Frequency-Domain Deep Guided Image Denoising

Zehua Sheng, Xiongwei Liu, Si-Yuan Cao, Hui-Liang Shen, and Huaqi Zhang

***Abstract*—Despite the tremendous advances in denoising techniques, it's still challenging to restore a clean image with salient structures based on one noisy observation, especially at high noise levels. In this work, we propose a frequency-domain guided denoising algorithm to conduct denoising with the help of a well-aligned guidance image. Thanks to their structural correlations, the frequency characteristics of the guidance image can indicate whether the frequency coefficients of the noisy target image are contributed by noise or textures. Therefore, the explicit frequency decomposition enables our denoising model to avoid over-smoothing detailed contents. However, as two input images are usually captured in different fields, their structures are not always consistent. Therefore, we model guided denoising with an optimization problem which considers both the representation model of the guidance image and the fidelity to the noisy target. Further, we design a convolutional neural network, called as FGDNet, to explore the optimal solution. Due to the visual masking phenomenon, human eyes are sensitive to noise in the flat areas, but may not perceive noise around edges or textures. Therefore, we expect to remove as much noise as possible to guarantee the spatial smoothness of flat contents, while also preserving high-frequency structures. Through frequency decomposition, our model can process the low-frequency and high-frequency contents separately. We also adopt a frequency-relevant loss function to train the network. Experimental results show that, compared with state-of-the-art guided and non-guided denoisers, our FGDNet achieves higher denoising accuracy and better visual quality in both flat and texture-rich regions.**

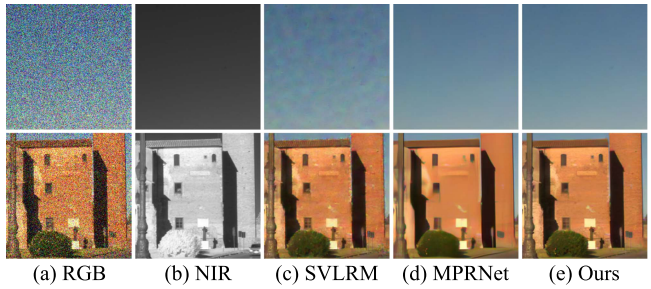
***Index Terms*—Frequency decomposition, guided image denoising, convolutional neural network.**

I. Introduction

RANDOM noise is one of the most common factors that degrade the quality of digital images in modern camera systems. For the last few decades, numerous studies have been conducted to restore clean images directly from their noisy observations. They usually take the advantage of various image priors, such as self-similarity [1], [2], [3], [4], [5], [6], [7], sparsity [3], [6], [8], and low-rankness [4], [5], to improve their capability of estimating image structures during noise removal. Recently, the exploitation of neural networks further improves the denoising performance [9], [10], [11], [12], [13], [14], [15], [16]. However, it's still difficult for these methods to distinguish between detailed contents and random noise, especially in the case of low signal-to-noise ratio (SNR).

Instead of exploring stronger image priors or designing more complicated network architectures, a popular trend is to take the original noisy image as the target and seek the help of external information provided by an aligned guidance image [17], [18], [19], [20] with high SNR. For instance, using an additional near-infrared (NIR) light source, we can acquire almost clean NIR images to guide the denoising process for the noisy RGB images captured in ambient light. To ensure that the captured RGB-NIR image pairs are well-aligned, the work [21] presents a dual-camera system. It contains two optically aligned digital cameras as well as a beam splitter to divide the incoming light into visible and NIR components. Since the images are simultaneously acquired in the same scene, their contents can be highly correlated in most areas.

To conduct guided denoising using a pair of aligned images, guided filtering [17] introduces an inspirational solution that the clean target image can be estimated by a certain representation of the guidance image. However, this model has two drawbacks. First, although it can restore fine structures and rich details according to the guidance image, its noise removal ability is somewhat limited. Two example results of NIR-guided RGB image denoising obtained by SVLRM [18], a follow-up work of the classical guided filtering, are illustrated in Fig. 1. Due to the visual masking phenomenon [22], human eyes may not be able to



(a) RGB (b) NIR (c) SVLRM (d) MPRNet (e) Ours

Fig. 1. Denoising results obtained by the guided restoration algorithm SVLRM [18], the non-guided denoising algorithm MPRNet [24], and our algorithm. SVLRM restores clear structures but produces splotched artifacts in the flat areas. MPRNet over-smooths the details. Our algorithm achieves the best visual results.

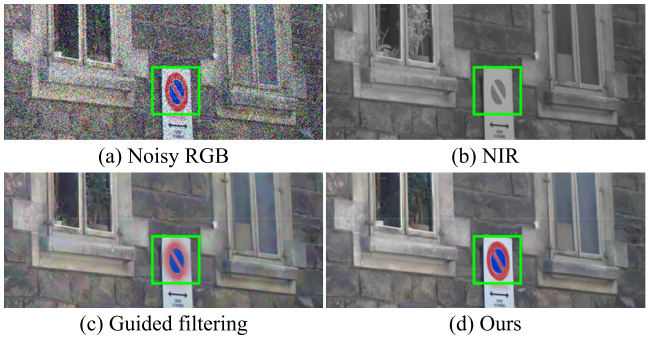
Manuscript received 31 March 2022; revised 10 July 2022, 23 August 2022, and 2 October 2022; accepted 8 October 2022. Date of publication 13 October 2022; date of current version 1 November 2023. This work was supported in part by the Ten Thousand Talents Program of Zhejiang Province under Grant 2020R52003 and in part by the ZJU-vivo Information Technology Joint Research Center. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Erkut Erdem. *(Corresponding author: Hui-Liang Shen.)*

Zehua Sheng, Xiongwei Liu, Si-Yuan Cao, and Hui-Liang Shen are with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: shengzehua@zju.edu.cn; liuxw11@zju.edu.cn; karlcao@hotmail.com; shenhl@zju.edu.cn).

Huaqi Zhang is with the Vivo Mobile Communication Company, Ltd., Hangzhou 310030, China (e-mail: zhanghuaqi@vivo.com).

Source codes are available at https://github.com/lustrouselixir/FGDNet.

Digital Object Identifier 10.1109/TMM.2022.3214375



(a) Noisy RGB (b) NIR

(c) Guided filtering (d) Ours

Fig. 2. Denoising results obtained by guided filtering [17] and our algorithm. In the result of guided filtering, the unique structures of RGB image is over-smoothed, and noise is not completely removed in the flat areas.

perceive the remaining noise around sharp edges. But it can be noticeable in flat areas, usually appearing as splotched artifacts. Therefore, to further improve the visual quality, it's essential to eliminate these artifacts.

Second, it cannot resolve the structural inconsistency issue, which basically exists in all kinds of practical guided denoising tasks when the input images have different modalities, such as RGB/NIR, flash/no-flash, depth/RGB, *etc*. For NIR-guided RGB image denoising, each image can have its unique textures because the same object can have different reflectances to the NIR and the visible light. As shown in Fig. 2, the unique structures of the target image are over-smoothed in the guided denoising result. In the follow-up work of guided filtering [18], [20], this problem is still not fully solved.

In this work, given a noisy target image and its aligned clean guidance image, we aim to generate a high-quality denoising result with rich details, while addressing the aforementioned two problems. For a denoising task, our goals of restoring flat and structurally rich areas are quite different. In the flat areas, we expect to remove as much noise as possible to ensure the spatial smoothness. When restoring image structures, due to the visual masking phenomenon, it's more important to preserve clear edges and details than to eliminate noise completely. From the perspective of image decomposition, flat areas are basically composed of low-frequency components, while edges and details are caused by the intensity changes and thus belong to the high frequencies. Hence, it motivates us to decompose the images into various frequency layers and process them with different denoising schemes accordingly.

Different from current guided restoration methods that are directly conducted in the spatial domain, we propose to split the input image pairs into various frequency layers based on patch-wise 2D discrete cosine transform (2D-DCT). Then, guided denoising is processed within each frequency layer independently. On one hand, in the low-frequency layers, we aim to recover the spatial smoothness of the base components to avoid splotched artifacts. On the other hand, we focus on reconstructing the high-frequency image structures according to the guidance image. The explicit division of the low and high frequencies ensures that the above two processes can be conducted simultaneously and individually.

The contribution of frequency analysis to guided denoising is beyond that. It also helps to distinguish weak details from random noise. According to the sparse coding theory [23], clean images can be linearly represented by a limit number of atoms, each of which records a specific pattern. Denote $\mathbf{y},\ \mathbf{g} \in \mathbb{R}^{s\times s}$ as a pair of patches extracted from the input target and guidance images at the same position. As they are structurally correlated and share many similar contents, it's natural to assume that they can be represented using the same set of atoms. That is, $\mathbf{y} = \mathbf{n} + \alpha_0 + \sum_{k=1}^{K} \alpha_k \mathbf{d}_k$ and $\mathbf{g} = \beta_0 + \sum_{k=1}^{K} \beta_k \mathbf{d}_k$, where $\{\mathbf{d}_k\}_{k=1,\dots,K}$ is the atom set, $\{\alpha_k\}_{k=1,\dots,K}$ and $\{\beta_k\}_{k=1,\dots,K}$ are the corresponding coefficients, α_0 and β_0 are two constants that denote the overall brightness levels, $\mathbf{n}$ is the noise component. After transforming $\mathbf{x}$ and $\mathbf{g}$ into the frequency domain, this linear representation model still holds, i.e., $\mathcal{T}(\mathbf{y}) = \mathcal{T}(\mathbf{n}) + \alpha_0\boldsymbol{\delta} + \sum_{k=1}^{K} \alpha_k \mathcal{T}(\mathbf{d}_k)$ and $\mathcal{T}(\mathbf{g}) = \beta_0\boldsymbol{\delta} + \sum_{k=1}^{K} \beta_k \mathcal{T}(\mathbf{d}_k)$, where $\mathcal{T}(\cdot)$ denotes the linear frequency transform function while $\boldsymbol{\delta}$ is the Dirac delta function whose values are zero except at zero frequency. Since the atoms are sparsely distributed in the frequency domain while noise is not, the frequency characteristics of the guidance image can well indicate whether the frequency coefficients of the noisy target image are contributed by noise or image structures, including both strong edges and detailed textures. In comparison, zero-frequency coefficients are less correlated across different modalities. Hence, it's essential to also adopt the intra-correlation information of the noisy target image to avoid severe intensity deviation.

To ensure that the guided denoising result is structurally faithful to the input target image, we explicitly include both images in the restoration framework. In this work, we construct an optimization model to obtain the mathematical form of our guided denoising model. Based on the above analysis, our proposed objective function is composed of three terms: a fidelity term to constrain the similarity between the denoising result and the input noisy target image, a linear representation term to reconstruct image structures from the guidance image, and a noise estimation term to further improve the accuracy of our guided denoising model. To avoid a time-consuming optimization process and empirical setting of the parameters, we design a convolutional neural network, called as FGDNet, to explore the optimal solution. A more detailed description is given in Section III-A.

In the training stage, we also introduce a frequency-relevant loss function to optimize the network by conducting different supervision processes on different frequency components. On low-frequency layers, we aim to remove as much noise as possible and eliminate artifacts in flat areas. On high-frequency layers, in contrast, our main purpose is to preserve edges and details. Previous studies on frequency learning [25], [26] demonstrate that, neural networks are prone to focusing on low-frequency contents in the training stage. Therefore, our frequency-domain learning strategy and frequency-relevant loss function can enable the network to also pay attention to high frequencies, which is beneficial for restoring images with more salient structures.

In summary, the main contributions of this work are as follows:

- We introduce frequency decomposition into guided image denoising to restore the low- and high-frequency contents in different manners. It guarantees the superior ability of noise removal, and also enables the denoising process to effectively distinguish details from random noise.
- We model guided denoising with an optimization problem to restore fine structures from the guidance image, while overcoming structural inconsistency issues across different modalities. We design a convolutional neural network, named as FGDNet, to explore its optimal solution.
- Combining the frequency-domain correlations of different modalities with the internal information learned from the noisy target image, our algorithm outperforms state-of-the-art denoising methods in terms of both accuracy and visual quality on various guided denoising tasks.

II. Related Work

A. Image Denoising

Image denoising is one of the most classical low-level vision tasks that has been well explored for decades of years. Techniques including filtering [27], [28], sparse representation [3], [8], and low-rank approximation [4], [5] are commonly used in designing traditional denoising algorithms. Based on the self-similarity assumption of natural images, denoising is significantly enhanced by the nonlocal framework [1], [2], [3], [4], [5], [6], [7]. That is, similar patches within adjacent areas are collected for collaborative noise reduction. However, these methods often require a long computational time to achieve high accuracy, and also show less flexibility when facing more complicated noise distributions.

Recently, numerous studies on deep learning have shown their powerful capability in noise removal. With large amounts of paired training data, the learning-based denoising methods outperform the traditional ones by a large margin. For real-world image denoising, MIRNet [29] introduces a multi-scale network architecture to receive contextual information from the low-resolution representations while maintaining spatially-precise high-resolution representations. NBNet [30] proposes to remove noise by learning a set of reconstruction basis in the feature space. Further, MPRNet [24] adopts a multi-stage architecture that separates the full restoration process into several manageable steps. In [31], Uformer introduces a local-enhanced window Transformer block to capture long-range dependencies with less computational cost compared to the original Transformer architecture. ADNet [12] uses an attention module to estimate the latent noise hidden in the complicated background. The model is relatively small and can achieve competitive denoising performance in blind denoising tasks [32].

However, acquiring extensive aligned noisy/clean image pairs for training is quite expensive in real-world photography. Therefore, some current studies focus on training denoising models without supervision, which have achieved competitive performance. In [33], the denoising model is designed based on the Stein's unbiased risk estimate theory. Assuming that the pixel values of clean images aren't statistically independent while noise is conditional pixel-wise independent given a clean image, Noise2Void [34] introduces a blind-spot network, presenting that the denoising model can be trained in a self-supervised way. Following this, FBI-Denoiser [35] aims to handle Poisson-Gaussian noise, AP-BSN [36] is developed for restoring clean sRGB images in real-world photography.

Overall, the denoising performance has been significantly improved, but it's still difficult for these approaches to preserve weak details especially at high noise levels. In comparison, our proposed guided denoising framework can well balance the above two issues, producing clean images with fine structures and rich details.

B. Guided Image Restoration

Since it's difficult to estimate accurate structural information based only on the noisy observations, current studies start to seek the help of external information. Classical guided filtering [17] assumes the filtering result can be linearly represented by the guidance image in local windows, so that it can remove noise while preserving fine structures. To avoid halo artifacts around sharp edges, the works [37], [38] introduce edge-aware constraints. In [39], a scale map is optimized to address the cross-field problems including gradient magnitude variation and gradient direction divergence.

Recently, deep learning is applied to the guided restoration tasks. DJF [40] builds an end-to-end joint filtering network that directly predicts the restoration results. To reduce the amount of computation, the work [41] designs a network that first generates a low-resolution result and then up-samples it with a guided filtering block. SVLRM [18] constructs a spatially variant linear representation model with learnable coefficients. Further, UMGF [20] introduces a simplified formulation of guided filtering inspired by the unsharp masking theory. Based on the convolutional sparse representation theory, CUNet [19] introduces a network that can not only process guided restoration but also deal with guided fusion tasks. Different from previous works combining the nonlinear activations of spatially invariant kernels to predict the final output, the work [42] learns spatially variant filtering kernels for each pixel. However, most of the guided restoration works cannot strike a good balance between detail preservation and noise removal, especially in the case of high noise levels.

C. Frequency-Relevant Image Restoration

In digital image processing, frequency decomposition is one of the basic techniques often employed in traditional image restoration works. A classical solution to obtain the restored images is designing adaptive filters in the frequency domain, which has been a key step in various denoising [2], [7] and deblurring [43], [44] algorithms.

Recently, frequency-domain analysis has also shown its potential in improving the performance of various learning-based restoration frameworks. For image deblurring, SDWNet [45] designs a wavelet reconstruction module that uses the information recovered in the frequency domain to complement the

spatial domain, so that the restored images can contain more high-frequency details. In [46], the authors introduce a Residual Fast Fourier Transform with Convolution (Res FFT-Conv) Block, which not only enables both low and high frequency learning, but also allows the image-wise receptive field since the Fourier transform is globally conducted.

For image super-resolution, the work [47] divides the feature maps into multiple components based on 2D-DCT. Features in different parts will be processed using different convolutional layers so that the super-resolution can be conducted in a more efficient way. In [48], the authors combine conditional learning with frequency coefficient analysis to address the over-fitting issue when dealing with blind super-resolution. This strategy can also be extended to handling blind denoising tasks.

Currently, frequency analysis is also applied in denoising networks. Considering the tradeoff between receptive field and efficiency, MWCNN [49] replaces the conventional pooling operation with 2D discrete wavelet transform (2D-DWT). Since 2D-DWT keeps both frequency and location information of feature maps, it can be helpful in preserving more detailed contents. By combining frequency domain analysis and attention mechanism, FAN [50] presents both good performance and interpretability in real-world image denoising.

Different from previous restoration methods, our work is the first to introduce frequency analysis into guided denoising tasks. It takes the advantage of the frequency-domain correlation between the input cross-modal image pairs, thereby effectively distinguishing weak textures from random noise. Based on this, our guided denoising model can restore noise-free images with fine structures and rich details from the noisy observations, even at high noise levels.

III. Proposed Algorithm

In this section, we first introduce the proposed frequency-domain optimization model for guided denoising. Then, we describe the architecture of our FGDNet constructed based on the solution form of the objective function. Finally, we discuss the loss function for training the network.

A. Frequency-Domain Optimization Model

Let $\mathbf{Y}$ denote the observed noisy target image corrupted by the additive noise. The corresponding clean image and noise component are denoted as $\mathbf{X}$ and $\mathbf{N}$, respectively. The noise model can thus be formulated as

$$\mathbf{Y} = \mathbf{X} + \mathbf{N}. \tag{1}$$

In actual application scenarios, the additive noise is usually modeled by the mixed Poisson-Gaussian distribution [51]. As our guided denoising process is identical in each of the three channels of the RGB image, we assume $\mathbf{Y}, \mathbf{X}, \mathbf{N} \in \mathbb{R}^{H\times W}$ for brevity, where H and W denote the height and width of the image respectively. The aligned guidance image is denoted as $\mathbf{G} \in \mathbb{R}^{H\times W}$.

Based on patch-wise 2D-DCT, we decompose the input image pair $\mathbf{Y}$ and $\mathbf{G}$ into various frequency layers. Here, 2D-DCT is a common technique that transforms images into the frequency domain using multiple basis functions. Each frequency layer aggregates the coefficients computed by the same basis function in sliding windows of size $s \times s$, and thus contains spatial features at one certain frequency. In this work, the stride of the sliding window is set to 1, and the images are processed with zero padding of size $s-1$. Those layers are then stacked into two frequency tensors $\mathbf{F}_Y \in \mathbb{R}^{H\times W\times L}$ and $\mathbf{F}_G \in \mathbb{R}^{H\times W\times L}$. Taking $\mathbf{Y}$ as an example, let $\mathbf{p}_{ij}$ be the patch of size $s \times s$ centered at position (i, j). We transform it into the frequency domain using 2D-DCT. The transformed patch is then vectorized and denoted as $\mathbf{u}_{ij} \in \mathbb{R}^{s^2\times 1}$, where the elements are arranged from low frequency to high frequency. The element value of $\mathbf{F}_Y$ at (i, j, l) is computed by $\mathbf{F}_Y(i, j, l) = \mathbf{u}_{ij}(l)$, $0 \le l \le s^2 - 1$. The layer number of $\mathbf{F}_Y$ is $L = s^2$, which is determined by the manually selected patch size. The other frequency tensor $\mathbf{F}_G$ is constructed in the same way. A visualization of our frequency decomposition process is illustrated in Fig. 3.

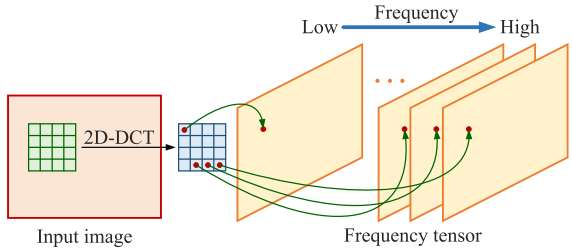


Fig. 3. A visualization of our frequency decomposition process. Setting the window size to $s \times s$, the computed frequency tensor has s^2 layers.

After transforming the input image pair into two frequency tensors, our guided denoising is then processed within each layer independently. Define $\mathbf{F}_Y^l, \mathbf{F}_G^l \in \mathbb{R}^{H\times W}$ as the l-th layers of the two tensors. Motivated by the guided filtering theory, the l-th layer of the restored frequency tensor $\hat{\mathbf{F}}_X$ can be estimated by the linear representation of the guidance layer, i.e.,

$$\hat{\mathbf{F}}_X^l = \mathbf{A}^l \odot \mathbf{F}_G^l, \tag{2}$$

where $\odot$ is the element-wise product operator and $\mathbf{A}^l$ is the linear coefficient matrix. However, due to the possible structural inconsistency between $\mathbf{Y}$ and $\mathbf{G}$, it's not sufficient to rely solely on the linear representation of the guidance layer to restore the unique structures of the target layer. To make the restored result structurally faithful to the target image, we propose to explicitly include its noisy observation in the representation model.

To some extent, guided denoising is not much different from the conventional denoising process, except that the guidance image can provide external information. In previous studies, internal priors such as sparsity, self-similarity, even implicit ones learned from training data, have shown their effectiveness in noise removal. The major purpose of exploiting external information is to reproduce the details that are concealed by the random noise. Hence, to ensure that our guided denoising can effectively handle different noise levels, it's beneficial to learn the noise features using some internal priors.

Based on the above analysis, we construct an optimization model to estimate the frequency tensor of the clean target image.

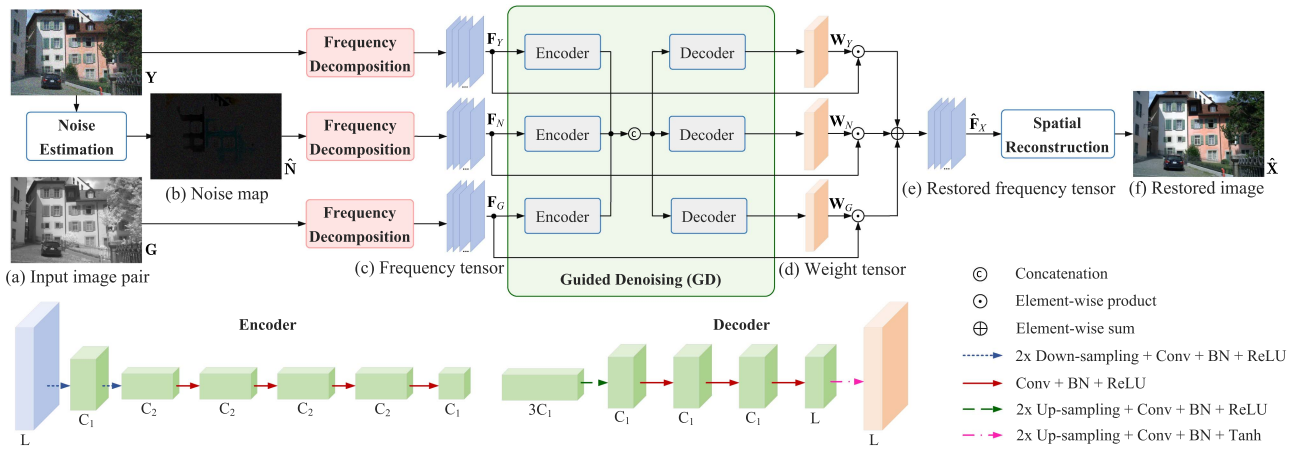


Fig. 4. Architecture of our proposed FGDNet. First, the noisy target image is fed into the noise estimation module to predict a rough noise map. Then, the frequency decomposition module transforms the noise map, the noisy target image and the guidance image into three frequency tensors. In the guided denoising module, three weight tensors are computed with three separate encoders and decoders to synthesize the frequency tensor of the denoised image. Finally, the spatial reconstruction module transforms the restored frequency tensor back to the spatial domain.

Stacking all the frequency layers together, the objective function is formulated as

$$[\mathbf{F}_X, \mathbf{F}_N, \mathbf{A}] = \underset{\mathbf{F}_X, \mathbf{F}_N, \mathbf{A}}{\arg\min} \underbrace{||\mathbf{\Lambda}_1 \odot (\mathbf{F}_X - \mathbf{F}_Y)||_F^2}_{\text{fidelity term}} + \underbrace{||\mathbf{\Lambda}_2 \odot (\mathbf{F}_X - \mathbf{A} \odot \mathbf{F}_G)||_F^2 + \Phi_1(\mathbf{A})}_{\text{linear representation term}} + \underbrace{||\mathbf{\Lambda}_3 \odot (\mathbf{F}_X - \mathbf{F}_Y + \mathbf{F}_N)||_F^2 + \Phi_2(\mathbf{F}_N)}_{\text{noise estimation term}}, \tag{3}$$

where $\mathbf{F}_N$ is the frequency tensor of noise, $\{\mathbf{\Lambda}_i\}_{i=1,2,3}$ are three parameter tensors, $\Phi_1(\cdot)$ is the regularization function of linear coefficient $\mathbf{A}$, and $\Phi_2(\cdot)$ is the internal prior of noise.

Suppose $\Phi_1(\cdot)$ and $\Phi_2(\cdot)$ are differentiable. To minimize this objective function, we compute its partial derivatives with respect to $\mathbf{A}$, $\mathbf{F}_N$ and $\mathbf{F}_X$, and let them be zero. Then we can get

$$\hat{\mathbf{A}} = \Psi_1(\mathbf{F}_G, \mathbf{F}_X), \tag{4}$$

$$\hat{\mathbf{F}}_N = \Psi_2(\mathbf{F}_Y, \mathbf{F}_X), \tag{5}$$

and

$$\hat{\mathbf{F}}_X = \Big[(\mathbf{\Lambda}_1 + \mathbf{\Lambda}_3) \odot \mathbf{F}_Y + \mathbf{\Lambda}_2 \odot \mathbf{A} \odot \mathbf{F}_G - \mathbf{\Lambda}_3 \odot \mathbf{F}_N\Big] \oslash (\mathbf{\Lambda}_1 + \mathbf{\Lambda}_2 + \mathbf{\Lambda}_3), \tag{6}$$

where $\Psi_1(\cdot)$ and $\Psi_2(\cdot)$ are related to the forms of $\Phi_1(\cdot)$ and $\Phi_2(\cdot)$, and $\oslash$ is the element-wise division operator. The values of $\{\mathbf{\Lambda}_i\}_{i=1,2,3}$ can be empirically determined by analyzing the structural correlation between the image pair. For complicated regularization terms, the optimization process may need to be performed iteratively.

In fact, the solution of $\mathbf{F}_N$ is equivalent to a conventional denoising process, and $\mathbf{F}_X$ can be regarded as a linear representation of the noisy observation, the guidance, and the noise component. In order to avoid empirically setting the values of parameter $\{\mathbf{\Lambda}_i\}_{i=1,2,3}$, we introduce a convolutional neural network, called as FGDNet, to accomplish this task. The forms of $\Psi_1(\cdot)$ and $\Psi_2(\cdot)$ can also be implicitly learned by training the network. Here, (6) can be re-written as

$$\hat{\mathbf{F}}_X = \mathbf{W}_Y \odot \mathbf{F}_Y + \mathbf{W}_N \odot \mathbf{F}_N + \mathbf{W}_G \odot \mathbf{F}_G. \tag{7}$$

Instead of directly computing the denoised image, FGDNet predicts three weight tensors $\mathbf{W}_Y$, $\mathbf{W}_N$, $\mathbf{W}_G \in \mathbb{R}^{H \times W \times L}$, and then compute the denoised frequency tensor using the linear representation model in (7). Based on the inverse 2D-DCT, we transform the restored frequency tensor back to the final denoising result $\hat{\mathbf{X}}$.

TABLE I
THE AVERAGE PSNR (DB), SSIM, AND LPIPS VALUES OF FGDNET ON IMAGES FROM THE RGB-NIR SCENE DATASET UNDER FREQUENCY DECOMPOSITION WITH DIFFERENT PATCH SIZES. NOISE IS GAUSSIAN WITH A STANDARD DEVIATION OF 0.2

Patch size	PSNR ↑	SSIM ↑	LPIPS ↓
3×3	30.28	0.9114	0.2409
7×7	30.60	0.9140	0.2201
11×11	30.56	0.9143	0.2364

B. Network Architecture

The architecture of FGDNet is displayed in Fig. 4. It is composed of four parts: the noise estimation module, the frequency decomposition module, the guided denoising module, and the spatial reconstruction module.

The noisy RGB image $\mathbf{Y}$ is first fed into the noise estimation module to obtain a rough noise map $\hat{\mathbf{N}}$. Here, we use ADNet [12] as the backbone architecture, except that the channel number of each feature map is reduced from the original 64 to 32. Its attention mechanism helps to estimate the latent noise hidden in the complicated background.

TABLE II
ABLATION STUDIES ON GUIDANCE IMAGES, NOISE ESTIMATION, FREQUENCY DECOMPOSITION, AND FREQUENCY-RELEVANT SUPERVISION UNDER GAUSSIAN NOISE ($\sigma = 0.2$) AND MIXED POISSON-GAUSSIAN NOISE ($\alpha = 0.02, \sigma = 0.2$)

Guidance Images	Noise Estimation	Frequency Decomposition	Frequency Loss	$\sigma = 0.2$			$\alpha = 0.02$, $\sigma = 0.2$		
				PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
✗	✓	✓	✓	28.93	0.8772	0.3633	28.49	0.8708	0.3731
✓	✗	✓	✓	30.32	0.9123	0.2326	29.97	0.9085	0.2387
✓	✓	✗	✗	30.11	0.9076	0.2473	29.69	0.9028	0.2552
✓	✓	✓	✗	30.55	0.9134	0.2385	30.17	0.9096	0.2448
✓	✓	✓	✓	**30.60**	**0.9140**	**0.2201**	**30.23**	**0.9102**	**0.2251**

TABLE III
THE AVERAGE PSNR (DB) AND SSIM VALUES OF THE PRE-DENOISED AND THE FINAL DENOISED IMAGES OBTAINED WITH DIFFERENT NOISE ESTIMATORS ON IMAGES FROM THE RGB-NIR SCENE DATASET IN THE CASES OF GAUSSIAN NOISE ($\sigma = 0.2$) AND MIXED POISSON-GAUSSIAN NOISE ($\alpha = 0.02, \sigma = 0.2$)

	$\sigma = 0.2$		$\alpha = 0.02$, $\sigma = 0.2$	
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
Pre-denoised (w/ *NEst-A*)	28.63	0.8661	28.16	0.8572
Pre-denoised (w/ *NEst-D*)	26.30	0.7875	25.76	0.7639
Final denoised (w/o estimator)	30.32	0.9123	29.97	0.9085
Final denoised (w/ *NEst-A*)	30.60	0.9140	30.23	0.9102
Final denoised (w/ *NEst-D*)	30.37	0.9128	30.04	0.9090

In the frequency decomposition module, the input RGB-NIR image pair $\mathbf{Y}$, $\mathbf{G}$, and the estimated noise map $\hat{\mathbf{N}}$ are decomposed into 3 frequency tensors $\mathbf{F}_Y$, $\mathbf{F}_G$ and $\mathbf{F}_N$. Since 2D-DCT is computed in sliding windows and its basis functions are fixed and spatially invariant, the decomposition process can be easily implemented using a convolution block with s^2 fixed kernels of size $s \times s$.

Guided denoising is the core module of the network. Each frequency tensor is fed into an individual encoder to compute a C_1-channel feature tensor. The encoder contains 6 convolution blocks with a kernel size of 3×3. Each convolution operation is followed by batch normalization and ReLU. Down-sampling in the first two blocks is implemented using max-pooling. Feature maps produced by the 2nd∼5th convolution blocks have C_2 channels. In this work, we set $C_1 = 48$ and $C_2 = 96$. The obtained three feature tensors are then concatenated and fed into three decoders with identical structures. Each decoder is also composed of 5 convolution blocks with a kernel size of 3×3. The first 4 convolution blocks are followed by batch normalization and ReLU, while the last one uses Tanh as the activation function. Up-sampling here is implemented by bilinear interpolation. The decoders predict three weight tensors to compute the restored frequency tensor $\hat{\mathbf{F}}_X$ according to (7).

Spatial reconstruction is the inverse process of frequency decomposition, which transforms the restored frequency tensor back to the spatial domain. Similarly, it can also be implemented using a convolution block.

C. Loss Function

In the training stage, the total loss function $\mathcal{L}$ is the summation of three terms, i.e., the restoration loss $\mathcal{L}_r$, the frequency loss $\mathcal{L}_f$, and the noise estimation loss $\mathcal{L}_n$, formulated as

$$\mathcal{L} = \mathcal{L}_r + \mathcal{L}_f + \mathcal{L}_n. \tag{8}$$

Restoration loss: The restoration loss function aims to ensure the overall denoising accuracy. In this work, it is implemented by the ℓ_1-norm, defined as

$$\mathcal{L}_r = ||\mathbf{X} - \hat{\mathbf{X}}||_1. \tag{9}$$

Noise estimation loss: To train the noise estimation module along with the entire network, we exploit a noise estimation loss to obtain the rough noise map $\hat{\mathbf{N}}$. It is formulated as

$$\mathcal{L}_n = \lambda_n ||\mathbf{X} - \left(\mathbf{Y} - \hat{\mathbf{N}}\right)||_1. \tag{10}$$

Frequency loss: Considering that the flat contents and the image structures belong to low and high frequencies, respectively, we adpot a frequency loss to restore them in different manners.

Here, to give a specific definition of the low-frequency layer, we estimate the energy distribution of those flat contents in the frequency domain. According to the DCT theory, for a patch $\mathbf{p}$ of size $s \times s$, its zero frequency coefficient is computed by $f_{\text{DC}} = s^{-1} \cdot \sum_{i=0}^{s^2-1} \mathbf{p}(i)$. If the patch is flat, the intensity of each pixel can be very close to the average value. Hence, the energy of $\mathbf{p}$ satisfies

$$E_p = \sum_{i=0}^{s^2-1} \mathbf{p}^2(i) \approx s^2 \cdot \left(\frac{1}{s^2} \sum_{i=0}^{s^2-1} \mathbf{p}(i) \right)^2 = f_{\text{DC}}^2. \tag{11}$$

Based on the Parseval theorem, we can conclude that the energy of flat areas is basically distributed at zero frequency. Therefore, it's natural to regard the zero frequency layer as the low-frequency layer to restore the desired flat contents. Besides, according to the frequency-domain sparse coding model presented in Section I, the zero-frequency coefficient records the overall brightness level of image. Therefore, it's reasonable to restore this single frequency layer individually.

The remaining frequency components record intensity variations (both soft and strong ones) among neighboring pixels, which describe the features of edges and details at different levels. Therefore, we treat them all as high frequencies where we concentrate on estimating image structures.

Hence, in this work, our frequency loss $\mathcal{L}_f$ is computed by the summation of a low-frequency loss $\mathcal{L}_{\text{LF}}$ and a high-frequency loss $\mathcal{L}_{\text{HF}}$.

TABLE IV
NUMBER OF PARAMETERS (M), FLOPS AND INFERENCE TIME (S) FOR DIFFERENT DENOISING NETWORKS

Algorithms	ADNet	MIRNet	NBNet	MPRNet	HINet	Uformer	DGUNet
Parameters (M)	0.52	31.79	10.45	15.73	88.67	20.60	17.20
FLOPs (G)	8.52	61.56	6.63	31.25	38.32	10.24	61.12
Inference time (ms)	4.14	96.22	23.61	74.46	17.51	40.17	74.37
Algorithms	NAFNet	SVLRM	CUNet	DKN	FGDNet-s	FGDNet-m	FGDNet
Parameters (M)	29.14	0.37	0.21	1.15	0.59	0.86	1.63
FLOPs (G)	4.04	6.10	3.40	2.05	4.36	4.97	6.60
Inference time (ms)	46.96	2.81	29.38	49.05	11.20	11.65	11.89

TABLE V
THE AVERAGE PSNR (DB), SSIM, AND LPIPS VALUES OF DIFFERENT ALGORITHMS ON IMAGES FROM THE RGB-NIR SCENE DATASET IN THE CASES OF GAUSSIAN NOISE ($\sigma = 0.1;\ 0.2$) AND MIXED POISSON-GAUSSIAN NOISE ($\alpha = 0.02, \sigma = 0.2$)

Algorithms	$\sigma = 0.1$			$\sigma = 0.2$			$\alpha = 0.02$, $\sigma = 0.2$		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
ADNet [12]	31.59	0.9186	0.2783	28.57	0.8655	0.3879	27.98	0.8533	0.4094
MIRNet [29]	32.04	0.9206	0.2679	29.29	0.8818	0.3486	28.90	0.8755	0.3588
NBNet [30]	31.67	0.9213	0.2728	28.83	0.8741	0.3607	28.39	0.8671	0.3695
MPRNet [24]	32.09	0.9265	0.2541	29.35	0.8732	0.3360	28.98	0.8779	0.3456
HINet [58]	32.33	0.9277	0.2557	29.59	0.8866	0.3362	29.21	0.8810	0.3452
Uformer [31]	31.95	0.9249	0.2676	29.22	0.8814	0.3460	28.82	0.8603	0.3522
DGUNet [59]	32.14	0.9265	0.2563	29.44	0.8846	0.3312	29.06	0.8786	0.3394
NAFNet [60]	32.36	0.9283	0.2513	29.63	0.8873	0.3323	29.26	0.8817	0.3411
GF [17]	29.76	0.9035	0.2798	26.86	0.8708	0.3203	26.24	0.8663	0.3320
CFJR [39]	29.49	0.8644	0.2034	27.12	0.8195	0.2651	26.42	0.8140	0.2644
SVLRM [18]	31.06	0.9195	0.2213	28.69	0.8822	0.2995	28.22	0.8759	0.3130
CUNet [19]	31.78	0.9313	0.2012	29.18	0.8948	0.2578	28.70	0.8873	0.2738
DKN [42]	29.98	0.8866	0.3170	26.46	0.7806	0.4350	25.85	0.7591	0.4515
UMGF [20]	30.59	0.9172	0.2455	28.42	0.8802	0.3072	28.08	0.8746	0.3174
FGDNet-s	32.87	0.9412	0.1905	30.14	0.9122	0.2404	30.02	0.9080	0.2468
FGDNet-m	32.95	0.9413	0.1884	30.47	0.9130	0.2325	30.07	0.9093	0.2373
FGDNet	**33.00**	**0.9415**	**0.1817**	**30.60**	**0.9140**	**0.2201**	**30.23**	**0.9102**	**0.2251**

1) *Low-frequency loss:* To restore flat areas that are spatially smooth without splotched artifacts, the low-frequency reconstruction error of each pixel is expected to be a constant value close to zero. In other words, the error of each pixel should be as small as possible, which can be handled by penalizing the ℓ_1-norm of the error map. Besides, as the reconstruction errors of adjacent overlapping patches should be close to each other, we penalize its successive difference as well. Hence, the low-frequency loss can be mathematically formulated as

$$\mathcal{L}_{\text{LF}} = \lambda_{l1}||\hat{\mathbf{F}}_X^0 - \mathbf{F}_X^0||_1 + \lambda_{l2}||D_h\left(\hat{\mathbf{F}}_X^0 - \mathbf{F}_X^0\right)||_2^2 + \lambda_{l2}||D_v\left(\hat{\mathbf{F}}_X^0 - \mathbf{F}_X^0\right)||_2^2, \tag{12}$$

where $D_h(\cdot)$ and $D_v(\cdot)$ denote the horizontal and the vertical successive difference functions, respectively.

2) *High-frequency loss:* Here, we use the ℓ_1-norm as the high-frequency loss, i.e.,

$$\mathcal{L}_{\text{HF}} = \sum_{l=1}^{s^2-1} \lambda_h||\hat{\mathbf{F}}_X^l - \mathbf{F}_X^l||_1. \tag{13}$$

The ℓ_1 loss leads to sparse solutions, so it's also beneficial for reducing high-frequency artifacts in the flat areas.

IV. EXPERIMENTS

In this section, we evaluate the performance of our proposed FGDNet and compare it with state-of-the-art guided and non-guided denoising methods on three practical guided denoising tasks including NIR-guided RGB image denoising, flash-guided no-flash image denoising, and RGB-guided depth image denoising. Three metrics, i.e., peak signal-to-noise ratio (PSNR), structural similarity index (SSIM) [52], and learned perceptual image patch similarity (LPIPS) [53] are adopted to assess the denoising performance. Here, LPIPS measures the perceptual similarity between the denoising result and the ground-truth image using a pre-trained VGGNet. Compared with PSNR and SSIM, LPIPS corresponds more to human perceptual judgments that rely on high-order image structures and are context-dependent, so it can be used to quantitatively evaluate the visual quality of the denoising results. Higher PSNR and SSIM, and lower LPIPS values indicate better performance.

A. Training Details and Parameter Settings

NIR-guided RGB denoising: is evaluated on the RGB-NIR Scene Dataset [54]. It consists of aligned RGB-NIR image pairs in 9 categories captured using modified digital single-lens reflex cameras. We randomly select 389 pairs for training and 45 pairs for testing. Each subset covers all 9 categories of images. We

TABLE VI
THE AVERAGE PSNR (DB), SSIM, AND LPIPS VALUES OF DIFFERENT ALGORITHMS ON IMAGES FROM THE FLASH AND AMBIENT ILLUMINATIONS DATASET IN THE CASES OF GAUSSIAN NOISE ($\sigma = 0.1;\ 0.2$) AND MIXED POISSON-GAUSSIAN NOISE ($\alpha = 0.02, \sigma = 0.2$)

Algorithms	$\sigma = 0.1$			$\sigma = 0.2$			$\alpha = 0.02$, $\sigma = 0.2$		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
ADNet [12]	34.15	0.9503	0.3070	30.92	0.9145	0.3758	30.15	0.9021	0.3952
MIRNet [29]	34.66	0.9613	0.3026	31.91	0.9301	0.3748	31.43	0.9255	0.3821
NBNet [30]	34.33	0.9530	0.3189	31.46	0.9253	0.3752	30.92	0.9193	0.3753
MPRNet [24]	34.73	0.9565	0.3134	32.02	0.9317	0.3662	31.55	0.9266	0.3719
HINet [58]	34.96	0.9577	0.3154	32.28	0.9344	0.3662	31.86	0.9304	0.3727
Uformer [31]	34.55	0.9555	0.3217	31.94	0.9303	0.3682	31.47	0.9253	0.3697
DGUNet [59]	34.72	0.9563	0.3074	32.02	0.9309	0.3499	31.51	0.9264	0.3550
NAFNet [60]	34.98	0.9575	0.3156	32.23	0.9339	0.3644	31.65	0.9297	0.3705
GF [17]	31.75	0.9356	0.2809	28.16	0.8918	0.3367	27.31	0.8918	0.3322
CFJR [39]	31.46	0.9222	**0.2538**	27.48	0.8779	0.3179	26.34	0.8574	0.3476
SVLRM [18]	34.56	0.9568	0.2652	32.04	0.9385	0.3190	31.48	0.9352	0.3301
CUNet [19]	34.82	0.9591	0.2761	32.44	0.9425	0.3016	31.93	0.9392	0.3060
DKN [42]	33.89	0.9479	0.2987	30.93	0.9143	0.3467	30.43	0.9072	0.3576
UMGF [20]	31.42	0.9299	0.3223	29.29	0.9047	0.3522	28.65	0.9000	0.3605
FGDNet-s	34.50	0.9591	0.2848	32.49	0.9431	0.3137	31.93	0.9405	0.3166
FGDNet-m	35.00	0.9606	0.2782	32.78	0.9451	0.3016	32.38	0.9427	0.3043
FGDNet	**35.33**	**0.9620**	0.2749	**33.08**	**0.9472**	**0.2921**	**32.68**	**0.9448**	**0.2940**

TABLE VII
THE AVERAGE PSNR (DB), SSIM, AND LPIPS VALUES OF DIFFERENT ALGORITHMS ON IMAGES FROM THE NYU V2 DATASET IN THE CASES OF GAUSSIAN NOISE ($\sigma = 0.1;\ 0.2$) AND MIXED POISSON-GAUSSIAN NOISE ($\alpha = 0.02, \sigma = 0.2$)

Algorithms	$\sigma = 0.1$			$\sigma = 0.2$			$\alpha = 0.02$, $\sigma = 0.2$		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
ADNet [12]	39.54	0.9876	0.2468	35.27	0.9701	0.4283	34.07	0.9601	0.4709
MIRNet [29]	40.66	0.9921	0.1774	37.06	0.9825	0.2217	36.44	0.9818	0.2291
NBNet [30]	40.55	0.9918	0.1696	37.01	0.9849	0.2533	36.34	0.9831	0.2827
MPRNet [24]	40.44	0.9922	0.1687	37.45	0.9865	0.2068	37.00	0.9856	0.2124
HINet [58]	40.40	0.9861	0.1811	37.12	0.9810	0.2107	36.59	0.9782	0.2142
Uformer [31]	40.37	0.9922	0.1696	37.24	0.9854	0.2326	36.76	0.9844	0.2547
DGUNet [59]	40.65	0.9922	0.1684	37.65	0.9853	0.2413	37.15	0.9852	0.2571
NAFNet [60]	40.88	0.9913	0.1716	37.35	0.9829	0.2451	36.82	0.9835	0.2673
GF [17]	37.20	0.9593	0.2469	34.21	0.9434	0.2626	34.07	0.9375	0.2890
CFJR [39]	34.57	0.8667	0.3793	31.94	0.8192	0.4222	31.20	0.7937	0.4430
SVLRM [18]	39.89	0.9890	0.1628	36.21	0.9800	0.2710	35.40	0.9783	0.2891
CUNet [19]	39.83	0.9886	0.1746	36.61	0.9796	0.2270	36.05	0.9778	0.2500
DKN [42]	40.14	0.9888	0.2102	36.57	0.9772	0.3824	35.94	0.9744	0.4176
UMGF [20]	39.69	0.9872	0.1781	36.29	0.9786	0.2244	35.70	0.9773	0.2338
FGDNet-s	40.69	0.9921	0.1638	37.42	0.9858	0.1901	36.74	0.9853	0.1943
FGDNet-m	40.64	0.9922	0.1603	37.55	0.9872	0.1875	36.92	0.9866	0.1924
FGDNet	**41.30**	**0.9929**	**0.1567**	**37.90**	**0.9877**	**0.1853**	**37.27**	**0.9870**	**0.1884**

crop the training set into 95,000 pairs of 128×128 patches as training data. The entire training stage takes $T = 100$ epochs.

Flash-guided no-flash denoising: is evaluated on the Flash and Ambient Illuminations Dataset [55] with image pairs in 6 categories. In our experiments, 1940 pairs are used for training while 558 pairs are used for testing. In the training stage, we randomly crop patches of size 128×128 from the input image pairs. It takes $T = 600$ epochs to finish the training process.

RGB-guided depth denoising: is conducted on the NYU v2 Dataset [56], where 1000 pairs of them are used for training and the remaining 449 ones are for evaluation. It also takes $T = 600$ epochs to accomplish training. Patches of size 128×128 are randomly cropped in each training epoch.

In this work, we focus on dealing with Poisson noise and Gaussian noise, which are two main noise types of modern camera sensors. Their intensities are indicated by α and σ, respectively. In each epoch, noisy patches are generated with random noise levels, with α ranging from 0 to 0.02 and σ ranging from 0 to 0.2. The network is trained using the Adam optimizer [57] with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate is initially set to 0.0003, and changed to 0.0001 after epoch $0.1 \cdot T$. For the last $0.7 \cdot T$ epochs, the learning rate is set to 0.00005. The weight decay is set to 0.0001, and the batch size is 24. We use 1 Nvidia GeForce GTX 2070 GPU to train the network.

After going through various parameter settings, we conclude that our algorithm is not parameter-sensitive. As shown in Table I, FGDNet obtains similar PSNR, SSIM and LPIPS values

Fig. 5. Denoising results of FGDNet with and without guidance image under mixed Poisson-Gaussian noise ($\alpha = 0.02, \sigma = 0.2$).

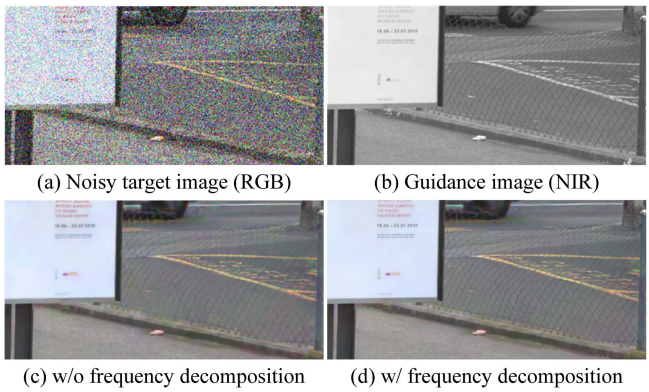


Fig. 7. Denoising results with and without frequency decomposition under mixed Poisson-Gaussian noise ($\alpha = 0.02, \sigma = 0.2$).

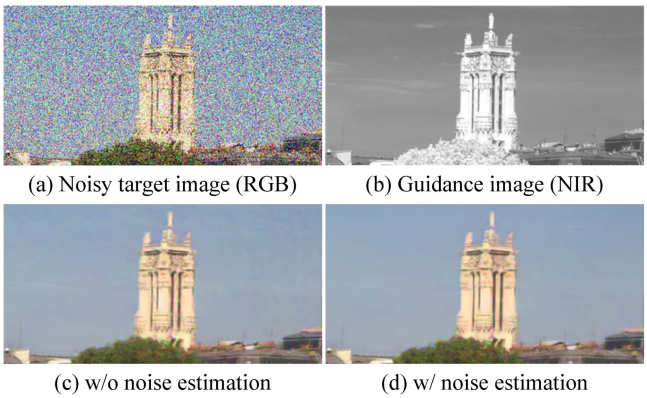


Fig. 6. Denoising results of FGDNet with and without noise estimation under mixed Poisson-Gaussian noise ($\alpha = 0.02, \sigma = 0.2$).

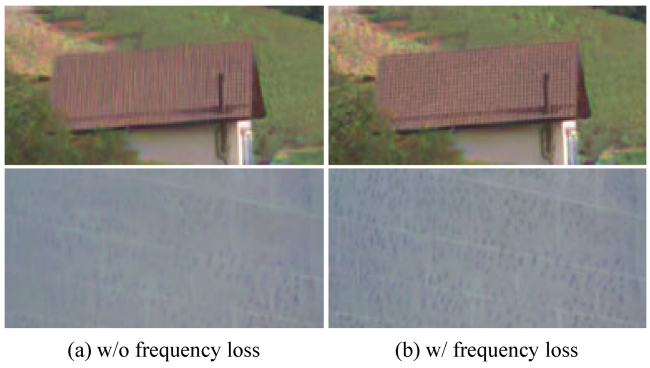


Fig. 8. Denoising results with and without frequency supervision under mixed Poisson-Gaussian noise ($\alpha = 0.02, \sigma = 0.2$).

under frequency decomposition with different patch sizes. Balancing denoising accuracy and model size, we set patch size $s = 7$. For loss function, we set $\lambda_n = 10$, $\lambda_{l1} = 0.5$, $\lambda_{l2} = 10$ and $\lambda_h = 2$.

B. Ablation Studies

In this section, we conduct ablation studies on image pairs from the RGB-NIR Scene Dataset [54] to validate the influence of guidance images, noise estimation, frequency decomposition, and frequency-relevant supervision to our algorithm.

Guidance image: To demonstrate the contribution of guidance images to denoising, we use the noisy target image to guide its own denoising process for comparison. Without changing the network architecture, the original guided denoising framework can be converted into a non-guided one. Table II shows that, in this case, the denoising accuracy is significantly decreased. Visual results in Fig. 5 show that, without additional guidance images, both edges and details are over-smoothed during noise removal.

Noise estimation: To show the necessity of noise estimation while keeping the same model size, we remove the noise estimation loss in the training stage so as not to force the output of the original noise estimation module to be noise. Table II demonstrates that the introduction of noise estimation contributes to higher denoising accuracy. As shown in Fig. 6, it can further reduce the splotched artifacts around flat areas.

To further show that an inaccurate noise estimation does not have a bad influence on denoising performance, we evaluate our proposed FGDNet in a more challenging situation. That is, we replace its backbone from the original modified ADNet (denoted as *NEst-A*) with a modified DnCNN [9] (denoted as *NEst-D*) which only contains three convolution blocks. By subtracting the estimated noise maps from the noisy target images, we can obtain the pre-denoised images. The similarity between the pre-denoised and the ground truth images can indicate the denoising abilities of the two noise estimators. As shown in Table III, the noise estimators only produce rough noise maps. However, even if the highly inaccurate estimated noise map obtained by *NEst-D* still contributes to a slight performance gain.

Frequency decomposition: Frequency-domain guided denoising is the core idea of our work. To evaluate its effectiveness, we conduct guided denoising without frequency decomposition for comparison. That is, the average weighting is computed in the spatial domain. To retain the same model size, we modify the frequency decomposition module and the spatial reconstruction module into two trainable convolution blocks with kernels of

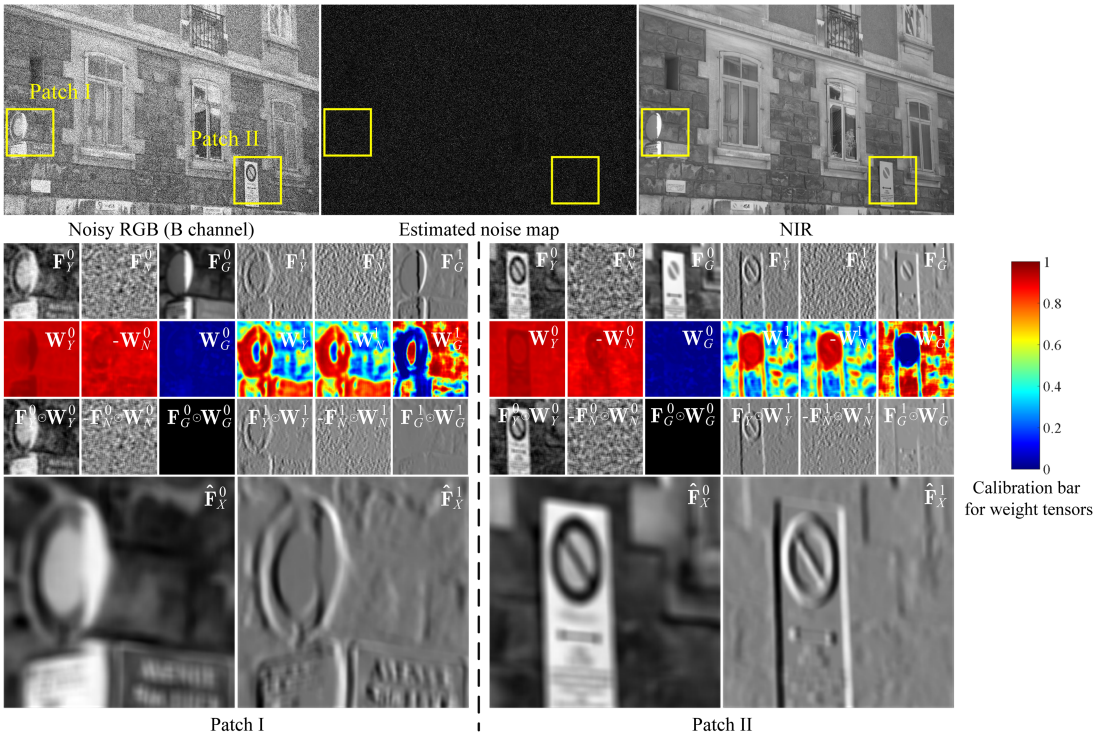


Fig. 9. Visualization of three frequency tensors, three weight tensors, and the restored frequency tensor on the zero frequency layer and the first high frequency layer. Frequency tensors are normalized between 0 and 1 for display, while weight tensors are displayed in pseudo color.

the same size. As shown in Table II, our frequency decomposition contributes to achieving higher accuracy. The visual results shown in Fig. 7 demonstrate that using frequency decomposition can not only restore sharper edges and finer details, but also alleviate artifacts around flat areas.

Frequency-relevant supervision: To demonstrate the contribution of frequency-domain supervision to guided denoising, we retrain our FGDNet without frequency loss. See Table II for the quantitative comparison. Both cases obtain similar PSNR and SSIM values. With frequency loss, our FGDNet can achieve lower LPIPS values, indicating better visual quality. As shown in Fig. 8, the frequency-relevant supervision helps guided denoising preserve richer and more accurate details.

C. Evaluation and Comparison

To demonstrate the contribution of each frequency tensor in regressing the restoration result, we visualize different frequency layers and their weight tensors in the NIR-guided RGB image denoising task. Denote $\psi = Y$, N, or G. As displayed in Fig. 9 , the zero frequency layer $\mathbf{F}^0_\psi$ records the base contents of the image. The weights of the noisy RGB image $\mathbf{W}^0_Y$ and the estimated noise map $-\mathbf{W}^0_N$ are close to each other with slight difference and also much larger than that of the NIR image $\mathbf{W}^0_G$. That is, the network implicitly takes the advantage of the input guidance image to further refine the denoising result without causing severe color deviation. As a result, the restored low-frequency contents are clean and spatially smooth.

In comparison, the high-frequency layers record structural information of the image. Taking the first one as an example, the restoration result is contributed by both input images. The guidance image plays the role of ensuring structural saliency and reducing unwanted high-frequency responses for the flat areas. Patch I and Patch II, as marked by the yellow boxes, display the structurally inconsistent contents between the input RGB and NIR images. When dealing with inconsistent structures, the weight values of the guidance image are close to zero to avoid transferring its unique contents to the denoising result. In the restored frequency layer $\hat{\mathbf{F}}^1_X = \mathbf{W}^1_Y \odot \mathbf{F}^1_Y + \mathbf{W}^1_N \odot \mathbf{F}^1_N + \mathbf{W}^1_G \odot \mathbf{F}^1_G$, the reconstructed structural information is salient and clean, and also faithful to the target image.

Further, we quantitatively evaluate our algorithm on three different guided denoising tasks: NIR-guided RGB image denoising, flash-guided no-flash image denoising, and RGB-guided depth image denoising. We also compare it to the state-of-the-art guided restoration algorithms including GF [17], CFJR [39], SVLRM [18], CUNet [19], DKN [42], and UMGF [20], as well as non-guided blind denoising algorithms including ADNet [12], MIRNet [29], NBNet [30], MPRNet [24], HINet [58], Uformer [31], DGUnet [59], and NAFNet [60]. For a fair comparison, we conduct an exhaustive search for the optimal parameter settings of GF and CFJR. The remaining comparative algorithms are learning-based ones. We retrain their network models using the official codes provided by the authors on the same training set as ours. We use an additional dataset that contains 3859 images provided by [12] to generate synthetic noisy/clean image pairs when training the non-guided denoising algorithms.

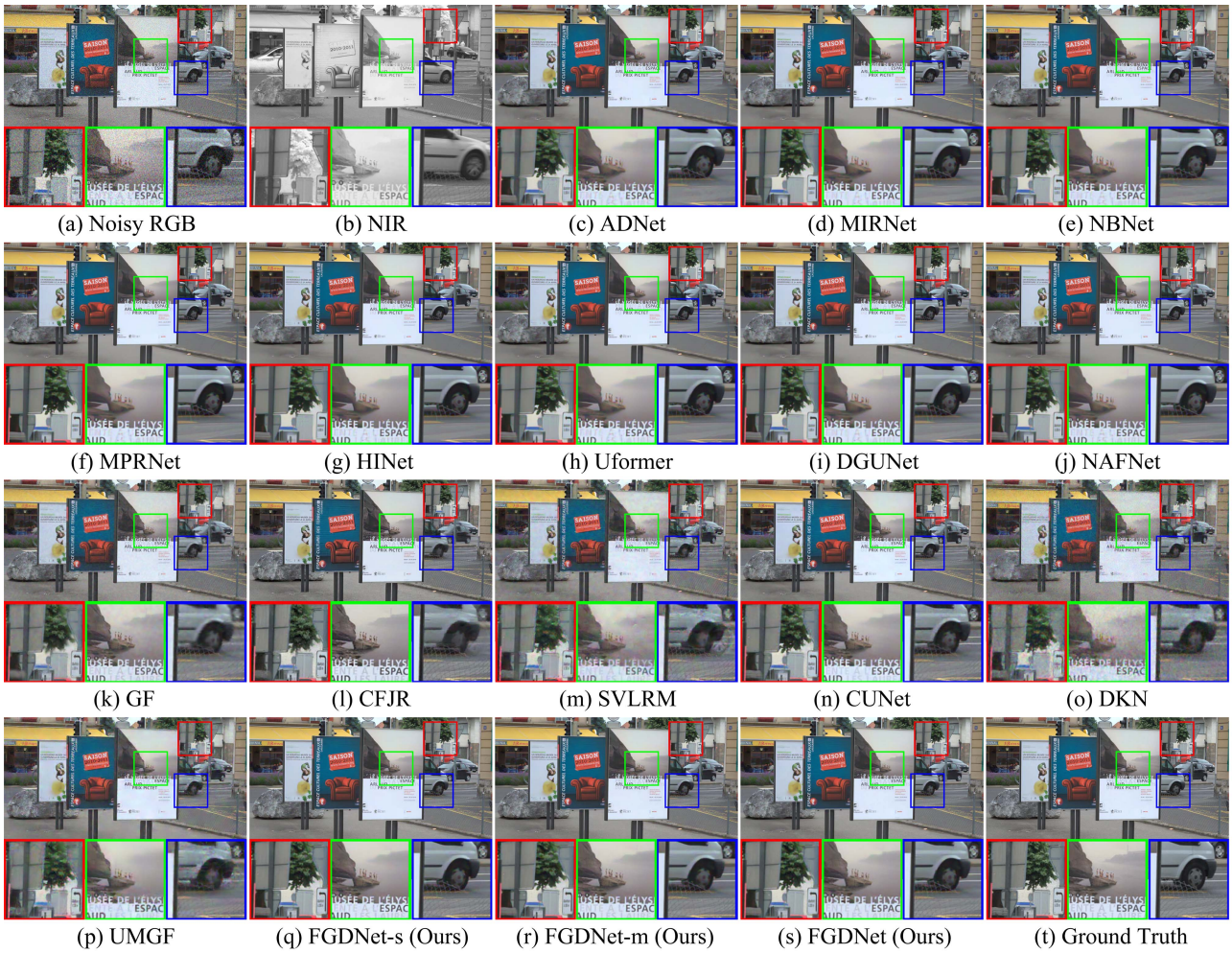


Fig. 10. Denoising results on the RGB-NIR Scene Dataset under Gaussian noise ($\sigma = 0.1$) obtained by the comparative non-guided and guided denoising methods, and our FGDNet with different model sizes.

The training parameters are exactly set according to the corresponding papers. For color image denoising, the comparative methods process each channel independently, which is the same as our FGDNet.

In addition, to show the influence of model size on the denoising accuracy, we conduct guided denoising using two alternative versions of FGDNet with different model sizes, i.e., FGDNet-m ($C_1 = 32, C_2 = 64$), and FGDNet-s ($C_1 = 24, C_2 = 48$). Their numbers of parameters, floating points per second (FLOPs), and inference time are listed in Table IV along with other comparative algorithms. Here, FLOPs and inference time are measured with input images of size 128×128. We can observe that the non-guided denoising models are basically much larger than the guided ones. In comparison, our FGDNet not only has smaller parameter numbers, but also has lower computational complexity and require less time to complete the denoising process. More specifically, our FGDNet spends 3.57 ms on noise estimation, and 8.32 ms on the remaining guided denoising step including frequency transformation. Here, frequency decomposition only takes about 0.13 ms.

NIR-guided RGB image denoising: is evaluated on the RGB-NIR Scene Dataset [54]. Table V lists the average PSNR, SSIM and LPIPS values on the test set. Among the competing algorithms, the non-guided approaches basically achieve higher PSNR and SSIM values than the guided ones. Here, PSNR is a per-pixel measure. SSIM evaluates similarity based on the local mean, variance, and covariance values, which are all statistical metrics. Hence, both PSNR and SSIM are sensitive to intensity changes, and can be employed to indicate whether noise is completely removed. Therefore, the non-guided denoisers have stronger noise removal abilities. In fact, the denoising accuracy can be affected by a series of complicated factors, such as the complexity of the network architecture, the size of the model, the training scheme, *etc.* These non-guided denoisers basically adopt more complicated network designs and larger model sizes. It's reasonable that they may have higher robustness when dealing with different noise levels. However, as discussed in [53], PSNR and SSIM cannot perceive high-order structures as in human judgments. A blurry output can also achieve high PSNR and SSIM values. As shown in Fig. 10, all non-guided models

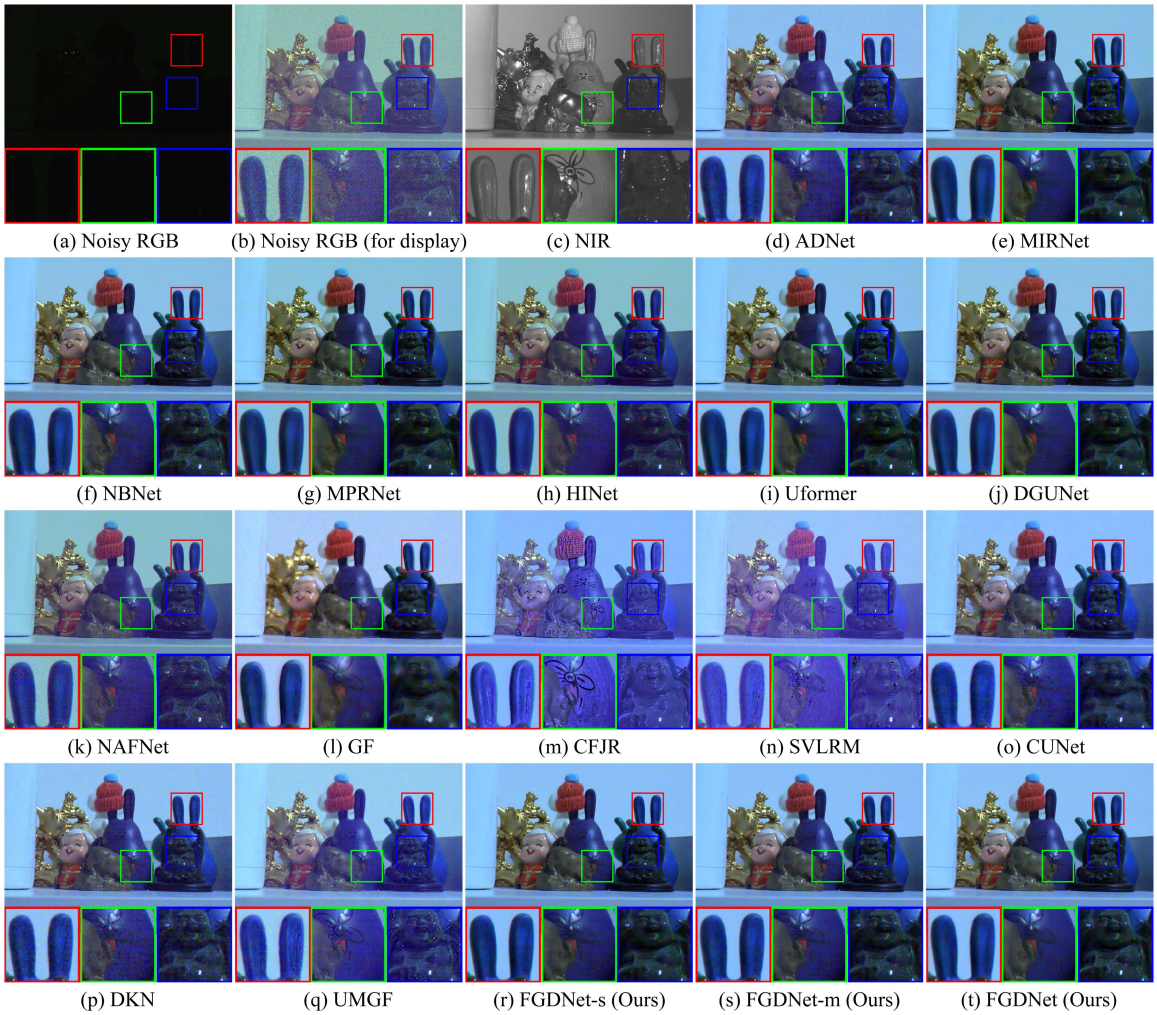


Fig. 11. Denoising results on the realistic RGB-NIR image pair obtained by the comparative non-guided and guided denoising methods, and our FGDNet with different model sizes.

produce very clean denoising results, but the detailed contents are over-smoothed at the same time.

Compared to non-guided denoisers, the superiority of competing guided denoising models including CFJR, SVLRM, CUNet, and UMGF is that they basically obtain lower LPIPS values, as listed in Table V, which indicates their better visual quality. The visual comparison in Fig. 10 demonstrates that, the superior visual quality lies in the fact that they can make the use of the additional structural information of the guidance image to restore the target image with richer details. However, when dealing with structurally inconsistent contents, the guided denoisers GF, CFJR, SVLRM and UMGF can produce ghosting artifacts. Different from the aforementioned guided denoising models, DKN exploits the guidance image only to compute the weights and offsets for a set of deformable kernels, and then uses them to filter the noisy target image to obtain the restoration result. It doesn't explicitly involve the guidance image into the restoration model, and therefore cannot make the best use of its structural information to restore clean images with salient structures.

In addition, the competing guided denoising algorithms including GF, SVLRM, and UMGF have a common drawback that they cannot remove noise completely, especially at high noise levels. Therefore, they basically obtain lower PSNR and SSIM values than the non-guided ones. More specifically, GF uses a local linear representation model of the guidance image to estimate the restored target image. Denote $\mathbf{y},\ \mathbf{g}$ as a pair of patches extracted from the input target and guidance images $\mathbf{Y}$ and $\mathbf{G}$ at the same position, respectively. The restored patch is computed by $\hat{\mathbf{x}} = a \cdot \mathbf{g} + b$, where $a = Cov(\mathbf{y}, \mathbf{g})/(v_g + \varepsilon)$ and $b = \overline{\mathbf{y}} - a \cdot \overline{\mathbf{g}}$. Here, $\overline{\mathbf{y}}$ and $\overline{\mathbf{g}}$ are the mean pixel values of $\mathbf{y}$ and $\mathbf{g}$, v_g is the variance of pixel values of patch $\mathbf{g}$, $Cov(\cdot)$ is the covariance function, and ε is a smoothness-related parameter. As discussed in [17], when $\mathbf{g}$ is flat, i.e., $v_g \approx 0 \ll \varepsilon$, we can obtain $a \approx 0$ and $b \approx \overline{\mathbf{y}}$. In other words, GF is equivalent to mean filtering in the flat areas, which is the reason for its limited denoising ability.

SVLRM predicts two linear coefficient maps $\mathbf{F}_a$ and $\mathbf{F}_b$ using a shared convolutional network and estimates the restored target image by $\hat{\mathbf{X}} = \mathbf{F}_a \odot \mathbf{G} + \mathbf{F}_b$. However, as analyzed in [20], it's

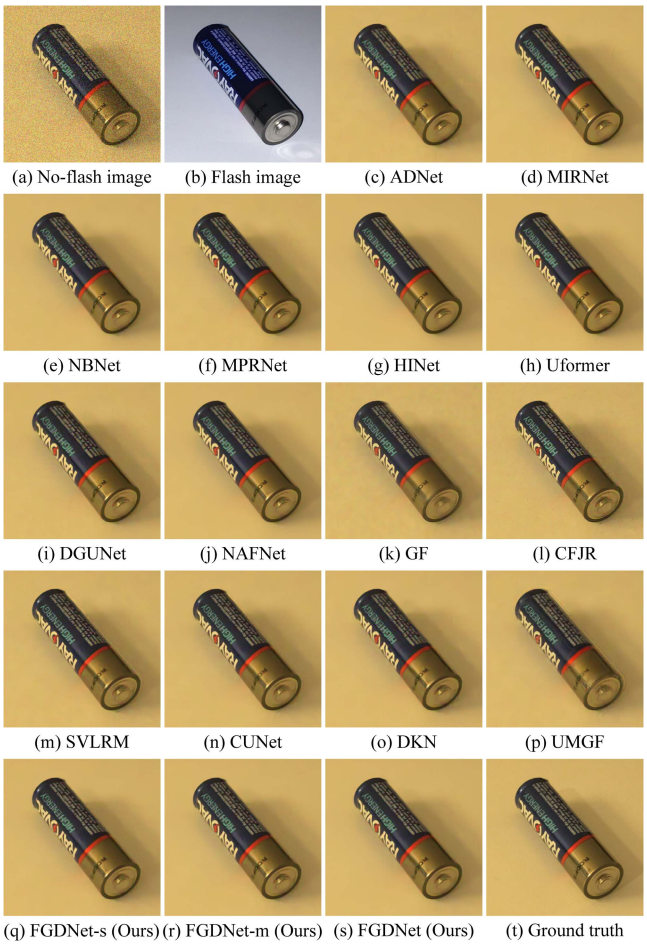


Fig. 12. Denoising results of flash/no-flash image pairs on the Flash and Ambient Illuminations Dataset ($\sigma = 0.1$).

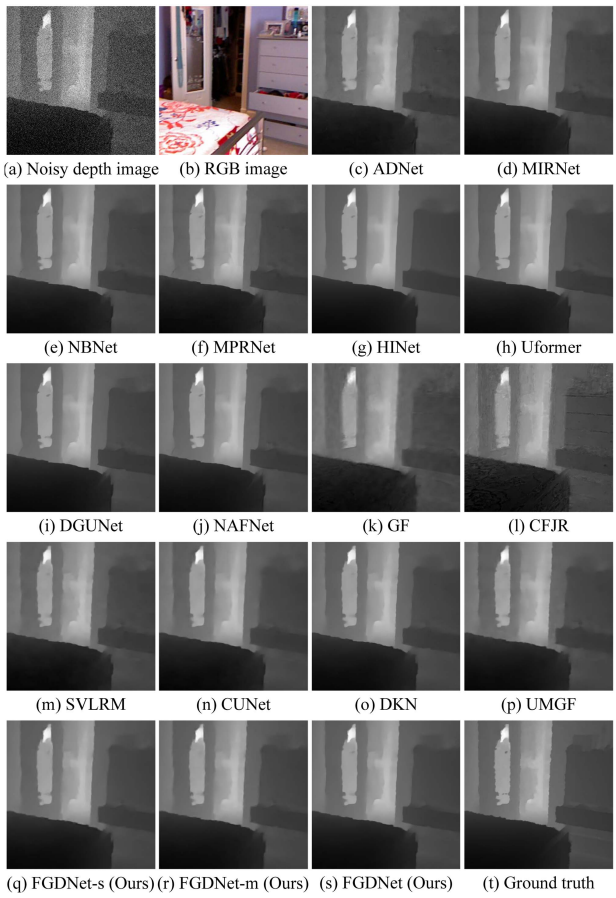


Fig. 13. Denoising results of depth/RGB image pairs on the NYU v2 Dataset ($\sigma = 0.1$).

actually difficult to disentangle the functions and the representations of $\mathbf{F}_a$ and $\mathbf{F}_b$ using a shared network, especially without corresponding supervisions. Hence, it cannot guarantee that $\mathbf{F}_a$ effectively accomplishes structure transfer from the guidance image and $\mathbf{F}_b$ retains the clean base components of the noisy target image since both of them are learned implicitly.

UMGF uses mean filtering to obtain the base layers $\mathbf{Y}_L$ and $\mathbf{G}_L$ of the noisy target image $\mathbf{Y}$ and the guidance image $\mathbf{G}$, respectively. Then, it computes the corresponding detail layers $\mathbf{Y}_m = \mathbf{Y} - \mathbf{Y}_L$ and $\mathbf{G}_m = \mathbf{G} - \mathbf{G}_L$. Here, UMGF directly uses $\mathbf{Y}_L$ as the base layer of the denoising result, and learns a linear coefficient map $\mathbf{F}$ to regress the restored detail layer by $\mathbf{F} \odot \mathbf{G}_m$. However, due to the limited denoising ability of mean filtering, in the training stage, the computation of $\mathbf{F} \odot \mathbf{G}_m$ not only has to complete the structure transfer, but also has to compensate for the inaccurate noise removal of the base layer. It's actually difficult to depend only on such a linear representation of the detail layer extracted from the guidance image to well balance the above two issues. Therefore, we can observe from the denoising results that, UMGF not only causes splotched artifacts but also slightly blurs the image structures.

In comparison, our FGDNet, even the smallest FGDNet-s, outperforms all guided and non-guided comparative methods with higher PSNR, SSIM and lower LPIPS values, showing that it achieves both the highest denoising accuracy and also the highest visual quality. Thanks to the explicit frequency decomposition, our FGDNet can simultaneously focus on the noise removal of low-frequency components and the structure reconstruction of high-frequency components. The noise estimation term can further ensure its robustness to different noise levels. The visual result in Fig. 10 shows that FGDNet not only can restore flat contents that are spatially smooth without splotched artifacts, but also reconstructs salient structures based on the guidance image. In addition, due to the data fidelity constraint in our restoration model, our FGDNet effectively avoids transferring inconsistent contents from the guidance image. As shown in Fig. 10, it can restore image structures faithfully to the noisy target input, even when severe inconsistencies occur, marked by the blue boxes.

We also evaluate our algorithm on the realistic RGB-NIR pairs obtained by our dual-camera system. As shown in Fig. 11, the image pair is captured in the low-light environment with additional NIR light provided by an NIR lamp. For clear illustration, the RGB input and the denoised images are processed with tone mapping. Similar to the restoration results on synthetic data, our algorithm can effectively remove noise and cause much fewer artifacts than the comparative methods. Non-guided denoising

methods tend to over-smooth the details. Highlights in the NIR image leave ghosting artifacts in the results obtained by guided denoising methods including GF, CFJR, SVLRM, and UMGF, while our algorithm is not affected.

Flash-guided no-flash image denoising: is validated on the Flash and Ambient Illuminations Dataset [55]. Table VI shows that our FGDNet basically achieves the highest accuracy under different noise levels. Visual results are displayed in Fig. 12. Similarly, all non-guided algorithms over-smooth the details. Guided denoising methods including GF, CFJR, SVLRM, CUNet and UMGF cause ghosting artifacts due to the shadows in the flash image. In comparison, our FGDNet not only removes noise completely, but also preserves fine structures faithfully to the target image.

RGB-guided depth image denoising: is validated on the NYU v2 Dataset [56]. Quantitative and visual results are shown in Table VII and Fig. 13, respectively. Compared with the state-of-the-art non-guided denoising algorithms, our FGDNet achieves higher denoising accuracy, and can recover depth-discontinuity pixels according to the guidance RGB image. Traditional guided denoising algorithms including GF and CFJR bring unwanted textures from the RGB image into the restored depth images. In comparison, our restored clean depth images have fewer artifacts and finer structures.

V. Conclusion

In this work, we propose a frequency-domain guided denoising algorithm to restore clean images from well-aligned image pairs. Different from common guided restoration approaches that are directly conducted on the image itself, we decompose the image pairs into various frequency layers and conduct guided denoising within each layer independently. To deal with the possible structural inconsistency problem between the image pair, we take into account both the representation model with respect to the guidance image and the structural fidelity to the target image. We construct an optimization function to explore its mathematical formulation, and solve it using a convolutional network, called as FGDNet. To restore flat contents without artifacts and also clear structures, we further introduce a frequency loss to conduct different supervision processes on the low-frequency layer and the high-frequency layers. Compared to the state-of-the-art approaches, our FGDNet can achieve higher accuracy and also better visual quality.

Acknowledgment

The authors would like to thank the anonymous reviewers and associate editor for their valuable comments.

References

[1] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, vol. 2, pp. 60–65.

[2] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[3] W. Dong, L. Zhang, G. Shi, and X. Li, "Nonlocally centralized sparse representation for image restoration," *IEEE Trans. Image Process.*, vol. 22, no. 4, pp. 1620–1630, Apr. 2013.

[4] W. Dong, G. Shi, and X. Li, "Nonlocal image restoration with bilateral variance estimation: A low-rank approach," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 700–711, Feb. 2013.

[5] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2862–2869.

[6] J. Xu, L. Zhang, and D. Zhang, "A trilateral weighted sparse coding scheme for real-world image denoising," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 20–36.

[7] Y. Hou et al., "NLH: A blind pixel-level non-local method for real-world image denoising," *IEEE Trans. Image Process.*, vol. 29, pp. 5121–5135, 2020.

[8] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[9] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[10] S. I. Cho and S.-J. Kang, "Gradient prior-aided CNN denoiser with separable convolution-based optimization of feature dimension," *IEEE Trans. Multimedia*, vol. 21, no. 2, pp. 484–493, Feb. 2019.

[11] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, "Toward convolutional blind denoising of real photographs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1712–1722.

[12] C. Tian et al., "Attention-guided CNN for image denoising," *Neural Netw.*, vol. 124, pp. 117–129, 2020.

[13] Y. Du, G. Han, Y. Tan, C. Xiao, and S. He, "Blind image denoising via dynamic dual learning," *IEEE Trans. Multimedia*, vol. 23, pp. 2139–2152, 2021.

[14] R. Ma, S. Li, B. Zhang, and Z. Li, "Towards fast and robust real image dnoising with attentive neural network and PID controller," *IEEE Trans. Multimedia*, vol. 24, pp. 2366–2377, 2022.

[15] J. Ma, C. Peng, X. Tian, and J. Jiang, "DBDnet: A deep boosting strategy for image denoising," *IEEE Trans. Multimedia*, vol. 24, pp. 3157–3168, 2022.

[16] C. Mou, J. Zhang, X. Fan, H. Liu, and R. Wang, "COLA-Net: Collaborative attention network for image restoration," *IEEE Trans. Multimedia*, vol. 24, pp. 1366–1377, 2022.

[17] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2010, pp. 1–14.

[18] J. Pan et al., "Spatially variant linear representation models for joint filtering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1702–1711.

[19] X. Deng and P. L. Dragotti, "Deep convolutional neural network for multi-modal image restoration and fusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3333–3348, Oct. 2021.

[20] Z. Shi, Y. Chen, E. Gavves, P. Mettes, and C. G. Snoek, "Unsharp mask guided filtering," *IEEE Trans. Image Process.*, vol. 30, pp. 7472–7485, 2021.

[21] X. Zhang, T. Sim, and X. Miao, "Enhancing photographs with near infrared images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[22] D. Kahneman, "Method, findings, and theory in studies of visual masking," *Psychol. Bull.*, vol. 70, no. 6pt.1, pp. 404–425, 1968.

[23] J. Wright et al., "Sparse representation for computer vision and pattern recognition," *IEEE Proc. IRE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.

[24] S. W. Zamir et al., "Multi-stage progressive image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14816–14826.

[25] N. Rahaman et al., "On the spectral bias of neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 5301–5310.

[26] D. Yin, R. G. Lopes, J. Shlens, E. D. Cubuk, and J. Gilmer, "A fourier perspective on model robustness in computer vision," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, vol. 32, pp. 13276–13286.

[27] R. A. Haddad et al., "A class of fast Gaussian binomial filters for speech and image processing," *IEEE Trans. Signal Process.*, vol. 39, no. 3, pp. 723–727, Mar. 1991.

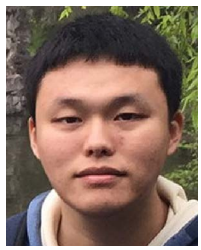
[28] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Trans. Image Process.*, vol. 11, no. 10, pp. 1141–1151, Oct. 2002.

[29] S. W. Zamir et al., "Learning enriched features for real image restoration and enhancement," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 492–511.

[30] S. Cheng et al., "NBNet: Noise basis learning for image denoising with subspace projection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4896–4906.

[31] Z. Wang et al., "Uformer: A general U-shaped transformer for image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17683–17693.

[32] C. Tian et al., "Deep learning on image denoising: An overview," *Neural Netw.*, vol. 131, pp. 251–275, 2020.
[33] S. Soltanayev and S. Y. Chun, "Training deep learning based denoisers without ground truth data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, vol. 31, pp. 3261–3271.
[34] A. Krull, T.-O. Buchholz, and F. Jug, "Noise2Void-learning denoising from single noisy images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2129–2137.
[35] J. Byun, S. Cha, and T. Moon, "FBI-Denoiser: Fast blind image denoiser for Poisson-Gaussian noise," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 5768–5777.
[36] W. Lee, S. Son, and K. M. Lee, "AP-BSN: Self-supervised denoising for real-world images via asymmetric PD and blind-spot network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17725–17734.
[37] Z. Li, J. Zheng, Z. Zhu, W. Yao, and S. Wu, "Weighted guided image filtering," *IEEE Trans. Image Process.*, vol. 24, no. 1, pp. 120–129, Jan. 2015.
[38] F. Kou, W. Chen, C. Wen, and Z. Li, "Gradient domain guided image filtering," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4528–4539, Nov. 2015.
[39] Q. Yan et al., "Cross-field joint image restoration via scale map," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1537–1544.
[40] Y. Li, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep joint image filtering," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2016, pp. 154–169.
[41] H. Wu, S. Zheng, J. Zhang, and K. Huang, "Fast end-to-end trainable guided filter," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1838–1847.
[42] B. Kim, J. Ponce, and B. Ham, "Deformable kernel networks for joint image filtering," *Int. J. Comput. Vis.*, vol. 129, no. 2, pp. 579–600, 2021.
[43] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-Laplacian priors," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, vol. 22, pp. 1033–1041.
[44] J. Kruse, C. Rother, and U. Schmidt, "Learning to push the limits of efficient FFT-based image deconvolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4586–4594.
[45] W. Zou et al., "SDWNet: A straight dilated network with wavelet transformation for image deblurring," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2021, pp. 1895–1904.
[46] X. Mao, Y. Liu, W. Shen, Q. Li, and Y. Wang, "Deep residual Fourier transformation for single image deblurring," 2021, *arXiv:2111.11745*.
[47] W. Xie et al., "Learning frequency-aware dynamic network for efficient super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 4308–4317.
[48] M. El Helou, R. Zhou, and S. Süsstrunk, "Stochastic frequency masking to improve super-resolution and denoising networks," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 749–766.
[49] P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo, "Multi-level wavelet-CNN for image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 773–782.
[50] H. Mo et al., "Frequency attention network: Blind noise removal for real images," in *Proc. Asian Conf. Comput. Vis.*, 2020, pp. 168–184.
[51] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, "Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data," *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1737–1754, Oct. 2008.
[52] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[53] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 586–595.
[54] M. Brown and S. Süsstrunk, "Multi-spectral SIFT for scene category recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 177–184.
[55] Y. Aksoy et al., "A dataset of flash and ambient illumination pairs from the crowd," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 634–649.
[56] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2012, pp. 746–760.
[57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.
[58] L. Chen, X. Lu, J. Zhang, X. Chu, and C. Chen, "HINet: Half instance normalization network for image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2021, pp. 182–192.
[59] C. Mou, Q. Wang, and J. Zhang, "Deep generalized unfolding networks for image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17399–17410.
[60] L. Chen, X. Chu, X. Zhang, and J. Sun, "Simple baselines for image restoration," in *Proc. Eur. Conf. Comput. Vis.*, 2022.



Zehua Sheng received the B.E. degree in 2017 from Zhejiang University, Hangzhou, China, where he is currently working toward the Ph.D. degree with the College of Information Science and Electronic Engineering, Zhejiang University. His research interests include image denoising and multimodal image processing.

Xiongwei Liu received the B.E. degree in 2020 from Zhejiang University, Hangzhou, China, where he is currently working toward the master's degree with the College of Information Science and Electronic Engineering, Zhejiang University. His research interests include multimodal image denoising and deep learning.

Si-Yuan Cao received the B.E. degree in electronic information engineering from Tianjin University, Tianjin, China, in 2016, and the Ph.D. degree in electronic science and technology from Zhejiang University, Hangzhou, China, in 2022. He is currently an Assistant Researcher with Ningbo Innovation Center, Zhejiang University. His research interests include multispectral/multimodal image registration, homography estimation, place recognition, and image processing.

Hui-Liang Shen received the B.E. and Ph.D. degrees in electronic engineering from Zhejiang University, Hangzhou, China, in 1996 and 2002, respectively. He was a Research Associate and Research Fellow with The Hong Kong Polytechnic University, Hong Kong, from 2001 to 2005. He is currently a Full Professor with the College of Information Science and Electronic Engineering, Zhejiang University. His research interests include multispectral imaging, image processing, computer vision, and machine learning.

Huaqi Zhang received the M.D. degree from Hangzhou Dianzi University, Hangzhou, China, in 2004. He is currentlywith the vivo Mobile Communication Company, Ltd. His main research interests include computer vision and machine learning.