# Learning Frequency Domain Priors for Image Demoireing

Bolun Zheng*, Shanxin Yuan*, Chenggang Yan†, Xiang Tian, Jiyong Zhang, Yaoqi Sun, Lin Liu, Aleš Leonardis, *Member, IEEE*, Gregory Slabaugh, *Senior Member, IEEE*

**Abstract**—Image demoireing is a multi-faceted image restoration task involving both moire pattern removal and color restoration. In this paper, we raise a general degradation model to describe an image contaminated by moire patterns, and propose a novel multi-scale bandpass convolutional neural network (MBCNN) for single image demoireing. For moire pattern removal, we propose a multi-block-size learnable bandpass filters (M-LBFs), based on a block-wise frequency domain transform, to learn the frequency domain priors of moire patterns. We also introduce a new loss function named Dilated Advanced Sobel loss (D-ASL) to better sense the frequency information. For color restoration, we propose a two-step tone mapping strategy, which first applies a global tone mapping to correct for a global color shift, and then performs local fine tuning of the color per pixel. To determine the most appropriate frequency domain transform, we investigate several transforms including DCT, DFT, DWT, learnable non-linear transform and learnable orthogonal transform. We finally adopt the DCT. Our basic model won the AIM2019 demoireing challenge. Experimental results on three public datasets show that our method outperforms state-of-the-art methods by a large margin.

**Index Terms**—Image Demoireing, Frequency Domain Prior, Learnable Bandpass Filter, Dilated Advanced Sobel Loss, Degradation Model, Learnable Orthogonal Transform, Two-step Color Restoration.

## 1 Introduction

DIGITAL screens are ubiquitous and have become one of the most popular devices from which people receive information. At the same time, mobile devices (e.g., smartphones) that include digital cameras are an increasingly essential tool for modern living. It is becoming a common practice to take pictures of screens to quickly save information. For example, when attending an academic conference one may want to take pictures of the slides displayed on a digital screen, and read them carefully later. Sometimes taking a photo is the only practical way to save information. Unfortunately, a common side effect is that moire patterns can appear, degrading the image quality of the photo, see Figure 1. Moire patterns arise when two repetitive patterns interfere with each other. In the case of taking pictures of screens, the camera's color filter array (CFA) interferes with the screen's subpixel layout. Moire patterns exhibit largely varying patterns of color, thickness and appear as ripples or stripes, which are sensitive to shooting distance and camera orientation. Moire patterns vary not only across different images but even within the same image. Additionally, due to differences between color systems on the screen and camera, color degradation is another degradation that appears along with moire patterns. That is to say, a general image demoireing algorithm should give consideration to both moire pattern removal and color restoration.

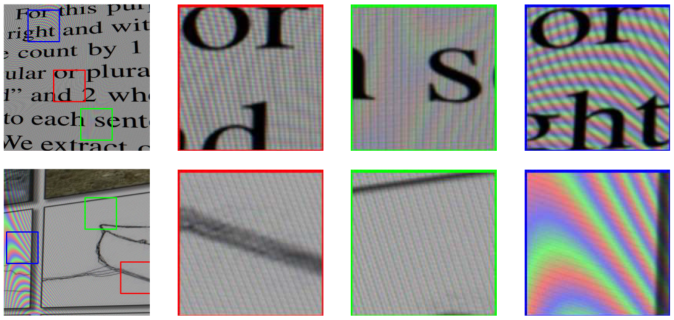

Fig. 1. Moire patterns of different scales, frequencies, and colors.

Although human observers can distinguish moire patterns, recovering the underlying clear image from an image contaminated by moire patterns is an ill-posed problem and poses a considerable challenge. Image priors are generally used to solve ill-posed problems. Color priors and texture priors are two sets of well studied priors for many different image processing tasks. Color priors are focused on the general color nature of natural images or degraded images. The dark channel prior [1], [2], color line prior [3], [4], and color ellipsoid prior [5] are representative color priors for image enhancement and restoration tasks. The spatial domain priors and transformation domain priors constitute the texture priors. The spatial domain priors like self similarity [6], [7] and shape priors [8], [9], are usually adopted for filtering based image manipulations. The transformation domain priors depend on the specific transformation. They are usually adopted by optimisation-based algorithms [7],

- *B. Zheng and S. Yuan contributed equally to this paper.*
- *Chenggang Yan (email: cgyan@hdu.edu.cn) is the corresponding author of this paper.*
- *B. Zheng, C. Yan, J. Zhang and Y. Sun are currently with Hangzhou Dianzi University.*
- *S. Yuan and A. Leonardis are currently with Huawei Noah's Ark Lab.*
- *X. Tian is currently with Zhejiang University.*
- *L. Liu is currently with University of Science and Technology of China. He is also a student research intern at Huawei Noah Ark's Lab.*
- *G. Slabaugh is currently with Queen Mary University of London. He was with Huawei Noah's Ark Lab when this work was conducted.*

[10]. The Fourier and wavelet domains are two commonly studied transformation domains in the image restoration literature. Because values in these two domains relate to frequency, we customarily call methods working in either of two domains as frequency domain techniques.

In recent years, deep learning based algorithms have demonstrated great ability to handle many image manipulation tasks. These algorithms are usually driven by enormous training data pairs, and targeted to build globally end-to-end or stage-wise neural network solutions. With deep neural networks, the priors for a task are learned from data and implicitly stored in the neural weights. Unlike conventional image priors discovered by human's experience, these data-driven priors are task-oriented. This way, traditional iterative optimization processes can be replaced by fixed forward inference. Inspired by this, several recent methods have been proposed that combine frequency domain transformation and a deep neural network within a unified architecture [11], [12], [13].

Because of the largely varying appearance of moire patterns, conventional image prior based [6], [10] algorithms are inadequate for the demoireing task. Only very recently, a few deep learning based attempts [13], [14], [15], [16], [17] have been made to address image demoireing. Recent work [14], [17], [18] tried to remove moire patterns of different frequency bands through multi-scale design. DMCNN [14] proposed to deal with moire patterns with a multi-scale CNN with multi-resolution branches and summed up the outputs from different scales to obtain a final output. MDDM [18] improved DMCNN by introducing adaptive instance normalization [19] based on a dynamic feature encoder. DCNN [15] proposed a coarse-to-fine structure to remove moire patterns from two scales. The coarse scale result was upsampled and concatenated with the fine scale input for further residual learning. MopNet [17] used a multi-scale feature aggregation sub-module to address the challenging frequencies of moire patterns, and two other sub-modules to address edges and pre-defined moire types. Though current algorithms show promising performance, the problem remains to a large extent unsolved, due to the large variation of moire patterns in terms of frequencies, shapes, and colors.

In this paper, we introduce a general degradation model to describe an image contaminated with moire patterns, and propose a novel *learnable bandpass filter* (LBF) to explicitly learn the frequency domain prior of moire patterns for single image demoireing. We also investigate the best frequency domain representation. We explicitly split the demoireing task into two sub-tasks — moire pattern removal and color restoration, and introduce a unified framework namely a multi-scale bandpass convolution neural network (MBCNN) to perform the two sub-tasks within the same model. The LBF is the core component for the moire pattern removal. The LBF introduces a learnable passband to learn the frequency prior, which could precisely separate moire patterns from normal image texture. The global and local tone mapping are included for accurate color restoration. The global tone mapping learns the global color shift from moire images to clean images, while the local tone mapping is to make a local fine-grained color restoration. To guide the LBF to efficiently learn the moire pattern's frequency domain prior, we introduce a multi-scale architecture (MBCNN) trained with a proposed *dilated advanced Sobel loss* (D-ASL) applied at each scale.

We note that this paper is an extended version of our conference paper [20] (called MBCNN-conf in this paper) that has been extended in the following substantial ways:

1. We improve the advanced Sobel loss (ASL) by introducing dilation rates to the Sobel filtering to construct the D-ASL. With the dilation rates, D-ASL can sense the structural high-frequency information in multiple scales, and significantly improve the light model's performance by 1.51dB, which outperforms MBCNN-conf.
2. We develop a learnable orthogonal transform (LOT). Then we investigate the effect of different transformation domains in the LBF, including discrete cosine transform (DCT) domain, discrete wavelet transform (DWT) domain, discrete Fourier transform (DFT) domain, and LOT. We demonstrate the DCT domain is the most suitable domain for learning the prior of moire patterns within the LBF.
3. We advance LBF by estimating transformation domain values from different block sizes. This multi-block-size structure brings observable performance gain with little additional computation for the DCT-domain based LBF.

## 2 Related Work

Image demoireing requires both texture restoration[1] and color restoration, rendering it a complex challenge. In this section, we briefly introduce several deep learning based methods in related tasks, where deep learning has made significant impact.

**Image restoration.** Dong *et al.* [21], [22] were the first to propose end-to-end convolutional neural networks for image super-resolution and compression artifact reduction. Subsequent research [23], [24], [25] further improved these models by increasing the network depth, introducing skip connections [26] and residual learning. Much deeper networks [27], [28], [29], [30] were then introduced. DRCN [27] proposed recursive learning for parameter sharing. Tai *et al.* [28], [29] introduced a recursive residual learning and proposed a memory block. Zhang *et al.* [30] replaced the recursive connection in the memory block by a dense connection [31]. Moreover, several studies focused on multi-scale CNNs inspired by high-level computer vision methods. Mao *et al.* [32] proposed a skip connection-based multi-scale autoencoder. Cavigelli *et al.* [33] introduced a multi-supervised network for compression artifact reduction.

**Frequency domain learning.** Several studies [11], [34], [35] focus on CNN-based frequency domain learning. Liu *et al.* [34] introduced a U-Net-like model that uses the discrete wavelet transform and its inverse to replace conventional downscaling and upscaling operations for image restoration. One benefit is that there is no information loss for downscaling. Luo *et al.* [36] later extended this idea to moire artifact removal for high-frequency natural images.

[1] In this paper, the term 'texture restoration' refers to the removal of moire patterns, unless otherwise specified.

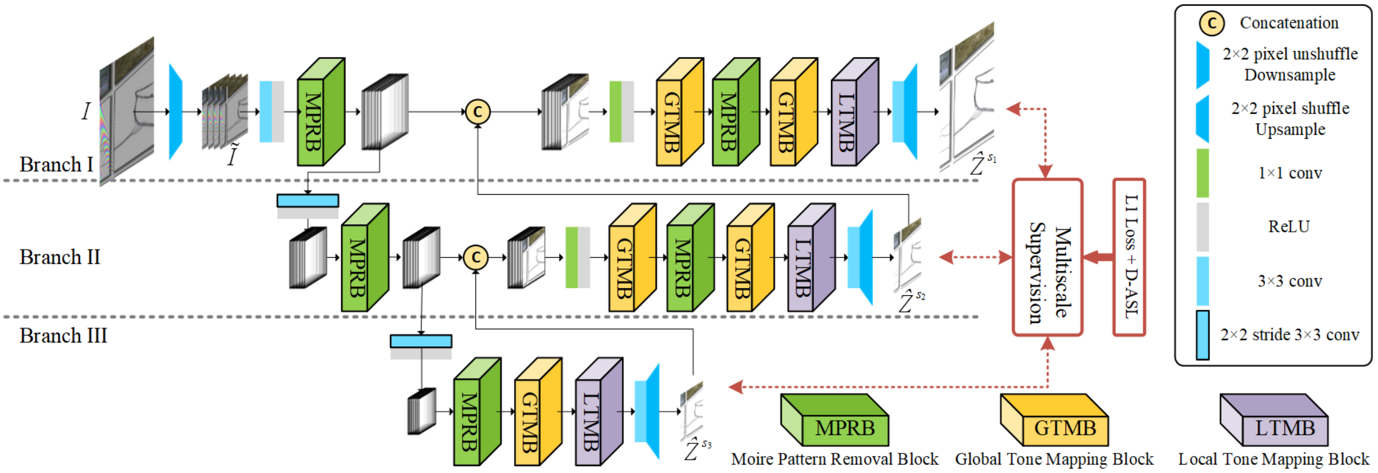

Fig. 2. The architecture of our MBCNN. The network is U-Net like and mainly consists of three branches. In each branch, the moire pattern removal block (MPRB), global tone mapping block (GTMB) and local tone mapping block (LTMB) are sequentially stacked and finally output a clean image at the corresponding scale. The additional GTMB and MPRB are introduced to branch I and II to reduce the texture and color errors caused by fusing the features of the current branch and the output of the coarser branch. The details of the three blocks are explained in Sec. 4.

Liu *et al.* [13] proposed Wavelet-based Dual-branch network (WDNet) for image demoireing by first transforming the original moire images into the wavelet domain, where a dual-branch network is used to remove moire patterns. Guo *et al.* [11] introduced a dual-domain representation network working in both the discrete cosine transform (DCT) and pixel domains for reduction of compression artifacts, where the DCT and inverse DCT (IDCT) are used to construct a sub-network to learn the DCT-domain prior knowledge of JPEG compression. Gia Vien *et al.* [37] tested a similar idea for moire artifact removal. Zheng et al. [35] introduced an implicit DCT to extend the DCT-domain learning to color image compression artifact reduction.

**Color restoration.** Image dehazing and image enhancement are two classic color restoration problems. Eilertsen *et al.* [38] proposed a gamma correction based loss function and trained a U-Net [39] based CNN for high dynamic range (HDR) image reconstruction. Gharbi *et al.* [40] proposed HDRNet to learn local piece-wise linear tone mapping. Inspired by the guided filter [41], Wu *et al.* [42] proposed an end-to-end trainable guided filter for image enhancement. Ren et al. [43] grouped a hazy image and several pre-enhanced images together as input, and proposed a symmetric autoencoder to learn a gated fusion for image dehazing. Zhang et al. [44] proposed a densely connected pyramid CNN for image dehazing. Remarkably, few of these color restoration methods include residual connections in their solutions.

**Loss function.** The loss function is one of the most important components of CNN-based low-level vision methods. With the same model, different loss functions used in training lead to greatly different results. Zhao *et al.* [45] conducted a comprehensive study of several common losses for image restoration tasks and demonstrated that using L1 loss and SSIM [46] loss are effective in several restoration tasks. Ledig *et al.* [47] and Wang *et al.* [48] introduced a GAN-based loss function for image super-resolution. Guo *et al.* [49] introduced a VGG-Net [50]-based perceptual loss to minimize the semantic distance between the input and output for JPEG compression artifact reduction. Other loss functions include Charbonnier loss [51], CORAL loss [52], differential content loss [53], etc. Recently several frequency domain loss functions were proposed, including wavelet loss functions, DCT loss functions, etc. We propose a dilated advanced Sobel loss (D-ASL) function that is suitable for image demoireing.

**Image demoireing.** Recently, several end-to-end image demoireing solutions have been proposed. Sun *et al.* [14] first introduced a CNN for image demoireing (DMCNN) and created an ImageNet [54]-based moire dataset for training and testing. Cheng *et al.* [18] improved DMCNN by introducing an adaptive instance normalization [19] based dynamic feature encoder. He *et al.* [17] introduced additional moire attribute labels based on shape, color, and frequency for more precise moire pattern removal. Liu *et al.* [13] proposed to transform the original moire images into the wavelet domain, where a dual-branch network is used to remove moire patterns. However, none of the existing methods modeled the moire patterns explicitly. We propose a novel *Learnable bandpass filter* to explicitly learn the frequency domain priors of moire patterns. We treat the image demoireing problem as two sub-problems: moire pattern removal and color restoration.

## 3 Image Degradation Model

An image captured by a digital camera of a screen usually has color degradation and contains moire patterns. The color degradation is caused by systematic color differences between the screen and the camera. The screen's display settings, *e.g.*, luminance, saturation, and contrast, have an effect, along with the camera's image signal processor pipeline. Moire patterns mainly result from interference between the screen's display grid pattern and the camera's CFA. The moire patterns can change dramatically with different viewpoints of the camera. All existing image demoireing methods [14], [17], [18] did not model the two problems explicitly, relying on the model to implicitly handle both problems. We argue that the color degradation and moire

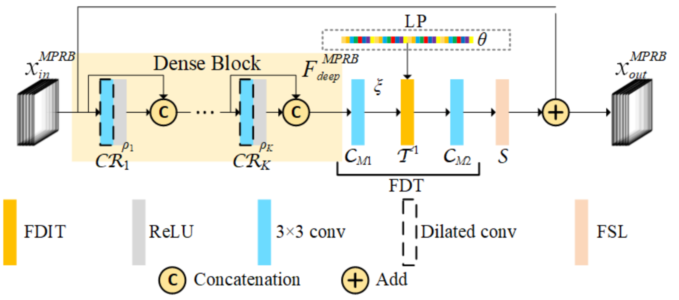

Fig. 3. The structure of moire pattern removal block (MPRB). The MPRB is formulated by a dense block, the frequency domain transform (FDT) and a feature scale layer (FSL). The FDT consists of two convolution layers $\mathcal{C}_{M1}$ and $\mathcal{C}_{M2}$, and a frequency domain inverse transform (FDIT) layer $\mathcal{T}^{-1}$ with the learnable passband (LP) $\theta$.

patterns can be separated and dealt with independently. Thus, we can model a moire image as:

$$I_{moire} = \psi(I_{clean}) + N_{moire} \tag{1}$$

where $I_{clean}$ is the clean image, $N_{moire}$ is the introduced moire pattern, and $\psi$ is the color degradation caused by the screen and the camera sensor. $I_{clean}$ can be then expressed as:

$$I_{clean} = \psi^{-1}(I_{moire} - N_{moire}) \tag{2}$$

where $\psi^{-1}$ is the inverse function of $\psi$, which is known as the tone mapping function in the image processing field. Modeled in this way, the image demoireing task can be divided into two steps, *i.e.*, moire pattern removal and tone mapping.

Moire patterns exhibit considerable variation in shape, frequency, color, *etc*. Some examples are shown in Figure 1, where the moire patterns clearly exhibit different frequencies and scales. Therefore, the moire patterns can be written as:

$$N_{moire} = \sum_i \sum_j N_{f_{ij}}^{s_i} \tag{3}$$

where $N_{f_{ij}}^{s_i}$ denotes the moire pattern component of scale $s_i$ and frequency $f_{ij}$. Given a frequency domain transformation $\mathcal{T}$, we can have the frequency domain expression of $N_{moire}$ as:

$$\begin{aligned} N_{moire} &= \sum_i \sum_j \mathcal{T}^{-1}(\mathcal{T}(N_{f_{ij}}^{s_i})) \\ &= \sum_i \mathcal{T}^{-1}(\sum_j \mathcal{T}(N_{f_{ij}}^{s_i})) \\ &= \sum_i \mathcal{T}^{-1}(FS^{s_i}) \end{aligned} \tag{4}$$

where $FS^{s_i} = \sum_j \mathcal{T}(N_{f_{ij}}^{s_i})$ is the frequency spectrum of $N_{moire}$ in the scale $s_i$, and $\mathcal{T}^{-1}$ denotes the inverse function of $\mathcal{T}$. Therefore, we can split moire patterns from a moire image by estimating the moire patterns' frequency spectrum.

# 4 Proposed Method

Though image demoireing can be divided into moire pattern removal and color restoration, it's difficult to obtain only moire pattern-contaminated images or only color degraded images to train two DNNs for the two sub-problems separably. Therefore, we propose to solve the two sub-problems within the same DNN. Our solution is to solve the two sub-problems in the feature domain without explicit supervision. The architecture of our MBCNN is shown in Figure 2. Our model works in three scales and has three different types of blocks, which are moire pattern removal block (MPRB), global tone mapping block (GTMB), and local tone mapping block (LTMB). The MPRB is for moire pattern removal, while the GTMB and the LTMB are for color restoration. The details of each block are described in Sec. 4.1 and Sec. 4.2.

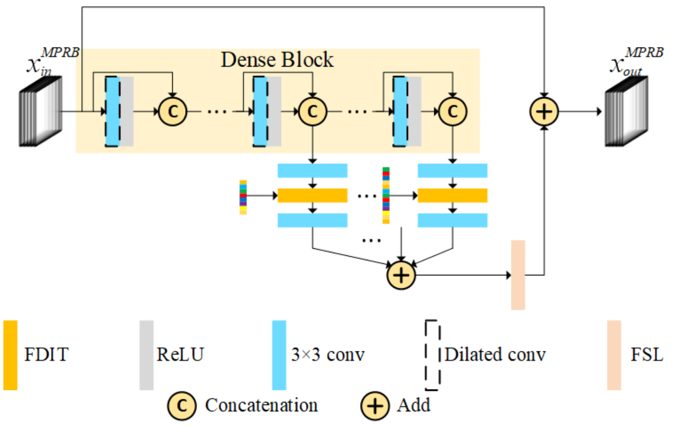

Fig. 4. The structure of moire pattern removal block constructed by MLBFs.

The input image $I$ with the shape of $h \times w \times c$ is first reversibly downsampled into four subimages $\tilde{I}$ with the shape of $\frac{h}{2} \times \frac{w}{2} \times 4c$. Because moire pattern is a shape-wise noise rather than pixel-wise noise (*e.g.* Gaussian noise), removing it from a larger scale is appropriate. Besides, this operation is a lossless transform and also helps to reduce computation. With the tensor $\tilde{I}$ as input, the following network consists of three branches, each to recover the clean image in a specific scale. Following Eq. 2, each branch sequentially executes the moire pattern removal and tone mapping, and finally outputs an up-scaled image to be fused in the finer scale branch. In branch I and II, after fusing the feature of current branch and the output of the coarser scale branch, additional GTMB and MPRB are stacked to remove the texture and color errors caused by the scale change.

## 4.1 Moire Pattern Removal

Given a color moire image patch $P$, we denote the moire patterns of each color channel as $N_P^c$, $c \in \{R, G, B\}$. Then the feature domain representation of the moire patterns $N_P$ is

$$\mathcal{C}(N_P) = \sum_{c \in \{R,G,B\}} \mathcal{C}(N_P^c) \tag{5}$$

where $\mathcal{C}$ denotes a learnable convolution. Based on Eq. 4, Eq. 5 can be rewritten as

$$\begin{aligned} \mathcal{C}(N_P) &= \sum_{c \in \{R,G,B\}} \mathcal{C}(\sum_i \mathcal{T}^{-1}(FS^{s_i}|_c)) \\ &= \sum_i \mathcal{C}(\mathcal{T}^{-1}(\sum_{c \in \{R,G,B\}} FS^{s_i}|_c)) \end{aligned} \tag{6}$$

where $\sum_{c\in\{R,G,B\}} FS^{s_i}\big|_c$ is the combined frequency spectrum of channel $c$ with the scale of $s_i$, which is defined as the implicit frequency spectrum (IFS), denoted as $\xi^{s_i}$. Now, we can have

$$\mathcal{C}(N_P) \quad = \textstyle\sum_i \mathcal{C}(\mathcal{T}^{-1}(\xi^{s_i})) \tag{7}$$

**Learnable Bandpass Filter.** Inspired by the implicit DCT [35], we can directly estimate $\xi^{s_i}$ with a deep CNN block. Since the frequency domain transforms $\mathcal{T}$ are always linear, we can use a a pre-defined $1 \times 1$ convolution layer, whose weights are fixed as the transform matrix to model them. However, it's difficult to get the accurate frequency spectrum, because there would be several frequencies in different scales and they can also affect each other. Therefore, noise inevitably exists in the estimated $\xi^{s_i}$. As the frequency spectrum of moire patterns covers a broad range [17], $\xi^{s_i}$ should distribute across the frequency domain. We define this frequency domain distribution as the frequency domain prior and use a frequency domain passband to describe the prior. With the passband, we can construct a bandpass filter to amplify certain frequencies in the estimated $\xi^{s_i}$ and diminish others. We use a DNN to learn the passbands from a large collection of moire and clean image pairs. Because the passbands are learnable, we name this bandpass filter as *learnable bandpass filter* (LBF). In this situation, the Eq. 7 can be rewritten as

$$\mathcal{C}(N_P) = \sum_i \mathcal{C}(\mathcal{T}^{-1}(\theta^{s_i} \cdot \xi^{s_i})) \tag{8}$$

where $\theta^{s_i}$ denotes the learnable passband for the scale $s_i$. Assuming the size of the frequency domain transformation is $p \times p$, then the corresponding frequency spectrum totally has $p^2$ frequencies, so the size of $\theta^{s_i}$ is $p^2$. All parameters of $\theta^{s_i}$ are initialized to be 1 and constrained to be non-negative. Thus all frequencies are initially passable.

**CNN Structure.** Following Eq. 8, we can respectively remove moire patterns from different scales. For each specific scale, we propose a moire pattern removal block (MPRB), see Figure 3.

Assuming the input of the MPRB is $x_{in}^{MPRB}$, a dense block is first used for feature extraction, which is denoted as $F_{deep}$. Then a $3\times3$ convolution layer estimates the IFS $\xi$ from $F_{deep}$. The dense block has $K$ densely connected [31] $3 \times 3$ $n_D$-channel dilated convolution [55] with ReLU activation ($Conv\_ReLU$) layers. We adopt dilated convolution rather than normal convolution to enlarge the receptive field of the dense block to produce $F_{deep}$, so that the $p^2$ sized $\xi$ can be easily estimated from the $F_{deep}$. After estimating $\xi$, the learnable weight $\theta$ and the block-wise Frequency Domain Inverse Transform (FDIT) layer $\mathcal{T}^{-1}$, a convolution layer $\mathcal{C}_{M2}$ is added as indicated in Eq. 8. Directly multiplying $\theta$ and $\xi$ will consume a large amount of calculations. Instead, we reshape $\theta$ to the size of $1 \times 1 \times p \times p$, and multiply it with the convolution kernel of $\mathcal{T}^{-1}$ layer, then the $\xi$ is directly sent to $\mathcal{T}^{-1}$ layer. In this way, the product $\theta \cdot \xi$ can be avoided. Considering that the $\mathcal{T}^{-1}$ might lead to large local output and produce excessive gradient, we stacked a feature scale layer (FSL) to linearly constrain the output of $\mathcal{C}_{M2}$. The FSL contains a learnable parameter $\alpha$ initialized to be 0.1, which will be updated along with other learnable parameters in the network during the training stage. Finally, we introduce the residual connection [56] to remove the moire patterns in the feature domain. Thus, the final output of MPRB $x_{out}^{MPRB}$ can be obtained by

$$x_{out}^{MPRB} = x_{in}^{MPRB} + \alpha \cdot (\mathcal{C}_{M2}(\mathcal{T}^{-1}(\theta \cdot \xi)) \tag{9}$$

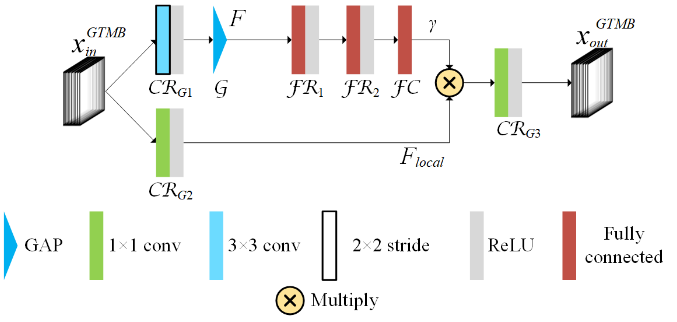

Fig. 5. The structure of global tone mapping block.

TABLE 1
Attributions of learnable layers in GTMB.

| Layer | $\mathcal{CR}_{G1}$ | $\mathcal{CR}_{G2}$ | $\mathcal{CR}_{G3}$ | $\mathcal{FR}_1$ | $\mathcal{FR}_2$ | $\mathcal{FC}$ |
|---|---|---|---|---|---|---|
| Stride | $2 \times 2$ | $1 \times 1$ | $1 \times 1$ | - | - | - |
| Kernel | $3 \times 3$ | $1 \times 1$ | $1 \times 1$ | - | - | - |
| Output Ch. | $n_G \cdot 2$ | $n_G \cdot 2$ | $n_G$ | $n_G \cdot 8$ | $n_G \cdot 4$ | $n_G \cdot 2$ |

**Multi-block-size LBFs.** Theoretically, we can use a passband of any block size to fit the frequency priors of moire patterns. In practice, one passband can only best match one specific frequency, making it hard to use one passband to fit all frequency priors. To solve this problem, we introduce Multi-block-size LBFs (M-LBFs) to fit the frequency priors via several LBFs of different block sizes. Since an LBF of a larger block size requires a larger receptive field, we use the final output of the dense block to build the largest size LBF and intermediate outputs to build LBFs of smaller sizes. Figure 4 illustrates the structure of MPRB constructed by M-LBFs. The output from the last dilated convolution layer of the dense block is used for building the largest sized LBF. Before that, each upcoming densely connected layer is used for building a smaller sized LBF. Finally, the outputs of all LBFs are added together and then sent to the FSL.

### 4.2 Tone Mapping

The RGB color space is an extremely large space containing $256^3$ colors, making it difficult to perform point-wise tone mapping. Observing that there are color shifts between the moire and clean images, we propose a two-step tone mapping strategy with two types of tone mapping blocks: Global Tone Mapping Block (GTMB) and Local Tone Mapping Block (LTMB).

**Global tone mapping block.** The GTMB is proposed to learn the global color shift, see Figure 5 for the detailed structure. Given the input $x_{in}^{GTMB}$, we first extract a global feature $F$ through a $3\times3$ $Conv\_ReLU$ layer with a stride of 2 and a global average pooling (GAP) layer. Then, to extract a deep global feature $\gamma$, we stack two fully connected (FC) layers with ReLU activation ($\mathcal{FR}_1$, $\mathcal{FR}_2$) and a FC layer

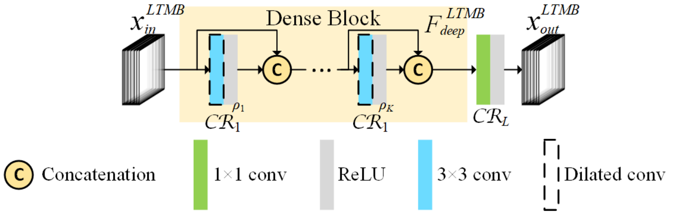

Fig. 6. The structure of local tone mapping block.

without ReLU activation ($\mathcal{FC}$). In addition, we use a $1 \times 1$ $Conv\_ReLU$ layer to extract the local feature $F_{local}$ from $x_{in}^{GTMB}$. The output $x_{out}^{GTMB}$ can be obtained as

$$x_{out}^{GTMB} = \mathcal{CR}_{G3}(\gamma \cdot F_{local}) \tag{10}$$

Assuming the $\mathcal{CR}_{G3}$ outputs a $n_G$-channel tensor, Table 1 lists the attributions of all learnable layers in GTMB.

**GTMB vs. Channel Attention.** The attention mechanism has proven to be effective in many tasks [57], [58], [59], [60], and several channel attention blocks have been proposed [61], [62]. Our GTMB can be viewed as a kind of channel attention block (also known as squeeze and excitation (SE) block [62]). However, GTMB is different from existing channel attention blocks in several aspects. First, existing channel attention blocks are always activated by a sigmoid unit, while there are no such constraints for $\gamma$ in the GTMB. Second, channel attention is directly applied on the input of the existing channel attention blocks, while the $\gamma$ in GTMB is applied on the local feature $F_{local}$. Finally, existing channel attention blocks are aimed at making an adaptive channel-wise feature re-calibration; the goal of GTMB is to make a global color shift and avoid the irregular and inhomogeneous local color artifacts (more analysis are described in Sec. 5.4).

**Local tone mapping block.** The LTMB is developed to fit a local fine-grained tone mapping function. As shown in Figure 6, the structure of LTMB is similar to MPRB. LTMB first takes a similar dense block in MPRB to extract the deep feature $F_{deep}^{LTMB}$ from the input of LTMB $x_{in}^{LTMB}$. Then, the output of LTMB is obtained by

$$x_{out}^{LTMB} = \mathcal{CR}_L(F_{deep}^{LTMB}) \tag{11}$$

where $\mathcal{CR}_L$ is a $1 \times 1$ convolution, and $x_{out}^{LTMB}$ has the same shape as $x_{in}^{LTMB}$.

### 4.3 Loss Function

In this paper, we use the L1 loss as the base loss function, as it has been proven [30], [45], [63] that L1 loss is more effective than L2 loss for image restoration tasks. However, the L1 loss itself is not enough as it is a point-wise loss that cannot provide structural information, while moire patterns are structural artifact. We propose an dilated advanced Sobel loss (D-ASL) to solve this problem. The D-ASL can be expressed as:

$$\text{D-}\mathcal{ASL}^{\{d_1,d_2,\cdots,d_n\}} = \sum_{i=1}^{n} \mathcal{ASL}|_{dilation\_rate=d_i} \tag{12}$$

TABLE 2
Comparison of MBCNNs constructed by LBF with different frequency domain transforms.

| Transform | IDWT | IDCT | IDFT | LNT | LOT |
|---|---|---|---|---|---|
| PSNR | 44.26 | **45.08** | 44.69 | 43.06 | 42.69 |
| SSIM | 0.9963 | **0.9967** | 0.9965 | 0.9898 | 0.9953 |

where $\mathcal{ASL}|_{dilation\_rate=d_i}$ denotes the ASL [20] with dilation rate of $d_i$. Because the ASL can be viewed as a $3 \times 3$ convolution, we can adjust the perceived frequency of the ASL by setting different dilation rates. We combine ASLs of different dilation rates to strengthen the basic ASL (whose dilation rate can be seen as 1). We combine the D-ASL and L1 losses as

$$Loss(\hat{Z}, Z) = \mathcal{L}1(\hat{Z}, Z) + \lambda \cdot \text{D-}\mathcal{ASL}^{\mathbf{D}}(\hat{Z}, Z) \tag{13}$$

where $\mathcal{L}1$ denotes the L1 loss, $\mathbf{D} = \{d_1, d_2, \cdots, d_n\}$, and $\lambda$ is a hyper-parameter to balance the L1 loss and D-ASL.

When training MBCNN, we adopt the multi-supervision strategy that supervises the outputs from all branches, which can be expressed as,

$$\begin{aligned} loss = Loss(\hat{Z}^{s_1}, Z^{s_1}) + Loss(\hat{Z}^{s_2}, Z^{s_2}) \\ + Loss(\hat{Z}^{s_3}, Z^{s_3}) \end{aligned} \tag{14}$$

where $s_1$, $s_2$, and $s_3$ indicate branch 1, 2, and 3, respectively.

## 5 Experiments

We have conducted extensive ablation studies and outperformed the state-of-the-art by large margins on three public datasets: *LCDMoire* [64], *TIP2018* [14] and *London's Buildings* [13]. The *LCDMoire* dataset consists of 10,200 synthetically generated image pairs with 10,000 training images, 100 validation images and 100 testing images. The *TIP2018* dataset consists of real photographs constructed by photographing images of the ImageNet [54] dataset displayed on computer screens with various combinations of different camera and screen hardware. It has 150,000 real clean and moire image pairs, split into 135,000 training images and 15,000 testing images. The *London's Buildings* is an urban-scene data set and its images contain bricks, windows and other regular patterns which are prone to generate moire patterns. It includes 400 training pairs and 60 testing pairs with about $2300 \times 1700$ resolution. All three datasets are used for comparison with state-of-the-art methods. *LCDMoire* dataset is also used for ablation studies. The ablation studies are conducted on the validation set, as the test dataset's ground truth is not available. Please note: the validation dataset is completely independent and not used in training.

### 5.1 Implementation Details

For the MBCNN model, we adopt the following settings, with $c = 3$, $n_D = 64$, $n_G = 2 \cdot n_D = 128$, $K = 5$. We sequentially introduced 3 LBFs with the size of 4, 6, and 8 to build the M-LBF in each MPRB. We set $\mathbf{D} = 1, 2, 3$ and followed Eqs. 12 to construct the D-ASL, then build the loss function based on Eq. 13. Adam [65] is used as our training optimizer. The learning rate is initialized to be $10^{-4}$. The validation was conducted after every training epoch. If the decrease in the validation loss was lower than

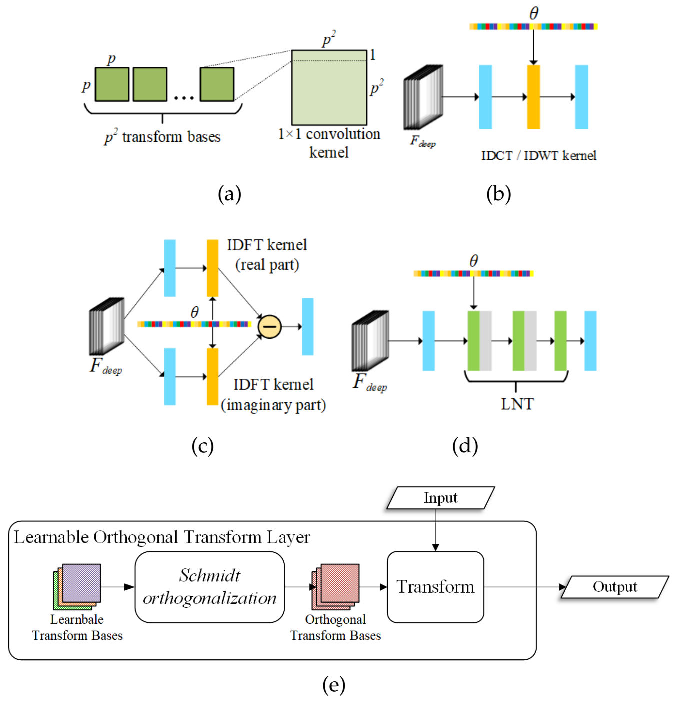

Fig. 7. Constructing the FDIT layer with IDCT, IDWT, IDFT and LNT.

0.001 dB for four consecutive epochs, the learning rate was halved. When the learning rate became lower than $10^{-6}$, the training procedure was completed. For all three datasets, we grouped training data into $256 \times 256$ patches, and set the batch size to 4. Training a MBCNN roughly takes 40 hours with a NVidia RTX2080Ti GPU.

### 5.2 Frequency Domain Transform

As described in Sec. 4.1, we introduced a block-wise FDIT layer to construct the LBF. In this subsection, we compare four different FDITs, including Inverse Discrete Cosine Transform (IDCT), Inverse Discrete Wavelet Transform (IDWT), Inverse Discrete Fourier Transform (IDFT), Learnable Non-linear Transform (LNT) and a new Learnable Orthogonal Transform (LOT). As shown in Figure 7(a) and Figure 7(b), we use $1 \times 1$ convolution layers to simulate the IDCT and IDWT, whose convolution kernel is fixed to the corresponding transform bases. Since the IDFT is a complex transform, we first use two convolution layers to estimate the real part and imaginary part from $F_{deep}$ separately, then calculate the final real output (shown in Figure 7(c)). In addition to the classic IDCT, IDWT and IDFT, we introduce the LOT and the LNT to investigate the performance of a totally learned transform. We construct the LNT by sequentially stacking two $1 \times 1$ *Conv_Relu* layers and a $1 \times 1$ convolution layer. All convolution layers in the LNT output a $p^2$-channel feature map. In this situation, the learnable passband $\theta$ is directly applied on the first $1 \times 1$ *Conv_Relu* layer (shown in Figure 7(d)). We construct the LOT using a $1 \times 1$ convolution layer with an orthogonality constraint. The orthogonality constraint ensures all learned bases are orthogonal to each other and that there will be no information loss after the transform. We adopt the method proposed in [66] to initialize orthogonal transform bases. Because the loss back-propagation is random in the training stage, we can regard the updated orthogonal bases as linearly independent. Then we adopt *Schmidt orthogonalization* to ensure the learnt transform bases are orthogonal during training procedure, where *Schmidt orthogonalization* is a differentiable operation that allows to generate transform bases from a set of linear independent vectors. The working flow of the LOT layer is shown in Figure 7(e).

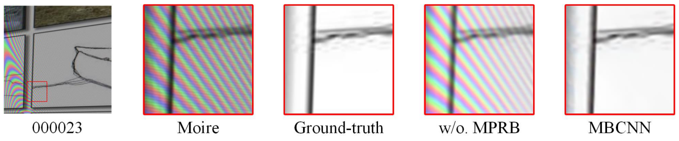

Fig. 8. Demoireing results produced by MBCNN with and without MPRB.

We adopt the settings listed in Sec. 5.1 and compare the five FDITs on the *LCDMoire* dataset. As shown in Table 2, all five transforms produce decent results, with the IDCT producing the best results. The LOT and LNT constructed LBFs have to learn the transform bases and corresponding priors at the same time, making it difficult and inefficient to learn. Both DCT and DFT are the discrete formations of the Fourier transform, where DCT is a special form of DFT as DCT contains only the real part. The DCT has better performance than DFT, we speculate for two reasons: first, estimating the complex response (as in the DFT) requires twice the outputs than estimating only the real part (as in the DCT), which makes it difficult to make an accurate estimation; second, the DCT achieves a better energy compaction than the DFT in the frequency domain, which make it easier to sense the moire patterns. We adopt the DCT-constructed LBF in the following experiments.

### 5.3 Ablation Studies

To verify the effectiveness of each component in our model, we conduct extensive ablation studies, including MPRB vs. GTMB and LTMB, learnable bandpass filter, and loss function. We adopt L1 + ASL as the training loss function in this subsection, except for the loss function study part.

#### 5.3.1 *MPRB vs. GTMB and LTMB*

As described in previous sections, the MPRB is designed to remove moire patterns, while GTMB and LTMB are designed for color restoration. We first investigate the effect of the MPRB using a trained MBCNN, and visualize the experimental results in Figure 8. Due to the residual connection in MPRB, we can separate the effect of MPRB from the two tone mapping blocks by forcing the $\alpha$ in the FSL to be zero. As shown in Figure 8, without MPRBs, the degraded color can still be well restored, and some of very high frequency moire patterns can also be well removed. However many high frequency image details are lost, and the low-frequency moire pattern largely remains. The result is mainly caused by two reasons. First, because $3 \times 3$ convolutions are used in GTMB and LTMB, the CNN has certain denoising and local smoothing capabilities. Second, although the proposed tone mapping blocks do have a great ability to restore color, the major contribution to moire pattern removal is made by MPRBs. This experiment demonstrates that the MPRBs have strong capability to remove moire patterns, while the GTMBs and LTMBs are good at restoring colors.

Moreover, we also conduct ablation experiments to investigate the performance of three blocks separately. Because our GTMB is similar to channel attention (CA), the CA is also included in the comparison. We adopt the CA proposed in [61], and set the reduction to be 0.25. The comparison results are shown in Table 3. Removing the GTMB or MPRB leads to a strong degradation in performance. However removing the LTMB yields a milder performance degradation, which is due to the LBF in the MPRB partly restoring the local color information when removing the moire pattern. On the other hand, our GTMB outperforms CA by 1.39dB/0.069 PSNR/SSIM. The channel reduction and the *Sigmoid* activation seriously weaken the global color adjustment ability of CA.

TABLE 3
Ablation investigation between MPRB, GTMB, LTMB and CA.

| MPRB | GTMB | LTMB | CA | PSNR | SSIM |
|---|---|---|---|---|---|
| × | ✓ | ✓ | × | 41.38 | 0.9869 |
| ✓ | × | ✓ | × | 40.02 | 0.9890 |
| ✓ | ✓ | × | × | 43.35 | 0.9895 |
| ✓ | × | ✓ | ✓ | 42.87 | 0.9893 |
| ✓ | ✓ | ✓ | × | 44.26 | 0.9962 |

### 5.3.2 Learnable Bandpass Filter

In this subsection, we investigate the contribution of LBF and M-LBFs from the structural contribution, and explain the reasons why we choose the relevant settings.

TABLE 4
Performance of MBCNN, MBCNN-nLP and MBCNN-nTDT on *LCDMoire* validation set.

| | LBF | | | MLBFs | |
|---|---|---|---|---|---|
| $p$ | 8 | 8 | 8 | $\{6, 8\}$ | $\{4, 6, 8\}$ |
| Structure | w/o. FDT | w/o. LP | FDT and LP | - | - |
| PSNR | 42.91 | 43.09 | 44.04 | 44.20 | 44.26 |
| SSIM | 0.9932 | 0.9936 | 0.9948 | 0.9958 | 0.9962 |

**Structural Contribution**. The M-LBFs contain several LBFs, which are constructed by two parts, namely Frequency Domain Transform (FDT) and the Learnable Passband (LP). To investigate the contribution of each component, we applied the settings described in Section 5.1.

As $p = 8$ has been prooved to be a best setting for LBF, we first used a single LBF ($p$=8) and respectively removed FDT and LP to investigate the importance of both components. We removed the entire FDT by replacing it by a $1 \times 1$ convolution layer to keep the output shape unchanged. In this case, the MPRB degenerates to a residual dense block. We removed the LP by keeping the entire FDT, but forcing all parameters in the passbands to be 1, so they will not be updated during training phase. We tested the performance of these three models on the validation set of *LCDMoire*. As shown in Table 4, introducing the FDT could provide a structural learning path and explicitly ensure the internal receptive field (block-IDCT size), and finally leads to a slight improvement of 0.18dB. Introducing LP makes it possible to learn the frequency prior of the moire patterns and leads a significant improvement of 0.95 dB from the FDT structure.

We then compared MLBFs of a single LBF with MLBFs of two LBFs ($p$ = 6, 8) and three LBFs ($p$ = 4, 6, 8). As shown in Table 4, introducing additionally sized LBFs can further strengthen the ability of learning frequency priors and lead a further improvement of 0.22 dB from the single-LBF model.

TABLE 5
Performance comparison of different loss functions.

| | Loss | $\lambda$ | PSNR (dB) | SSIM |
|---|---|---|---|---|
| Other | L1 | - | 42.19 | 0.9941 |
| | L1 + Sobel | 0.5 | 43.43 | 0.9956 |
| | L1 + Laplace | 0.5 | 43.02 | 0.9950 |
| | L1 + SSIM | 0.2 | 43.25 | 0.9958 |
| | L1 + perceptual | 1.0 | 44.39 | 0.9961 |
| | L1 + Wavelet | 0.6 | 40.66 | 0.9925 |
| Proposed | L1 + ASL | 0.25 | 44.26 | 0.9962 |
| | L1 + D-ASL$^{\{1,2\}}$ | 0.25 | 44.78 | 0.9964 |
| | L1 + D-ASL$^{\{1,2,3\}}$ | 0.25 | **45.08** | **0.9967** |
| | L1 + D-ASL$^{\{1,2,3,4\}}$ | 0.25 | 44.76 | 0.9964 |

### 5.3.3 Study of the Loss Function

In this part, we investigate the contribution from the loss functions. To demonstrate the effectiveness of the proposed ASL and D-ASL, we compare them with several related and well-known loss functions, including Sobel loss, Laplace loss, SSIM loss [45], [46], wavelet loss [67] and perceptual loss based on a pre-trained Vgg16 network [68]. Generally, all loss functions are applied through the multi-supervision strategy stated in Eq. 14 and finally measured by an MAE function. To balance these losses and the L1 loss, we assigned different values of $\lambda$ (in Eq. 13) to different losses.

As shown in Table 5, the structural high frequency loss provided by the Sobel loss leads to a significant improvement of 1.24 dB, and the additional two directional filters from ASL further improve the performance of 0.83dB. Although the Laplace loss is also a high frequency descriptor, because it has a much higher weight on the center pixel than the neighbouring pixels, it behaves similar to the L1 loss. The SSIM loss and perceptual loss also can improve the performance. The SSIM loss behaves similar to Sobel loss. Benefiting from the strong feature extraction ability of the VGG16 network, the perceptual loss achieved a good performance similar to ASL. As for the wavelet loss, we find it doesn't work as desired, possibly for two reasons. First, because the L1 loss already sufficiently captures the low frequency loss, the additional low frequency loss from the wavelet loss may mislead the network to overfit on the training set. Second, the Haar wavelet decomposition only focuses on a $2 \times 2$ neighborhood, which cannot provide much structural information for learning the frequency domain priors. Moreover, stacking multiple ASLs of different dilation rates to build D-ASL can capture much richer frequency information and greatly improve the performance. However, introducing an over-dilated ASL will lead a performance reduction. The larger dilation rate provides the information of lower frequency. As the moire patterns are mainly high-frequency artifacts, the lowest-frequency information hardly provides any help for learning the frequency priors.

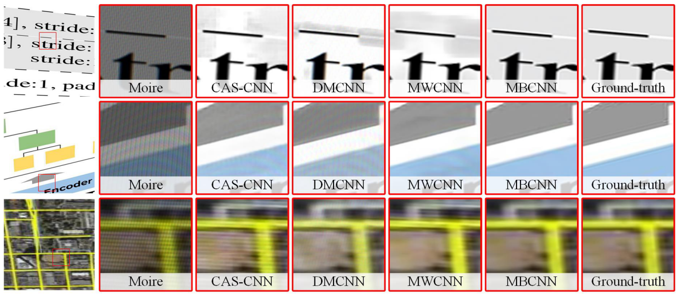

Fig. 9. Demoireing results on the validation set of *LCDMoire* produced by proposed methods and other prior methods.

TABLE 6
Performance comparison of MBCNN models and other prior work on the validation set of *LCDMoire*.

| Model | CAS-CNN | MWCNN | DMCNN | MBCNN-conf | MBCNN |
|---|---|---|---|---|---|
| PSNR | 36.16 | 28.93 | 35.48 | 44.04 | 45.08 |
| SSIM | 0.9873 | 0.9698 | 0.9785 | 0.9948 | 0.9967 |
| Parameters | 4.9M | 15.4M | 1.4M | 13.9M | 14.9M |
| Time(s) | 0.14 | 0.14 | 0.10 | 0.25 | 0.29 |

### 5.4 Comparison With Prior Work

In this subsection, we compare the proposed method with several most related prior work.

**Comparison on LCDMoire dataset**.

Since the ground-truth of the *LCDMoire* hold-out testing set is not released, we also compare with several related methods that did not participate in the challenge, on the validation set of *LCDMoire*. The compared methods are CAS-CNN [33], MWCNN [34], DMCNN [14]. We retrained all compared methods with the same training settings described in Sec. 5.1 for fair comparison. The result and average running time per image are shown in Table 6. Limited by the global residual connection, MWCNN fails to solve the image demoireing problem, while CAS-CNN achieves a very close performance to DMCNN. Owing to the moire image degradation model, M-LBFs and D-ASL, the proposed MBCNN method clearly outperforms the other methods, with a significant performance gain of $9.60\text{dB}/0.094$ PSNR/SSIM than CAS-CNN.

From the visualized results shown in Figure 9, our MBCNN accurately removes moire patterns and restores most image details. Owing to the global tone mapping components, our method can effectively handle the global color degradation. Other methods all suffer from irregular and inhomogeneous local artifacts. The single step training tries to balance the network for all pixels. Because normal 2D convolution shares the same weights in local neighbors, it difficult to use one group of convolutions to address all color degradations. Thus, although all of the participated methods have relatively large receptive fields, they all have difficulty in solving the global color degradation.

**Comparison on the TIP2018 dataset**. Since some related work is evaluated on the *TIP2018* dataset, we further evaluated our MBCNN on the *TIP2018* dataset to compare with several related methods including DnCNN [69], VDSR [71], EDSR [63], UNet [39], DMCNN [14], MopNet [17], WDNet [13]. We retrained all compared methods with the same settings provided in [14] except for MopNet, that the result of MopNet is reported in original paper. We exhibit the Parameters, GFlops and PSNR/SSIM results of compared methods in Table 8. Our MBCNN beats the second best method by 2.28 dB, in terms of PSNR, and achieved the second best SSIM result which is only 0.004 lower than the best. Moreover, the visualized results shown in Figure 10 also demonstrates the proposed method outperformed other compared methods. MBCNN is able to remove the moire patterns of large range of frequencies and at the same time keep the real high frequency patterns.

TABLE 7
Performance comparison of MBCNN models using different number of feature channels on *TIP2018* dataset.

| Model | MBCNN-16 | MBCNN-32 | MBCNN-64 | MBCNN-128 |
|---|---|---|---|---|
| PSNR | 29.80 | 30.36 | 30.41 | 30.48 |
| SSIM | 0.892 | 0.897 | 0.900 | 0.901 |
| Parameters | 1.45M | 4.42M | 14.9M | 54.3M |

We also conducted the experiments to test the effects of number of channels, by setting $n_D$ to 16, 32, 64, and 128, respectively, while keeping other settings unchanged. Table 7 shows the quantitative results of the four models. On one hand, not surprisingly, larger number of channels leads to larger model and better performance. On the other hand, increasing the number of channels leads to better improvement for the smaller model, *e.g.*, by changing 16 to 32, the PNSR improved by 0.56dB, while changing 64 to 128, the PSNR improved by 0.07dB. Note that our smaller and efficient model (MBCNN-16) has only 1.45M parameters, but still beats several most recently proposed methods with great margins.

**Comparison on the London's Buildings dataset**. The most recent dataset is *London's Buildings* dataset [13], which is also the most challenging dataset, on which the best model in the literature can only achieve 25.41dB PSNR so far. Table 9 shows quantitative comparison between MBCNN and five state-of-the-art methods, including UNet [39], DMCNN [70], ResNet-34 [56], CFNet [15], and WDNet [13]. MBCNN achieves the best PSNR of 25.82, which is 0.41dB higher than the second best. We note that WDNet tends to have better SSIM values on both *TIP2018* and *London's Buildings* datasets. WDNet also reported an interesting phenomenon that by increasing the level of wavelet transform of WDNet, the PSNR gets higher but the SSIM gets lower. [72] shows that perceptual quality (SSIM) and distortion (PSNR) may conflict with each other, especially in image restoration. Figure 11 shows qualitative results comparison. MBCNN can remove the moire patterns well and also have better color restoration.

**Generalization to new real data.** To test our model's ability to generalize to new data, we took a few new pictures using an iPhone6S from an AOC 27B2H screen, which are not used in the *TIP2018* dataset, and tested our model that was trained on *TIP2018* dataset. As shown in Figure 12, our model was able to remove moire patterns successfully, even though the moire patterns are very strong and come from a new set of camera and screen.

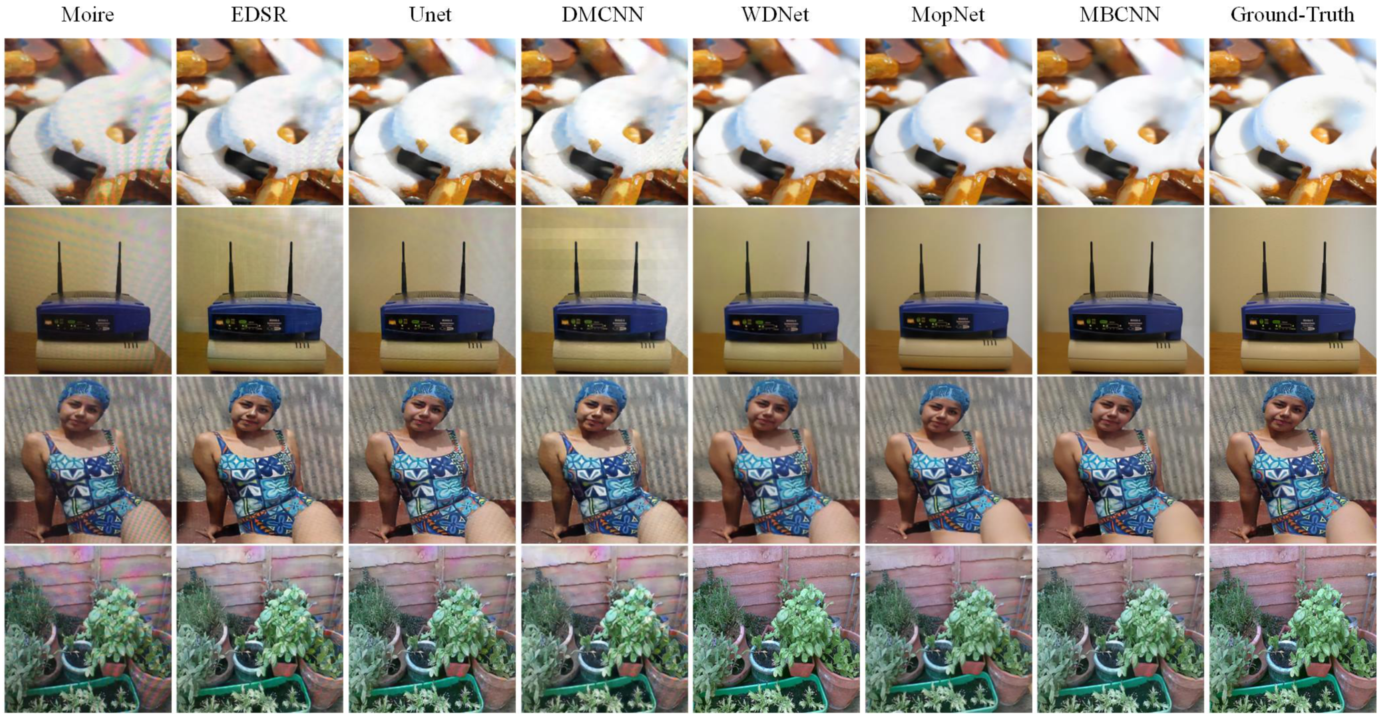

Fig. 10. Qualitative comparison on *TIP2018* dataset.

TABLE 8
Performance comparison of MBCNN models and other related works on *TIP2018* dataset. The Flops and Times are measured by processing a $256 \times 256$ sized color image.

| | DnCNN [69]<br>*TIP'16* | VDSR [24]<br>*CVPR'16* | EDSR [63]<br>*CVPRW'17* | UNet [39]<br>*MICCAI'15* | DMCNN [70]<br>*TIP'18* | MopNet [17]<br>*ICCV'19* | WDNet [13]<br>*ECCV'20* | MBCNN-conf [20]<br>*CVPR'20* | **MBCNN**<br>*proposed* |
|---|---|---|---|---|---|---|---|---|---|
| PSNR | 24.54 | 24.68 | 26.82 | 26.49 | 26.77 | 27.75 | 28.08 | 30.03 | **30.41** |
| SSIM | 0.834 | 0.837 | 0.853 | 0.864 | 0.871 | 0.895 | **0.904** | 0.893 | 0.900 |
| Parameters | 0.6M | 0.7M | 12.2M | 8.6M | 1.4M | 12.4M | 5.7M | 13.5M | 14.9M |
| Flops | 72.2G | 87.5G | 160.3G | 32.8G | 49.3G | 396.1G | 42.9G | 125.3G | 148.6G |
| Time | 16.4ms | 10.5ms | 118ms | 6.2ms | 9.8ms | 123.4ms | 25.1ms | 19.4ms | 22.9ms |

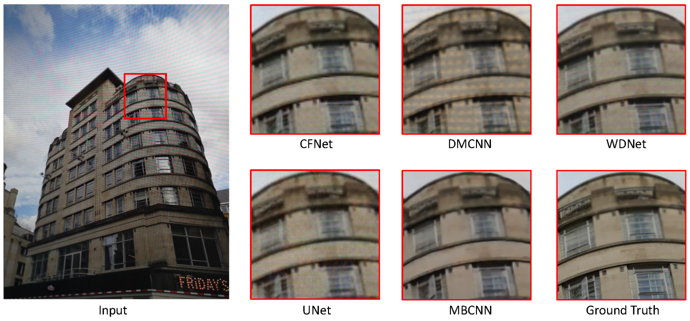

Fig. 11. Visual comparison on the London's Buildings data set.

TABLE 9
Quantitative comparison on London's Buildings.

| | UNet | DMCNN | ResNet-34 | CFNet | WDNet | MBCNN |
|---|---|---|---|---|---|---|
| PSNR | 23.48 | 23.64 | 23.87 | 23.22 | 25.41 | **25.82** |
| SSIM | 0.790 | 0.791 | 0.780 | 0.764 | **0.839** | 0.816 |

$^*$All the results except for MBCNN are reported from [13].

## 6 Conclusion

In this paper, we propose a multi-scale bandpass CNN (MBCNN) for image demoireing, and significantly outperform state-of-the-art methods by more than 2dB in terms of PSNR. Multi-block-size learnable bandpass filters (MLBFs) and dilated advanced Sobel loss (D-ASL) are proposed to learn the frequency prior of moire patterns. Our model has two steps: moire pattern removal and tone mapping. A MLBFs-based residual CNN block is used for moire pattern removal, and another two CNN blocks for global and local tone mappings. We firstly compare the performances of four frequency domain transforms for the frequency domain priors learning, including IDWT, IDCT, IDFT, LNT and LOT, and demonstrate the IDCT is the most appropriate choice. Then, the ablation study was conducted to show the importance of the components in the network. We have also clarified the the effects of the block-IDCT sizes in the MLBFs, and the dilation rates in the D-ASL. We demonstrated that the block-IDCT sizes of $\{4, 6, 8\}$ to formulate the MLBFs, and the dilation rates of $\{1, 2, 3\}$ to construct the D-ASL, are the best settings for the image demoireing task. Finally, experiments on three datasets show that our model outper-

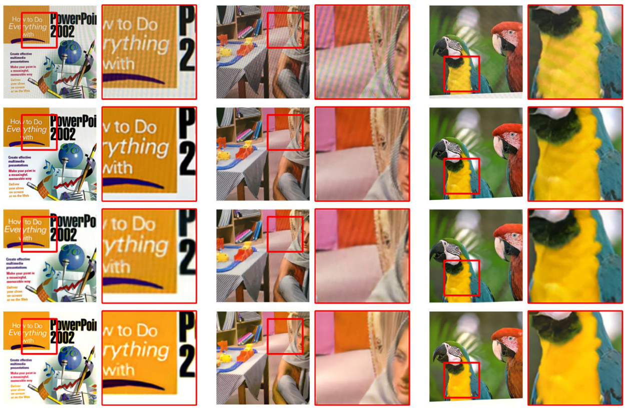

Fig. 12. Visualized results produced by MBCNN on real moire images. From top to bottom are the input moire images, and the results of MopNet, WDnet, and MBCNN. The moire images are captured with iPhone6S and AOC 27B2H.

forms state-of-the-art methods by large margins, and the light versions (MBCNN-16 and MBCNN-32) of our model can achieve an great balance between performance and efficiency.

## 7 Acknowledgement

This work is supported by The National Key Research and Development Program of China SY2020YFB1406604, National Nature Science Foundation of China (62001146, 61931008, 61671196, 61701149, 61801157, 61971268, 61901145, 61901150, 61972123), National Natural Science Major Foundation of Research Instrumentation of PR China under Grants 61427808, Zhejiang Province Nature Science Foundation of China (LR17F030006, Q19F010030), 111 Project, No. D17019.

## References

[1] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *TPAMI, 2010*.

[2] J. Pan, D. Sun, H. Pfister, and M. Yang, "Deblurring images via dark channel prior," *TPAMI, 2018*.

[3] N. Joshi, C. L. Zitnick, R. Szeliski, and D. J. Kriegman, "Image deblurring and denoising using color priors," in *CVPR, 2009*.

[4] R. Fattal, "Dehazing using color-lines," *ACM transactions on graphics (TOG)*, vol. 34, no. 1, pp. 1–14, 2014.

[5] T. M. Bui and W. Kim, "Single image dehazing using color ellipsoid prior," *TIP, 2017*.

[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *TIP, 2007*.

[7] X. Liu, G. Cheung, X. Wu, and D. Zhao, *TIP, 2017*.

[8] K. He, J. Sun, and X. Tang, "Guided image filtering," *TPAMI, 2012*.

[9] H. Cho, H. Lee, H. Kang, and S. Lee, "Bilateral texture filtering," *TOG, 2014*.

[10] T. S. Cho, C. L. Zitnick, N. Joshi, S. B. Kang, R. Szeliski, and W. T. Freeman, *TPAMI, 2012*.

[11] J. Guo and H. Chao, "Building dual-domain representations for compression artifacts reduction," in *ECCV, 2016*.

[12] K. Xu, M. Qin, F. Sun, Y. Wang, Y.-K. Chen, and F. Ren, "Learning in the frequency domain," in *CVPR*, 2020, pp. 1740–1749.

[13] L. Liu, J. Liu, S. Yuan, G. Slabaugh, A. Leonardis, W. Zhou, and Q. Tian, "Wavelet-based dual-branch network for image demoireing," in *ECCV, 2020*.

[14] Y. Sun, Y. Yu, and W. Wang, "Moire photo restoration using multiresolution convolutional neural networks," *TIP, 2018*.

[15] B. Liu, X. Shu, and X. Wu, "Demoireing of camera-captured screen images using deep convolutional neural network," *arXiv, 2018*.

[16] T. Gao, Y. Guo, X. Zheng, Q. Wang, and X. Luo, "Moiré pattern removal with multi-scale feature enhancing network," in *ICMEW, 2019*.

[17] B. He, C. Wang, B. Shi, and L.-Y. Duan, "Mop moire patterns using mopnet," in *ICCV, 2019*.

[18] X. Cheng, Z. Fu, and J. Yang, "Multi-scale dynamic feature encoding network for image demoiréing," in *ICCVW, 2019*.

[19] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *ICCV, 2017*.

[20] B. Zheng, S. Yuan, G. Slabaugh, and A. Leonardis, "Image demoireing with learnable bandpass filters," in *CVPR, 2020*.

[21] C. Dong, Y. Deng, C. C. Loy, and X. Tang, "Compression artifacts reduction by a deep convolutional network," in *ICCV, 2015*.

[22] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *ECCV, 2014*.

[23] P. Svoboda, M. Hradis, D. Bařina, and P. Zemcík, "Compression artifacts removal using convolutional neural networks," *Journal of WSCG*, vol. 24, pp. 63–72, 05 2016.

[24] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *CVPR, 2016*.

[25] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," in *CVPR, 2017*.

[26] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR, 2015*.

[27] J. Kim, J. Kwon Lee, and K. Mu Lee, "Deeply-recursive convolutional network for image super-resolution," in *CVPR, 2016*.

[28] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *CVPR, 2017*.

[29] Y. Tai, J. Yang, X. Liu, and C. Xu, "MemNet: A persistent memory network for image restoration," in *ICCV, 2017*.

[30] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *CVPR, 2018*.

[31] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *CVPR, 2017*.

[32] L.-F. Dong, Y.-Z. Gan, X.-L. Mao, Y.-B. Yang, and C. Shen, "Learning deep representations using convolutional auto-encoders with symmetric skip connections," in *ICASSP, 2018*.

[33] L. Cavigelli, P. Hager, and L. Benini, "CAS-CNN: A deep convolutional neural network for image compression artifact suppression," in *IJCNN, 2017*.

[34] P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo, "Multi-level wavelet-cnn for image restoration," in *CVPRW, 2018*.

[35] B. Zheng, Y. Chen, X. Tian, F. Zhou, and X. Liu, "Implicit dual-domain convolutional network for robust color image compression artifact reduction," *TCSVT, 2019*.

[36] X. Luo, J. Zhang, M. Hong, Y. Qu, Y. Xie, and C. Li, "Deep wavelet network with domain adaptation for single image demoireing," in *CVPRW, 2020*.

[37] A. Gia Vien, H. Park, and C. Lee, "Dual-domain deep convolutional neural networks for image demoireing," in *CVPRW, 2020*.

[38] G. Eilertsen, J. Kronander, G. Denes, R. K. Mantiuk, and J. Unger, "Hdr image reconstruction from a single exposure using deep cnns," *TOG, 2017*.

[39] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI, 2015*.

[40] M. Gharbi, J. Chen, J. T. Barron, S. W. Hasinoff, and F. Durand, "Deep bilateral learning for real-time image enhancement," *TOG, 2017*.

[41] K. He, J. Sun, and X. Tang, "Guided image filtering," in *ECCV, 2010*.

[42] H. Wu, S. Zheng, J. Zhang, and K. Huang, "Fast end-to-end trainable guided filter," in *CVPR, 2018*.

[43] W. Ren, L. Ma, J. Zhang, J. Pan, X. Cao, W. Liu, and M.-H. Yang, "Gated fusion network for single image dehazing," in *CVPR, 2018*.

[44] H. Zhang and V. M. Patel, "Densely connected pyramid dehazing network," in *CVPR, 2018*.

[45] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *TCI, 2016*.

[46] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *TIP, 2004*.

[47] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *CVPR, 2017*.

[48] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy, "Esrgan: Enhanced super-resolution generative adversarial networks," in *ECCV, 2018*.

[49] J. Guo and H. Chao, "One-to-many network for visually pleasing compression artifacts reduction," in *CVPR, 2017*.

[50] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv, 2014*.

[51] D. Sun, S. Roth, and M. J. Black, "Secrets of optical flow estimation and their principles," in *CVPR 2010*.

[52] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *ECCV, 2016*.

[53] M. Cheon, J.-H. Kim, J.-H. Choi, and J.-S. Lee, "Generative adversarial network-based image super-resolution using perceptual content losses," in *ECCV, 2018*.

[54] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *IJCV, 2015*.

[55] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *ICLR, 2016*.

[56] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR, 2016*.

[57] C. Yan, Y. Tu, X. Wang, Y. Zhang, X. Hao, Y. Zhang, and Q. Dai, "Stat: spatial-temporal attention mechanism for video captioning," *TMM, 2019*.

[58] X. Wang, K. C. Chan, K. Yu, C. Dong, and C. Change Loy, "Edvr: Video restoration with enhanced deformable convolutional networks," in *CVPRW, 2019*.

[59] C. Yan, B. Gong, Y. Wei, and Y. Gao, "Deep multi-view enhancement hashing for image retrieval," *TPAMI, 2020*.

[60] C. Yan, B. Shao, H. Zhao, R. Ning, Y. Zhang, and F. Xu, "3d room layout estimation from a single rgb image," *TMM, 2020*.

[61] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *ECCV, 2018*.

[62] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *CVPR, 2018*.

[63] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *CVPRW, 2017*.

[64] S. Yuan, R. Timofte, G. Slabaugh, and A. Leonardis, "Aim 2019 challenge on image demoreing: dataset and study," in *ICCVW, 2019*.

[65] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2014.

[66] A. M. Saxe, J. L. Mcclelland, and S. Ganguli, "Exact solutions to the nonlinear dynamics of learning in deep linear neural network," in *ICLR, 214*.

[67] X. Cheng, Z. Fu, and J. Yang, "Improved multi-scale dynamic feature encoding network for image demoiréing," *PR, 2021*.

[68] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *ECCV, 2016*.

[69] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *TIP, 2017*.

[70] X. Zhang, W. Yang, Y. Hu, and J. Liu, "Dmcnn: Dual-domain multi-scale convolutional neural network for compression artifacts removal," in *ICIP, 2018*.

[71] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *CVPR, 2016*.

[72] Y. Blau and T. Michaeli, "The perception-distortion tradeoff," in *CVPR, 2018*.

**Bolun Zheng** received the B.S. and Ph.D. degrees in electronic information technology and instrument from Zhejiang University in 2014 and 2019, respectively. He is currently a lecturer with Hangzhou Dianzi University. His research interests are computer vision, pattern recognition, image processing and embedded parallel computing.

**Shanxin Yuan** is a Senior Research Scientist in Computer Vision for Huawei Technologies, R&D. He received the PhD degree from Imperial College London, the MSc degree from the University of Chinese Academy of Sciences, and the BSc degree from China Agricultural University. His research interests are machine learning and computer vision, and he is currently working on low-level vision.

**Chenggang Yan** received the B.S. degree in computer science from Shandong University in 2008, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences in 2013. He is currently a Professor with Hangzhou Dianzi University. Before that, he was an Assistant Research Fellow of Tsinghua University. His research interests include intelligent information processing, machine learning, image processing, computational biology, and computational photography.

**Xiang Tian** received B.Sc. and Ph.D. degrees in signal processing from Zhejiang University, Hangzhou, China, in 2001 and 2007, respectively. He is currently an Associate Professor at Zhejiang University. His research focuses on the fields of signal processing, video coding, image processing and computer vision.

**Jiyong Zhang** received the B.S. and M.S. degrees in computer science from Tsinghua University in 1999 and 2001, respectively, and the Ph.D. degree in computer sciences from the Swiss Federal Institute of Technology (EPFL), Lausanne, in 2008. He is currently a Distinguished Professor with Hangzhou Dianzi University. His research interests include intelligent information processing, machine learning, data sciences, and recommender systems.

**Yaoqi Sun** received the B.S. degree from the Zhejiang University of Science and Technology, Zhejiang, China, in 2012. He is currently pursuing the M.E. degree with Hangzhou Dianzi University, Zhejiang. His research interests include intelligent information processing, machine learning and pattern recognition.

**Lin Liu** received the B.S. degree from University of Science and Technology of China, in 2019. He is currently pursuing the M.E. degree with University of Science and Technology of China. His research interests include computer vision and machine learning. Now he is a student research intern at Huawei working on low-level vision tasks.

**Aleš Leonardis** is a Senior Research Scientist and Computer Vision Team Leader at Huawei Technologies Noah's Ark Lab (London, UK). He is also Chair of Robotics at the School of Computer Science, University of Birmingham and Professor of Computer and Information Science at the University of Ljubljana. Previously, he held research positions at GRASP Laboratory at the University of Pennsylvania and at Vienna University of Technology. He is a Fellow of the IAPR.

**Gregory Slabaugh** is Professor of Computer Vision and AI and Director of the Digital Environment Research Institute (DERI) at Queen Mary University of London. Previously, he was Chief Scientist in Computer Vision (EU) for Huawei Technologies R&D. His research interests include computational photography, medical image computing, and applications of deep learning.