Applied Artificial Intelligence Research

INNOVIAIR

A Learnable Frequency Gated DCT Denoising Network

Haonian Cao[a,*]

[a]Guangdong Polytechnic Normal University, Guangzhou 510000, China

ARTICLEINFO

Keywords:
Image Denoising
Discrete Cosine Transform
Frequency Domain Modeling
Multi Scale Attention
Global Residual
Image Restoration

ABSTRACT

DCT-based denoising often relies on fixed rules such as thresholding or retaining only low-frequency components, which easily blur textures or leave residual noise on natural images. To address this limitation, we develop a learnable denoising model that keeps the structure of the DCT domain while allowing the network to adaptively regulate each frequency band. With differentiable DCT/IDCT layers, the method applies trainable masks to high- and low-frequency coefficients, and a lightweight global attention block at reduced resolution provides broader contextual cues at low cost. An image-level residual path further aids reconstruction. Under a compact setup, the model improves PSNR and SSIM from 11.89 dB and 0.2915 (pure DCT) to 25.05 dB and 0.945, showing that replacing fixed heuristics with learnable frequency modulation leads to substantially better restoration quality.

1. Introduction

DCT is a classic tool in compression and filtering. For natural image denoising, fixed thresholds or a simple keep low frequency policy often remove noise but also blur textures. Per block Wiener filters can keep details, but may create artifacts. Deep models do well on complex noise, but they are harder to read from a physical point of view.

The core idea of our approach is straightforward. Rather than discarding the DCT prior, we aim to integrate it more effectively. Let the data learn which frequency bands to keep or suppress, and how to fuse them. We make orthogonal DCT and IDCT into differentiable layers, learn masks for high and low sub bands, add a low resolution attention branch for cheap global context, and keep an image level skip. This retains the structural advantages of DCT while enabling end-to-end trainability.

2.Materials and Methods

2.1.Related Work

2.1.1.Classical denoising: non-locality, self-similarity, sparsity

Non-local Means (NLM) uses all similar patches across the image to average and preserve detail[1];BM3D:block matching + 3D transform-domain collaborative filtering is a milestone for Gaussian noise[2];Sparse/low-rank priors also matter a lot: K-SVD learns a dictionary with sparse coding[3];EPLL uses a GMM prior over patches to restore full images[4];WNNM models grouped patches with weighted nuclear norms[5].

2.1.2.Learning-based denoising: CNN to Transformer

In learning based denoising, DnCNN first systematically used residual learning for end to end denoising by regressing noise as the residual and became a strong baseline[6]; Then FFDNet introduced a noise level map and downsampled sub images to balance speed, flexibility, and accuracy, and to support variable noise strength[7]; In the Transformer era, Restormer applied content adaptive channel attention and efficient feed forward networks to high resolution restoration and achieved SOTA on multiple tasks[8]; Uformer proposed a U shaped structure with locally enhanced window attention to keep global dependencies while controlling computational complexity[9].

2.1.3.Frequency or wavelet with differentiable transforms

In the fusion of prior and network, MWCNN embeds the wavelet transform into a U Net framework to naturally combine downsampling and a large receptive field, balancing efficiency and accuracy, and it is widely used in denoising, super resolution, and compression artifact removal[10]; Compared with wavelets, DCT has long been used in compression and filtering due to its energy compaction. In recent years there is a stronger trend of combining frequency domain and deep networks, FcaNet points out theoretically that global average pooling can be regarded as a low frequency special case of DCT and proposes multi band channel attention to improve channel compression and

* Corresponding author.
E-mail addresses:chn2112501025@stu.gpnu.edu.cn.
https://doi.org/10.65455/p24ghq94
Received 11 November 2025; Received in revised form 14 November 2025; Accepted 20 November 2025; Available online 9 December 2025

attention modeling using DCT frequency components[11]; This provides a principled foundation and practical methodology for selectively using DCT bands within learning frameworks.

2.2.Overall Framework

Because DCT has good prior interpretability for energy compaction and band separation[12], as shown in Fig 1, DCT and IDCT need no training and are cheap to compute, but on complex natural images they easily cause detail loss and blocking artifacts. When texture and noise bands overlap, fixed rules struggle to balance denoising and texture fidelity[13].

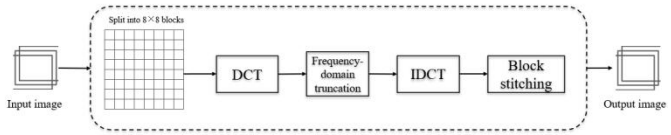

Fig 1: Block-based DCT filtering pipeline

Based on the above observations, we propose a DCT-based learnable frequency-gated denoising network, termed DCT-GatingNet, whose architecture is shown in Fig 2. The network embeds differentiable orthogonal DCT/IDCT layers into an end-to-end learning framework. Concretely, the input image is first fed into a shared shallow network, i.e., the stem, to extract basic features. These features are then sent into two parallel branches corresponding to high- and low-frequency components. Each branch applies a DCT transform, and the resulting coefficients are multiplied pointwise with a learnable gating mask, implementing adaptive frequency-domain filtering. The processed coefficients are transformed back to the pixel domain via IDCT, and the two branches are summed to form the fused representation.

To model global context under limited GPU memory, we further introduce a low-resolution attention module, which can be instantiated as either standard multi-head self-attention (MSA) or AgentAttention. The fused feature map is first downsampled with a scale factor of 4 or 8, global attention is computed in this low-resolution space, and the result is then upsampled and injected back into the main path through a residual connection. In this way we capture long-range dependencies with only a small extra inference cost. On top of this, we adopt an image-level global residual connection, i.e., the network learns the “detail + noise residual” to be corrected rather than the full clean image. This design improves training stability and the fidelity of fine details, and lets the model focus on learning the residual that needs compensation.

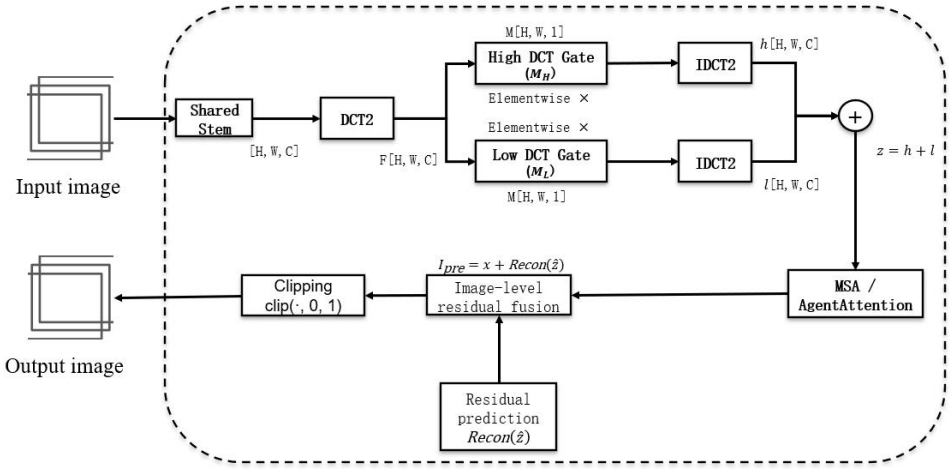

Fig 2. Overall architecture of DCT-GatingNet

2.2.1.Classical denoising: non-locality, self-similarity, sparsity

A core consideration in our design is to avoid parameter redundancy and unstable interfaces. If the high- and low-frequency branches each adopt an independent shallow convolutional stem (a dual-stem design), computation is duplicated and the statistical mismatch between the two initial feature representations can also hurt the stability of the subsequent gating learning. Therefore, we adopt a shared-stem scheme: the input is first mapped by a unified shallow projection to obtain a common feature representation, which is then reused by the two branches, ensuring consistency and efficiency at the sourcee.

Let the input image be $x \in R^{H \times W \times 3}$,We denote the shared stem as S_θ:

$$f=S_\theta(x), f \in \mathbb{R}^{H \times W \times C} \quad (1)$$

After that, the high- and low-frequency branches perform their own subsequent processing on this shared feature. It is worth emphasizing that the role of the shared stem is only to map the input from RGB space to a unified intermediate representation; it does not itself distinguish between high- and low-frequency paths. All operations related to frequency characteristics are carried out independently in the downstream branches.

In a traditional dual-stem design, if each stem consists of convolution kernels of size with channels, the number of parameters and dominant FLOPs are approximately:

$$\text{Params}_{\text{two}}=2\cdot(K^2\cdot 3\cdot C), \text{FLOPs}_{\text{two}} \propto 2\cdot(K^2\cdot 3\cdot C\cdot H\cdot W) \quad (2)$$

With the shared stem, both quantities are roughly halved:

$$\text{Params}_{\text{shared}}=K^2\cdot 3\cdot C, \text{FLOPs}_{\text{shared}} \propto K^2\cdot 3\cdot C\cdot H\cdot W \quad (3)$$

In short, using a shared stem cuts both measures roughly in half. Without changing the structure of the subsequent frequency-domain transforms and attention modules, this effectively reduces redundant computation and provides a unified feature interface, which in turn significantly improves the learning stability of the frequency gating module and the early convergence speed of the model.

In implementation, we use a Conv3×3–Norm–GELU block as S_θ, as shown in Fig 3. This module preserves the same

spatial resolution as the input (no downsampling), and the default number of output channels C is set to 64. The normalization layer (Norm) can be flexibly chosen as batch normalization (BN) or group normalization (GN). The rest of the network remains unchanged; we only need to replace the two original independent stems ($S_{\theta h}$,$S_{\theta l}$)with a single shared parameter set $S_{\theta to}$ realize the shared-stem design.

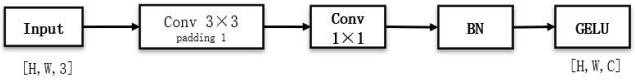

Fig 3: Shared stem module

2.2.2.Learnable High/Low Mask on DCT

Classical DCT-based denoising methods usually rely on hand-crafted and fixed band-partition rules, such as keeping only low frequencies or applying a uniform threshold, which limits their adaptability to diverse image content and noise levels. To address this, we introduce a learnable frequency-domain gating mechanism in the orthogonal DCT space, as illustrated in Figure. 4. The DCT coefficients of the shared feature are softly selected at each frequency point by this mechanism, and the high- and low-frequency information is then distributed to the two processing branches in a data-driven manner. Compared with fixed rules, this design allows the network to automatically learn an adaptive band allocation strategy during training, while the gating operation is fully decoupled from the subsequent processing modules, which improves both flexibility and interpretability.

Let the output of the shared stem be $f\in\mathbb{R}^{H\times W\times C}$, and denote the 2D DCT and IDCT by $D(\cdot)$ and $D^{-1}(\cdot)$, respectively. The frequency-domain representation is given by.

$$F=D(f)\in\mathbb{R}^{H\times W\times C} \tag{4}$$

For the high- and low-frequency branches, we introduce learnable masks $M_{\mathrm{h}},M_l\in[0,1]^{H\times W}$(broadcast along channels), which are obtained from trainable logits $W_{\mathrm{h}},W_l\in\mathbb{R}^{H\times W}$via a temperature-controlled sigmoid:

$$M_{\mathrm{h}}=\sigma\left(\frac{W_{\mathrm{h}}}{\tau}\right),M_l=\sigma\left(\frac{W_l}{\tau}\right),\tau>0 \tag{5}$$

The high- and low-frequency branches then perform pointwise frequency gating with these masks and are each mapped back to the spatial domain:

$$\mathrm{h}=D^{-1}(M_{\mathrm{h}}\odot F),l=D^{-1}(M_l\odot F),z=\mathrm{h}+l \tag{6}$$

Unlike a fixed k×klow-pass filter or a fixed-threshold scheme, (Mh,Ml)learn during training, for every frequency component, the relative strength of preservation, suppression, and redistribution, allowing a single network to simultaneously handle both fine textures and smooth regions.

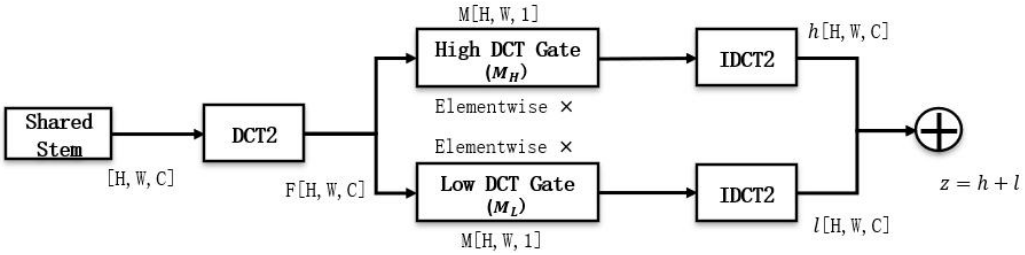

Fig 4: Learnable High/Low Mask on DCT

2.2.3.MSA / AgentAttention

Standard full-resolution self-attention has very high computational and memory complexity in the spatial domain, which makes it difficult to train stably together with the frequency-domain modules under limited GPU memory. Therefore, we adopt a low-resolution attention mechanism to introduce global dependencies with controllable cost, as illustrated in the Figure 5: the intermediate feature representation is first downsampled, then multi-head self-attention (MSA) or AgentAttention is applied in the low-resolution space, and finally the result is upsampled and injected back into the main network through a residual connection. This design efficiently supplements the model with long-range contextual information while keeping the computational burden within an acceptable range.

Let the feature obtained after the shared stem and frequency gating be $z\in\mathbb{R}^{H\times W\times C}$. We define a downsampling operator $\mathrm{Pool}_d:\mathbb{R}^{H\times W\times C}\to\mathbb{R}^{\frac{H}{d}\times\frac{W}{d}\times C}$ average pooling with both stride and kernel size equal to d), an upsampling operator Up_d(bilinear interpolation), and an attention module Attn($\cdot$), which can be instantiated as standard MSA or AgentAttention. Then

$$z_{\downarrow}=\mathrm{Pool}_d(z),g=\mathrm{Attn}(z_{\downarrow}),z'=z+\mathrm{Up}_d(g). \tag{7}$$

where the residual connection preserves the original spatial information, and Up_d maps the global context back to the original resolution. If Attn is instantiated as standard MSA, then

$$\mathrm{MSA}(X)=\mathrm{Concat}(\mathrm{h}_1,\ldots,\mathrm{h}_{N_{\mathrm{h}}})\,W_o,\mathrm{h}_i=\mathrm{softmax}\left(\frac{Q_iK_i^{\top}}{\sqrt{d_k}}\right)V_i \tag{8}$$

where $X\in\mathbb{R}^{N\times C}$and $N=\frac{H}{d}\cdot\frac{W}{d}$. AgentAttention interacts a small number of "agent tokens" with dense tokens, which can further reduce the quadratic complexity; in our implementation it is fully compatible with the implementation of MSA and shares the same interface.

For memory consumption, let the downsampling ratio be $d$and the number of low-resolution tokens be $N=(HW)/d^2$. The dominant attention complexity changes from $O(N^2C)$to

$$O\left(\frac{(HW)^2}{d^4}C\right). \tag{9}$$

and the memory cost scales approximately with N^2as well. Empirically, for 256×256 inputs with C=64, d=4 provides finer global modeling and yields a slight performance gain compared with d=8, at the cost of increased memory usage.

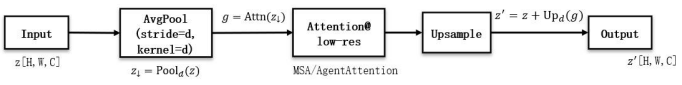

Fig 5: MSA / AgentAttention

2.2.4.Image-Level Skip

Finally, we adopt an image-level residual reconstruction scheme, as illustrated in the Figure 6. Based on the intermediate representation $z\in\mathbb{R}^{H\times W\times C}$ obtained after frequency gating and low-resolution attention, a lightweight reconstruction head Recon$(\cdot)$ predicts a residual

Δ=Recon(z) , which is directly added to the input image$x \in \mathbb{R}^{H\times W\times 3}$.

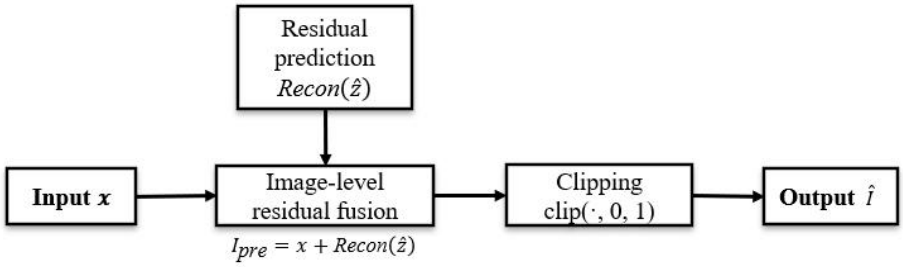

Fig 6: Image-level Skip

3.Results and Discussion

The experiments run on Ubuntu 22.04, using PyTorch 2.2.0. We use Adam as the optimizer. The initial learning rate is 1e-4, and a cosine annealing schedule reduces it to 1e-5 over 60 epochs. To keep things simple and reproducible, data augmentation is limited to basic random flips and random crops.

We use a self-built dataset called toy_datasets. It contains 1000 natural images. We split it into training and validation sets by a fixed ratio. During training we add Gaussian noise with sigma 25 on the fly for supervision, and the validation set uses the same noise level.

3.1.Metrics

To objectively measure denoising at sigma 25, we use Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) as the main metrics. Both are computed on RGB images normalized to the range 0 to 1, first per image and then averaged over the dataset.

1) PSNR

Given a reference clean image $I$and a restored image $\hat{I}$, the mean squared error (MSE) is defined in :

$$\mathrm{MSE}(I,\hat{I})=\frac{1}{N}\sum_{p=1}^{N}\left(I_p-\hat{I}_p\right)^2 \tag{11}$$

Here N is the product of the number of pixels and the number of channels, so for RGB we count all three channels together. Since pixels are already normalized to the range [0,1], the dynamic range is L=1. PSNR is then defined in :

$$\mathrm{PSNR}(I,\hat{I})=10\log_{10}\left(\frac{L^2}{\mathrm{MSE}(I,\hat{I})}\right)=-10\log_{10}\left(\mathrm{MSE}(I,\hat{I})\right) \tag{12}$$

For a dataset Dwith Mimages, we compute PSNR for each image and then take the arithmetic mean, as in:

$$\overline{\mathrm{PSNR}}=\frac{1}{M}\sum_{i=1}^{M}\mathrm{PSNR}\left(I^{(i)},\hat{I}^{(i)}\right) \tag{13}$$

PSNR is reported in dB, and a larger value means less distortion. With the combination of differentiable orthogonal DCT and IDCT, learnable frequency gating in the DCT domain, low resolution attention, and the global residual design, our method can suppress noise while keeping high frequency details, so the PSNR becomes higher.

2) SSIM

SSIM measures the similarity of two images from three aspects: luminance, contrast, and structure. For each window (or the whole image), the statistics are the means $\mu_I$and $\mu_{\hat{I}}$, the variances $\sigma_I^2$and $\sigma_{\hat{I}}^2$, and the covariance $\sigma_{I\hat{I}}$.

SSIM is defined as:

$$\mathrm{SSIM}(I,\hat{I})=\frac{(2\mu_I\mu_{\hat{I}}+C_1)(2\sigma_{I\hat{I}}+C_2)}{(\mu_I^2+\mu_{\hat{I}}^2+C_1)(\sigma_I^2+\sigma_{\hat{I}}^2+C_2)} \tag{14}$$

The stability constants C1 and C2 are defined as $C_1=(k_1L)^2, C_2=(k_2L)^2$， where we usually set $k_1=0.01, k_2=0.03$ and $L=1$ as the pixel dynamic range. In practice, SSIM is computed on local regions with a Gaussian window and then averaged to get the SSIM of the whole image. In this work we also report SSIM per image and then take the arithmetic mean:

$$\overline{\mathrm{SSIM}}=\frac{1}{M}\sum_{i=1}^{M}\mathrm{SSIM}\,!\left(I^{(i)},\widehat{I^{(i)}}\right) \tag{15}$$

SSIM focuses more on consistency of structure and contrast. In our method, the learnable frequency gating adapts to keep the bands that are most sensitive to structure, for example the mid and high frequencies where textures live, and the global residual helps avoid over smoothing. As a result, we keep PSNR while raising SSIM.

3.2.Main Results

From Table 1, under the same training and inference settings, our DCT-based learnable frequency gating model outperforms both traditional DCT methods and other learning baselines on all main metrics. In detail, the method reaches 25.05 dB PSNR and 0.945 SSIM. Compared with the pure CNN baseline without DCT, the gains are +0.12 dB and +0.002. Compared with the version without attention, the gains are +0.02 dB and +0.001. Most notably, against the pure DCT baseline the improvements reach +13.16 dB and +0.653.

These results make it clear that while keeping the sparsity and interpretability of the DCT prior, adding learnable high and low frequency gating and low resolution multi head attention improves the model’s ability to tell noise from real image details. The method delivers steady gains on objective metrics. In high noise and complex texture scenes, the recovery of high frequency details is more complete. This supports that the mix of learnable frequency selection and global residual fusion is both effective and has good generalization.

Table 1 Main results

Method	AVG PSNR (dB)	AVG SSIM
DCT	11.8912	0.2915
DCT-Block-LP	5.3090	0.0146
DCT-Block-Wiener	5.6218	0.0266
CNN	24.93	0.943

Method	AVG PSNR (dB)	AVG SSIM
No Attention(attn=none)	24.93	0.943
MSA(ds=8)	24.89	0.943
Ours + residual(attn=none)	25.03	0.944
Ours + residual(MSA ds=8)	25.02	0.944
Ours + residual(MSA ds=4)	**25.05**	**0.945**

3.3.Ablation Study

To evaluate how each submodule contributes to the model, we run an ablation (Table 2) with a controlled variables setup. Each run disables only one target module, and everything else stays the same as the main model. The main model reaches 25.05 dB/0.945. The results and brief notes are:

1) When the global residual connection is removed, performance drops to 24.95 dB/0.944. Without the y = x + recon(z) path, the network has to fit the full clean image instead of the simpler noise residual. This makes learning harder and slightly hurts edges and fine details. The result shows the global residual gives a stable gain of about +0.1 dB.

2)When the low resolution attention is turned off, performance goes to 24.93 dB/0.944. Without long range dependency modeling, the reconstruction of large structures and texture consistency gets weaker. Even with a low compute setting like ds = 4, global context still brings a clear benefit.

3) When the gating is disabled, performance falls to 24.89 dB/0.944. This is the single biggest drop. Replacing the data driven adaptive gate with a fixed hand crafted mask prevents the model from adjusting band selection to image content. This confirms the learnable gating is central in this design.

4) When the shared stem is removed, performance is 25.00 dB/0.944. Giving the high and low frequency branches separate shallow extractors adds parameter and compute redundancy, and the mismatch in feature statistics slightly disturbs later fusion. A shared stem offers a unified feature interface with better efficiency.

Table 2 Ablation results

Method	PSNR (dB)	SSIM
Disable global residual connection	24.95	0.944
Disable MSA	24.93	0.944
Disable gating	24.89	0.944
Disable shared stem	25.00	0.944
Ours	**25.05**	**0.945**

4.Conclusions

This work presents a denoising model that merges the structure of the DCT domain with adaptable, data-driven components. By allowing the network to adjust the contribution of different frequency bands through learnable gating, the model can better distinguish informative high-frequency details from noise. The low-resolution attention branch offers additional global context without imposing a large computational burden, and the image-level residual design helps stabilize optimization and enhances the reconstruction of fine structures.

Across quantitative evaluations, the approach consistently performs better than both fixed-rule DCT filtering and comparable lightweight CNN baselines. The analysis further indicates that the gating temperature and data scale can influence the behavior of frequency selection, especially in images dominated by repetitive textures or extremely noisy regions. Although the attention module increases memory usage slightly, the overall system remains efficient, and the DCT components themselves are computationally inexpensive.

Overall, enabling learnable frequency selection within a DCT-based framework provides a flexible and interpretable solution for image denoising. Future work will explore extending this approach to real-world noisy datasets and examining its adaptability to related restoration tasks such as deblurring and super-resolution.

References

[1] BUADES A, COLL B, MOREL J M. A non-local algorithm for image denoising//2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Diego, CA, USA: IEEE, 2005: 60-65.

[2] AHMED N, NATARAJAN T, RAO K R. Discrete cosine transform. IEEE Transactions on Computers, 2006, 100(1): 90-93.

[3] AHARON M, ELAD M, BRUCKSTEIN A. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing, 2006, 54(11): 4311-4322.

[4] ZORAN D, WEISS Y. From learning models of natural image patches to whole image restoration//2011 International Conference on Computer Vision. Barcelona, Spain: IEEE, 2011: 479-486.

[5] GU S, ZHANG L, ZUO W, et al. Weighted nuclear norm minimization with application to image denoising//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA: IEEE, 2014: 2862-2869.

[6] ZHANG K, ZUO W, CHEN Y, et al. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. IEEE Transactions on Image Processing, 2017, 26(7): 3142-3155.

[7] ZHANG K, ZUO W, ZHANG L. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. IEEE Transactions on Image Processing, 2018, 27(9): 4608-4622.

[8] ZAMIR S W, ARORA A, KHAN S, et al. Restormer: Efficient transformer for high-resolution image restoration//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans, LA, USA: IEEE, 2022: 5728-5739.

[9] WANG Z, CUN X, BAO J, et al. Uformer: A general u-shaped transformer for image restoration//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans, LA, USA: IEEE, 2022: 17683-17693.

[10] LIU P, ZHANG H, ZHANG K, et al. Multi-level wavelet-CNN for image restoration//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. Salt Lake City, UT, USA: IEEE, 2018: 773-782.

[11] QIN Z, ZHANG P, WU F, et al. Fcanet: Frequency channel attention networks//Proceedings of the IEEE/CVF International Conference on Computer Vision. Montreal, Canada: IEEE, 2021: 783-792.

[12] AHMED N, NATARAJAN T, RAO K R. Discrete cosine transform. IEEE Transactions on Computers, 2006, 100(1): 90-93.

[13] GONZALEZ R C. Digital image processing. New York, USA: Pearson Education, 2009.