FSR: A General Frequency-Oriented Framework to Accelerate Image Super-resolution Networks

Jinmin Li[1,2], Tao Dai[2*], Mingyan Zhu[1,4], Bin Chen[3,4], Zhi Wang[1], Shu-Tao Xia[1,4]

[1] Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen, China
[2] College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China
[3] Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China
[4] Research Center of Artificial Intelligence, Peng Cheng Laboratory
{ljm22,zmy20}@mails.tsinghua.edu.cn, daitao.edu@gmail.com, chenbin2021@hit.edu.cn,
{wangzhi,xiast}@sz.tsinghua.edu.cn

Abstract

Deep neural networks (DNNs) have witnessed remarkable achievement in image super-resolution (SR), and plenty of DNN-based SR models with elaborated network designs have recently been proposed. However, existing methods usually require substantial computations by operating in spatial domain. To address this issue, we propose a general frequency-oriented framework (FSR) to accelerate SR networks by considering data characteristics in frequency domain. Our FSR mainly contains dual feature aggregation module (DFAM) to extract informative features in both spatial and transform domains, followed by a four-path SR-Module with different capacities to super-resolve in the frequency domain. Specifically, DFAM further consists of a transform attention block (TABlock) and a spatial context block (SCBlock) to extract global spectral information and local spatial information, respectively, while SR-Module is a parallel network container that contains four to-be-accelerated branches. Furthermore, we propose an adaptive weight strategy for a trade-off between image details recovery and visual quality. Extensive experiments show that our FSR can save FLOPs by almost 40% while reducing inference time by 50% for other SR methods (e.g., FSRCNN, CARN, SRResNet and RCAN). Code is available at https://github.com/THU-Kingmin/FSR.

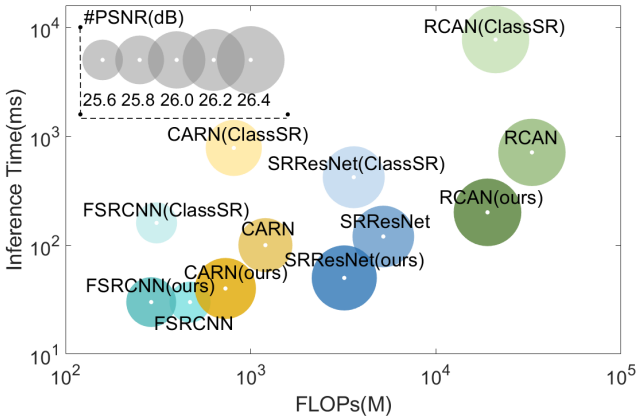

Figure 1: Comparison of PSNR results and computations of ClassSR (Kong et al. 2021) and our FSR over other SR networks on DIV8K dataset in $\times 4$ SR. ClassSR can accelerate other SR networks with fewer FLOPs, while it requires higher inference time. By contrast, our FSR succeeds in not only reducing both FLOPs and inference time, but obtaining better performance.

Introduction

Image super-resolution (SR), which pursues to reconstruct high-resolution images from their corresponding low-resolution versions, has been widely applied in various computer vision tasks like medical diagnostic imaging and object recognition. Recently, deep neural network (DNN) based SR methods have received much attention, and different advanced deep SR models with elaborated network designs have been proposed.

Generally, most existing DNN-based SR methods focus on deeper architecture design to achieve better performance. For example, Dong et al. (2014) first designed a shallow-three-layer convolutional network (SRCNN) for SR. Later, deeper VDSR (Kim, Lee, and Lee 2016) with more than 16 layers is developed based on residual learning. To further improve the performance, Lim et al. (2017) designed a very deep and wide network EDSR (Lim et al. 2017) with more than 60 layers. More recently, more advanced SR networks like RCAN (Zhang et al. 2018) and SAN (Dai et al. 2019) contain hundreds of layers and obtain impressive performance by considering feature correlation. Despite the remarkable performance, these methods usually suffer from heavy computational costs in practice (see Fig. 1), which hinders the applications of SR models in resource-constrained devices. For example, as shown in Fig. 1, RCAN requires high computational costs of 32.6G FLOPs for input images with 32×32 cropped from DIV8K dataset.

To accelerate SR networks, ClassSR (Kong et al. 2021) has been recently developed based on the observation that different image regions contain different reconstruction difficulties. In this way, SR networks can be accelerated by handling image regions with different capacities. Although

*Corresponding author: Tao Dai

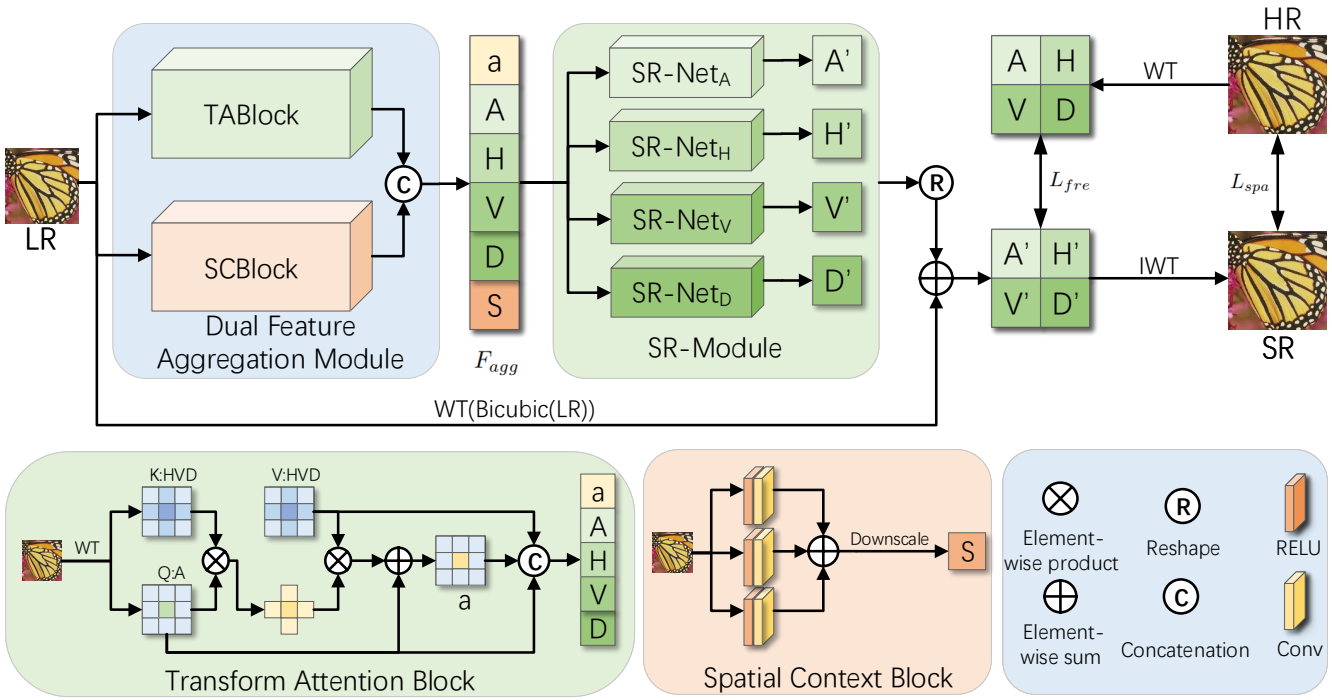

Figure 2: The overall framework of the proposed FSR. WT: Wavelet transform. IWT: Inverse wavelet transform. SR-Net: Existing SR networks. [A, H, V, D]: Wavelet coefficients of four sub-bands. F_{agg}: Aggregation features. L_{fre}: Frequency reconstruction loss. L_{spa}: Spatial reconstruction loss. WT(Bicubic(LR)): Upscale and then wavelet transform. A': $\hat{Y}^A$. H': $\hat{Y}^H$. V': $\hat{Y}^V$. D': $\hat{Y}^D$.

reducing FLOPs, ClassSR still suffers from high inference time by performing classification of numerous image regions in the spatial domain (see Fig. 1), which limits its usage in real-world applications. On the other hand, it is well-known that natural images can be sparsely represented in a frequency domain. Specifically, smooth image contents would lie in low-frequency spectra, while those edge and texture structures would correspond to high-frequency spectra. Surprisingly, we can roughly classify image structures by converting the spatial images into the frequency domain.

The above observations naturally motivate us to handle different frequency spectra with different capacities. To this end, in this paper, we propose a general frequency-oriented framework (FSR) to accelerate SR networks by operating in the frequency domain. As shown in Fig. 2, our FSR mainly consists of a dual feature aggregation module (DFAM) and a four-path SR-Module. Our FSR first converts input images into different frequency sub-bands through DFAM, followed by SR networks with different capacities to process. Specifically, DFAM is composed of a transform attention block (TABlock) and a spatial context block (SCBlock) to capture frequency global features and spatial local features. In this way, DFAM can produce more semantic features by fusing complementary frequency global and spatial local features. After that, SR-Module with different capacities is used to handle different frequency sub-bands to accelerate SR networks. Besides, to achieve a trade-off between the textures and pixel-wise reconstruction, we propose an adaptive weight strategy for network training to control the recovery of image structures.

In summary, the main contributions are as follows:

- We propose FSR, a general and efficient framework, to accelerate SR networks. To the best of our knowledge, it is the first frequency-based general framework to accelerate existing SR networks.
- We propose dual feature aggregation module to generate semantic features, which contain complementary global frequency and local spatial information by a transform attention block and a spatial context block. Besides, we propose an adaptive weight strategy to control the image structure recovery during training.
- Extensive experiments demonstrate the effectiveness of our FSR in accelerating SR networks by significantly reducing both FLOPs and inference time, while enhancing the model performance.

Related Works

Deep Image Super-resolution

Deep image SR networks have achieved excellent performance based on deep learning. Among them, SRCNN (Dong et al. 2014) is the pioneered work that applies convolutional neural networks for image SR. Later, deeper networks like VDSR (Kim, Lee, and Lee 2016) and SRRes-

Net (Ledig et al. 2017) stack many convolutional layers with residual learning to facilitate the network training. Recently, other advanced methods like RCAN (Zhang et al. 2018) and SAN (Dai et al. 2019) build very deep networks with channel attention to enhance the feature expression ability. More recently, transformer-based SwinIR (Liang et al. 2021) has obtained remarkable performance for image restoration by modelling long-range dependency. Despite the impressive performance, these methods suffer from high computational costs, thus hindering their usage in real applications. Instead, we propose an efficient frequency-oriented framework (FSR) to accelerate existing SR networks by reducing FLOPs and inference time.

Lightweight Super-resolution

Recently, a number of lightweight SR works have been proposed to relieve the problem of computational cost (Dong, Loy, and Tang 2016; Hui et al. 2019; Luo et al. 2020). For example, FSRCNN (Dong, Loy, and Tang 2016) attempts to upsample images at the end of the networks to reduce the size of intermediate layers. Later, CARN (Ahn, Kang, and Sohn 2018) designs efficient cascading residual block with group convolution, while IMDN (Hui et al. 2019) utilizes channel split to learn information in the multi-distillation network. To further reduce computations, the lightweight frequency-aware network (FADN) (Xie et al. 2021) is developed to restore distinct frequency signals with different operations. Instead of designing a lightweight SR network, a general framework (ClassSR) (Kong et al. 2021) is developed to accelerate SR networks. Among them, ClassSR is most related to our method with the same goal of accelerating SR networks, but there are distinct differences. Specifically, ClassSR operates in the spatial domain and requires a pre-trained classification network to realize image region classification before feeding to SR networks, thus leading to high inference time (See Fig. 1). By contrast, our method is more efficient by operating in the frequency domain without the need of a classification network.

Wavelet-based Super-resolution

Transforming images to the frequency domain has recently received much attention in various computer vision tasks, including image SR. Recently, several wavelet-based works are proposed for image super-resolution. Among them, Wavelet-SRNet (Huang et al. 2017) learns wavelet coefficients independently to solve the face SR problem, while SRCliqueNet (Zhong et al. 2018) studies the relationship among the four sub-bands of wavelet transform. Unlike these methods for the purpose of better performance, our method focuses on designing an efficient and general framework to accelerate SR networks based on 2D fast wavelet transform (FWT) (see Fig. 3).

Methodology

Observation

It is known that image contents with different characteristics correspond to different frequency spectra. Specifically, smooth area contents would lie in low-frequency spectra, while edge and texture structures would correspond to high-frequency spectra. Such observations motivate us to handle different frequency spectra with SR networks with different capacities.

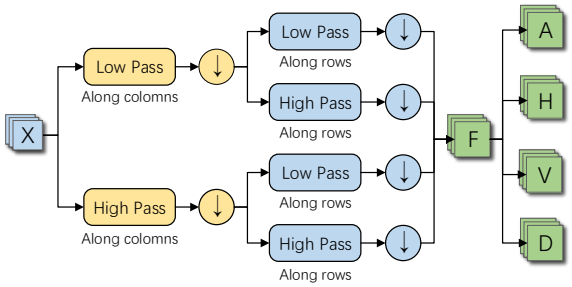

Figure 3: Illustration of 2D fast discrete wavelet transform (FDWT (Mallat 1989)). We use two Haar-based fillers iteratively to compute wavelet coefficients of images, where $Low\ Pass = (1/2, 1/2)$, $High\ Pass = (1/2, -1/2)$.

Model	Feature Channels	FLOPs	PSNR
FSRCNN	$N_A = 64, N_1 = 64$	146M	28.91
	$N_A = 64, N_1 = 56$	144M	28.87
	$N_H = 48, N_1 = 64$	110M	36.59
	$N_H = 16, N_1 = 56$	48M	36.59
	$N_V = 48, N_1 = 64$	110M	35.96
	$N_V = 16, N_1 = 56$	48M	35.95
	$N_D = 44, N_1 = 64$	102M	41.57
	$N_D = 14, N_1 = 56$	44M	41.53
	$N_T = 204, N_1 = 64$	468M	25.85
	$N_T = 110, N_1 = 56$	285M	25.83

Table 1: Results of FSRCNN with different capacities on different frequency sub-bands with $\times 4$ SR on TEST2K dataset. N_A, N_H, N_V, N_D: the number of feature channels in the middle layers for four branches. N_1: the number of feature channels in the first layer for four branches. N_T: the total of feature channels for FSRCNN.

To investigate the effect of frequency spectra for SR, we use 2D FWT to transfer the input images into four frequency sub-bands, followed by FSRCNN with varying capacities to super-resolve in the frequency domain. The results are shown in Table 1, from which we can see that FSRCNN with low capacities (e.g., $N_H = 16$) achieves similar performance to the original FSRCNN (e.g., $N_H = 48$), when performing in the high-frequency sub-bands. In other words, we can save about 40% FLOPs by processing different sub-bands for SR networks with different capacities. These observations inspire us to super-resolve frequency sub-bands with varying capacities to accelerate SR networks.

Overview of Frequency-oriented Framework

As shown in Fig. 2, our proposed general frequency-oriented framework (FSR) consists of a dual feature aggregation module (DFAM) and a SR-Module, where DFAM further contains transform attention block (TABlock) and spatial context block (SCBlock). Specifically, given an LR image X, DFAM utilizes TABlock to generate global features

$[a, A, H, V, D]$ and SCBlock to learn local features S. Then the aggregated features F_{agg} are obtained by fusing the captured global and local features. Then, F_{agg} is fed into SR-Module with four SR-Nets (f_{SR}^{A}, f_{SR}^{H}, f_{SR}^{V}, f_{SR}^{D}), and outputs four super-resolved sub-bands features ($\hat{Y}^A$, $\hat{Y}^H$, $\hat{Y}^V$, $\hat{Y}^D$), as shown in Equ. (10). After that, we reshape the four wavelet sub-bands features and add the wavelet transform features of X to it: $\hat{Y}_W = \text{Reshape}([\hat{Y}^A, \hat{Y}^H, \hat{Y}^V, \hat{Y}^D]) + \text{WT}(\text{Bicubic}(X))$. Finally, we get the SR image by inverse wavelet transform (IWT): $\hat{Y} = \text{IWT}(\hat{Y}_W)$.

Dual Feature Aggregation Module

DFAM consists of a transform attention block (TABlock) and spatial context block (SCBlock) to extract the global spectral information and local spatial information.

Transform Attention Block The architecture of TABlock is given in Fig. 2. Firstly, we apply a 2D wavelet transform (WT) as the first layer to generate four wavelet sub-bands feature, as illustrated in Fig. 3. It can explicitly decompose LR image X into a low-pass feature: A, horizontal, vertical and diagonal directions of high-frequency spectra features: H, V, D. To be specific, WT transforms X with of height $2H$, width of $2W$ and channel of C into a tensor of shape of $(H, W, 4C)$. The output of WT is denoted as

$$[A, H, V, D] = \text{WT}(X) \tag{1}$$

Then we compute the attention feature upon four wavelet sub-band features. To reduce the computations, we adopt efficient Criss-Cross attention proposed in CcNet (Huang et al. 2019). Specifically, we show how to generate the attention feature a^A of SR-Net$_\text{A}$. First, we apply three frequency feature transform networks (FFT) to generate Q, K and V, as denoted by

$$\begin{aligned} Q^A &= \text{FFT}\,(A) \\ K^A &= \text{FFT}\,([H, V, D]) \\ V^A &= \text{FFT}\,([H, V, D]) \end{aligned} \tag{2}$$

where $Q^A, K^A, V^A \in \mathbb{R}^{3\times H\times W}$. FFT contains two convolutional layers with 1 × 1 filter and one RELU layer. The Q^H, K^H, V^H, concerning the attention feature a^H of SR-Net$_\text{H}$ are similarly generated:

$$\begin{aligned} Q^H &= \text{FFT}\,(H) \\ K^H &= \text{FFT}\,([A, V, D]) \\ V^H &= \text{FFT}\,([A, V, D]) \end{aligned} \tag{3}$$

Likewise, we also generate attention feature of SR-Net$_\text{V}$ and SR-Net$_\text{D}$. After that, the residual attention operation is defined as:

$$a^i = \text{Attention}\left(Q^i, K^i, V^i\right) + Q^i \tag{4}$$

where $i \in \{A, H, V, D\}$ belongs to four branches, $a^i \in \mathbb{R}^{3\times H\times W}$ and attention operation is denoted as $a : \text{Attention}(Q, K, V)$. At each position u in the spatial dimension of output feature a_u, we calculate attention maps just at the range of criss-cross area. Thus we can obtain the related $Q_u \in \mathbb{R}^{1\times 3}$, $K_u \in \mathbb{R}^{(H+W-1)\times 3}$, $V_u \in \mathbb{R}^{(H+W-1)\times 3}$. Then each position u of attention maps a_u can be denoted as

$$\begin{aligned} a_u &= \text{Attention}(Q_u, K_u, V_u) \\ &= (Q_u \times K_u^T) \times V_u + Q_u \end{aligned} \tag{5}$$

where $a_u \in \mathbb{R}^{1\times 3}$ and T denotes transposition of matrix.

Finally, we aggregate four sub-bands features of WT and the attention feature to obtain global features, as denoted as

$$F_{global}^{i} = \text{Concat}(a^i, A, H, V, D) \tag{6}$$

where $i \in \{A, H, V, D\}$ belongs to four branches.

Spatial Context Block As shown in Fig. 2, spatial context block (SCBlock) is designed as a simple and effective network that contains three small kernel convolutional layers, three RELU layers and a downscale layer. Three RELU layers follow three convolutions layers, respectively. We adopt $1*1, 3*3, 5*5$ kernel for convolutional layers to extract local spatial feature S. Then we downscale S with factors 0.5 to keep the same size as the F_{global} of TABlock. The output of SCBlock F_{local} is obtained as

$$\begin{aligned} S =& \text{RELU}(\text{Conv}(X, 1)) + \text{RELU}(\text{Conv}(X, 3)) \\ &+ \text{RELU}(\text{Conv}(X, 5)) \end{aligned} \tag{7}$$

$$F_{local} = \text{Downscale}(S) \tag{8}$$

Finally, the aggregated feature is obtained as

$$F_{agg}^{i} = \text{Concat}(F_{global}^{i}, F_{local}) \tag{9}$$

SR-Module

The SR-Module consists of four-path of SR-Net (f_{SR}^{A}, f_{SR}^{H}, f_{SR}^{V}, f_{SR}^{D}) with different capacities. To accelerate SR networks, we simply reduce the number of feature channels in the original networks (e.g., FSRCNN and RCAN). Next, the SR-Module takes the aggregated feature F_{agg}^{A}, F_{agg}^{H}, F_{agg}^{V}, F_{agg}^{D} as input of different SR-Net. The output is denoted as

$$\hat{Y}^i = f_{\text{SR}}^{i}(F_{agg}^{i}) \tag{10}$$

where $i \in \{A, H, V, D\}$ belongs to four branches.

Model	Parameters	Test2K		Test4K		Test8K	
		PSNR	FLOPs	PSNR	FLOPs	PSNR	FLOPs
FSRCNN-O	25K	25.61	468M(100%)	26.90	468M(100%)	32.66	468M(100%)
ClassSR-FSRCNN	113K	25.61	311M(67%)	26.91	286M(61%)	32.73	**239M(51%)**
FSR-FSRCNN	154K	**25.83**	**285M(61%)**	**27.01**	**285M(61%)**	**32.75**	285M(61%)
CARN-O	295K	25.95	1.15G(100%)	27.34	1.15G(100%)	33.18	1.15G(100%)
ClassSR-CARN	645K	26.01	814M(71%)	27.42	742M(65%)	**33.24**	**607M(53%)**
FSR-CARN	745K	**26.17**	**730M(63%)**	**27.44**	**730M(63%)**	33.20	730M(63%)
SRResNet-O	1.5M	26.19	5.20G(100%)	27.65	5.20G(100%)	33.50	5.20G(100%)
ClassSR-SRResNet	3.1M	26.20	3.62G(70%)	27.66	3.30G(63%)	**33.50**	**2.69G(52%)**
FSR-SRResNet	3.66M	**26.31**	**3.24G(62%)**	**27.66**	**3.24G(62%)**	33.47	3.24G(62%)
RCAN-O	15.6M	26.39	32.6G(100%)	27.89	32.6G(100%)	**33.76**	32.6G(100%)
ClassSR-RCAN	30.1M	26.39	21.2G(65%)	27.88	19.5G(60%)	33.73	**16.4G(50%)**
FSR-RCAN	36.9M	**26.39**	**19.4G(60%)**	**27.89**	**19.4G(60%)**	33.73	19.4G(60%)

Table 2: PSNR and FLOPs on Test2K, Test4K and Test8K with 4X SR. -O: the original networks. ClassSR-: ClassSR framework with the baselines. FSR-: frequency-oriented framework with the baselines (ours). The best results are highlighted in bold.

Model	Average Inference Time(s)		
	TEST2K	TEST4K	TEST8K
FSRCNN-O	0.03	0.12	0.60
ClassSR-FSRCNN	0.16	0.69	8.90
FSR-FSRCNN	0.03	0.12	0.60
FSR*-FSRCNN	**0.03**	**0.12**	**0.58**
CARN-O	0.10	0.28	0.88
ClassSR-CARN	0.78	2.87	14.85
FSR-CARN	0.10	0.26	0.77
FSR*-CARN	**0.04**	**0.13**	**0.66**
SRResNet-O	0.12	0.26	0.85
ClassSR-SRResNet	0.42	1.40	8.59
FSR-SRResNet	0.11	0.25	0.81
FSR*-SRResNet	**0.05**	**0.15**	**0.65**
RCAN-O	0.71	1.53	4.41
ClassSR-RCAN	7.67	28.59	226.78
FSR-RCAN	0.66	1.51	4.37
FSR*-RCAN	**0.20**	**0.67**	**2.06**

Table 3: Average inference time on Test2K, Test4K and Test8K with 4X SR. -O: the original networks. ClassSR-: ClassSR framework. FSR-: frequency-oriented framework. FSR*-: Largest branch of frequency-oriented framework.

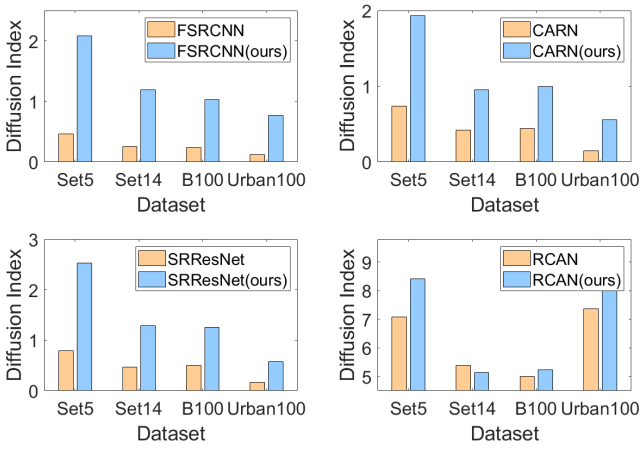

Figure 4: The quantitative comparison (average diffusion index) between baselines (FSRCNN, CARN, SRResNet and RCAN) and FSR on Set5, Set14, B100 and Urban100.

Loss Functions

Our FSR is a frequency-oriented framework that reconstructs HR images from the frequency features and spatial features in the LR images. Thus, the loss function consists of a frequency reconstruct loss and spatial reconstruct loss

$$L = L_{fre} + \alpha_{ada} L_{spa} \tag{11}$$

where α_{ada} is an adaptive weight to balance the two losses. Specifically, we apply Charbonnier loss as frequency loss L_{fre} due to its robustness to frequency noise, while L_2 Loss is applied as the spatial loss to improve the visual quality of the reconstructed images. The frequency loss L_{fre} and the spatial loss L_2 can be formulated as

$$\begin{aligned} L_{fre} =& \beta_A \|\hat{Y}^A - Y^A\|_c + \beta_H \|\hat{Y}^H - Y^H\|_c + \\ & \beta_V \|\hat{Y}^V - Y^V\|_c + \beta_D \|\hat{Y}^D - Y^D\|_c \end{aligned} \tag{12}$$

$$L_{spa} = \|\hat{Y} - Y\|_2 \tag{13}$$

where $\|\cdot\|_c$ is denoted as Charbonnier loss, $\|\cdot\|_2$ is denoted as L_2 loss, $\beta^A, \beta^H, \beta^V, \beta^D$ are hyper-parameters. Besides, $\hat{Y}^A, \hat{Y}^H, \hat{Y}^V, \hat{Y}^D$ are four sub-bands of SR image, Y^A, Y^H, Y^V, Y^D are four sub-bands of HR image, and $\hat{Y}$ is the restored image and Y is the corresponding HR image.

During training, the frequency loss L_{fre} aims to guarantee the recovery of high-frequency image details, while the spatial loss aims to improve the visual quality. To have a trade-off between image detail recovery and visual quality, in our experiments, we design an adaptive weight strategy for L_{fre} and L_{spa} as

$$\alpha_{ada} = min(\alpha_{max}, \alpha_{init} + k \frac{E_{cur}}{E_{max}} \cdot \frac{L_{spa}}{L_{fre}}) \tag{14}$$

where α_{max} is the maximum value, α_{init} is the initial value and k controls the growth rate. E_{cur} is the current epoch and E_{max} is the maximum epoch.

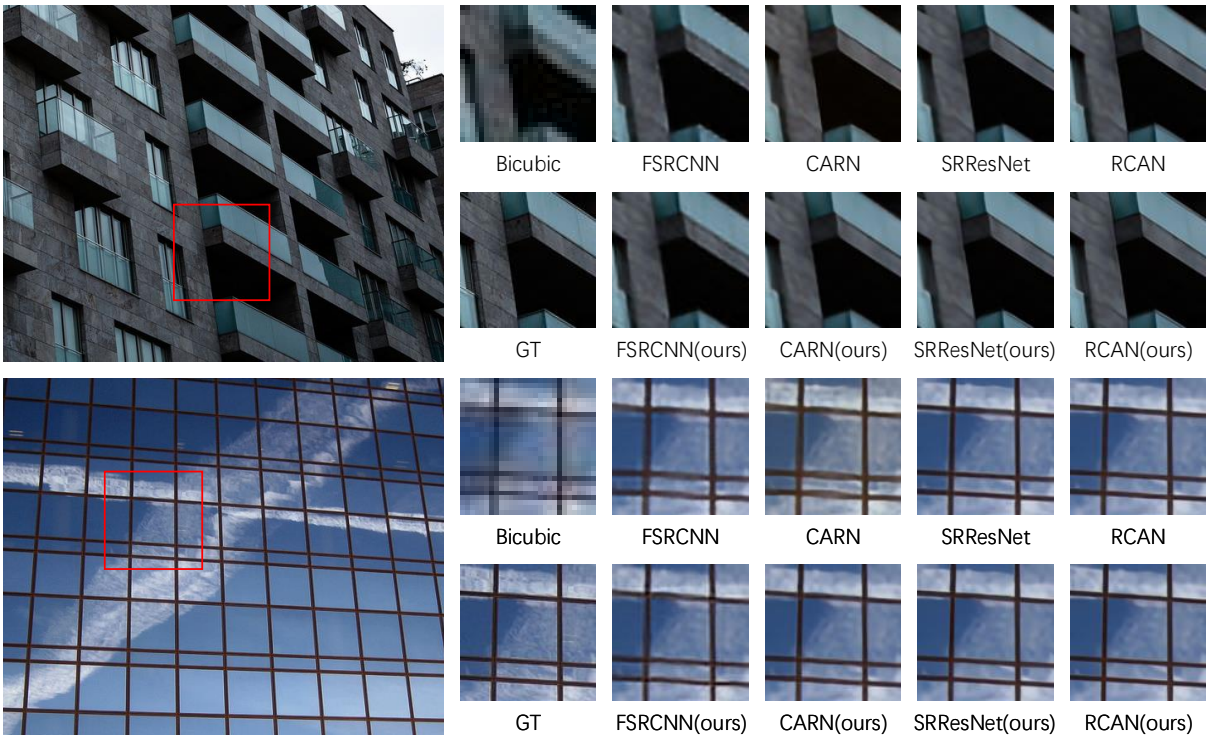

Figure 5: Visual results of our FSR and the baselines with ×4 SR. The right images are 128×128

Model	Para	FLOPs	PSNR/SSIM		
			TEST2K	TEST4K	TEST8K
FSR-FSRCNN w/o TABlock	136K(88%)	276M(97%)	25.78/0.8490	26.95/0.8809	32.58/0.9355
FSR-FSRCNN w/o SCBlock	137K(89%)	276M(97%)	25.77/0.8488	26.94/0.8806	32.67/0.9352
FSR-FSRCNN	154K(100%)	285M(100%)	25.83/0.8489	27.01/0.8820	32.75/0.9362

Table 4: Ablation study of the effect of proposed Transform Attention Block (TABlock) and Spatial Context Block (SCBlock)

Experiments

Settings

Training Data The DIV2K dataset (Agustsson and Timofte 2017) is adopted to train our FSR framework. Firstly, we prepare the HR images by downsampling the original images with scaling factors 0.6, 0.7, 0.8, 0.9. These HR images are downsampled with scaling factors 0.25 to obtain the LR images. Then we obtain the training sub-images with size 64×64 by cropping from LR images. Finally, We augment these sub-images by flipping and rotating.

Testing Data We evaluate our method with PSNR (RGB channels) and SSIM (Y channel) metrics on DIV8K (Gu et al. 2019) instead of commonly used benchmarks: Set5 (Bevilacqua et al. 2012), Set14 (Yang et al. 2010), on account of these images are too small to be decomposed. To be specific, we choose 300 images (index 1201-1500) from the DIV8K to generate TEST2K, TEST4K and TEST8K, which follows ClassSR (Kong et al. 2021). During testing, we further crop the LR images into 64×64 sub-images with stride 58. Then we combine these SR sub-images to SR images by averaging overlapping areas. Besides, We evaluate computations and inference time on DIV8K. For a fair comparison with baselines and ClassSR, all sub-images are with size 32×32 and we calculate a mean within a test set.

Training Details We apply the proposed frequency-oriented framework on FSRCNN, CARN, SRResNet and RCAN. Our FSR is trained using the ADAM (Kingma and Ba 2014) optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.99$. The batch size is set to 64. We adopt the cosine annealing learning strategy to adjust the learning rate. Specifically, the initial learning rate is set to 2×10^{-4} and the minimum is set to 1×10^{-7}. α_{max} is set to 25, α_{init} is set to 1 and k is set to 100, E_{max} is set to 4000. β_A is set to 5 whereas $\beta_H, \beta_V, \beta_D$ are set to 1.

Comparison with State-of-the-art Methods

Our FSR is a general framework to accelerate existing SR networks thus we do not compare FSR with other networks directly. Instead, we compare with State-of-the-arts framework ClassSR (Kong et al. 2021) on prominent FSRCNN, CARN, SRResNet (middle scale) and RCAN. We visualize the SR resulting images as shown in Fig. 5 including two common scenes.

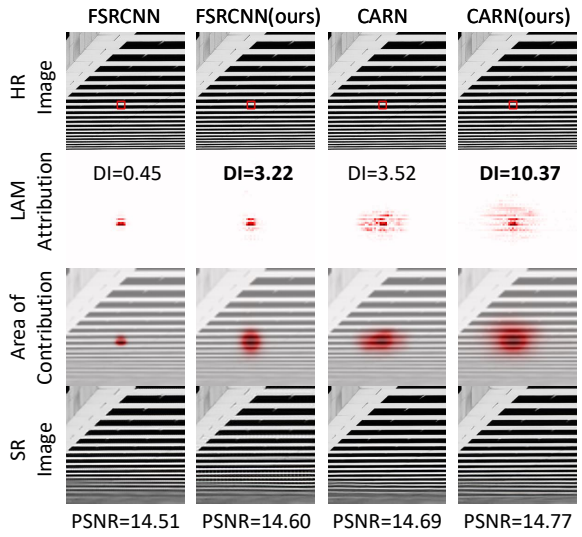

Figure 6: Visual results of Local Attribution Maps (LAM (Gu and Dong 2021)) for original networks and FSR (ours).

Performance Comparison As shown in Table 2, our FSR reduces FLOPs by average 39% while keeping and even increasing the PSNR. To be specific, the average PSNR of four baselines on TEST2K with $\times 4$ SR is improved by 0.14 dB while being improved by 0.06 dB on TEST4K. Besides, FSR achieves similar PSNR to ClassSR on TEST8K whereas the latter saves more FLOPs on TEST8K. ClassSR accelerates baselines by data characteristics. Thus it is efficient on smooth data (e.g., TEST8K) but has no significant effect on complex data (e.g., TEST2K, B100, Urban100). However, our FSR is more general and robust on any data because it is data independent and deals with images in both frequency and spatial domains.

As shown in Table 3, we also evaluate the inference time of our FSR, ClassSR and four baselines. The experimental results show that the proposed FSR reduces inference time by almost 50% while ClassSR increases four times inference time. FSR has low computations (only 6% FLOPs of the whole framework) to generate the aggregation feature of dual domain. In addition, four SR-Nets of SR-Module are mutually independent so they can be executed efficiently with high parallelism.

Interpretation with Local Attribution Maps (LAM) To better understand why our FSR achieves higher quantitative results, reduces FLOPs and inference time at the same time, we analyze the diffusion index (DI) of original networks and FSR on Set5 (Bevilacqua et al. 2012), Set14 (Yang et al. 2010), B100 (Martin et al. 2001) and Urban100. DI is applied to measure local attribution maps (LAM (Gu and Dong 2021)) from input image to output image of networks. As shown in Fig. 4, compared to four baselines, our FSR achieves three times of DI. As shown in Fig. 6, we can draw the observation that the better performance of SR networks has the higher diffusion index of LAM. Also, Our FSR leverages the advantages of frequency domain and spatial domain features which are global information and local information.

Model	FLOPs	PSNR		
		TEST2K	TEST4K	TEST8K
FSRCNN-$\times 2$	141M	30.14	31.89	38.23
FSR-$\times 2$	**85M**	**30.16**	**32.01**	**39.33**
FSRCNN-$\times 4$	468M	25.61	26.90	32.66
FSR-$\times 4$	**285M**	**25.83**	**27.01**	**32.75**
FSRCNN-$\times 8$	1.75G	23.00	23.88	28.73
FSR-$\times 8$	**942M**	**23.23**	**24.23**	**28.86**

Table 5: Evaluation of FSR in the FSRCNN networks with $\times 2$, $\times 4$, $\times 8$ upscale factors

Ablation Study

In the ablation study, we apply our FSR on FSRCNN to evaluate the effect of different modules and settings.

Dual Feature Aggregation Module First, we evaluate the effect of TABlock by removing it from FSR. As shown in Table 4, TABlock increases 12% parameters and 3% computations. Nevertheless, this block brings the benefit that the PSNR increases by 0.06 dB, 0.07 dB and 0.08 dB on TEST2K, TEST4K and TEST8K, respectively. This demonstrates that adding a lightweight TABlock is quite beneficial because it extracts shadow global information for SR-Module.

Then we evaluate the effect of SCBlock by removing it from FSR. Our experiments in Table 4 demonstrate that SCBlock improves performance significantly whereas just bears small parameters and computations. This is because it extracts shadow local information for SR-Module.

Different Upscale Factors Furthermore, we conduct experiments with different upscale factors $\times 2$, $\times 4$ and $\times 8$. Our experiments in Table 5 indicate that our FSR also achieves predominant performance on DIV8K with $\times 2$, $\times 8$ SR besides $\times 4$ SR. For instance, the FLOPs of FSR on TEST4K with $\times 8$ SR are almost reduced by 50% while PSNR is improved by 0.35 dB. The FLOPs of FSR on TEST8K with $\times 2$ SR are reduced by 40% while PSNR is improved by 1.1 dB.

Conclusion

In this paper, we develop an efficient yet general frequency-oriented framework (FSR) to accelerate SR networks. Our FSR adopts dual feature aggregation module (DFAM) to generate semantic features, followed by different SR networks with different capacities to super-resolve in the frequency domain. Specifically, DFAM captures complementary global spectral information and local spatial context information with transform attention block and spatial context block. Besides, we propose an adaptive weight strategy to achieve a proper balance with between image details recovery and visual quality. Extensive experiments demonstrate that our FSR can accelerate mainstream SR networks on public datasets by saving about 40% FLOPs and reducing about 50% inference time, while improving the model performance.

Acknowledgments

This work is supported in part by the National Natural Science Foundation of China under Grant 62171248, the Natural Science Foundation of Guangdong Province 2021A1515011807, Shenzhen Science and Technology Program (Grant No.RCYX20200714114523079, JCYJ20220818101012025) and the PCNL KEY project (PCL2021A07), and Shenzhen Science and Technology Innovation Commission (Research Center for Computer Network(Shenzhe) Ministry of Education).

References

Agustsson, E.; and Timofte, R. 2017. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 126–135.

Ahn, N.; Kang, B.; and Sohn, K.-A. 2018. Fast, accurate, and lightweight super-resolution with cascading residual network. In *Proceedings of the European conference on computer vision (ECCV)*, 252–268.

Bevilacqua, M.; Roumy, A.; Guillemot, C.; and line Alberi Morel, M. 2012. Low-Complexity Single-Image Super-Resolution based on Nonnegative Neighbor Embedding. In *Proceedings of the British Machine Vision Conference*, 135.1–135.10. BMVA Press. ISBN 1-901725-46-4.

Dai, T.; Cai, J.; Zhang, Y.; Xia, S.-T.; and Zhang, L. 2019. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11065–11074.

Dong, C.; Loy, C. C.; He, K.; and Tang, X. 2014. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, 184–199. Springer.

Dong, C.; Loy, C. C.; and Tang, X. 2016. Accelerating the super-resolution convolutional neural network. In *European conference on computer vision*, 391–407. Springer.

Gu, J.; and Dong, C. 2021. Interpreting super-resolution networks with local attribution maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9199–9208.

Gu, S.; Lugmayr, A.; Danelljan, M.; Fritsche, M.; Lamour, J.; and Timofte, R. 2019. Div8k: Diverse 8k resolution image dataset. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 3512–3516. IEEE.

Huang, H.; He, R.; Sun, Z.; and Tan, T. 2017. Wavelet-srnet: A wavelet-based cnn for multi-scale face super resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, 1689–1697.

Huang, Z.; Wang, X.; Huang, L.; Huang, C.; Wei, Y.; and Liu, W. 2019. Ccnet: Criss-cross attention for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 603–612.

Hui, Z.; Gao, X.; Yang, Y.; and Wang, X. 2019. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the 27th acm international conference on multimedia*, 2024–2032.

Kim, J.; Lee, J. K.; and Lee, K. M. 2016. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1646–1654.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kong, X.; Zhao, H.; Qiao, Y.; and Dong, C. 2021. Classsr: A general framework to accelerate super-resolution networks by data characteristic. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12016–12025.

Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. 2017. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4681–4690.

Liang, J.; Cao, J.; Sun, G.; Zhang, K.; Van Gool, L.; and Timofte, R. 2021. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1833–1844.

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

Luo, X.; Xie, Y.; Zhang, Y.; Qu, Y.; Li, C.; and Fu, Y. 2020. Latticenet: Towards lightweight image super-resolution with lattice block. In *European Conference on Computer Vision*, 272–289. Springer.

Mallat, S. G. 1989. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE transactions on pattern analysis and machine intelligence*, 11(7): 674–693.

Martin, D.; Fowlkes, C.; Tal, D.; and Malik, J. 2001. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, 416–423. IEEE.

Xie, W.; Song, D.; Xu, C.; Xu, C.; Zhang, H.; and Wang, Y. 2021. Learning frequency-aware dynamic network for efficient super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4308–4317.

Yang, J.; Wright, J.; Huang, T. S.; and Ma, Y. 2010. Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11): 2861–2873.

Zhang, Y.; Li, K.; Li, K.; Wang, L.; Zhong, B.; and Fu, Y. 2018. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European conference on computer vision (ECCV)*, 286–301.

Zhong, Z.; Shen, T.; Yang, Y.; Lin, Z.; and Zhang, C. 2018. Joint sub-bands learning with clique structures for wavelet domain super-resolution. *Advances in neural information processing systems*, 31.