

ShapeMoiré: Channel-Wise Shape-Guided Network for Image Demoiréing

JINMING CAO and SICHENG SHEN, National University of Singapore, Singapore, Singapore
QIU ZHOU, Beijing University of Posts and Telecommunications, Beijing, China
YIFANG YIN, Institute for Infocomm Research (I2R), A*STAR, Singapore, Singapore
YANGYAN LI, Alibaba Group, Hangzhou, China
ROGER ZIMMERMANN, National University of Singapore, Singapore, Singapore

Photographing optoelectronic displays often introduces unwanted moiré patterns due to analog signal interference between the pixel grids of the display and the camera sensor arrays. This work identifies two problems that are largely ignored by existing image demoiréing approaches: (1) moiré patterns vary across different channels (RGB); (2) repetitive patterns are constantly observed. However, employing conventional convolutional (CNN) layers cannot address these problems. Instead, this article presents the use of our recently proposed *Shape* concept. It was originally employed to model consistent features from fragmented regions, particularly when identical or similar objects coexist in an RGB-D image. Interestingly, we find that the *Shape* information effectively captures the moiré patterns in artifact images. Motivated by this discovery, we propose a new method, ShapeMoiré, for image demoiréing. Beyond modeling shape features at the patch level, we further extend this to the global image level and design a novel Shape-Architecture. Consequently, our proposed method, equipped with both ShapeConv and Shape-Architecture, can be seamlessly integrated into existing approaches without introducing any additional parameters or computation overhead during inference. We conduct extensive experiments on four widely used datasets, and the results demonstrate that our ShapeMoiré achieves state-of-the-art performance, particularly in terms of the PSNR metric. We then apply our method across four popular architectures to showcase its generalization capabilities. Moreover, to further validate its generality beyond the demoiréing task, we apply ShapeMoiré to the image deblurring task, where it continues to deliver consistent performance gains. Finally, experiments on real-world images captured by smartphones confirm the robustness and practical applicability of ShapeMoiré in challenging demoiréing scenarios. We open sourced an implementation of ShapeMoiré in PyTorch at <https://github.com/SichengS/ShapeMoiré>.

CCS Concepts: • **Computing methodologies** → **Machine learning approaches**;

Additional Key Words and Phrases: Moiré Pattern, Image Demoiréing, Image Restoration, Shape Information

This work is fully supported by the Advanced Research and Technology Innovation Centre (ARTIC), the National University of Singapore under Grant (project number: ELDT-RP2).

Authors' Contact Information: Jinming Cao (corresponding author), National University of Singapore, Singapore, Singapore; e-mail: jinming.cao@gmail.com; Sicheng Shen, National University of Singapore, Singapore, Singapore; e-mail: sshen@u.nus.edu; Qiu Zhou, Beijing University of Posts and Telecommunications, Beijing, China; e-mail: zhouqiulv@gmail.com; Yifang Yin, Institute for Infocomm Research (I2R), A*STAR, Singapore, Singapore; e-mail: yin_yifang@i2r.a-star.edu.sg; Yangyan Li, Alibaba Group, Hangzhou, China; e-mail: yangyan.lee@gmail.com; Roger Zimmermann, National University of Singapore, Singapore, Singapore; e-mail: rogerz@comp.nus.edu.sg.



This work is licensed under Creative Commons Attribution International 4.0.

© 2025 Copyright held by the owner/author(s).

ACM 1551-6865/2025/11-ART348

<https://doi.org/10.1145/3748657>

ACM Reference format:

Jinming Cao, Sicheng Shen, Qiu Zhou, Yifang Yin, Yangyan Li, and Roger Zimmermann. 2025. ShapeMoiré: Channel-Wise Shape-Guided Network for Image Demoiréing. *ACM Trans. Multimedia Comput. Commun. Appl.* 21, 12, Article 348 (November 2025), 20 pages.
<https://doi.org/10.1145/3748657>

1 Introduction

Moiré artifacts manifest as wavy or rippled distortions in digital photographs, arising from the interference between a camera’s color filter array and the sub-pixel structure of electronic displays. These unwanted patterns significantly degrade visual quality, posing challenges in multimedia applications such as mobile photography [22] and cross-device content reproduction [21]. Figure 1 illustrates a digital photo with such moiré patterns ($I_{\text{moiré}}$), which make the original image (I) appear opaque and distorted when displayed on a screen. Notably, the irregular shape and broad frequency ranges of moiré patterns distinguish demoiréing from other image restoration tasks such as **super-resolution (SR)** [29, 46] and reducing JPEG compression artifacts [6].

Moiré pattern removal has attracted extensive research interests over the past few years [11, 12, 34, 35, 37]. Initially, efforts were devoted to collecting synthetic data [36] or genuine images with low resolution [24]. Subsequent datasets expanded the image resolution to 1080P [12] or even ultra-high-definition (4K) [35], posing greater challenges in removing moirés without compromising the pristine features of the images. To address this, prior methods have mostly relied on rich side information, such as complementary visual attributes [11] and aliasing frequencies [20, 47]. Despite the advancements in existing methods, they often treat the input image from a holistic perspective, especially without discriminating between different channels (i.e., RGB channels). However, as observed in Figure 1, unlike the uniform and smooth color distribution in each channel (R , G , and B) of the original image I , these three input channels ($R_{\text{moiré}}$, $G_{\text{moiré}}$, and $B_{\text{moiré}}$) typically exhibit distinct moiré patterns. As a result, processing them equally with convolutional layers confuses the model’s focus on demoiréing, resulting in sub-optimal outcomes. Furthermore, existing literature often overlooks the repetitive patterns that are consistently observed.

Our idea in this work is inspired by a recent prevailing concept—*Shape*—in RGB-D segmentation [2]. Unlike traditional CNNs, ShapeConv [2] can distinguish between different channels and is adept at modeling repetitive shapes, such as two identical chairs in a depth image from RGB-D segmentation. In the context of moiré patterns in an image, as demonstrated in Figure 1, the moiré image ($I_{\text{moiré}}$) exhibits frequent similar or repetitive patterns, unlike the uniform color distribution in the original image (I). This type of information is precisely what *shape* excels at expressing. Additionally, the features of moiré patterns in the *shape* space ($R_{\text{moiré}}^{\text{Shape}}$, $G_{\text{moiré}}^{\text{Shape}}$, and $B_{\text{moiré}}^{\text{Shape}}$) remain consistent across different channels of the original moiré image ($R_{\text{moiré}}$, $G_{\text{moiré}}$ and $B_{\text{moiré}}$), but exhibit enhanced structural integrity. This observation motivates us to approach the image demoiré problem from the *shape* perspective.

Therefore, we propose a novel method named ShapeMoiré to extend our ShapeConv for image demoiréing. Our ShapeMoiré operates on both the patch¹-level and image-level shapes. This results in two key components of our proposed ShapeMoiré: the **shape-aware convolutional layer (ShapeConv)** and the **shape-aware network architecture (Shape-Architecture)**. Specifically, at the patch level, we first decompose a patch into two separate components, i.e., a *base component* and a *shape component*. We then employ two operations, namely, base product and shape product, to process these components separately, with two learnable weights: the base kernel and the shape

¹The operational unit of input features for the convolutional layer, whose spatial size matches that of the convolution kernel.

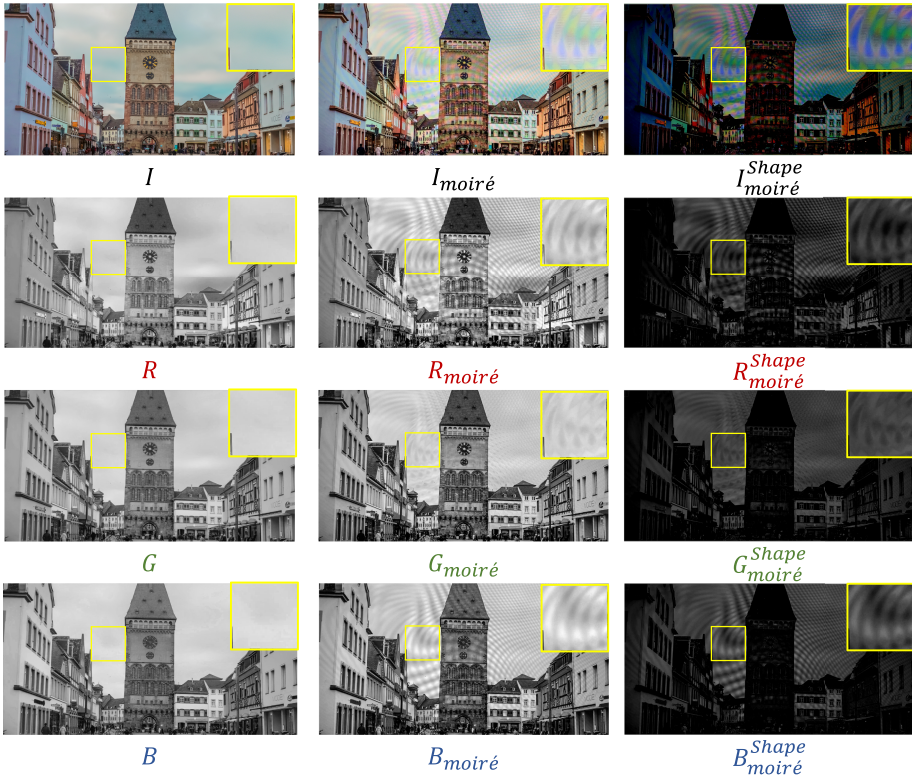


Fig. 1. Visual demonstration of the moiré pattern in the given image and its separate manifestations in the individual RGB channels. Left column: the original image I , where the color in each channel is uniformly and smoothly distributed; Middle column: it can be observed that the three channels of $I_{\text{moiré}}$ display distinct moiré patterns; Right column: the shapes of moiré patterns in the *shape* space ($R_{\text{moiré}}^{\text{Shape}}$, $G_{\text{moiré}}^{\text{Shape}}$ and $B_{\text{moiré}}^{\text{Shape}}$) remain consistent with the patterns in the different channels of the original moiré image ($R_{\text{moiré}}$, $G_{\text{moiré}}$ and $B_{\text{moiré}}$), but with enhanced structural integrity.

kernel. The outputs from these operations are then combined in an additive manner to form a shape-aware patch, which is subsequently convolved with a standard convolutional kernel. Unlike the original patch, the shape-aware patch is capable of adaptively learning the shape characteristics with the shape kernel, while the base kernel serves to balance the contributions of the *shape* and the *base* components in the final feature representation. At the extended image level, we incorporate an additional training stream to enhance the model's focus on shape information during training. This is achieved by adding a loss function for the shape branch, thereby updating the parameters with an emphasis on shape information. In a nutshell, the former replaces the conventional convolutional layer with a delicately designed ShapeConv that involves the modeling of shape features, and the latter employs the shape information of raw pixels to further enhance the baseline architecture. It is worth noting that our ShapeMoiré can be seamlessly integrated into most baseline architectures that are equipped with convolutional layers as building blocks. In addition, benefiting from its original virtue, our ShapeMoiré results in no incremental parameters and only negligible computational cost.

To evaluate the effectiveness of our proposed method, we conducted extensive experiments on four well-established datasets: LCDMoiré [36], TIP2018 [24], FHDmi [12], and **ultra-high-**

definition demoiréing (UHDM) [35]. Our experimental results demonstrate that ShapeMoiré consistently outperforms existing state-of-the-art methods by a significant margin. To assess its generalization capabilities, we applied ShapeMoiré to various widely adopted demoiréing architectures [20, 24, 35] and observed improved results compared to baselines. Furthermore, to validate the method’s broader applicability beyond demoiréing, we extended our experiments to an additional image restoration task—image deblurring. Results on the DPDD dataset [1] confirm that ShapeMoiré remains effective in this new context, indicating good cross-task generalization. Notably, qualitative results highlight that ShapeMoiré remarkably enhances moiré pattern removal, even when applied to real-world images captured using a smartphone. This underscores both the effectiveness on public datasets and the practical applicability of our ShapeMoiré in real-world scenarios.

The work on ShapeConv has previously been published at ICCV [2]. This extended manuscript expands upon the original version in several key aspects:

- We broaden the application of the notion of *Shape* beyond its original use in RGB-D segmentation. To the best of our knowledge, this is the first time the concept has been effectively applied to both the moiré pattern removal and image deblurring tasks in 2D image restoration.
- Beyond employing patch-level ShapeConv, we introduce an image-level Shape-Architecture for image demoiréing. Through comprehensive ablation studies and cross-task experiments, we demonstrate that integrating *Shape* information at multiple levels consistently enhances performance on diverse image restoration tasks.
- The proposed ShapeMoiré method, which incorporates both ShapeConv and Shape-Architecture, represents a versatile approach. We conduct experiments using various existing methods as baselines, all of which effectively enhance baseline performance. Notably, both ShapeConv and Shape-Architecture introduce ZERO additional parameters and incur only negligible computational overhead during inference.

2 Related Work

2.1 Image Demoiréing

Removing moiré patterns constitutes a form of image denoising aimed at mitigating the destructive artifacts arising from frequency interference during image capture. Unlike other types of noise, such as flickering² induced by photography, there is currently no standardized tool for effectively eliminating moiré patterns. Recently, numerous benchmark datasets and approaches have been introduced to address the challenge of image demoiréing.

Datasets. Liu et al. [19] initiated the task by curating a synthetic dataset that simulates the process of photographing with a camera. Building upon this approach, Yuan et al. [36] developed a large-scale synthetic dataset, namely, LCDMoiré, for further image demoiréing [4, 36, 47]. However, the transition from synthetic training to real-world scenarios presents a significant obstacle due to the inherent gap between simulation and reality. Sun et al. [24] addressed this challenge by introducing the TIP2018 dataset, which utilizes real-world moiré images. Following this approach, Yue et al. [37] compiled a well-aligned dataset of raw moiré image. Nevertheless, these images often have low resolutions, making it difficult for the accompanying methods to adapt to high-resolution images. To tackle this issue, He et al. [12] curated the FHDMi dataset with a resolution of 1080P. However, as the demand for image quality continues to increase, the practicality of 1080P resolution may diminish, especially when considering the ultra-high-definition (4K) images captured by modern

²<https://digitalanarchy.com/Flicker/main.html>.

mobile cameras. Recognizing these challenges and limitations, Yu et al. [35] collected the first UHDM dataset, which contains 5,000 real-world 4K resolution images.

Methods. Pertaining to moiré pattern removal methods, current approaches often endeavor to design specific architectures for this task. For example, He et al. [11] proposed to classify moiré patterns using manually annotated category labels. Some more approaches [20, 47] have focused on the frequency domain via de-aliasing. In order to remove moiré without compromising the clarity of underlying image details, Sun et al. [24] implemented a multi-scale network, DMCNN, effectively addressing moiré patterns in practical scenarios. Building upon DMCNN, He et al. [12] developed the multi-stage framework FHD²Net to handle a wider pattern scale range and preserve fine details. For 4K images, ESDNet [35] implements an efficient module that is semantic-aligned and scale-aware to address the scale variation of moiré patterns.

Unlike previous methods, which treat the entire image as a whole without distinguishing between different channels, Yang et al. [34] also observed differences in moiré patterns across different RGB channels. Specifically, they found that the red (R) and blue (B) channels are more heavily affected by moiré artifacts compared to the green (G) channel. Therefore, they proposed to remove moiré artifacts in the R and B channels using guided filtering based on the texture layer obtained from the G channel. In contrast to the method in [34], our approach does not involve special treatment of a specific channel but instead presents a universal method that discriminates between different channels. Moreover, [34] is a specialized design, while our proposed ShapeMoiré stands out as a versatile approach. It can seamlessly integrate with existing methods, enhancing model performance with ZERO increase of memory and computation.

2.2 Image Restoration

Image restoration aims to reconstruct the original, clean, and high-quality image from degraded versions, serving as a core task in computer vision and multimedia computing. This task encompasses various processes, including image SR [15], image denoising [3, 31, 33], reducing JPEG compression artifacts, and so on. A plethora of learning-based image restoration models [17, 39, 39] have gained popularity due to their impressive performance. These models typically leverage large-scale paired datasets to learn mappings between low-quality and high-quality images, employing encoder-decoder structures [23] or hierarchical architectures [24, 41]. Building upon pioneering works such as SRCNN [7] (for SR), DnCNN [42] (for denoising), and ARCNN [6] (for reducing JPEG artifacts), numerous CNN-based models [9, 16, 26, 43] have emerged. These models enhance representational capacity by incorporating larger and deeper neural architectures, such as residual blocks [13], dense blocks [14], and others [16]. In recent years, transformer-based and attention-driven frameworks have further advanced the field. Restormer [38] proposes a multi-Dconv head transposed attention mechanism tailored for image restoration, achieving state-of-the-art results on multiple benchmarks. Uformer [28] unifies transformer and U-Net architectures to improve global context modeling while maintaining efficiency. Guo et al. [8] introduce a simple but effective baseline, named MambaIR, which introduces both local enhancement and channel attention, yet surprisingly achieves competitive performance on several restoration tasks. Lin et al. [18] map the degraded images into textual representations for removing the degradations and then convert the restored textual representations into a guidance image for assisting image restoration.

However, while these methods target general image restoration tasks—such as image SR, image denoising, and JPEG compression artifact reduction—the denoising types they address are mostly spatial-domain noises like Gaussian color image denoising [8, 38], real image denoising [8], deraining [28], and deblurring [18]. Although moiré patterns are also a type of noise, they are frequency-domain artifacts with unique structural characteristics and thus have not been effectively tackled by these existing methods.

2.3 Shape Concept and Its Applications

The concept of *Shape* was initially introduced in the domain of RGB-D segmentation in a previously published paper [2], where it is defined as the disparity between local depth values and the overall distance of the region from the observation point. One key observation from this article is that, in spaces where identical or similar objects coexist, *Shape* can extract more consistent features compared to raw pixel values. To leverage this insight, we designed ShapeConv to replace traditional convolution layers, allowing for the processing of data from various modalities while emphasizing *Shape* attributes. Subsequent studies [30, 48] extended the application to remote sensing data, showcasing *Shape*'s superiority in feature extraction across RGB+nDSM (primarily representing height features) data modalities. Specifically, Xiong et al. [30] utilized ShapeConv for semantic segmentation tasks on the their collected GAMUS dataset, which features numerous repetitive buildings. Zhong et al. [48] created an ocean coral dataset characterized by repetitive texture patterns within the corals. By integrating ShapeConv as a fundamental component of their model, they achieved superior performance. Beyond the RGB+X data, [49] employs ShapeConv for 3D object detection and occupancy.

This article represents the first application of the *Shape* notion to 2D tasks. In particular, we observe that the moiré patterns tend to be repetitive across channel, aligning with the features that *Shape* learns. Additionally, we propose an application of *Shape* at the global image level, namely the Shape-Architecture, to complement the patch-level ShapeConv. Our experimental results demonstrate a distinct advantage of *Shape* when the input data changes from 3D to 2D. This showcases the versatility of the *Shape* concept and holds promise for broader applications.

3 Method

In this section, we first provide the basic formulation of the ShapeConv, followed by its application in the training and inference phase. Subsequently, we delve into the comprehensive framework of the Shape-Architecture, elucidating its dataflow dynamics during both training and inference stages. Finally, we present the ShapeMoiré method, equipped with ShapeConv and Shape-Architecture.

3.1 ShapeConv

Method Intuition. Given an input image with a moiré pattern, $I_{\text{moiré}}$, image demoiréing aims to remove these unwanted patterns and thereby restore the original pristine image. Existing methods [12, 20, 24, 35] often involve an encoder-decoder framework, wherein the conventional convolutional layer serves as the core building block. A convolutional layer takes as input a feature map, either raw pixels or intermediate image features, in a sliding window manner. It typically comprises multiple convolution kernels, which are applied to a patch of images at each window position. To facilitate our discussion, we conceptualize one image patch as a 3D tensor $\mathbb{P} \in R^{K_h \times K_w \times C_{in}}$, K_h , and K_w are the spatial dimensions of the kernel and C_{in} represents the channel numbers in the input feature map. The output features from the vanilla convolution layer are obtained by,

$$\mathbb{F} = \text{Conv}(\mathbb{K}, \mathbb{P}), \quad (1)$$

where $\mathbb{K} \in R^{K_h \times K_w \times C_{in} \times C_{out}}$ denotes the learnable weights of kernels in a convolutional layer (the bias terms are not included for simplicity) and C_{out} represents the number of channels in the output feature map. Each element of $\mathbb{F} \in R^{C_{out}}$ is calculated as,

$$\begin{cases} \mathbb{F} = \mathbb{K} \circ \mathbb{P} \\ \mathbb{F}_{C_{out}} = \sum_i^{K_h \times K_w \times C_{in}} (\mathbb{K}_{i, C_{out}} \times \mathbb{P}_i), \end{cases} \quad (2)$$

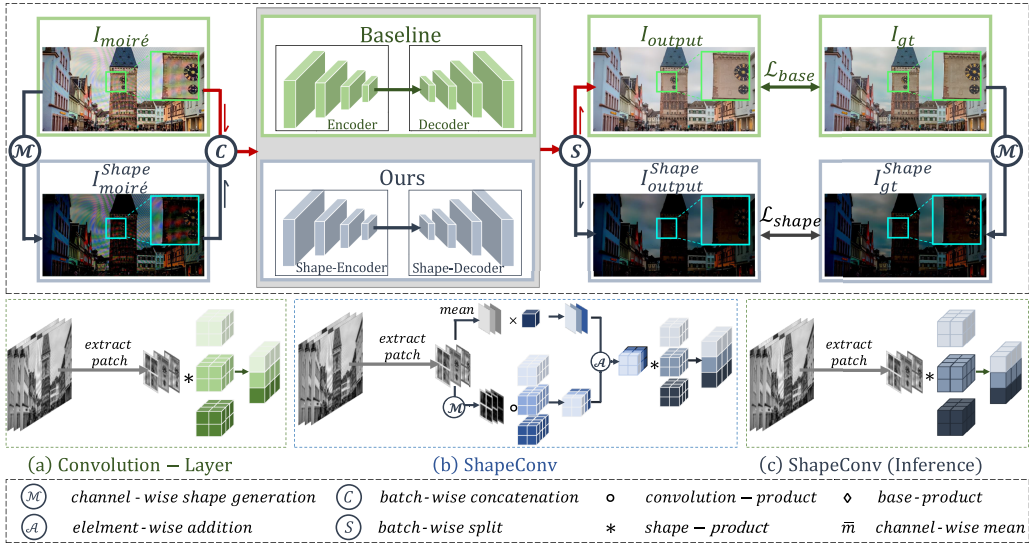


Fig. 2. Overview of ShapeMoiré. The top part depicts the network’s overall architecture. The green and blue parts denote the baseline and ShapeMoiré components, respectively, and the red line flow represents ShapeMoiré during inference. The bottom part shows network layers: (a) convolution layer (consistent in both training and testing), and (b) and (c) illustrate ShapeConv during training and inference, respectively. For both ShapeConv and Shape-Architecture, our method introduces no parameters and negligible computational cost during inference.

where the $\circ$ denotes the convolution-product. We show one example in Figure 2(a) (best view in color), where $K_h = K_w = 2$, $C_{in} = 3$ (featuring distinct input channels represented by various square frame colors), and $C_{out} = 3$.

ShapeConv Formulation. Based on the aforementioned analysis, in this article, we offer to decompose an input patch into two components: a base component $\mathbb{P}_B$ and a shape component $\mathbb{P}_S$. Specifically, we refer to the mean of patch values to be $\mathbb{P}_B$, and its relative values to be as $\mathbb{P}_S$:

$$\begin{aligned}\mathbb{P}_B &= \bar{m}(\mathbb{P}), \\ \mathbb{P}_S &= \mathcal{M}(\mathbb{P}) = \mathbb{P} - \bar{m}(\mathbb{P}),\end{aligned}\tag{3}$$

where $\bar{m}(\mathbb{P})$ is the channel-wise mean function on $\mathbb{P}$ (over the $K_h \times K_w$ dimensions), with $\mathbb{P}_B \in \mathbb{R}^{1 \times 1 \times C_{in}}$, and $\mathbb{P}_S \in \mathbb{R}^{K_h \times K_w \times C_{in}}$.

Note that directly convolved $\mathbb{P}_S$ with $\mathbb{K}$ in Equation (1) is sub-optimal, as the values from $\mathbb{P}_B$ contribute significantly to key discrimination across patches. Therefore, our ShapeConv approach utilizes two learnable weights, $\mathbb{W}_B \in \mathbb{R}^1$ and $\mathbb{W}_S \in \mathbb{R}^{K_h \times K_w \times K_h \times K_w \times C_{in}}$, to separately process these two components. The resulting features are then combined through element-wise addition, creating a new channel-wise shape-aware patch with the same size as the original $\mathbb{P}$. The formulation of ShapeConv is given as,

$$\begin{aligned}\mathbb{F} &= \text{ShapeConv}(\mathbb{K}, \mathbb{W}_B, \mathbb{W}_S, \mathbb{P}) \\ &= \text{Conv}(\mathbb{K}, \mathbb{W}_B \diamond \mathbb{P}_B + \mathbb{W}_S * \mathbb{P}_S) \\ &= \text{Conv}(\mathbb{K}, \mathbb{P}_B + \mathbb{P}_S) \\ &= \text{Conv}((\mathbb{K}, \mathbb{P}_{BS}),\end{aligned}\tag{4}$$

where $\diamond$ and $*$ denote the base-product and shape-product operator, respectively, which are defined as,

$$\begin{cases} \mathbf{P}_B = \mathbb{W}_B \diamond \mathbb{P}_B \\ \mathbf{P}_{B_{1,1,c_{in}}} = \mathbb{W}_B \times \mathbb{P}_{B_{1,1,c_{in}}}, \end{cases} \quad (5)$$

$$\begin{cases} \mathbf{P}_S = \mathbb{W}_S * \mathbb{P}_S \\ \mathbf{P}_{S_{k_h,k_w,c_{in}}} = \sum_i^{K_h \times K_w} (\mathbb{W}_{S_{i,k_h,k_w,c_{in}}} \times \mathbb{P}_{S_{i,c_{in}}}), \end{cases} \quad (6)$$

where c_{in} , k_h , and k_w are the indices of the elements in C_{in} , K_h , and K_w dimensions, respectively.

We reconstruct the channel-wise shape-aware patch $\mathbf{P}_{BS}$ through the addition of $\mathbf{P}_B$ and $\mathbf{P}_S$, where $\mathbf{P}_{BS} \in R^{K_h \times K_w \times C_{in}}$, enabling it to be smoothly convolved by the kernel $\mathbb{K}$ of a vanilla convolutional layer. However, $\mathbf{P}_{BS}$ is equipped with the important shape information which is learned by two additional weights. The operation of the ShapeConv is depicted at the bottom of Figure 2(b).

3.2 ShapeConv in Training and Inference

Training Phase. The proposed ShapeConv can effectively leverage the *shape* information of patches. However, replacing the vanilla convolutional layer with ShapeConv in CNNs introduces more computational cost due to the two *product* operations in Equations (5) and (6). To tackle this problem, we propose shifting these two operations from patches to kernels,

$$\begin{cases} \mathbf{K}_B = \mathbb{W}_B \diamond \mathbb{K}_B \\ \mathbf{K}_{B_{1,1,c_{in},c_{out}}} = \mathbb{W}_B \times \mathbb{K}_{B_{1,1,c_{in},c_{out}}}, \end{cases}$$

$$\begin{cases} \mathbf{K}_S = \mathbb{W}_S * \mathbb{K}_S \\ \mathbf{K}_{S_{k_h,k_w,c_{in},c_{out}}} = \sum_i^{K_h \times K_w} (\mathbb{W}_{S_{i,k_h,k_w,c_{in}}} \times \mathbb{K}_{S_{i,c_{in},c_{out}}}), \end{cases}$$

where $\mathbb{K}_B \in R^{1 \times 1 \times C_{in} \times C_{out}}$ and $\mathbb{K}_S \in R^{K_h \times K_w \times C_{in} \times C_{out}}$ denote the base component of kernels and shape component, respectively, and $\mathbb{K} = \mathbb{K}_B + \mathbb{K}_S$.

We therefore re-formalize ShapeConv in Equation (4) to:

$$\begin{aligned} \mathbf{F} &= \text{ShapeConv}(\mathbb{K}, \mathbb{W}_B, \mathbb{W}_S, \mathbf{P}) \\ &= \text{Conv}(\mathbb{W}_B \diamond \overline{m}(\mathbb{K}) + \mathbb{W}_S * \mathcal{M}(\mathbb{K}), \mathbf{P}) \\ &= \text{Conv}(\mathbb{W}_B \diamond \overline{m}(\mathbb{K}) + \mathbb{W}_S * (\mathbb{K} - \overline{m}(\mathbb{K})), \mathbf{P}) \\ &= \text{Conv}(\mathbb{W}_B \diamond \mathbf{K}_B + \mathbb{W}_S * \mathbf{K}_S, \mathbf{P}) \\ &= \text{Conv}(\mathbf{K}_B + \mathbf{K}_S, \mathbf{P}) \\ &= \text{Conv}(\mathbf{K}_{BS}, \mathbf{P}), \end{aligned} \quad (7)$$

where $\overline{m}(\mathbb{K})$ is the channel-wise mean function on $\mathbb{K}$ (over the $K_h \times K_w$ dimensions). And we require $\mathbf{K}_{BS} = \mathbf{K}_B + \mathbf{K}_S$, $\mathbf{K}_{BS} \in R^{K_h \times K_w \times C_{in} \times C_{out}}$.

In fact, the two formulations of Shpe-Layer, i.e., Equations (4) and (7) are mathematically equivalent, i.e.,

$$\begin{aligned} \mathbf{F} &= \text{ShapeConv}(\mathbb{K}, \mathbb{W}_B, \mathbb{W}_S, \mathbf{P}) \\ &= \text{Conv}(\mathbb{K}, \mathbf{P}_{BS}) \\ &= \text{Conv}(\mathbf{K}_{BS}, \mathbf{P}), \end{aligned} \quad (8)$$

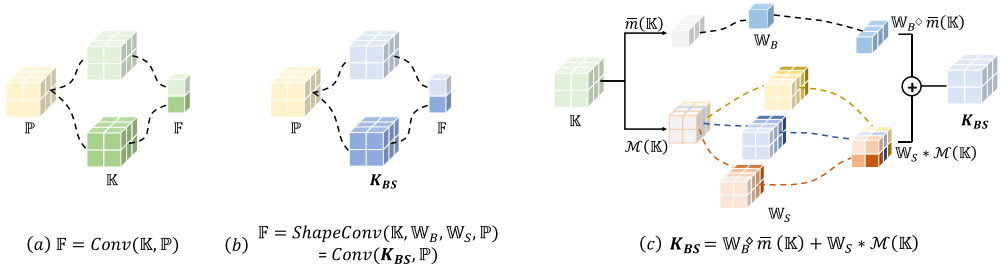


Fig. 3. Comparison of vanilla convolution layer and ShapeConv within a patch P . In this figure, $K_h = K_w = 2$, $C_{in} = 3$, and $C_{out} = 2$, “+” denotes element-wise addition. (a) Vanilla convolution with kernel K ; (b) ShapeConv with folding the W_B and W_S into K_{BS} ; (c) The computation of K_{BS} from K , W_B and W_S .

$$\begin{aligned}
 F_{C_{out}} &= \sum_i^{K_h \times K_w \times C_{in}} (K_{i, C_{out}} \times P_{BS_i}) \\
 &= \sum_i^{K_h \times K_w \times C_{in}} (K_{BS_{i, C_{out}}} \times P_i),
 \end{aligned} \tag{9}$$

please refer to the Appendix for detailed proof. In this way, we utilize the ShapeConv in Equation (7) in our implementation as illustrated in Figure 3(b) and (c).

Inference Phase. During inference, since the two additional weights i.e., W_B and W_S , become constants, we can fuse them into K_{BS} as shown in Figure 3(c) with $K_{BS} = W_B \diamond K_B + W_S * K_S$. And K_{BS} shares the same tensor size with K in Equation (1); thus, our ShapeConv is actually the same as the vanilla convolutional layer in Figure 3(a). In this way, this operation introduces negligible computational and memory costs compared to the original convolution layer (see Figure 2(a) and (c) for a visual comparison).

3.3 Shape-Architecture

Method Intuition. As depicted in Figure 2 (top), the baseline network architecture (shown in green boxes) takes the image with moiré patterns, denoted as $I_{moiré}$, as input and produces the restoration image, denoted as I_{output} . Subsequently, the $\mathcal{L}_{base}$ is computed by comparing I_{output} with the ground truth, represented as I_{gt} . The network parameters can then be updated using typical gradient descent algorithms. Current prevailing methods commonly utilize pixel-wise loss functions, such as L_1 loss [25] and L_2 loss [24], or a combination with the feature-based perceptual loss L_p [10, 11, 20, 35]. In this context, we take the combination loss as an example,

$$\mathcal{L}_{base} = L_1(I_{output}, I_{gt}) + \lambda L_p(I_{output}, I_{gt}), \tag{10}$$

where λ is a hyperparameter that balances between L_1 and L_p .

Shape-Architecture. The above ShapeConv primarily operates on intermediate features. To further explore the utility of *Shape* features, we leverage a Shape-Architecture to extract these features from the image level. The Shape-Architecture complements the baseline network architecture with an additional *Shape* stream, as illustrated in the blue box and solid dataflow in Figure 2. To implement this, we first acquire the shape feature $I_{moiré}^{Shape}$ from the input image. Thereafter, we input $I_{moiré}^{Shape}$ to the same model and employ the ground-truth shape features, I_{gt}^{Shape} which are extracted from I_{gt} ,

for supervision,

$$\begin{aligned} I_{\text{moiré}}^{\text{Shape}} &= \mathcal{M}(I_{\text{moiré}}), \\ I_{\text{gt}}^{\text{Shape}} &= \mathcal{M}(I_{\text{gt}}), \end{aligned} \quad (11)$$

where $\mathcal{M}$ aligns with the definition in ShapeConv, i.e., the subtraction from the channel-wise mean. Subsequently, $I_{\text{moiré}}^{\text{Shape}}$ is batch-wisely concatenated with $I_{\text{moiré}}$ and fed into the model, ensuring that these two streams remain uninterfered with each other. The model output is then split into I_{output} and $I_{\text{output}}^{\text{Shape}}$. Pertaining to the shape stream, the loss function $\mathcal{L}_{\text{shape}}$ is obtained by,

$$\mathcal{L}_{\text{shape}} = L_1(I_{\text{output}}^{\text{Shape}}, I_{\text{gt}}^{\text{Shape}}) + \lambda L_p(I_{\text{output}}^{\text{Shape}}, I_{\text{gt}}^{\text{Shape}}). \quad (12)$$

The final loss function $\mathcal{L}_{\text{total}}$ is:

$$\mathcal{L}_{\text{total}} = \alpha \mathcal{L}_{\text{base}} + \beta \mathcal{L}_{\text{shape}}, \quad (13)$$

where α and β are the weighting coefficients for the loss terms $\mathcal{L}_{\text{base}}$ and $\mathcal{L}_{\text{shape}}$, respectively.

Shape-Architecture during Inference. Note that during training, the two streams share the same parameters. As a result, no parameters are introduced by our Shape-Architecture. We leverage $\mathcal{L}_{\text{shape}}$ to make the parameters more aware of the shape features. Regarding the inference stage, in order to align with the structure of the baseline network architecture, we efficiently remove the shape stream from our Shape-Architecture, as illustrated in the red line flow in Figure 2. Moreover, we employ this strategy to allow for seamless integration with baselines and avoid introducing any computational overheads.

Relationship between ShapeConv and Shape-Architecture. While ShapeConv and Shape-Architecture operate at different levels—patch level and image level, respectively—they are intrinsically aligned through the shared notion of shape decomposition. Specifically, both components rely on the same shape operator $\mathcal{M}$ to extract the relative information from input data by subtracting the channel-wise mean. ShapeConv uses this operator within local patches to enhance feature discrimination, whereas Shape-Architecture applies it globally to form a parallel shape stream at the image level. This design ensures consistency in how shape information is perceived and utilized throughout the network. Together, they form a hierarchical shape-aware framework, where ShapeConv captures local fine-grained shape variations and Shape-Architecture reinforces the model's global structural awareness. This joint modeling helps the network better suppress moiré patterns that exhibit both local distortions and large-scale repetitive interference.

3.4 ShapeMoiré

We term the approach equipped with both ShapeConv and Shape-Architecture as ShapeMoiré. This means our ShapeMore operates on two hierarchical levels: (1) *Patch Level*: we replace all the convolutional layers with our ShapeConv; (2) *Image Level*: we complement the baseline network architecture with another shape-aware image stream. It is noteworthy that the two components, ShapeConv and Shape-Architecture, are independent of each other and can be separately added to baseline methods to enhance performance. As mentioned earlier, neither of these components introduces additional parameters or computational overhead during inference. Therefore, our approaches introduce ZERO parameters and incur negligible computation compared to the baseline method during inference. We will later demonstrate the complementary effect of these two components in our experiments.

4 Experiments

Datasets. Our experiments were conducted using four publicly accessible datasets:

- (1) *LCDMoiré Dataset* [36]. As a part of the AIM19 image demoiréing challenge, the LCDMoiré dataset comprises 10,200 synthetically generated image pairs. It is structured to include 10,000 images for training purposes, 100 images for validation, and another 100 for testing. Each image is of high resolution, with dimensions of $1,024 \times 1,024$ pixels.
- (2) *TIP2018 Dataset* [24]. This dataset consists of real photographs obtained by capturing images from ImageNet [5] displayed on computer screens, ensuring each image has a resolution of 256×256 pixels.
- (3) *FHDMi Dataset* [12]. It includes a substantial collection of 9,981 training image pairs and 2,019 testing pairs. All images in this dataset are in full high-definition resolution, measuring $1,920 \times 1,080$ pixels.
- (4) *UHDM Dataset* [35]. It stands out for its 4K resolution demoiréing capability. It is an ultra-high-resolution dataset featuring 5,000 image pairs that span a broad spectrum of scenes, including landscapes, sports, video clips, and documents. The dataset is notable for its diverse moiré patterns, produced using various device combinations and viewpoints. Each image in this dataset is $3,840 \times 2,160$ pixels.

These datasets, ranging from synthetically generated pairs to ultra-high-resolution real photographs, offer a comprehensive platform for evaluating the effectiveness of our methods across different scenarios and image qualities.

Evaluation Protocols. For quantitative evaluation, we employed widely recognized metrics, including **peak signal-to-noise ratio (PSNR)**, **structure similarity (SSIM)** [27], and **learned perceptual image patch similarity (LPIPS)** [44]. In addition to these performance-related metrics, we also considered the network's parameter count, denoted as Param.(M), which indicates the number of network parameters in millions. This metric is pertinent to the overall memory usage of the network. It is important to note that previous methods [4, 11, 20, 24, 35] have exclusively reported PSNR and SSIM metrics on the TIP2018 and LCDMoiré datasets. In line with this convention, we adhere to the same setup for our comparisons.

Implementation Details. We adopted several popular approaches [20, 24, 35] as our baseline to demonstrate both the effectiveness and generalization capability of our proposed method. We strictly followed the original implementations of these baseline methods, except for the model components of ShapeMoiré. This guarantees that the observed performance improvements are solely attributable to the application of ShapeMoiré, yet not influenced by other factors. Unless otherwise noted, the baseline model is ESDNet-L [35]. Detailed implementations of our method can be found in our released code.

4.1 Comparison with State-of-the-Art Methods

We compared our method with state-of-the-art techniques and reported the results in Table 1. As evident from the table, our ShapeMoiré significantly outperforms existing methods across most scenarios. It is noteworthy that our approach establishes a new SOTA across all four datasets in terms of the PSNR metric, which quantifies the level of noise or distortion compared to the original undistorted image. The results highlight that the modeled shape features in our Shapemoiré play a crucial role in identifying the causal factors essential for image demoiréing. Furthermore, ShapeMoiré demonstrates superior performance compared to existing approaches on the UHDM dataset, emphasizing the effectiveness of our proposed method for ultra-high-definition images.

4.2 Generalization to Diverse Architectures

The proposed ShapeMoiré is a versatile approach that can be seamlessly integrated into existing methods for image demoiréing. To test its generalization capability, we applied our ShapeMoiré to

Table 1. Performance Comparison with State-of-the-Art Methods on LCDMoiré, TIP2018, FHDMi, and UHDM Datasets

Dataset	Metrics	Input	DMCNN (TIP18) [24]	MopNet (ICCV19) [11]	WDNet (ECCV20) [20]	MBCNN (CVPR20) [47]	ESDNet (ECCV22) [35]	SEDNet-L (ECCV22) [35]	DDA (ICLR23) [45]	RRID (arXiv24) [32]	ShapeMoiré
LCDMoiré [36]	PSNR↑	10.44	35.48	-	29.66	44.04	44.83	45.34	41.68	-	46.56
	SSIM↑	0.5717	0.9785	-	0.9670	0.9948	0.9963	0.9966	0.9869	-	0.9930
TIP2018 [24]	PSNR↑	20.3	26.77	27.75	28.08	30.03	29.81	30.11	-	-	30.16
	SSIM↑	0.7380	0.8710	0.8950	0.9040	0.8930	0.9160	0.9200	-	-	0.8795
FHDMi [12]	PSNR↑	17.97	21.54	22.76	-	22.31	24.5	24.88	23.62	24.39	25.06
	SSIM↑	0.7033	0.7727	0.7958	-	0.8095	0.8351	0.8440	0.8293	0.8300	0.8474
	LPIPS↓	0.2837	0.2477	0.1794	-	0.1980	0.1354	0.1301	-	-	0.1306
UHDM [35]	PSNR↑	17.12	19.91	19.49	20.36	21.41	22.12	22.42	-	-	22.95
	SSIM↑	0.5089	0.7575	0.7572	0.6497	0.7932	0.7956	0.7985	-	-	0.8021
	LPIPS↓	0.5314	0.3764	0.3857	0.4882	0.3318	0.2551	0.2454	-	-	0.2415

PSNR, peak signal-to-noise ratio; SSIM, structure similarity; LPIPS, learned perceptual image patch similarity.

(↑) and (↓) indicate better performance with larger and smaller numbers, respectively. Red: best and Blue: second best.

Table 2. Performance Comparison with Different Baseline Methods on Four Datasets

Architecture	Method	UHDM [35]			FHDMi [12]			TIP2018 [24]		LCDMoiré [36]		Params. (M)
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	PSNR↑	SSIM↑	
ESDNet [35]	Baseline ^a	22.253	0.7974	0.2511	24.393	0.8392	0.1366	29.791	0.8753	45.286	0.9921	5.934
	ShapeMoiré	22.597	0.8007	0.2490	24.629	0.8402	0.1336	29.862	0.8758	45.537	0.9925	5.934
	Δ	0.344↑	0.0033↑	0.0021↓	0.236↑	0.0010↑	0.0030↓	0.071↑	0.0005↑	0.251↑	0.0004↑	0
ESDNet-L [35]	Baseline ^a	22.554	0.7997	0.2468	24.808	0.8435	0.1321	30.096	0.8789	45.544	0.9925	10.623
	ShapeMoiré	22.948	0.8021	0.2415	25.064	0.8474	0.1306	30.161	0.8795	46.558	0.993	10.623
	Δ	0.394↑	0.0024↑	0.0053↓	0.256↑	0.0039↑	0.0015↓	0.065↑	0.0006↑	1.014↑	0.0005↑	0
WDNet [20]	Baseline ^a	19.181	0.6417	0.4271	21.161	0.7714	0.2411	27.812	0.8381	37.324	0.9721	3.36
	ShapeMoiré	19.882	0.7442	0.3272	22.182	0.7901	0.2134	28.312	0.8535	38.408	0.9782	3.36
	Δ	0.701↑	0.1025↑	0.0999↓	1.021↑	0.0187↑	0.0277↓	0.500↑	0.0154↑	1.084↑	0.0061↑	0
DMCNN [24]	Baseline ^a	17.812	0.7294	0.5561	19.313	0.7375	0.3395	24.518	0.8073	31.425	0.9552	1.426
	ShapeMoiré	18.036	0.7336	0.5185	19.615	0.7495	0.2889	25.381	0.8234	31.789	0.9562	1.426
	Δ	0.224↑	0.0042↑	0.0376↓	0.302↑	0.0120↑	0.0506↓	0.863↑	0.0161↑	0.364↑	0.0010↑	0

^aThe performance we reproduced on the respective datasets using the source code provided by different methods, noting slight discrepancies from the original publications.

four representative demoiréing methods, namely ECDNet [35], ECDNet-L [35], WDnet [20], and DMCNN [24]. Table 2 demonstrates that ShapeMoiré consistently delivers significant performance improvements across ALL settings. Moreover, our method introduces no additional inference cost compared to baselines, i.e., the added parameters are zero. It is worth noting that, as observed in Table 2, the performance of current methods mostly correlates positively with the number of parameters. However, ShapeMoiré demonstrates the ability to enhance performance without increasing the network’s parameter count. For instance, our method achieves comparable (or even higher PSNR) performance to ESDNet-L (which has almost twice the parameter count of ESDNet) on the UHDM dataset, even when keeping the parameter count consistent with ESDNet.

4.3 Generalization to Image Deblurring

To further evaluate the generalization ability of the proposed method beyond the demoiréing task, we extend our experiments to the dual-pixel defocus deblurring task using the DPDD dataset [1]. We adopt MIRNet-v2 [40] as the baseline architecture and integrate our proposed ShapeConv and Shape-Architecture modules. As presented in Table 3, the ShapeMoiré-enhanced model consistently outperforms the baseline across all three metrics—PSNR, SSIM, and LPIPS—on both indoor and outdoor subsets. Specifically, on the combined test set, our method improves PSNR from 26.23 to

Table 3. Dual-Pixel Defocus Deblurring Comparisons on the DPDD Dataset [1]

Architecture	Method	Indoor Scenes			Outdoor Scenes			Combined		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
MIRNet-v2 [40]	Baseline	28.95	0.8800	0.1585	23.66	0.7549	0.2109	26.23	0.8158	0.1854
	ShapreMoiré	29.05	0.8841	0.1507	23.77	0.7569	0.1974	26.34	0.8188	0.1747
	Δ	0.10↑	0.0041↑	0.0078↓	0.11↑	0.020↑	0.0135↓	0.11↑	0.0030↑	0.0107↓

The test set of DPDD contains 37 indoor scenes and 39 outdoor scenes.

Table 4. Ablation Study of ShapreMoiré on the UHDM Dataset

ShapeConv	Shape-Architecture	PSNR↑	SSIM↑	LPIPS↓
×	×	22.253	0.7974	0.2511
✓	×	22.469	0.7990	0.2424
×	✓	22.315	0.7984	0.2471
✓	✓	22.597	0.8007	0.2490

The bold indicates the best result.

26.34, SSIM from 0.8158 to 0.8188 and reduces LPIPS from 0.1854 to 0.1747. These improvements indicate not only better pixel-level fidelity but also enhanced perceptual quality.

The consistent performance gains across different domains and evaluation metrics suggest that ShapeMoiré is a versatile and effective enhancement module that can be generalized to broader image restoration tasks beyond demoiréing.

4.4 Ablation Study

The two components we propose, ShapeConv and Shape-Architecture, are independent of each other. This means that adding either one of them to the baseline method alone can enhance the model's performance. To validate the effectiveness of both components, we conducted detailed ablation studies. The results of these studies are presented in Table 4. One can observe that the removal of both ShapeConv and Shape-Architecture leads to degraded model performance. Introducing either the ShapeConv or the Shape-Architecture yields a positive outcome. Combining these two together, i.e., our final ShapeMoiré, achieves the best results, highlighting their complementary effect on the overall model performance.

Additionally, we computed the inference time of both the baseline and our ShapeMoiré method. Specifically, we conducted this experiment on a single NVIDIA A40 GPU with 48 GB memory and ran both models for three runs. Subsequently, we calculated the average inference time for 5,000 images and presented the **milliseconds (ms)** per image in Table 5. It is evident that our ShapeMoiré method imposes a very negligible overhead on the baseline model in most cases and even runs faster than the baseline due to clock time oscillation. This results validate the efficiency of the proposed ShapeMoiré method.

We conducted experiments with three different settings of the loss weights (α, β), as shown in Table 6. The results demonstrate that across all configurations, our method consistently outperforms the baseline, indicating that the proposed framework is not highly sensitive to the specific choice of weight parameters and exhibits robust performance. While there are slight performance variations depending on the weighting (e.g., slightly better LPIPS with $\alpha = 10$), we attribute these differences to potential dataset-specific characteristics. To maintain fairness and reproducibility, we adopt the balanced setting ($\alpha = 1, \beta = 1$) for all experiments reported in the main paper.

Table 5. Inference Time Comparison of Baseline and Our ShapeMoiré on the UHDM Dataset

Architecture	Method	ms/Image		
		Run1	Run2	Run3
ESDNet [35]	Baseline	7.70	8.00	7.80
	ShapeMoiré	8.30	8.04	7.98
WDNet [20]	Baseline	6.79	7.07	6.62
	ShapeMoiré	6.72	7.01	7.06
DMCNN [24]	Baseline	2.17	1.96	1.91
	ShapeMoiré	1.95	2.07	2.21

Table 6. Ablation Study on the Weighting Coefficients for the Loss Terms

α	β	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
Baseline		31.425	0.9552	0.093
0.1	1.0	31.975	0.9616	0.085
1.0	1.0	31.789	0.9562	0.088
10	1.0	32.077	0.9609	0.070

Experiments are conducted using DMCNN as the baseline method on the LCDMoiré dataset.

4.5 Visualization

In Figure 4, we present qualitative comparisons across four sample cases—(a), (b), (c), and (d)—each visualized with moiré-contaminated input, ground truth, baseline output, and our ShapeMoiré output. We include two representative baselines, WDNet [20] and ESDNet-L [35], to assess the generalizability of our method. For each baseline, its corresponding ShapeMoiré variant is denoted as ShapeMoiré_A and ShapeMoiré_B, respectively. First, we observe that stronger baselines (e.g., ESDNet-L) generally produce better restoration results, as seen by comparing the green boxes of (b) and (d) with those of (a) and (c). However, regardless of the baseline strength, our ShapeMoiré consistently demonstrates superior moiré removal and better structural detail recovery. For instance, in subfigure (a), ShapeMoiré_A (blue box) more accurately reconstructs the sailboat and the surrounding sky compared to the baseline (green box). Similarly, in subfigure (b), ShapeMoiré_B produces more uniform color on the lake surface. In subfigures (c) and (d), our method further reduces residual moiré artifacts while better preserving textures such as road surfaces. These observations collectively highlight the robustness and effectiveness of our method across varying baseline strengths and image characteristics.

To further validate the practicality of our proposed method, we captured real-world images using smartphones and applied both the baseline and ShapeMoiré methods to these cases, as depicted in Figure 5. It is important to note that the patterns of moiré are not fixed; for the same original image, different devices may capture different moiré patterns, as illustrated by the two images with moiré, where $I_{moiré}^A$ is the original input image in the UHDM dataset [35] and $I_{moiré}^B$ is from our captured smartphone input image. It can be observed that $I_{moiré}^A$ exhibits more widespread moiré artifacts across the entire region ($Pattern^A$), while in $I_{moiré}^B$, the artifacts are concentrated in the sky area at the center of the image ($Pattern^B$). From the results in Figure 5, several observations can be made: (1) our ShapeMoiré consistently outperforms the baseline method, regardless of the



Fig. 4. Visualization results from the UHDM dataset. Subfigures (a), (b), (c), and (d), respectively, showcase different sample cases. In each subfigure, the $I_{\text{moiré}}$, I_{gt} , Baseline , and ShapeMoiré correspond to the moiré-contaminated input image, ground truth, baseline output, and ShapeMoiré output, respectively. For each case, Baseline_A and ShapeMoiré_A denote results based on WDNNet [20], while Baseline_B and ShapeMoiré_B are derived using ESDNet-L [35] as the baseline. The larger boxes display magnified versions of the corresponding colored boxes, providing a clearer view of fine-grained differences.

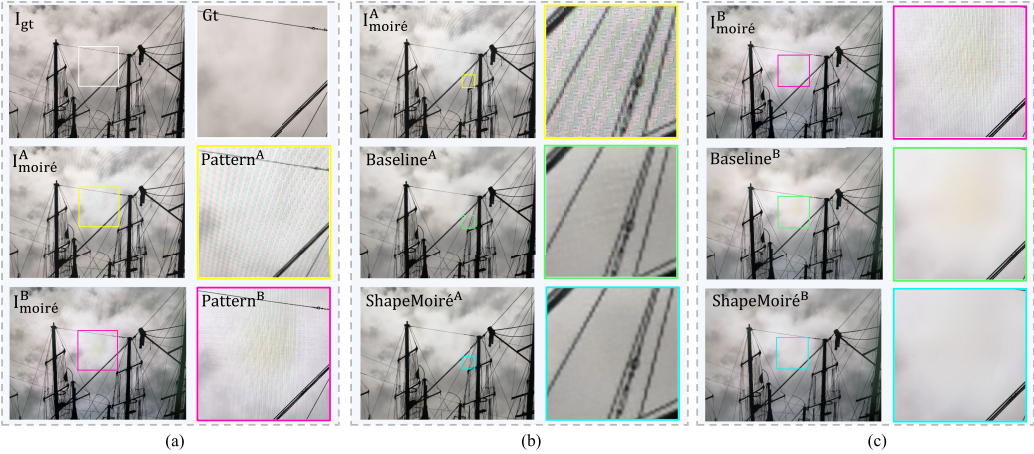


Fig. 5. Qualitative results of image demoiréing for real-world cases. (a) Given the same ground-truth image I_{gt} , different moiré-contaminated inputs $I_{\text{moiré}}^A$ and $I_{\text{moiré}}^B$ are obtained by photographing with different smartphones. (b) and (c) show the demoiréing results for $I_{\text{moiré}}^A$ and $I_{\text{moiré}}^B$ respectively, comparing a baseline method and our proposed ShapeMoiré. Larger boxes represent enlarged views of the corresponding colored small boxes at the borders, providing a clearer visualization of the moiré removal effectiveness.

two morphologies of moiré, $I_{\text{moiré}}^A$ and $I_{\text{moiré}}^B$; our method exhibits superior moiré removal effects. (2) Our method demonstrates robustness against different moiré patterns, even helping alleviate color distortions. As shown in example $I_{\text{moiré}}^B$ of Figure 5, due to severe moiré in the blank sky area at the center, noticeable color deviations occur. ShapeMoiré not only excels in removing Moiré patterns but also effectively restores the true colors of the sky.

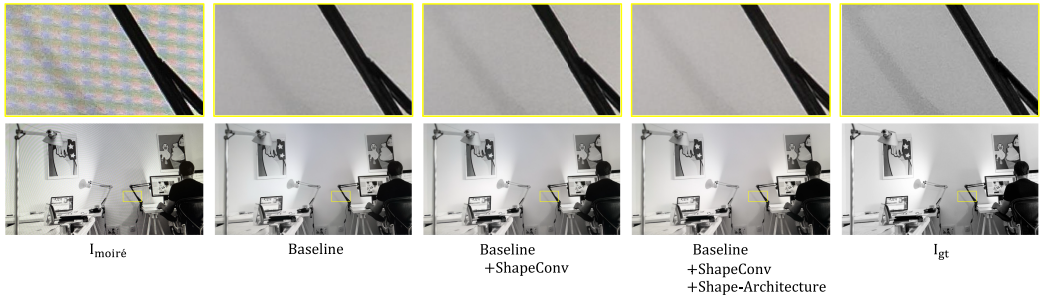


Fig. 6. Qualitative ablation results on the UHDM dataset using ESDNet as the baseline. From left to right: moiré-contaminated input, baseline result, baseline with ShapeConv, ShapeMoiré, and ground truth. The top row of each image shows a zoom-in of the highlighted region, allowing for closer inspection of visual differences.

In Figure 6, we present qualitative comparisons to visually examine the effects of ablating key components, including ShapeConv and the Shape-Architecture, based on the strong ESDNet [35] baseline. As shown, while the baseline already delivers competitive results due to its advanced design, further improvements can still be observed when our proposed modules are added. Specifically, adding ShapeConv leads to better restoration of fine-grained details. For instance, in the shadow area beneath the lamp structure, ShapeConv enables more accurate reconstruction compared to the baseline alone. This suggests that the patch-level shape-aware representation contributes to finer spatial precision. Moreover, incorporating the Shape-Architecture introduces a noticeable enhancement in global image appearance. Compared to the baseline's overall cooler tone, the Shape-Architecture helps correct color bias, producing a warmer tone that more closely aligns with the ground truth. This demonstrates its ability to recover higher-level structural and color features from the image-level shape stream. Together, these qualitative results visually confirm the effectiveness of each component, serving as a strong perceptual complement to our quantitative evaluations.

5 Conclusion

In this article, we extend our previously accepted ShapeConv to the image demoiréing domain. Unlike existing methods, our proposed ShapeMoiré addresses the channel inconsistency problem of moiré patterns using *shape* features. The key extension in the method is the introduction of the Shape-Architecture, which models the *shape* information from the image level and acts as a complement to the patch-level ShapeConv. The experimental results demonstrate its superiority over existing state-of-the-arts on four public popular datasets. Furthermore, we verify that our method can be painlessly integrated into four strong baselines to improve their performance, while introducing negligible computational cost during inference. Additionally, our ShapeMoiré yields significant promise in real-world cases with irregular moiré patterns, making it potentially suitable for edge device application and deployment. Beyond demoiréing, we have also applied ShapeMoiré to the image deblurring task and observed promising improvements, suggesting its potential generalizability to broader image restoration problems.

Despite the encouraging results, our method has certain limitations. For instance, the current design primarily targets repetitive moiré patterns and may be less effective for more complex or irregular noise types that do not exhibit clear shape structures. Also, while computational overhead during inference is minimal, training with ShapeConv and Shape-Architecture requires additional resources. In future work, we plan to explore adapting and extending the framework to other

image restoration tasks such as general denoising, deraining, and compression artifact removal. Furthermore, integrating more advanced frequency-domain analysis and lightweight model designs could improve both effectiveness and efficiency, making the approach more practical for diverse real-world applications.

References

- [1] Abdullah Abuolaim and Michael S. Brown. 2020. Defocus deblurring using dual-pixel data. In *Proceedings of the 16th European Conference on Computer Vision (ECCV '20)*. Springer, 111–126.
- [2] Jinming Cao, Hanchao Leng, Dani Lischinski, Daniel Cohen-Or, Changhe Tu, and Yangyan Li. 2021. Shapeconv: Shape-aware convolutional layer for indoor rgb-d semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7088–7097.
- [3] Bo-Hao Chen and Shih-Chia Huang. 2015. An advanced visibility restoration algorithm for single hazy images. *ACM Transactions on Multimedia Computing, Communications, and Applications* 11, 4 (2015), 1–21.
- [4] Xi Cheng, Zhenyong Fu, and Jian Yang. 2019. Multi-scale dynamic feature encoding network for image demoiréing. In *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE, 3486–3493.
- [5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 248–255.
- [6] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. 2015. Compression artifacts reduction by a deep convolutional network. In *Proceedings of the IEEE International Conference on Computer Vision*, 576–584.
- [7] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2014. Learning a deep convolutional network for image super-resolution. In *Proceedings of the 13th European Conference on Computer Vision (ECCV '14)*. Springer, 184–199.
- [8] Hang Guo, Jinmin Li, Tao Dai, Zhihao Ouyang, Xudong Ren, and Shu-Tao Xia. 2024. Mambair: A simple baseline for image restoration with state-space model. In *European Conference on Computer Vision*. Springer, 222–241.
- [9] Yangyang Guo, Liqiang Nie, Harry Cheng, Zhiyong Cheng, Mohan Kankanhalli, and Alberto Del Bimbo. 2023. On modality bias recognition and reduction. *ACM Transactions on Multimedia Computing, Communications and Applications* 19, 3 (2023), 1–22.
- [10] Yangyang Guo, Liqiang Nie, Zhiyong Cheng, Qi Tian, and Min Zhang. 2021. Loss re-scaling VQA: Revisiting the language prior problem from a class-imbalance view. *IEEE Transactions on Image Processing: A Publication of the IEEE Signal Processing Society* 31 (2021), 227–238.
- [11] Bin He, Ce Wang, Boxin Shi, and Ling-Yu Duan. 2019. Mop moire patterns using mopnet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2424–2432.
- [12] Bin He, Ce Wang, Boxin Shi, and Ling-Yu Duan. 2020. Fhde 2 net: Full high definition demoiréing network. In *Proceedings of the 16th European Conference on Computer Vision (ECCV '20)*. Springer, 713–729.
- [13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.
- [14] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4700–4708.
- [15] Junjun Jiang, Jiayi Ma, Chen Chen, Xinwei Jiang, and Zheng Wang. 2016. Noise robust face image super-resolution through smooth sparse representation. *IEEE Transactions on Cybernetics* 47, 11 (2016), 3991–4002.
- [16] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. 2017. Deep Laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 624–632.
- [17] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. 2021. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1833–1844.
- [18] Jingbo Lin, Zhilu Zhang, Yuxiang Wei, Dongwei Ren, Dongsheng Jiang, Qi Tian, and Wangmeng Zuo. 2024. Improving image restoration through removing degradations in textual representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2866–2878.
- [19] Bolin Liu, Xiao Shu, and Xiaolin Wu. 2018. Demoiréing of camera-captured screen images using deep convolutional neural network. arXiv:1804.03809. Retrieved from <https://arxiv.org/abs/1804.03809>
- [20] Lin Liu, Jianzhuang Liu, Shanxin Yuan, Gregory Slabaugh, Aleš Leonardis, Wengang Zhou, and Qi Tian. 2020. Wavelet-based dual-branch network for image demoiréing. In *Proceedings of the 16th European Conference on Computer Vision (ECCV '20)*. Springer, 86–102.
- [21] Pai Chet Ng, James She, Kang Eun Jeon, and Matthias Baldauf. 2017. When smart devices interact with pervasive screens: A survey. *ACM Transactions on Multimedia Computing, Communications, and Applications* 13, 4 (2017), 1–23.
- [22] Yogesh Singh Rawat and Mohan S. Kankanhalli. 2015. Context-aware photography learning for smart mobile devices. *ACM Transactions on Multimedia Computing, Communications, and Applications* 12, 1s (2015), 1–24.

- [23] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of the 18th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI '15)*. Springer, 234–241.
- [24] Yujing Sun, Yizhou Yu, and Wenping Wang. 2018. Moiré photo restoration using multiresolution convolutional neural networks. *IEEE Transactions on Image Processing* 27, 8 (2018), 4160–4172.
- [25] H. Wang, Q. Tian, L. Li, and X. Guo. 2021. Image demoiréing with a dual-domain distilling network. In *Proceedings of the 2021 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1–6.
- [26] Longguang Wang, Yingqian Wang, Zhengfa Liang, Zaiping Lin, Jungang Yang, Wei An, and Yulan Guo. 2019. Learning parallax attention for stereo image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12250–12259.
- [27] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. 2004. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing: A Publication of the IEEE Signal Processing Society* 13, 4 (2004), 600–612.
- [28] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. 2022. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17683–17693.
- [29] Pengxu Wei, Ziwei Xie, Guanbin Li, and Liang Lin. 2023. Taylor neural network for real-world image super-resolution. *IEEE Transactions on Image Processing: A Publication of the IEEE Signal Processing Society* 32 (2023), 1942–1951.
- [30] Zhitong Xiong, Sining Chen, Yi Wang, Lichao Mou, and Xiao Xiang Zhu. 2023. Gamus: A geometry-aware multi-modal semantic segmentation benchmark for remote sensing data. arXiv:2305.14914. Retrieved from <https://arxiv.org/abs/2305.14914>
- [31] Kang Xu, Weixin Li, Xia Wang, Xiaoyan Hu, Ke Yan, Xiaojie Wang, and Xuan Dong. 2023. CUR transformer: A convolutional unbiased regional transformer for image denoising. *ACM Transactions on Multimedia Computing, Communications and Applications* 19, 3 (2023), 1–22.
- [32] Shuning Xu, Binbin Song, Xiangyu Chen, and Jiantao Zhou. 2023. Image Demoiréing in RAW and sRGB domains. arXiv:2312.09063. Retrieved from <https://arxiv.org/abs/2312.09063>
- [33] Chenggang Yan, Zhisheng Li, Yongbing Zhang, Yutao Liu, Xiangyang Ji, and Yongdong Zhang. 2020. Depth image denoising using nuclear norm and learning graph model. *ACM Transactions on Multimedia Computing, Communications, and Applications* 16, 4 (2020), 1–17.
- [34] Jingyu Yang, Fanglei Liu, Huanjing Yue, Xiaomei Fu, Chunping Hou, and Feng Wu. 2017. Textured image demoiréing via signal decomposition and guided filtering. *IEEE Transactions on Image Processing* 26, 7 (2017), 3528–3541.
- [35] Xin Yu, Peng Dai, Wenbo Li, Lan Ma, Jiajun Shen, Jia Li, and Xiaojuan Qi. 2022. Towards efficient and scale-robust ultra-high-definition image demoiréing. In *European Conference on Computer Vision*. Springer, 646–662.
- [36] Shanxin Yuan, Radu Timofte, Gregory Slabaugh, Aleš Leonardis, Bolun Zheng, Xin Ye, Xiang Tian, Yaowu Chen, Xi Cheng, Zhenyong Fu, et al. 2019. Aim 2019 challenge on image demoiréing: methods and results. In *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE, 3534–3545.
- [37] Huanjing Yue, Yijia Cheng, Yan Mao, Cong Cao, and Jingyu Yang. 2022. Recaptured screen image demoiréing in raw doma. *IEEE Transactions on Multimedia* 25 (2022), 5589–5600.
- [38] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. 2022. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5728–5739.
- [39] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. 2021. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14821–14831.
- [40] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. 2022. Learning enriched features for fast image restoration and enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 2 (2022), 1934–1948.
- [41] Hongguang Zhang, Yuchao Dai, Hongdong Li, and Piotr Koniusz. 2019. Deep stacked hierarchical multi-patch network for image deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5978–5986.
- [42] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. 2017. Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing: A Publication of the IEEE Signal Processing Society* 26, 7 (2017), 3142–3155.
- [43] Kai Zhang, Wangmeng Zuo, and Lei Zhang. 2018. Learning a single convolutional super-resolution network for multiple degradations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3262–3271.
- [44] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 586–595.

- [45] Yuxin Zhang, Mingbao Lin, Xunchao Li, Han Liu, Guozhi Wang, Fei Chao, Shuai Ren, Yafei Wen, Xiaoxin Chen, and Rongrong Ji. 2023. Real-time image demoiréing on mobile devices. arXiv:2302.02184. Retrieved from <https://arxiv.org/abs/2302.02184>
- [46] Yongqin Zhang, Jiaying Liu, Wenhan Yang, and Zongming Guo. 2015. Image super-resolution based on structure-modulated sparse representation. *IEEE Transactions on Image Processing: A Publication of the IEEE Signal Processing Society* 24, 9 (2015), 2797–2810.
- [47] Bolun Zheng, Shanxin Yuan, Gregory Slabaugh, and Ales Leonardis. 2020. Image demoiréing with learnable bandpass filters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3636–3645.
- [48] Jiageng Zhong, Ming Li, Hanqi Zhang, and Jiangying Qin. 2023. Combining photogrammetric computer vision and semantic segmentation for fine-grained understanding of coral reef growth under climate change. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 186–195.
- [49] Qiu Zhou, Jinming Cao, Hanchao Leng, Yifang Yin, Yu Kun, and Roger Zimmermann. 2024. SOGDet: Semantic-occupancy guided multi-view 3D object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 7668–7676.

Appendix

A Proof for Equation (9)

PROOF.

$$\begin{aligned}
\mathbb{F}_{cout} &= \sum_k^{K_h \times K_w \times C_{in}} (\mathbb{K}_{k,cout} \times \mathbb{P}_{BS_k}) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{i,j,cout} \times \mathbb{P}_{BS_{i,j}}) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{i,j,cout} \times (\mathbb{P}_{B_{1,j}} + \mathbb{P}_{S_{i,j}})) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{i,j,cout} \times (\mathbb{W}_B \times \mathbb{P}_{B_{1,j}} + \sum_m^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{P}_{S_{m,j}}))) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{W}_B \times \mathbb{K}_{i,j,cout} \times \mathbb{P}_{B_{1,j}} + \sum_m^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{i,j,cout} \times \mathbb{P}_{S_{m,j}})) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{W}_B \times \mathbb{K}_{B_{1,j},cout} \times \mathbb{P}_{i,j} + \sum_m^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{i,j,cout} \times (\mathbb{P}_{m,j} - \mathbb{P}_{B_{1,j}}))) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{B_{1,j},cout} \times \mathbb{P}_{i,j}) + \sum_i^{K_h \times K_w} \sum_j^{C_{in}} \sum_m^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{i,j,cout} \times \mathbb{P}_{m,j} - \mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{i,j,cout} \times \mathbb{P}_{B_{1,j}}) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{B_{1,j},cout} \times \mathbb{P}_{i,j}) + \sum_i^{K_h \times K_w} \sum_j^{C_{in}} \sum_m^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{i,j,cout} \times \mathbb{P}_{m,j} - \mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{B_{1,j},cout} \times \mathbb{P}_{m,j}) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{B_{1,j},cout} \times \mathbb{P}_{i,j}) + \sum_i^{K_h \times K_w} \sum_j^{C_{in}} \sum_m^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times (\mathbb{K}_{i,j,cout} - \mathbb{K}_{B_{1,j},cout}) \times \mathbb{P}_{m,j}) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{B_{1,j},cout} \times \mathbb{P}_{i,j}) + \sum_i^{K_h \times K_w} \sum_j^{C_{in}} \sum_m^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{S_{i,j},cout} \times \mathbb{P}_{m,j}) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{B_{1,j},cout} \times \mathbb{P}_{i,j}) + \sum_m^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{S_{m,j},cout} \times \mathbb{P}_{m,j}) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{B_{1,j},cout} \times \mathbb{P}_{i,j} + \mathbb{K}_{S_{i,j},cout} \times \mathbb{P}_{i,j}) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} ((\mathbb{K}_{B_{1,j},cout} + \mathbb{K}_{S_{i,j},cout}) \times \mathbb{P}_{i,j}) \\
&= \sum_i^{K_h \times K_w} \sum_j^{C_{in}} (\mathbb{K}_{BS_{i,j},cout} \times \mathbb{P}_{i,j}) \\
&= \sum_k^{K_h \times K_w \times C_{in}} (\mathbb{K}_{BS_{k,cout}} \times \mathbb{P}_k).
\end{aligned}$$

Received 15 February 2025; revised 9 June 2025; accepted 29 June 2025