels of the target. In the LAM of LKFN, besides the red-boxed region, the pixels of the entire image have almost the same contribution with a slight red color tone. This validates that the attention map obtained from the frequency domain has a global view.

Study of design in FPA. During the development of the FPA module, we tried other possibilities. Fig. 7(c) does not use an attention mechanism at all. Fig. 7(a) does not use an attention mechanism but instead adds the feature maps. Fig.7(b) completely abandons frequency-domain process-
Table 2. Ablation study on different FPA design.

<table>
<thead>
<tr>
<th>Method</th>
<th>Set5</th>
<th>Set14</th>
<th>B100</th>
<th>Urban100</th>
<th>Manga109</th>
</tr>
</thead>
<tbody>
<tr>
<td>non-attention</td>
<td>37.86 / 0.9602</td>
<td>33.66 / 0.9180</td>
<td>32.16 / 0.8996</td>
<td>32.25 / 0.9292</td>
<td>38.60 / 0.9771</td>
</tr>
<tr>
<td>non-attention (add)</td>
<td>37.88 / 0.9603</td>
<td>33.74 / 0.9187</td>
<td>32.17 / 0.8998</td>
<td>32.33 / 0.9299</td>
<td>38.66 / 0.9773</td>
</tr>
<tr>
<td>spatial-enhancement</td>
<td>37.89 / 0.9603</td>
<td>33.73 / 0.9185</td>
<td>32.18 / 0.8998</td>
<td>32.21 / 0.9291</td>
<td>38.42 / 0.9769</td>
</tr>
<tr>
<td>standard FPA</td>
<td>37.88 / 0.9603</td>
<td>33.78 / 0.9189</td>
<td>32.21 / 0.9003</td>
<td>32.49 / 0.9315</td>
<td>38.72 / 0.9772</td>
</tr>
</tbody>
</table>

Table 3. Ablation study on PLKB.

<table>
<thead>
<tr>
<th>Method</th>
<th>Params[k]</th>
<th>Set5</th>
<th>Set14</th>
<th>B100</th>
<th>Urban100</th>
<th>Manga109</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSRB</td>
<td>305</td>
<td>32.24 / 0.8960</td>
<td>28.87 / 0.7832</td>
<td>27.61 / 0.7375</td>
<td>26.22 / 0.7891</td>
<td>30.69 / 0.9106</td>
</tr>
<tr>
<td>RBSB</td>
<td>305</td>
<td>32.29 / 0.8963</td>
<td>28.70 / 0.7837</td>
<td>27.63 / 0.7377</td>
<td>26.27 / 0.7906</td>
<td>30.76 / 0.9113</td>
</tr>
<tr>
<td>PLKB</td>
<td>309</td>
<td>32.34 / 0.8963</td>
<td>28.71 / 0.7836</td>
<td>27.65 / 0.7385</td>
<td>26.35 / 0.7930</td>
<td>30.80 / 0.9119</td>
</tr>
</tbody>
</table>

4.4. Comparison with State-of-the-art Methods

We compare our LKFN with several state-of-the-art efficient super-resolution models on 2×, 3×, and 4× scales, and the quantitative results are shown in Table 4. As we just analyzed, our method stands out on Urban100. We also made some interesting findings when considering the results across different scale factors. As scale factor decreases, our method’s leading advantage on Urban100 gradually increases. We believe that obtaining attention maps from the spatial domain always involves defining the kernel size, stride, dilation rate of the convolution kernels and pooling layer size (if exist) in the attention module in advance. For convenience, we usually optimize and decide the structural hyperparameters of the model only on one scale (commonly 4×) during the model development stage and then apply them directly to other scales. This leads to the optimal structure at one scale factor not necessarily being optimal at other scale factors. In contrast, our method uses a 1×1 convolution uniformly after Fourier transform processing, making it more adaptable and flexible in handling different scale factors. Qualitative comparisons on visual results can be found in Fig. 9, where it can be clearly observed that our method exhibits the best performance for this type of repeated pattern structure.

4.5. NTIRE 2024 Efficient SR Challenge

The aim of this challenge [36] is to devise a network that reduces one or several metrics such as runtime, parameters, and FLOPs of the baseline RLFN [20], while maintaining PSNR of around 26.90 dB on the DIV2K_LSDIR_valid dataset, and 26.99 dB on the DIV2K_LSDIR_test dataset.

Our solution, LKFN-S, for the NTIRE 2024 Efficient SR Challenge has proven to be both efficient and effective for super-resolution tasks, achieving competitive performance with just 90K parameters and 5.51G FLOPs for SR×4. We won the 3rd place and 4th place in the Parameters sub-track and FLOPs sub-track, respectively.

5. Conclusion

In this paper, we propose the large kernel frequency-enhanced network (LKFN) that adopts the framework design of LKDN. We directly introduce large kernel convolution into the deep feature extraction module and combine it with partial convolution to better preserve information brought by large receptive fields from different levels, while effectively controlling model complexity. We also propose a frequency-domain-based pixel attention mechanism. It
Table 4. Quantitative comparison (average PSNR/SSIM) with state-of-the-art methods, and multiply-accumulate operations is evaluated
and rigorous analysis, our LKFN achieves SOTA in terms
attention maps. Through comparisons with other methods
achieve a global receptive field, improving the quality of
not only has a simple and compact structure but can truly
achieve a global receptive field, improving the quality of
attention maps. Through comparisons with other methods
and rigorous analysis, our LKFN achieves SOTA in terms
of parameters, Multi-Adds operations, and model perfor-
mance, while achieving a balance between performance and
complexity. In addition, a variant of our LKFN, LKFN-S,
participated in the NTIRE 2024 efficient super-resolution
challenge and won the third place in the FLOPs sub-track.


[40] Long Sun, Jiangxin Dong, Jinhui Tang, and Jinshan Pan. Spatially-adaptive feature modulation for efficient image super-resolution. In ICCV, 2023. 3, 8


[43] Chengxing Xie, Xiaoming Zhang, Linze Li, Haiteng Meng, Tianlin Zhang, Tianrui Li, and Xiaole Zhao. Large kernel distillation network for efficient single image super-resolution. In CVPRW, pages 1283–1292, 2023. 1, 3, 4, 8


