Image Demoireing with Learnable Bandpass Filters

Bolun Zheng
Hangzhou Dianzi University
zhengbolun1024@163.com

Shanxin Yuan
Huawei Noah's Ark Lab
shanxin.yuan@huawei.com

Gregory Slabaugh
Huawei Noah's Ark Lab
gregory.slabaugh@huawei.com

Aleš Leonardis
Huawei Noah's Ark Lab
ales.leonardis@huawei.com

Abstract

Image demoireing is a multi-faceted image restoration task involving both texture and color restoration. In this paper, we propose a novel multiscale bandpass convolutional neural network (MBCNN) to address this problem. As an end-to-end solution, MBCNN respectively solves the two sub-problems. For texture restoration, we propose a learnable bandpass filter (LBF) to learn the frequency prior for moire texture removal. 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. Through an ablation study, we demonstrate the effectiveness of the different components of MBCNN. Experimental results on two public datasets show that our method outperforms state-of-the-art methods by a large margin (more than 2dB in terms of PSNR).

1. Introduction

Digital screens are ubiquitous in modern daily life. We have TV screens at home, laptop/desktop screens in the office, and large LED screens in public spaces. It is becoming common practice to take pictures of these screens to quickly save information. 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. Moire patterns appear 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.

Unlike other image restoration problems, including denoising [44], demosaicing [9], color constancy [1], sharpening [28], etc., much less attention has been paid to image demoireing, which is to recover the underlying clean image from an image contaminated by moire patterns. Only very recently, a few attempts [31, 24, 8, 12] have been made to address image demoireing. However, the problem remains to a large extent an unsolved problem, due to the large variation of moire patterns in terms of frequencies, shapes, colors, etc.

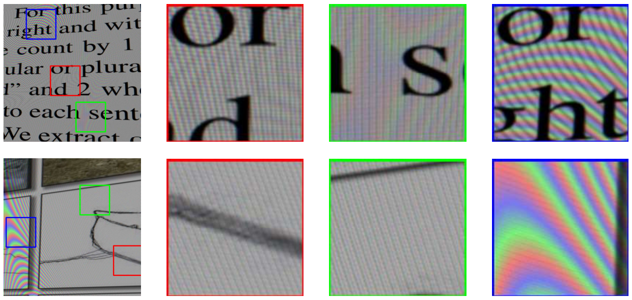


Figure 1. Moire texture of different scales, frequencies, and colors.

Recent works [31, 3, 12] tried to remove moire patterns of different frequency bands through multi-scale design. DMCNN [31] 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 [3] improved DMCNN by introducing an adaptive instance normalization [17] based on a dynamic feature encoder. DCNN [24] 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 [12] used a multi-scale feature aggregation sub-module to address the complex frequency, and two other sub-modules to address edges and pre-defined moire types. Our model also adopts a multi-scale design with three branches for three different scales. Among different scales, our model adopts a gradual upsampling strategy to smoothly increase the resolution.

Generally, none of the existing methods tried to model the moire patterns explicitly. In our model, we explicitly model the moire patterns by learning the frequency prior of moire patterns and respectively restore the moire image from texture and color. Our contributions are as follows.

- We introduce a unified framework namely multi-scale bandpass CNN (MBCNN) for image demoireing. The network performs both texture restoration and color restoration within the same model.

- We propose a learnable bandpass filter (LBF) for efficient moire texture removal. The LBF introduces a learnable bandpass to learn the frequency prior, which could precisely separate moire texture from normal image texture.

- Our method includes global/local tone mapping 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.

- We also propose an advanced Sobel loss (ASL) to learn the structural high-frequency information. With the ASL, we develop a multi-scale supervision to remove moire patterns in three scales.

2. Related work

Image demoireing requires both texture and color restoration, rendering it a complex challenge. In this section, we make a brief introduction of several CNN-based methods in related tasks, where deep learning has made significant impact.

Image restoration. Dong *et al.* [4, 5] were the first to propose end-to-end convolutional neural networks for image super-resolution and compression artifact reduction. Subsequent research [32, 19, 45] further improved these models by increasing the network depth, introducing skip connections [26] and residual learning. Much deeper networks [21, 33, 34, 47] were then introduced. DRCN [21] proposed recursive learning for parameter sharing. Tai *et al.* [33, 34] introduced a recursive residual learning and proposed a memory block. Zhang *et al.* [47] replaced the recursive connection in the memory block by a dense connection [16]. Moreover, several studies focused on multi-scale CNNs inspired by high-level computer vision methods. Mao *et al.* [6] proposed a skip connection-based multi-scale autoencoder. Cavigelli *et al.* [2] introduced a multi-supervised network for compression artifact reduction.

Frequency domain learning. Several studies [25, 11, 49] focus on frequency domain. Liu *et al.* [25] introduced the discrete wavelet transform and its inverse to replace conventional upscaling and downscaling operations for image restoration. Guo *et al.* [11] introduced convolution-based window sampling, Discrete Cosine Transform (DCT) and inverse DCT (IDCT) to construct a DCT-domain learning network. Zheng et al. [49] introduced 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. [7] proposed a Gamma correction based loss function and trained a U-Net [29] based CNN for high dynamic range (HDR) image reconstruction. Gharbi *et al.* [10] proposed HDRNet to learn local piece-wise linear tone mapping. Inspired by the guided filter [13], Wu *et al.* [36] proposed an end-to-end trainable guided filter for image enhancement. Ren et al. [27] 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. [43] proposed a densely connected pyramid CNN for image dehazing. Remarkably, few of these color restoration methods introduce residual connection in their solutions.

Image demoireing. Recently, several end-to-end image demoireing solutions have been proposed. Sun *et al.* [31] first introduced a CNN for image demoireing (DMCNN) and created an ImageNet [30]-based moire dataset for training and testing. Cheng *et al.* [3] improved DMCNN by introducing an adaptive instance normalization [17] based dynamic feature encoder. He *et al.* [12] introduced additional moire attribute labels based on shape, color, and frequency for more precise moire pattern removal. None of the existing methods modeled the moire patterns explicitly. We treat the image demoireing problem as moire texture removal and color restoration.

3. Proposed method

A moire image captured by a digital camera can be modeled as:

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

where I_{clean} is the clean image displayed on the screen, N_{moire} is the introduced moire texture, and ψ 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 ψ^{-1} is the inverse function of ψ, 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 texture removal and tone mapping.

3.1. Multiscale bandpass CNN

We propose a Multi-scale Bandpass CNN (MBCNN) to do image demoireing, *i.e.*, to recover the underlying clean image from the moire image. Our model works in three scales and has three different types of blocks, which are moire texture removal block (MTRB), global tone mapping block (GTMB), and local tone mapping block (LTMB). The details of each block are described in Sec. 3.2 and Sec. 3.3.

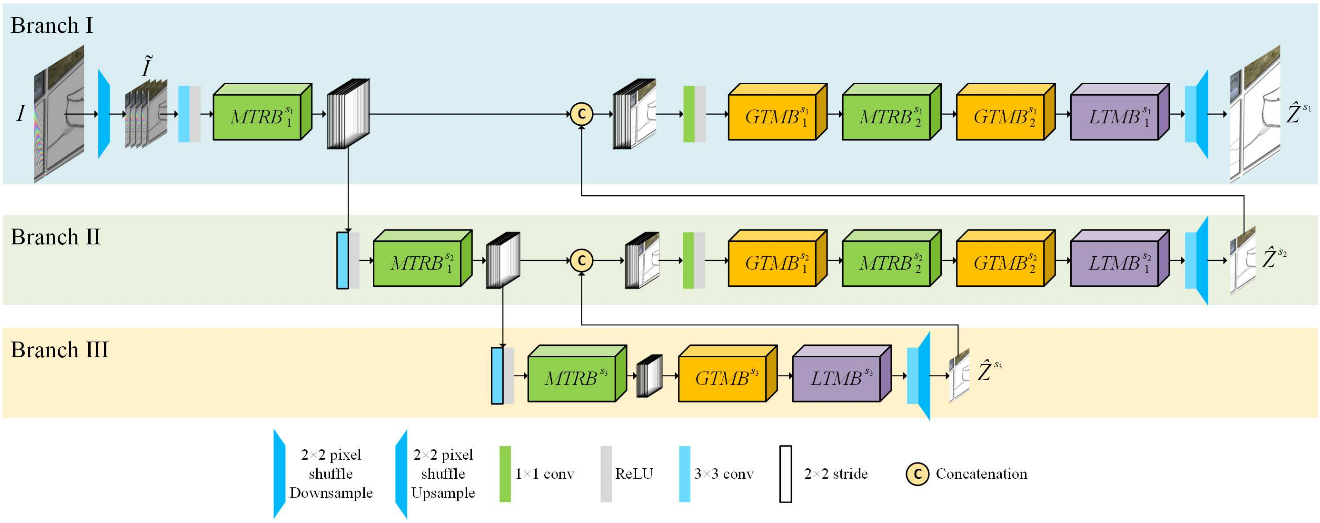


Figure 2. The architecture of our multi-scale bandpass CNN.

The architecture of MBCNN is shown in Figure 2. 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 $\frac{h}{2} \times \frac{w}{2} \times 4c$. With the tensor $\tilde{I}$ as input, the following network consists of three branches, each to recover the moire image in a specific scale. Following Eq. 2, each branch sequentially executes the moire texture 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 MTRB are stacked to remove the texture and color errors caused by the scale change.

3.2. Moire texture removal

Moire patterns exhibit considerable variation in shape, frequency, color, *etc*. Some examples are shown in Figure 1, where the moire patterns have different characteristics. The moire texture can be written as:

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

where $N^{s_i}_{f_{ij}}$ denotes the moire texture component of scale s_i and frequency f_{ij}. Following this formulation, we can first estimate the components of moire texture at different scales and frequencies, and then reconstruct the moire texture based on all the estimated components.

Block-DCT is an effective way for handling frequency related problems. Assuming that the frequency spectrum in block-DCT domain of each $N^{s_i}_{f_{ij}}$ is $FS^{s_i}_{f_{ij}}$, then Eq. 3 can be rewritten as

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

where $\mathcal{D}^{-1}$ denotes the block-IDCT function.

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

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

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

$$\begin{aligned} \mathcal{C}(N_P) &= \sum_{c \in \{R,G,B\}} \mathcal{C}(\mathcal{D}^{-1}(\sum_i \sum_j FS^{s_i}_{f_{ij}}))\Big|_c \\ &= \sum_i \mathcal{C}(\mathcal{D}^{-1}(\sum_{c \in \{R,G,B\}} \sum_j FS^{s_i}_{f_{ij}}\big|_c)) \\ &= \sum_i \mathcal{C}(\mathcal{D}^{-1}(\sum_{c \in \{R,G,B\}} FS^{s_i}\big|_c)) \end{aligned} \tag{6}$$

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

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

Learnable Bandpass Filter. Inspired by the implicit DCT [49], we can directly estimate ξ^{s_i} with a deep CNN block. Since the transforms presented in Eq. 7 are all linear, they can be modeled by a simple convolution layer. As the frequency spectrum of moire texture is always regular, we can use a bandpass filter to amplify certain frequencies and diminish others. However, it's difficult to get the frequency spectrum prior modeling the moire texture, because there would be several frequencies in different scales and they can also affect each other. To solve this problem, we

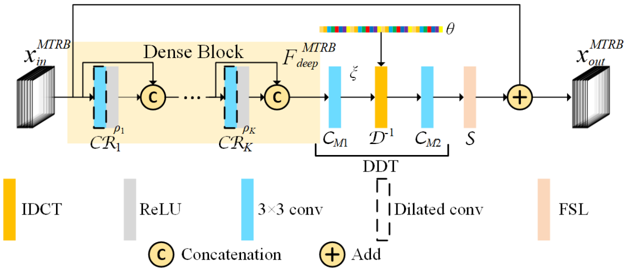


Figure 3. The structure of moire texture removal block.

propose a learnable bandpass filter (LBF) to learn the prior from moire images. LBF introduces a learnable weights for each frequency, which can be expressed as

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

where θ^{s_i} denotes the learnable weights of DCT domain frequencies for the scale s_i.

Assuming the size of block-IDCT is $p \times p$, then the corresponding DCT domain frequency spectrum totally has p^2 frequencies, so the size of θ^{s_i} is p^2. All parameters of θ^{s_i} are initialized to be 1 and constrained to be non-negative, the passbands are learned from the image data during training. $\mathcal{D}^{-1}$ can be implemented by a predefined 1×1 convolution layer, whose weights are fixed as the IDCT matrix.

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

Assuming the input of the MTRB is x_{in}^{MTRB}, a dense block is first used for feature extraction, which is denoted as F_{deep}. Then a 3×3 convolution layer estimates the IFS ξ from F_{deep}. The dense block has K densely connected [16] 3×3 n_D-channel dilated convolution [40] 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 ξ can be easily estimated from the F_{deep}. After estimating ξ, the learnable weight θ and the block-IDCT layer $\mathcal{D}^{-1}$, a convolution layer $\mathcal{C}_{M2}$ is added as indicated in Eq. 8.

Considering that the $\mathcal{D}^{-1}$ might lead to large local output and produce excessive gradient, we stacked a Feature Scale Layer (FSL) to linearly constrain the output of $\mathcal{C}_{M2}$. Finally, we introduce the residual connection [14] to remove the moire texture in convolution domain. Thus, the final output of MTRB x_{out}^{MTRB} can be obtained by

$$x_{out}^{MTRB} = x_{in}^{MTRB} + \mathcal{S}(\mathcal{C}_{M2}(\mathcal{D}^{-1}(\theta \cdot \xi))) \tag{9}$$

where $\mathcal{S}$ denotes the FSL.

Directly multiplying θ and ξ will consume large amount of calculations. Instead, we reshape θ to the size of $1 \times 1 \times p \times p$, and multiply it with the convolution kernel of $\mathcal{D}^{-1}$ layer, then the ξ is directly sent to $\mathcal{D}^{-1}$ layer. In this way, the product $\theta \cdot \xi$ can be avoided.

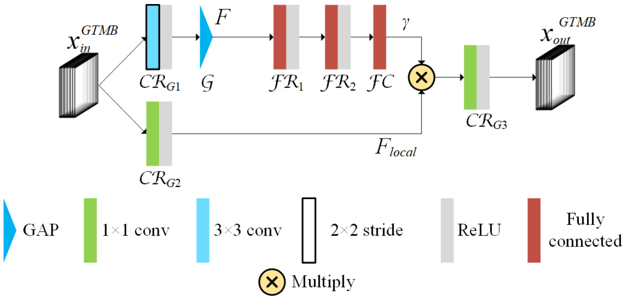


Figure 4. The structure of global tone mapping block.

3.3. Tone mapping

The RGB color space is an extremely large space containing 256^3 colors, making it difficult to do 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).

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

Table 1. Attributions of learnable layers in GTMB.

Global tone mapping block. The GTMB is proposed to learn the global color shift, see Figure 4 for the detailed structure. Given the input x_{in}^{GTMB}, we first extract a global feature F through a 3×3 $Conv\_ReLU$ layer with the stride of 2 and a global average pooling (GAP) layer. Then, to extract a deep global feature γ, we stack two fully connected (FC) layers with ReLU activation ($\mathcal{FR}_1$, $\mathcal{FR}_2$) and a FC layer without ReLU activation ($\mathcal{FC}$). Besides, we use an 1×1 $Conv\_ReLU$ layer extracts the local feature F_{local} from x_{in}^{GTMB}. The output of GTMB 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[39, 35, 37, 38], and several channel attention blocks have been proposed [46, 15]. Our GTMB can be view as a channel attention block. 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 the γ in

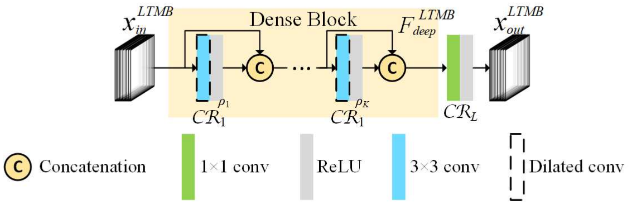
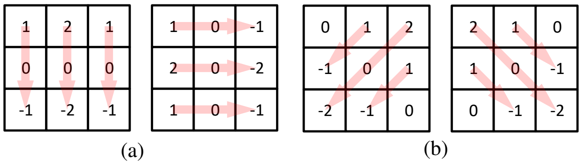


Figure 5. The structure of local tone mapping block.



1	2	1
0	0	0
-1	-2	-1

1	0	-1
2	0	-2
1	0	-1

(a)

0	1	2
-1	0	1
-2	-1	0

2	1	0
1	0	-1
0	-1	-2

(b)

Figure 6. Details of advanced Sobel loss. (a) Classic Sobel filters. (b) Two additional filters for advanced Sobel filters.

GTMB. Second, channel attention is directly applied on the input of the existing channel attention blocks, while the γ 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. 4.3.1).

Local tone mapping block. The LTMB is developed to fit a local fine-grained tone mapping function. As shown in Figure 5, the structure of LTMB is similar to MTRB. LTMB first takes a similar dense block in MTRB 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 1×1 convolution, and x_{out}^{LTMB} has the same shape with x_{in}^{LTMB}.

3.4. Loss function

In this paper, we use the L1 loss as the base loss function, as it has been proven [23, 47, 48] 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 Advanced Sobel Loss (ASL) to solve this problem. The proposed ASL can be expressed as

$$\mathcal{ASL}(\hat{Z}, Z) = \frac{1}{N}\sum \left|Sobel^*(Z) - Sobel^*(\hat{Z})\right| \tag{12}$$

where Z denotes the groundtruth, $\hat{Z}$ denotes the output of CNN, and $Sobel^*$ denotes the advanced Sobel filtering. Figure 6 illustrates the details of ASL. Compared to classic Sobel filters (Figure 6(a)), the advanced Sobel filters provide two additional filters of 45° directions (Figure 6(b)), which could provide richer structure information. We combine ASL and L1 loss as the final loss function, which can be expressed as,

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

where $\mathcal{L}1$ denotes the L1 loss, $\mathcal{ASL}$ denotes the ASL, and λ is a hyper-parameter to balance the L1 loss and ASL.

When training MBCNN, we adopt the multi-supervising strategy that supervising 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.

4. Experiments

We have conducted extensive ablation studies and outperformed state-of-the-art by large margins on two public datasets: *LCDMoire* [41] and *TIP2018* [31] The *LCDMoie* 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 [30] 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. Both *LCDMoire* and *TIP2018* datasets are used to do comparison with state-of-the-art methods. *LCDMoire* dataset is also used for ablation study. The ablation study is 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.

4.1. Implementation details

For the MBCNN model, we adopt the following settings, with $c = 3$, $n_G = 128$, $n_D = 64$, $K = 5$. Adam [22] 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 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 *LCDMoire* dataset, we 128×128 patches were randomly cropped from the images, with the batch size set to 16. When the 128×128 patch trained model converged, we re-grouped the training data into 256×256 patches for fine-tuning the model. This time, the learning rate was set to 10^{-5}, the batch size was set to 4. Training a MBCNN

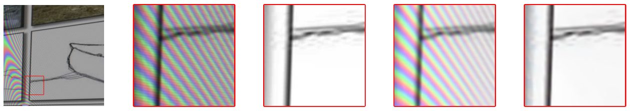

000023 Moire Ground-truth w/o. MTRB MBCNN

Figure 7. Demoireing results produced by MBCNN with and without MTRB.

roughly takes 40 hours with a NVidia RTX2080Ti GPU. For *TIP2018* dataset, we follow [31] and set the patch size as 256 × 256 through out the training.

4.2. Ablation Study

To verify the effectiveness of each component in our model, we conduct extensive ablation studies, including evaluation of MTRB vs. GTMB and LTMB, learnable bandpass filter, and loss function.

4.2.1 MTRB vs. GTMB and LTMB

As described in previous sections, the MTRB is designed for removing moire texture, GTMB and LTMB are designed for color restoration. We investigate the effect of the MTRB using a trained MBCNN, and visualize the experimental results in Figure. 7. Due to the residual connection in MTRB, we can separate the effect of MTRB from the two tone mapping blocks by forcing the learned scale in the feature scaling layer to be zero. As shown in Figure 7, without MTRBs, the degraded color can still be well restored, and some of very high frequency moire texture can also be well removed. However many high frequency image details are lost, and the low-frequency moire texture largely remains. The result is mainly caused by two reasons. First, because 3 × 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 texture removal is made by MTRBs. This experiment demonstrates that the MTRBs have strong capability to do moire texture removing, while the GTMBs and LTMBs are good at restoring colors.

4.2.2 Learnable bandpass filter

In this section, we investigate the contribution of LBF and explain the reasons why we choose the relevant settings.

Model	MBCNN-nDDT	MBCNN-nLP	MBCNN
PSNR/SSIM	42.91/0.9932	43.09/0.9936	44.04/0.9948

Table 2. Performance of MBCNN, MBCNN-nLP and MBCNN-nDDT on *LCDMoire* validation set.

Structural contribution. The LBF is constructed by two parts, DCT domain transform (DDT) and the learnable passband (LP). We applied the settings described in Section 4.1, and respectively removed the DDT and LP from the MTRBs to conduct the investigation. We removed the entire DDT by replacing it by a 1 × 1 convolution layer to keep the output shape unchanged. In this case, the MTRB degenerates to a residual dense block (RDB). We removed the LP by keeping the entire DDT, but forcing all parameters in the passbands to be 1, which will not be updated during training phase.

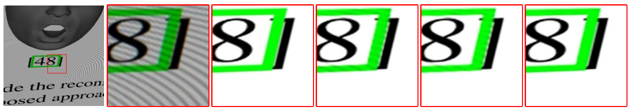

000041 Moire Ground-truth MBCNN-nDDT MBCNN-nLP MBCNN

Figure 8. Demoireing results produced by MBCNN-nDDT, MBCNN-nLP and MBCNN.

We denote the networks constructed without LP or DDT as MBCNN-nLP and MBCNN-nDDT, respectively. We tested the performance of these three models on the validation set of *LCDMoire*. As shown in Table 2, MBCNN-nLP introduces the DDT which 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 from MBCNN-nDDT. MBCNN introduces the learnable bandpass to learn the frequency prior of the moire texture and leads a significant improvement of 0.95 dB from MBCNN-nLP. Some demoireing results produced by these three models are shown in Figure 8. The LBFs enable the MBCNN to better sense the moire texture and recover more accurate details from moire images.

Model	MBCNN-6	MBCNN-8	MBCNN-10	MBCNN-12
PSNR/SSIM	43.25/0.9937	44.04/0.9948	43.45/0.9939	43.17/0.9937

Table 3. Comparison of MBCNNs with different p values.

Block-IDCT size p. p is a very important parameter for DDT. With a larger p, the LBF can learn a more accurate and more complete frequency prior. We denoted the MBCNN constructed with the block-IDCT size of p as MBCNN-p. We respectively validated the performance of MBCNNs constructed with $p = 6, 8, 10, 12$. $p = 8$ is found to be the best for moire texture removal. As shown in Table 3, larger p doesn't always lead to a better result. There are two reasons for this observation. First, enlarging p increases the complexity and difficulty of the frequency prior learning. Second, the receptive field provided by the front dense block cannot support a p that is too large. We visualize the learned passbands in the LBFs from an MBCNN-8 model in Figure 9. The LBFs perform band suppression mainly at the beginning of the branches. The LBFs at the end of the branches are primarily avoiding over-smoothing caused by concatenating the output from the upper scale.

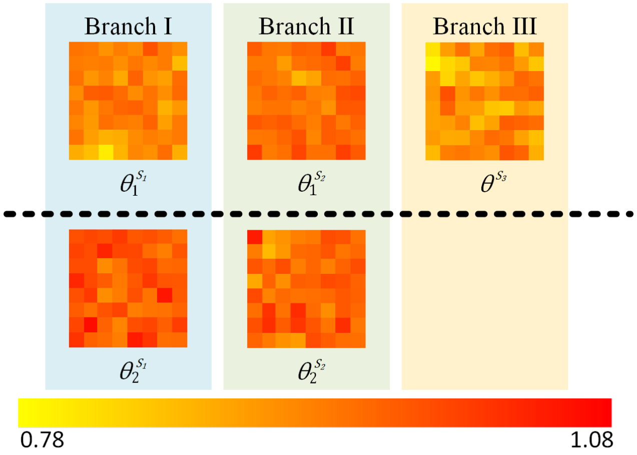


Figure 9. The learned frequency domain priors from the LBFs in different MTRBs.

4.2.3 Study of the loss function

In this subsection, we investigate the contribution from the loss functions. To demonstrate the effectiveness of the proposed ASL, we compare it with several related and well-known loss functions, including Sobel loss, Laplace loss, SSIM loss [48] and perceptual loss basing on pre-trained Vgg16 network [18]. Generally, all loss function are loaded through the multi-supervising strategy stated in Eq. 14 and finally measured by an MAE function. To balance the outputs of these losses and L1 loss, we assigned different λ (in Eq, 13) to different losses. As shown in Table 4, the structural high frequency loss provided by the Sobel loss leads to a significant improvement of 1.81dB, and the additional two directional filters from ASL further improve the performance of 0.40dB. Though 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. Besides, the SSIM loss and perceptual loss also can improve the performance. The SSIM loss behaves similar to Laplace loss, while the perceptual loss is the second best loss function which is only 0.21 dB inferior to ASL. Generally, our ASL is an simple and effective loss function for image demoireing task.

Loss	λ	PSNR (dB)	SSIM
L1	-	41.83	0.9905
L1 + Sobel	0.5	43.64	0.9945
L1 + Laplace	0.5	42.92	0.9927
L1 + SSIM	0.2	43.36	0.9946
L1 + perceptual	1.0	43.83	0.9946
L1 + ASL	0.25	44.04	0.9948

Table 4. Performance comparison of MBCNN models trained with different loss functions.

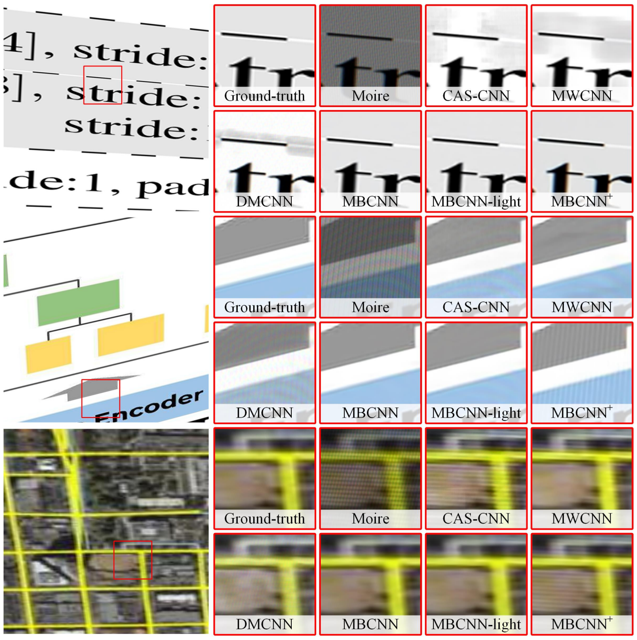


Figure 10. Demoireing results on the validation set of *LCDMoire* produced by proposed methods and other prior mehods.

4.3. Comparison with prior work

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

4.3.1 Comparison on LCDMoire dataset

We first compare with the participating methods in the AIM19 image demoireing challenge [42]. The results on the validation set (again, independent and not used in training) is shown in Table 5. Since the ground-truth of the *LCDMoire* testing set is not released, we provide the performance on the *LCDMoire* validation set. We also compared with several methods that did not participate in the challenge, including CAS-CNN [2], MWCNN [25], DMCNN [31]. The result and average running time per image are shown in Table 6. Because we have demonstrated the superiority of the ASL, we trained the methods (CAS-CNN, MWCNN, DMCNN) with L1 loss plus ASL. Limited by the global residual connection, MWCNN fails to solve the image demoireing problem, while CAS-CNN achieves a very close performance to DMCNN. The proposed MBCNN method clearly outperforms these other methods, with a significant performance gain of +7.88dB/+0.075 PSNR than CAS-CNN. From the visualized results shown in Figure 10, our MBCNN accurately removes moire texture and restores most image details.

However, since MBCNN consumes considerable parameters compared to several compared methods, we propose a light version of MBCNN (MBCNN-light) by setting $n_G =$

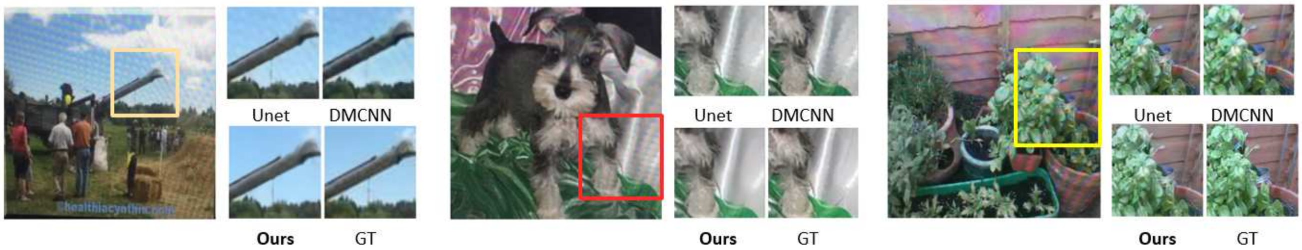


Figure 11. Qualitative comparison on *TIP2018* dataset.

Model	IPCV IITM	PCALab	IAIR	XMU-VIPLab	KU-CVIP	MoePhoto	Islab-zju	MBCNN
PSNR/SSIM	32.23/0.96	32,39.0.97	35.27/0.97	39.21/0.99	40.17/0.98	41.91/0.99	42.90/0.99	**44.04/0.9948**

Table 5. Performance comparison of MBCNN models and the top 7 participating methods in the AIM19 demoireing challenge.

Model	CAS-CNN	MWCNN	DMCNN	MBCNN	MBCNN-light	MBCNN$^+$
PSNR	36.16	28.93	35.48	44.04	42.81	33.65
SSIM	0.9873	0.9698	0.9785	0.9948	0.9940	0.9859
Time(s)	0.14	0.14	0.10	0.25	0.12	1.14

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

	DnCNN	VDSR	EDSR	UNet	DMCNN	MopNet	**MBCNN**
PSNR	24.54	24.68	26.82	26.49	26.77	27.75	**30.03**
SSIM	0.834	0.837	0.853	0.864	0.871	**0.895**	0.893

Table 7. Performance comparison of MBCNN models and other related works on *TIP2018* dataset.

64, $n_D = 32$, while keeping other settings unchanged. As shown in Table 6, the fewer parameters leads to a performance reduction of -1.46 dB/-0.028 from MBCNN. Nevertheless, MBCNN-light still outperforms other participating methods even in this reduced form of the method.

Recently, several studies have reported that the geometric self-ensemble could reasonably enhance the performance in the final testing phase. We adopted this strategy during testing time by rotating the input image by 90°, 180° and 270° to generate three augmented input images, and calculating the mean image of the original output and three augmented outputs (rotated back) as the final output. We denoted this self-ensemble MBCNN as MBCNN$^+$. Perhaps surprisingly, this strategy leads to a dramatic reduction in performance. We speculate that because the moire texture is a strongly direction-aware artifact, changing the direction would mislead the network to make an inaccurate restoration.

4.3.2 Comparison on 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 [44], VDSR [20], EDSR [23], UNet [29], DMCNN [31], MopNet [12]. As shown in Table 7, our proposed MBCNN beats the second best method by $+2.28$ dB, in terms of PSNR, and achieved the second best SSIM result which is only 0.002 lower than the best. Moreover, the visualized results shown in Figure 11 also demonstrates the proposed method outperformed other compared methods. More qualitative examples are shown in the supplementary material.

5. Conclusion

In this paper, we propose a multiscale bandpass CNN (MBCNN) for image demoireing, and significantly outperform state-of-the-art methods by more than 2dB in terms of PSNR. A learnable bandpass filter (LBF) is proposed to learn the frequency prior. Our model has two steps: moire texture removal and tone mapping. A LBF-based residual CNN block is used for moire texture removal, and another two CNN blocks for global and local tone mappings. An ablation study was conducted to show the importance of the components in the network. We have also clarified the the effect of the block-IDCT size in the LBF, and demonstrated that the block-IDCT size of 8 is the best for the image demoireing task. Experiments on two public datasets show that our model outperformed state-of-the-art methods by large margins.

References

[1] Jonathan Barron and Yun-Ta Tsia. Fast fourier color constancy. In *CVPR*, 2017. 1

[2] Lukas Cavigelli, Pascal Hager, and Luca Benini. CAS-CNN: A deep convolutional neural network for image compression artifact suppression. In *IJCNN, 2017*. 2, 7

[3] Xi Cheng, Zhenyong Fu, and Jian Yang. Multi-scale dynamic feature encoding network for image demoiréing. In *ICCVW, 2019*. 1, 2

[4] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. Compression artifacts reduction by a deep convolutional network. In *ICCV, 2015*. 2

[5] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *ECCV, 2014*. 2

[6] Lian-Feng Dong, Yuan-Zhu Gan, Xiao-Liao Mao, Yu-Bin Yang, and Chunhua Shen. Learning deep representations using convolutional auto-encoders with symmetric skip connections. In *ICASSP, 2018*. 2

[7] Gabriel Eilertsen, Joel Kronander, Gyorgy Denes, Rafał K Mantiuk, and Jonas Unger. Hdr image reconstruction from a single exposure using deep cnns. *TOG, 2017*. 2

[8] Tianyu Gao, Yanqing Guo, Xin Zheng, Qianyu Wang, and Xiangyang Luo. Moiré pattern removal with multi-scale feature enhancing network. In *ICMEW, 2019*. 1

[9] Michael Gharbi, Gaurav Chaurasia, Sylvain Paris, and Fredo Durand. Deep joint demosaicking and denoising. In *Siggraph Asia*, 2016. 1

[10] Michaël Gharbi, Jiawen Chen, Jonathan T. Barron, Samuel W. Hasinoff, and Frédo Durand. Deep bilateral learning for real-time image enhancement. *TOG, 2017*. 2

[11] Jun Guo and Hongyang Chao. Building dual-domain representations for compression artifacts reduction. In *ECCV, 2016*. 2

[12] Bin He, Ce Wang, Boxin Shi, and Ling-Yu Duan. Mop moire patterns using mopnet. In *ICCV, 2019*. 1, 2, 8

[13] Kaiming He, Jian Sun, and Xiaoou Tang. Guided image filtering. In *ECCV, 2010*. 2

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR, 2016*. 4

[15] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR, 2018*. 4

[16] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *CVPR, 2017*. 2, 4

[17] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV, 2017*. 1, 2

[18] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV, 2016*. 7

[19] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *CVPR, 2016*. 2

[20] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *CVPR, 2016*. 8

[21] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *CVPR, 2016*. 2

[22] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2014. 5

[23] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPRW, 2017*. 5, 8

[24] Bolin Liu, Xiao Shu, and Xiaolin Wu. Demoireing of camera-captured screen images using deep convolutional neural network. *arXiv, 2018*. 1

[25] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo. Multi-level wavelet-cnn for image restoration. In *CVPRW, 2018*. 2, 7

[26] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR, 2015*. 2

[27] Wenqi Ren, Lin Ma, Jiawei Zhang, Jinshan Pan, Xiaochun Cao, Wei Liu, and Ming-Hsuan Yang. Gated fusion network for single image dehazing. In *CVPR, 2018*. 2

[28] Yaniv Romano, John Isidoro, and Peyman Milanfar. Raisr: rapid and accurate image super resolution. *IEEE Transactions on Computational Imaging*, 3(1):110–125, 2016. 1

[29] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, 2015. 2, 8

[30] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV, 2015*. 2, 5

[31] Yujing Sun, Yizhou Yu, and Wenping Wang. Moire photo restoration using multiresolution convolutional neural networks. *TIP, 2018*. 1, 2, 5, 6, 7, 8

[32] Pavel Svoboda, Michal Hradis, David Bařina, and Pavel Zemcík. Compression artifacts removal using convolutional neural networks. *Journal of WSCG*, 24:63–72, 05 2016. 2

[33] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *CVPR, 2017*. 2

[34] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Mem-Net: A persistent memory network for image restoration. In *ICCV, 2017*. 2

[35] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *CVPRW, 2019*. 4

[36] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. Fast end-to-end trainable guided filter. In *CVPR, 2018*. 2

[37] C. Yan, B. Gong, Y. Wei, and Y. Gao. Deep multi-view enhancement hashing for image retrieval. *TPAMI, 2020*. 4

[38] C. Yan, B. Shao, H. Zhao, R. Ning, Y. Zhang, and F. Xu. 3d room layout estimation from a single rgb image. *TMM, 2020*. 4

[39] Chenggang Yan, Yunbin Tu, Xingzheng Wang, Yongbing Zhang, Xinhong Hao, Yongdong Zhang, and Qionghai Dai. Stat: spatial-temporal attention mechanism for video captioning. *TMM, 2019*. 4

[40] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 4

[41] Shanxin Yuan, Radu Timofte, Gregory Slabaugh, and Ales Leonardis. Aim 2019 challenge on image demoreing: dataset and study. In *ICCVW, 2019*. 5

[42] Shanxin Yuan, Radu Timofte, Gregory Slabaugh, Ales Leonardis, and etc. Aim 2019 challenge on image demoreing: methods and results. In *ICCVW, 2019*. 7

[43] He Zhang and Vishal M Patel. Densely connected pyramid dehazing network. In *CVPR, 2018*. 2

[44] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *TIP, 2017*. 1, 8

[45] K. Zhang, W. Zuo, S. Gu, and L. Zhang. Learning deep cnn denoiser prior for image restoration. In *CVPR, 2017*. 2

[46] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV, 2018*. 4

[47] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *CVPR, 2018*. 2, 5

[48] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *TCI, 2016*. 5, 7

[49] Bolun Zheng, Yaowu Chen, Xiang Tian, Fan Zhou, and Xuesong Liu. Implicit dual-domain convolutional network for robust color image compression artifact reduction. *TCSVT, 2019*. 2, 3