Laplacian Gradient Consistency Prior for Flash Guided Non-Flash Image Denoising

Jingyi Xu, Xin Deng, *Member, IEEE*, Chenxiao Zhang, *Student Member, IEEE*, Shengxi Li, *Member, IEEE*, and Mai Xu, *Senior Member, IEEE*

***Abstract*— For flash guided non-flash image denoising, the main challenge is to explore the consistency prior between the two modalities. Most existing methods attempt to model the flash/non-flash consistency in pixel level, which may easily lead to blurred edges. Different from these methods, we have an important finding in this paper, which reveals that the modality gap between flash and non-flash images conforms to the Laplacian distribution in gradient domain. Based on this finding, we establish a Laplacian gradient consistency (LGC) model for flash guided non-flash image denoising. This model is demonstrated to have faster convergence speed and denoising accuracy than the traditional pixel consistency model. Through solving the LGC model, we further design a deep network namely LGCNet. Different from existing image denoising networks, each component of the LGCNet strictly matches the solution of LGC model, giving the network good interpretability. The performance of the proposed LGCNet is evaluated on three different flash/non-flash image datasets, which demonstrates its superior denoising performance over many state-of-the-art methods both quantitatively and qualitatively. The intermediate features are also visualized to verify the effectiveness of the Laplacian gradient consistency prior. The source codes are available at https://github.com/JingyiXu404/LGCNet.**

***Index Terms*— Guided image denoising, modality gap, interpretable network.**

I. Introduction

FLASH guided non-flash image denoising aims to remove the noise in non-flash image with the guidance of the corresponding flash image [1], [2], [3]. However, due to different imaging conditions, there exist many inconsistent features between flash and non-flash images, such as red eyes, hard shadows, and specular highlights [4], [5], [6]. That poses a serious challenge to the flash guided non-flash image denoising task. In order to improve the denoising performance, it is of significant importance to establish the suitable cross-modality relationship between the two image modalities, so that the consistent features can be fully utilized while the inconsistent features can be suppressed.

Traditional guided image denoising methods often rely on handcrafted heuristics [1], [2] or guided image filtering [3], [4], [7], [8], to address the inconsistency between flash and non-flash images. Specifically, Petschnigg et al. [1] performed pre-processing operations like shadow masking and red-eye correction, before transferring details from flash image to the non-flash one. Guo et al. [4] proposed a mutually guided image filter to jointly preserve cross-modality structures, to avoid the misleading structures in the flash image. However, these model based methods are usually time-consuming and the hand-crafted features may not be adaptable to complex real-world situations. To overcome this drawback, some deep learning based methods have been proposed to preserve consistent structures while preventing misleading discrepancies between flash and non-flash images [5], [6], [9], [10]. For instance, Deng and Dragotti [9] split common and unique features from two modalities, and only utilized the common features in flash modality to assist the denoising of non-flash images. Oh et al. [6] proposed to infer the consistent flash image patches through network, and reconstructed the clean non-flash image from these consistent patches with learned combination weights. However, these works established the consistency model between flash and non-flash images from the pixel perspective, making them sensitive to mutagenic pixels and difficult to preserve sharp edges.

To address these issues, in this paper, we provide a new perspective to deal with the inconsistency between flash and non-flash image pairs. Instead of the constraint in pixel-level, we first find the consistency constraint between flash and non-flash images in gradient level, i.e., the gradient modality gap between flash and non-flash images conforms to the Laplacian distribution. Based on this statistical finding, we establish a novel Laplacian gradient consistency (LGC) model for flash guided non-flash image denoising task. In this model, two important elements are considered, i.e., the gradient dependency between the flash and non-flash image modalities, as well as the noise difference between the noisy non-flash input and the clean non-flash output. In this way, we are able to effectively utilize consistent information in flash image to guide the restoration of non-flash image, while avoiding the interference of inconsistent information. The proposed LGC model is solved through the alternating direction method of multipliers (ADMM) algorithm [11], leading to a set of

Received 28 May 2024; revised 11 October 2024; accepted 27 October 2024. Date of publication 7 November 2024; date of current version 11 November 2024. This work was supported in part by NSFC under Grant 62372024, Grant 62250001, Grant 62231002, and Grant 62206011; and in part by the Young Elite Scientists Sponsorship Program by China Association for Science and Technology (CAST) under Grant 2022QNRC001. The associate editor coordinating the review of this article and approving it for publication was Dr. Lizhi Wang. *(Corresponding author: Xin Deng.)*

Jingyi Xu, Xin Deng, Shengxi Li, and Mai Xu are with the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China (e-mail: jingyixu@buaa.edu.cn; cindydeng@buaa.edu.cn; ShengxiLi2014@gmail.com; MaiXu@buaa.edu.cn).

Chenxiao Zhang is with the School of Cyber Science and Technology, Beihang University, Beijing 100191, China (e-mail: sy2339221@buaa.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TIP.2024.3489275, provided by the authors.

Digital Object Identifier 10.1109/TIP.2024.3489275

iterative solution steps. For learning efficiency, we further unfold these iterative solution steps into a learnable neural network named LGCNet. The experimental results on various datasets show that the proposed LGCNet significantly advances the flash guided non-flash image denoising performance. The main contributions of this paper are summarized as follows,

- We have an interesting finding which first reveals that the gradient modality gap between flash and non-flash image pair conforms to the Laplacian distribution.
- Based on the statistical finding, we propose a novel Laplacian gradient consistency (LGC) model for flash guided non-flash image denoising.
- We design an interpretable guided image denoising network named LGCNet, by mapping the solution of LGC model into learnable network modules.

II. Related Work

A. Single Image Denoising

Image denoising has been well explored for decades as one of the most fundamental problems in low-level image processing. The existing image denoising methods can be broadly classified into two categories, the traditional model based methods and the recent deep learning based methods. The early image denoising methods are based on on techniques like spatial or transform domain filtering [12], [13], [14], [15], sparse representation [16], [17], [18], [19] and low-rank factorization [20], [21], [22]. Later, Efros and Leung [23] demonstrated that the inherent self-similarity of natural images is beneficial for texture restoration. Following this insight, the non-local strategy has been widely explored in image denoising task [24], [25], [26], [27]. For example, Dabov et al. [25] proposed a classical block matching 3D collaborative filtering (BM3D) approach through combining similar image patches for non-local grouping, which achieved significant improvement in image denoising. However, these model based methods are often computationally heavy, which makes them not suitable for practical applications.

Recently, many deep learning based works have demonstrated great capability in addressing inverse problems, particularly through the integration of model-based optimization and learning-driven network training [28], [29], [30], [31]. For example, [28] learned a dynamic kernel prior based on Markov Chain Monte Carlo sampling, which is subsequently used to optimize a blur kernel estimation network. Specially, in the context of noise removing task, several methods have combined model priors with neural networks through different unfolding strategies [32], [33], [34], [35]. For instance, Simon and Elad [32] applied the convolutional sparse coding (CSC) model to natural image denoising from a Bayesian perspective. Then they used ISTA [36] unfolding algorithm to solve the model, and designed a network through stride convolutions. Ren et al. [35] proposed an enhanced latent space model for real-world image denoising, and solved it by the self-correction alternative optimization algorithm. A denoising network was then developed based on the algorithm for image denoising. However, single image input is limited by its restricted information, especially at high noise levels. Therefore, in some certain cases, other image modalities are introduced as guidance to provide more complete information during denoising process, such as flash guided non-flash image denoising.

B. Flash Guided Non-Flash Image Denoising

Flash photography enables the capture of images without noise in low-light conditions, which are commonly used as reference images for ambient (non-flash) image denoising. For example, Petschnigg et al. [1] and Eisemann and Durand [2] introduced additional information guidance from flash images for noise removal in the non-flash images. Inspired by the pioneering works, many model based methods [4], [37], [38] have been proposed to improve the denoising performance through considering the cross-modality correlation between flash and non-flash modalities. Particularly, Yan et al. [38] constructed an optimal scale map on basis of adaptive smoothing, edge preservation, and guidance strength between flash and non-flash images. The scale map is then employed to restore the target image in an iterative way.

In recent years, several works have been proposed to use deep neural networks to solve the flash guided non-flash image denoising problem. For example, Li et al. [39] proposed to use two encoders to separately extract features from flash and non-flash images, and selectively combined these features to reconstruct the target high-quality non-flash image. Following [39], many deep learning based methods have been proposed by employing the two-branch feature extraction network, but with different fusion strategies [40], [41], [42], for flash guided non-flash image denoising. However, these methods may overlook the redundant and misleading association between flash and non-flash modalities, which can impact the denoising performance, especially in cases where significant modality differences exist between flash and non-flash image pairs.

In order to deal with misleading information contained in flash images, some methods have been proposed to tackle the modal inconsistency in either local or global manners [5], [6], [9], [10], [43]. For instance, Deng and Dragotti [9] proposed to separate the common and unique features between the two modalities, and only utilized the common feature in flash images as consistent guidance to assist the denoising of non-flash images. Xu et al. [5] employed Gaussian distribution to represent the consistent part in flash and non-flash image pairs, while utilizing non-Gaussian distribution to represent the inconsistent part. They then developed a mixture network (MN) to simultaneously deal with both cases. Oh et al. [6] locally inferred consistent flash image patches through learned filters, which were then used as denoising guidance to prevent misleading information in the flash image. Consistency priors have been effectively applied in range of image processing tasks [44], [45]. For example, Lin et al. [44] leveraged the consistency of gradient direction and magnitude within intra-modality local regions to enhance interesting point detection, resulting in significant improvements in corner and blob identification. Different from these works, in this paper, we have an important observation to reveal the modality consistency

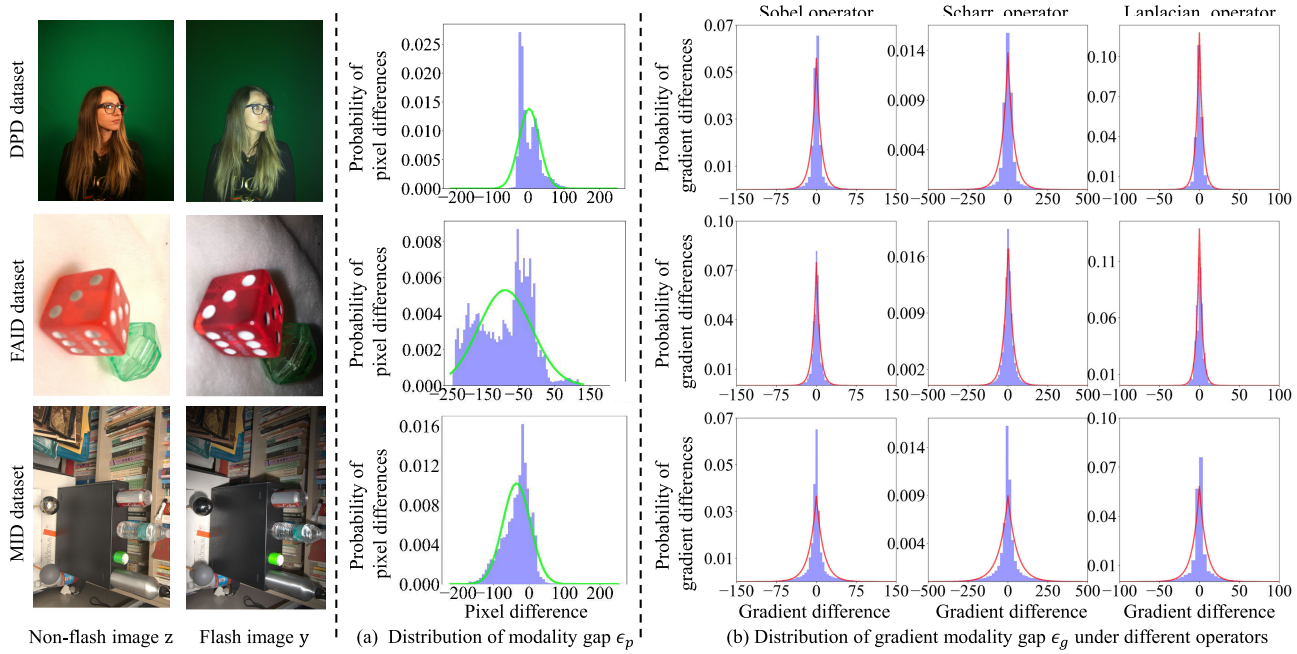


Fig. 1. Illustration of modality gap distribution between non-flash and flash image pairs on DPD, FAID and MID datasets. (a) distribution of the pixel modality gap $\boldsymbol{\epsilon_p}$ and its corresponding fitted Gaussian distribution represented by the green curves, (b) distribution of the gradient modality gap $\boldsymbol{\epsilon_g}$ under different operators and their corresponding fitted Laplacian distribution represented by the red curves.

in gradient domain, and propose a new flash guided non-flash image denoising method by utilizing the gradient consistency prior.

III. Proposed Method

The problem of flash guided non-flash image denoising can be stated as follows: Given a noisy non-flash image $\boldsymbol{x} \in \mathbb{R}^{h\times n}$, and a clean flash image $\boldsymbol{y} \in \mathbb{R}^{h\times n}$, we aim to restore a clean version of $\boldsymbol{x}$ with the guidance of $\boldsymbol{y}$. Here, we use symbol $\boldsymbol{z} \in \mathbb{R}^{h\times n}$ to represent the restored clean version of the non-flash image $\boldsymbol{x}$. In most existing works, the noise in $\boldsymbol{x}$ and the difference between $\boldsymbol{y}$ and $\boldsymbol{z}$ are assumed to follow Gaussian distribution, and the flash guided non-flash image denoising task can be cast into the following optimization problem [5]:

$$\underset{z}{\operatorname{argmin}} \frac{\lambda_1}{2}\|z-\boldsymbol{x}\|_2^2 + \frac{\lambda_2}{2}\|z-\boldsymbol{y}\|_2^2 + \gamma g(z). \tag{1}$$

Here, $g(\cdot)$ is a penalty imposed on $\boldsymbol{z}$ to constrain the effect of denoising, γ is the corresponding regularization parameter. λ_1 and λ_2 are regularization parameters to adjust the importance of $\boldsymbol{x}$ and $\boldsymbol{y}$ to restore $\boldsymbol{z}$. However, the local brightness inconsistency can easily occur, due to different object materials and shapes. Therefore, the distribution of the pixel difference $\boldsymbol{\epsilon_p} = \boldsymbol{z} - \boldsymbol{y}$ may not conform to the Gaussian distribution. As shown in Fig. 1 (a), the distribution of $\boldsymbol{\epsilon_p}$ can be asymmetric, heavy-tailed or with multiple peaks, which is difficult to be modelled though a simple Gaussian distribution. To solve this problem, it is necessary to establish a model that is more consistent with the distribution gap between flash and non-flash images. In this paper, we have an important finding which reveals that the gradient modality gap between flash and non-flash images conforms to the Laplacian distribution. Based on this finding, we further propose a Laplacian gradient consistency (LGC) model for flash guided non-flash image denoising. Next, we introduce this finding in details.

A. Finding: The Gradient Modality Gap Between Flash and Non-Flash Image Pair Conforms to Laplacian Distribution

To verify this finding, we conduct the experiment on a total of more than 1,800 image pairs from three flash and non-flash datasets, including the Flash and Ambient Illuminations Dataset (FAID) [46], the Multi-Illumination Dataset (MID) [47] and the DeepFlash Portrait Dataset (DPD) [48]. Specifically, we first transform the colored flash and non-flash image pairs into grayscale images. After that, three edge detection operators, i.e., Sobel et al. [49], Scharr [50], and Laplacian operators [51], are convoluted with both flash and non-flash images to obtain the gradient image pairs in a consistent manner. The gradient modality gap $\boldsymbol{\epsilon_g}$ between flash and non-flash images is then obtained by calculating the difference between the gradient image pairs. After that, the distribution of $\boldsymbol{\epsilon_g}$ is modelled as the Laplacian distribution through data fitting.

To evaluate the fitting accuracy, we employ the Kolmogorov-Smirnov test [52] to evaluate the consistency between $\boldsymbol{\epsilon_g}$ and the corresponding fitted Laplacian distribution. According to the universally used threshold [53], when the P-value of the Kolmogorov-Smirnov test exceeds 0.05, the two distributions can be considered to be close to each other. As presented in Table I, the P-values are larger than 0.05 on all the three datasets with different gradient operators, indicating the high modelling accuracy of the Laplacian distribution. In contrast, the P-values between $\boldsymbol{\epsilon_g}$ and the fitted Gaussian distribution are much lower than 0.05, suggesting that the Laplacian distribution provides a superior fitting capability compared to the Gaussian distribution. Moreover, we also compare the standard deviation of $\boldsymbol{\epsilon_g}$ and $\boldsymbol{\epsilon_p}$ on all samples in each dataset. As shown in Table II, the standard deviation of gradient difference $\boldsymbol{\epsilon_g}$ is way smaller than that of pixel difference $\boldsymbol{\epsilon_p}$. Specifically, the standard deviation of $\boldsymbol{\epsilon_g}$ with

TABLE I

THE AVERAGE P-VALUE OF THE KOLMOGOROV-SMIRNOV TEST BETWEEN ϵ_g AND THE FITTED LAPLACIAN OR GAUSSIAN DISTRIBUTION, RESPECTIVELY ON FAID, MID AND DPD DATASETS

Datasets	Fitted Laplacian Distribution			Fitted Gaussian Distribution		
	Sobel	Scharr	Laplacian	Sobel	Scharr	Laplacian
DPD	0.2002	0.1836	0.2046	0.0050	0.0042	0.0063
FAID	0.0879	0.1247	0.1283	0.0023	0.0027	0.0028
MID	0.0925	0.1210	0.1142	0.0056	0.0073	0.0124

TABLE II

THE COMPARISON BETWEEN THE STANDARD DEVIATION OF ϵ_g AND ϵ_p ON FAID, MID AND DPD DATASETS

Datasets	ϵ_p	Sobel ϵ_g	Scharr ϵ_g	Laplacian ϵ_g
DPD	0.1928	0.0257	0.0250	0.0183
FAID	0.2393	0.0627	0.0612	0.0974
MID	0.2099	0.0468	0.0454	0.0318

Sobel operator is 7.5×, 3.8× and 4.5× smaller than that of ϵ_p on DPD, FAID and MID datasets, respectively. These results indicate that the distribution of gradient difference ϵ_g is tighter than that of pixel difference ϵ_p, which brings more benefits to correlate the two modalities for guided image denoising.

Furthermore, we also visualize distribution difference between ϵ_p and ϵ_g in Fig. 1. As can be seen, the distribution of the pixel difference ϵ_p varies significantly from one image to another. In contrast, the distribution of the gradient difference ϵ_g is more regular across different images with different gradient operators, which is well suited with the Laplacian distribution curve. These results indicate that compared to pixel domain, the gradient domain modelling can bring more consistency between flash and non-flash modalities, which is beneficial for the guided denoising performance.

B. The Laplacian Gradient Consistency (LGC) Model

Based on the finding established in Section III-A, we model the relationship between the noisy non-flash image $\boldsymbol{x}$, clean flash image $\boldsymbol{y}$, and the target clean non-flash image $\boldsymbol{z}$ as follows,

$$\boldsymbol{x} = \boldsymbol{z} + \boldsymbol{\epsilon_n}, \tag{2a}$$

$$\boldsymbol{s} * \boldsymbol{y} = \boldsymbol{s} * \boldsymbol{z} + \boldsymbol{\epsilon_g}, \tag{2b}$$

where $\boldsymbol{s}$ represents the gradient operator, $\boldsymbol{\epsilon_n}$ denotes the noise, $\boldsymbol{\epsilon_g}$ denotes the gradient modality gap between flash and non-flash images following Laplacian distribution. To restore the clean image $\boldsymbol{z}$, we can cast the gradient consistency model in Eq. (2) into the following optimization problem,

$$\underset{z}{\operatorname{argmin}} \frac{\lambda_1}{2}\|\boldsymbol{z}-\boldsymbol{x}\|_2^2 + \rho\|\boldsymbol{s}*\boldsymbol{z}-\boldsymbol{s}*\boldsymbol{y}\|_1^1 + \gamma g(\boldsymbol{z}). \tag{3}$$

Here, $\|\cdot\|_1^1$ represents the ℓ_1 norm. λ_1, ρ and γ are regularization parameters. Due to the nonlinearity of $g(\boldsymbol{z})$, Eq. (3) does not have a closed-form solution. To solve this problem, we apply the alternating direction method of multipliers (ADMM) algorithm [11], to obtain the following optimization form:

$$\underset{\boldsymbol{z},\boldsymbol{a},\boldsymbol{m}}{\operatorname{argmin}} \frac{\lambda_1}{2}\|\boldsymbol{z}-\boldsymbol{x}\|_2^2 + \rho\|\boldsymbol{a}\|_1^1 + \gamma g(\boldsymbol{m}),$$
$$s.t. \quad \boldsymbol{a} = \boldsymbol{s}*\boldsymbol{z}-\boldsymbol{s}*\boldsymbol{y}, \, and \, \boldsymbol{m} = \boldsymbol{z}, \tag{4}$$

where $\boldsymbol{m}$ is an auxiliary variable decoupling the denoising penalty $g(\cdot)$ from the modality constraints, $\boldsymbol{a}$ is an auxiliary variable decoupling the gradient modality gap constraint $\|\boldsymbol{s}*\boldsymbol{z}-\boldsymbol{s}*\boldsymbol{y}\|_1^1$ from the noise constraint $\|\boldsymbol{z}-\boldsymbol{x}\|_2^2$. Then the augmented Lagrangian version of Eq. (4) is written as follows,

$$\begin{aligned}\underset{\boldsymbol{z},\boldsymbol{a},\boldsymbol{m},\boldsymbol{w},\boldsymbol{u}}{\operatorname{argmin}} & \frac{\lambda_1}{2}\|\boldsymbol{z}-\boldsymbol{x}\|_2^2 + \rho\|\boldsymbol{a}\|_1^1 + \gamma g(\boldsymbol{m}) \\ & + \frac{\beta}{2}\|\boldsymbol{s}*\boldsymbol{z}-\boldsymbol{s}*\boldsymbol{y}-\boldsymbol{a}+\boldsymbol{w}\|_2^2 - \frac{\beta}{2}\|\boldsymbol{w}\|_2^2 \\ & + \frac{\eta}{2}\|\boldsymbol{z}-\boldsymbol{m}+\boldsymbol{u}\|_2^2 - \frac{\eta}{2}\|\boldsymbol{u}\|_2^2. \end{aligned} \tag{5}$$

In Eq. (5), $\boldsymbol{w}$ and $\boldsymbol{u}$ are Lagrangian multipliers. Both β and η are regularization parameters. According to the principle of alternating optimization, Eq. (5) can be split into five iteratively updated subproblems, with the $(j+1)$-th iteration as follows:

$$\begin{aligned}\boldsymbol{z}^{(j+1)} = & \underset{z}{\operatorname{argmin}} \frac{\lambda_1}{2}\|\boldsymbol{z}-\boldsymbol{x}\|_2^2 + \frac{\eta}{2}\|\boldsymbol{z}-\boldsymbol{m}^{(j)}+\boldsymbol{u}^{(j)}\|_2^2 \\ & + \frac{\beta}{2}\|\boldsymbol{s}*\boldsymbol{z}-\boldsymbol{s}*\boldsymbol{y}-\boldsymbol{a}^{(j)}+\boldsymbol{w}^{(j)}\|_2^2, \end{aligned} \tag{6a}$$

$$\boldsymbol{a}^{(j+1)} = \underset{\boldsymbol{a}}{\operatorname{argmin}} \rho\|\boldsymbol{a}\|_1^1 + \frac{\beta}{2}\|\boldsymbol{s}*\boldsymbol{z}^{(j+1)}-\boldsymbol{s}*\boldsymbol{y}-\boldsymbol{a}+\boldsymbol{w}^{(j)}\|_2^2, \tag{6b}$$

$$\boldsymbol{w}^{(j+1)} = \underset{\boldsymbol{w}}{\operatorname{argmin}} \frac{\beta}{2}\|\boldsymbol{s}*\boldsymbol{z}^{(j+1)}-\boldsymbol{s}*\boldsymbol{y}-\boldsymbol{a}^{(j+1)}+\boldsymbol{w}\|_2^2 - \frac{\beta}{2}\|\boldsymbol{w}\|_2^2, \tag{6c}$$

$$\boldsymbol{m}^{(j+1)} = \underset{\boldsymbol{m}}{\operatorname{argmin}} \gamma g(\boldsymbol{m}) + \frac{\eta}{2}\|\boldsymbol{z}^{(j+1)}-\boldsymbol{m}+\boldsymbol{u}^{(j)}\|_2^2, \tag{6d}$$

$$\boldsymbol{u}^{(j+1)} = \underset{\boldsymbol{u}}{\operatorname{argmin}} \frac{\eta}{2}\|\boldsymbol{z}^{(j+1)}-\boldsymbol{m}^{(j+1)}+\boldsymbol{u}\|_2^2 - \frac{\eta}{2}\|\boldsymbol{u}\|_2^2. \tag{6e}$$

These five sub-problems should be solved iteratively until convergence to generate the clean non-flash image $\boldsymbol{z}$. This process can be very time-consuming. In this paper, we propose to map these solution steps into a learnable network to speed up the solving process. Next, we will introduce how the network is designed by solving the five sub-problems.

C. The Proposed LGCNet

Based on the LGC model in Section III-C, we design a network named LGCNet to solve the flash guided non-flash image denoising task. As shown in Fig. 2, the LGCNet is composed of one Initialization Module (INM) and J Guided Denoising Modules (GDM), with each GDM corresponding to one iteration in the LGC model. In each GDM, we have a Regularization Parameter Predicting (RPP) unit, a Non-Flash Restoration (NFR) unit, an Auxiliary Variable Updating (AVU) unit and a Denoising Penalty Updating (DPU) unit. Next, we will introduce the detailed structures of these units.

1) The Initialization Module (INM): The initialization module (INM) is used to initialize the parameters $\boldsymbol{z}^{(0)}$, $\boldsymbol{a}^{(0)}$, $\boldsymbol{w}^{(0)}$, $\boldsymbol{m}^{(0)}$ and $\boldsymbol{u}^{(0)}$. The architecture of INM module is shown in Fig. 3 (a). It consists of two convolutional layers with ReLU activation, with noisy non-flash image $\boldsymbol{x}$ and the corresponding flash image $\boldsymbol{y}$ as inputs.

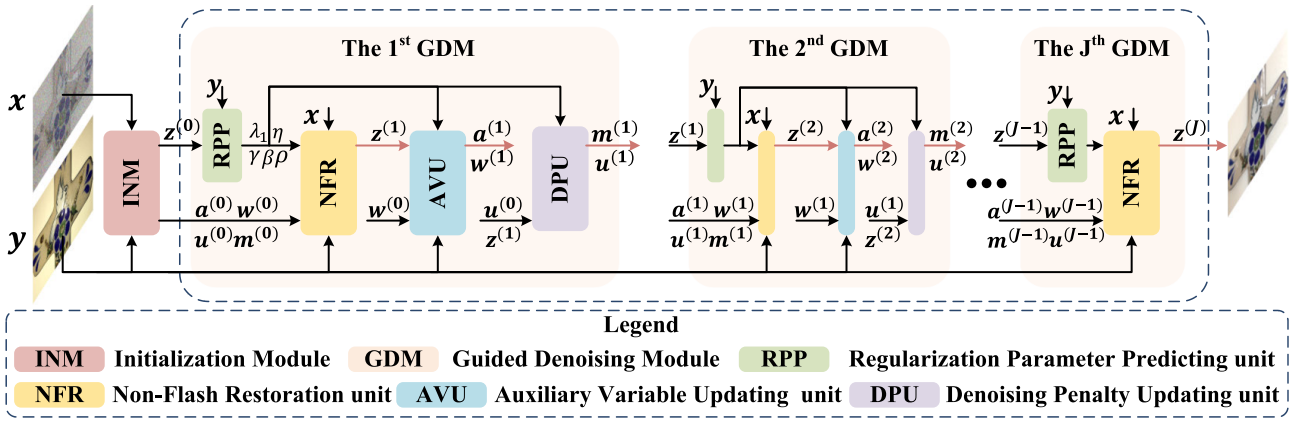


Fig. 2. The framework of the proposed LGCNet for flash guided non-flash image denoising. The network is composed of one INM module for initialization and several GDM modules for restoration. Each GDM module consists of one RPP unit for calculating regularization parameters, one NFR unit for updating z, one AVU unit for updating a and w, and one DPU unit for updating m and u. The pink arrows highlight the updated variables for each GDM module while the black arrows represent other internal data flows.

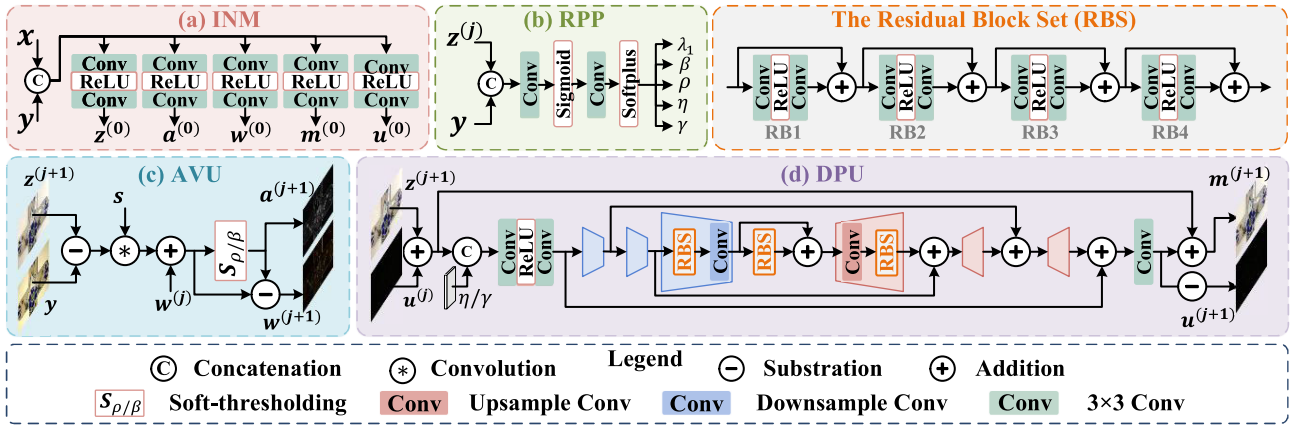


Fig. 3. The network architecture of (a) the initialization module (INM), (b) the regularization parameter predicting (RPP) unit, (c) the auxiliary variable updating (AVU) unit, and (d) the denoising penalty updating (DPU) unit.

2) The Regularization Parameter Predicting (RPP) Unit: The regularization parameters $\lambda_1, \rho, \gamma, \beta, \eta$ can affect the denoising performance of the proposed algorithm. Thus, we design a RPP unit to predict these hyper-parameters. In the $(j+1)$-th GDM module, the inputs to the RPP unit are the previously updated non-flash image $\boldsymbol{z}^{(j)}$ and the original flash image $\boldsymbol{y}$. Each RPP unit has two convolution layers, with Sigmoid and Softplus activations, respectively. The architecture of the RPP unit is shown in Fig. 3 (b).

3) The Non-Flash Restoration (NFR) Unit: The NFR unit corresponds to the solution of Eq. (6a), which aims to update the non-flash image $\boldsymbol{z}^{(j+1)}$. In order to solve Eq. (6a), we apply Fast Fourier Transform (FFT) to the variables following [54]. Specifically, we generate $\{\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{M}, \boldsymbol{A}, \boldsymbol{W}, \boldsymbol{U}, \boldsymbol{S}\} = \mathcal{F}(\{\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{m}, \boldsymbol{a}, \boldsymbol{w}, \boldsymbol{u}, \boldsymbol{s}\})$ in Fourier domain. Then, the derivative of Eq. (6a) with respect to $\boldsymbol{Z}$ can be calculated. After that, through utilizing the Sherman-Morrison formula [55], the closed-form solution of Eq. (6a) can be obtained as follows:

$$\boldsymbol{Z}^{(j+1)} = \frac{\beta}{\lambda_1 + \eta}\left(\bar{\boldsymbol{Z}} - \frac{\boldsymbol{S} \circ \bar{\boldsymbol{S}} \circ \bar{\boldsymbol{Z}}}{\frac{\lambda_1+\eta}{\beta} + \bar{\boldsymbol{S}} \circ \boldsymbol{S}}\right), \tag{7}$$

where $\circ$ denotes the Hadamard operator, $\bar{\boldsymbol{Z}}$ is defined as $\bar{\boldsymbol{Z}} = \frac{\lambda_1}{\beta}\boldsymbol{X} + \frac{\eta}{\beta}(\boldsymbol{M}^{(j)} - \boldsymbol{U}^{(j)}) + \boldsymbol{S} \circ (\bar{\boldsymbol{S}} \circ \boldsymbol{Y} + \boldsymbol{A}^{(j)} - \boldsymbol{W}^{(j)})$. Finally, the non-flash image $\boldsymbol{z}^{(j+1)}$ can be updated through the Inverse Fast Fourier Transform (IFFT) following $\boldsymbol{z}^{(j+1)} = \mathcal{F}^{-1}(\boldsymbol{Z}^{(j+1)})$. Due to space limit, the details of the above derivation process about NFR unit are presented in the supplementary material.

4) The Auxiliary Variable Updating (AVU) Unit: The AVU unit corresponds to Eq. (6b) and Eq. (6c), which aims to solve auxiliary variables $\boldsymbol{a}$ and $\boldsymbol{w}$. As we can see, Eq. (6b) is a least absolute shrinkage and selection operator (LASSO) problem with respect to the auxiliary variable $\boldsymbol{a}$, which can be solved through proximal gradient descent algorithm [56]. The iterative solution is given as follows,

$$\boldsymbol{a}^{(j+1)} = S_{\frac{\rho}{\beta}}(\boldsymbol{s} * \boldsymbol{z}^{(j+1)} - \boldsymbol{s} * \boldsymbol{y} + \boldsymbol{w}^{(j)}). \tag{8}$$

Here, $S_{\frac{\rho}{\beta}}$ indicates the soft-thresholding operator determined by regularization parameters ρ and β. After the updating of $\boldsymbol{a}^{(j+1)}$, the corresponding Lagrangian multiplier $\boldsymbol{w}^{(j+1)}$ can then be updated from Eq. (6c), as follows:

$$\boldsymbol{w}^{(j+1)} = \boldsymbol{w}^{(j)} + \boldsymbol{s} * \boldsymbol{z}^{(j+1)} - \boldsymbol{s} * \boldsymbol{y} - \boldsymbol{a}^{(j+1)}. \tag{9}$$

Following Eq. (8) and Eq. (9), the architecture of the AVU unit is designed as shown in Fig. 3 (c).

5) The Denoising Penalty Updating (DPU) Unit: The DPU unit is designed to update $\boldsymbol{m}$ and $\boldsymbol{u}$ according to Eq. (6d) and Eq. (6e). For Eq. (6d), we learn a proximal operator $G(\cdot)$ corresponding to the restoration constraint $g(\cdot)$. The solution of $\boldsymbol{u}^{(j+1)}$ in Eq. (6e) can be obtained in a similar way as

Algorithm 1 The Overall Network Design Procedure of the Proposed LGCNet

Input: Input images $\boldsymbol{x}$ and $\boldsymbol{y}$, number of iterations J.
Output: The predicted denoised image $\boldsymbol{z}^{(J)}$.
Initialize $\boldsymbol{z}^{(0)}, \boldsymbol{a}^{(0)}, \boldsymbol{w}^{(0)}, \boldsymbol{m}^{(0)}, \boldsymbol{u}^{(0)}$ by INM module.
for $j = 0, ..., J-1$ **do**
  calculate hyper-parameters $\lambda_1, \rho, \gamma, \beta, \eta$ by RPP unit;
  update $\boldsymbol{z}^{(j+1)}$ following Eq. (7);
  update $\boldsymbol{a}^{(j+1)}$ following Eq. (8);
  update $\boldsymbol{w}^{(j+1)}$ following Eq. (9);
  update $\boldsymbol{m}^{(j+1)}$ following Eq. (10a);
  update $\boldsymbol{u}^{(j+1)}$ following Eq. (10b);
end
return $\boldsymbol{z}^{(J)}$

Eq. (9). In this way, the auxiliary variable $\boldsymbol{m}^{(j+1)}$ and the corresponding Lagrangian multiplier $\boldsymbol{u}^{(j+1)}$ are updated as follows,

$$\boldsymbol{m}^{(j+1)} = G(\boldsymbol{z}^{(j+1)} + \boldsymbol{u}^{(j)}, \frac{\eta}{\gamma}), \tag{10a}$$

$$\boldsymbol{u}^{(j+1)} = \boldsymbol{u}^{(j)} + \boldsymbol{z}^{(j+1)} - \boldsymbol{m}^{(j+1)}. \tag{10b}$$

The proximal operator $G(\cdot)$ in Eq. (10) learns an implicit prior on the denoised non-flash image $\boldsymbol{z}^{(j+1)}$. The DPU unit is designed with a U-Net style architecture [57] to solve Eq. (10a), due to it effectiveness in restoration tasks [42], [58], [59], [60]. Fig. 3 (d) illustrates the network architecture of the DPU unit, which accepts $\boldsymbol{z}^{(j+1)} + \boldsymbol{u}^{(j)}$ and $\frac{\eta}{\gamma}$ as inputs, and output the updated $\boldsymbol{m}^{(j+1)}$ and $\boldsymbol{u}^{(j+1)}$. The detailed description of the DPU unit structure is provided in the supplementary material.

The above steps complete the introduction of the $(j+1)$-th GDM module to update the clean non-falsh image $\boldsymbol{z}^{(j+1)}$. We have in total J iterations to obtain $\boldsymbol{z}^{(J)}$ as the final denoising output. The whole design procedure is summarized in Algorithm 1 and the framework of our proposed LGCNet is shown in Fig. 2.

To train the proposed LGCNet, we minimize the mean absolute error (MAE) loss between the denoised non-flash image sequence $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(J)}\}$ and the ground-truth image $\boldsymbol{z}_{gt}$, with the loss function defined as follows,

$$L_{\mathrm{DN}} = L_{\mathrm{MAE}}(\boldsymbol{z}^{(J)}, \boldsymbol{z}_{gt}) + \frac{1}{J-1}\sum_{j=1}^{J-1} L_{\mathrm{MAE}}(\boldsymbol{z}^{(j)}, \boldsymbol{z}_{gt}). \tag{11}$$

In the above equation, the MAE loss between the final restored image $\boldsymbol{z}^{(J)}$ and $\boldsymbol{z}_{gt}$ is assigned a full weight, while the weight of other intermediate restored images is set to $\frac{1}{J-1}$.

IV. Experiment

A. Experimental Settings

1) Dataset: In this paper, we adopt the Flash and Ambient Illuminations Dataset (FAID) [46], the Multi-Illumination Dataset (MID) [47] and the DeepFlash Portrait Dataset (DPD) [48] for training and testing. For the FAID dataset, we randomly selected 404 flash and non-flash image pairs for training, and used the remaining 12 image pairs for testing. The test image pairs cover image content from different categories. The MID dataset captured 1,016 scenes under 25 lighting conditions, of which 984 scenes were used as training set and 30 scenes were used as test set. The DPD dataset contains 429 flash and non-flash image pairs. Each pair consists of one photograph of a subject's face taken with the camera flash enabled, and the other using a photographic studio-lighting setup. We randomly selected 20 image pairs for testing while using the remaining 409 pairs for training. Moreover, to evaluate the performance of the methods in real noisy scenarios, two real noisy flash and non-flash datasets are utilized for testing: the RIDFnF dataset [6], captured using a Galaxy S21+ with a fixed ISO of 1600 under low light conditions, and our own newly proposed dataset,[1] captured using an iPhone 12 with an ISO setting of 3168.

2) Implementation Details: The Laplacian operator is chosen as $\boldsymbol{s}$. Specifically, $\boldsymbol{s}$ is a 3-dimensional matrix containing three identical channels $\{\boldsymbol{s}', \boldsymbol{s}', \boldsymbol{s}'\}$, each of which consisting of a 2-dimensional matrix as shown below:

$$\boldsymbol{s}' = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}. \tag{12}$$

The number J of GDM modules is set to 3, with parameters in different GDM modules learned independently. The size of convolutional filters used in the network is 3×3. The network is initialized through uniform distribution and trained for around $5e^3$ epochs. The Adam optimizer with default parameters (ρ_1=0.9, ρ_2=0.999, δ=1×10^{-8}) is used to train the network. The initial learning rate is 5×10^{-4}, which is halved every $1e^3$ epochs. For all three datasets, the training patch size is 128×128, and the number of training patches for FAID, MID and DPD datasets is around $2e^6$, $5e^6$ and $2e^6$, respectively.

3) Comparison Methods: To demonstrate the effectiveness of our method, we compare it with both single and guided image denoising methods, including DnCNN [61], MAXIM [62] and DKP [28] for single image denoising, DJFR [40], MuGIF [4], CU-Net [9], DKN [41], MM-CNN [64], UMGF [63], MN [5], FGDNet [65], RIDFnF [6], FRL [66] and DeepM^{2}CDL [42] for guided image denoising. The results of these comparison approaches are either obtained through the official codes provided by the authors, or directly from the published article.

4) Evaluation Metrics: We adopt two widely used metrics to evaluate the quality of the restored non-flash images, including Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index (SSIM) [67]. Given a denoised image $\boldsymbol{I}_d \in \mathbb{R}^{h \times n}$ and the ground-truth image $\boldsymbol{I}_{gt} \in \mathbb{R}^{h \times n}$, these two metrics are defined as follows:

- PSNR metric: As a widely used image quality evaluation metric, PSNR is defined as follows:

$$\mathrm{PSNR}(\boldsymbol{I}_{\mathrm{d}}, \boldsymbol{I}_{\mathrm{gt}}) = 20\log_{10}\left(\frac{255}{\sqrt{\frac{1}{\mathrm{hn}}\sum_{\mathrm{i}=1}^{\mathrm{h}}\sum_{\mathrm{j}=1}^{\mathrm{n}}(\boldsymbol{I}_{\mathrm{d}} - \boldsymbol{I}_{\mathrm{gt}})^2}}\right). \tag{13}$$

- SSIM metric: SSIM [67] assesses the image quality based on the degradation of structural information. It mainly

[1]The proposed dataset is available for download at https://drive.google.com/drive/folders/10NQONsMlIeA72LK70mUoE1roRfOvbvbJ?usp=sharing

TABLE III

THE NUMERICAL RESULTS FOR FLASH GUIDED NON-FLASH IMAGE DENOISING IN TERMS OF PSNR (DB) ON FAID DATASET, WITH THE BEST RESULTS IN BOLD AND SECOND BESTS UNDERLINED

σ=25	Minion	Towel	Elmo	Pendant	Book	Tampax	Typewriter	Pot	Plant	Flower	Aloe	Cactus	Average
DnCNN [61]	34.13	37.58	36.65	39.78	35.61	36.55	34.87	34.35	34.42	36.26	32.70	31.62	35.38
MAXIM [62]	34.68	38.23	37.35	40.24	37.50	37.41	35.25	34.69	34.82	36.70	33.35	31.98	36.02
DKP [28]	30.78	31.17	34.30	38.89	35.19	30.50	30.90	28.63	30.21	35.15	33.40	33.39	32.71
DJFR [40]	31.59	36.86	34.69	36.83	34.16	31.45	32.81	33.26	31.90	34.69	30.78	31.13	33.76
MuGIF [4]	30.49	35.42	33.75	35.78	32.62	33.46	31.51	31.82	30.95	33.24	29.49	30.88	32.45
CU-Net [9]	34.24	37.99	36.82	39.95	36.86	36.97	35.07	35.52	34.42	36.40	32.83	33.26	35.86
DKN [41]	29.84	35.42	33.38	31.11	31.63	32.84	31.15	32.63	30.45	33.29	24.04	30.63	31.37
UMGF [63]	32.77	35.90	35.33	38.43	36.63	35.11	33.52	34.81	32.84	34.49	31.19	33.19	34.52
MM-CNN [64]	34.52	38.43	37.05	40.26	37.10	37.22	35.54	35.95	34.74	36.91	33.07	33.62	36.21
MN [5]	34.38	38.34	36.98	40.21	37.75	37.27	35.09	35.99	34.56	36.58	32.98	33.62	36.15
FGDNet [65]	32.83	37.73	35.62	39.52	36.33	36.31	33.70	35.37	32.83	35.31	31.20	33.16	34.99
RIDFnF [6]	34.56	38.32	37.34	40.16	38.03	37.59	35.17	36.04	34.35	36.67	32.88	33.88	36.25
DeepM^{2}CDL [42]	35.08	38.98	37.86	40.95	39.32	38.47	35.62	36.94	34.88	37.16	33.40	33.66	36.86
Ours	**35.67**	**39.41**	**38.57**	**41.32**	**40.50**	**39.01**	**35.87**	**37.45**	**35.10**	**37.36**	**33.85**	**34.97**	**37.42**
σ=50	Minion	Towel	Elmo	Pendant	Book	Tampax	Typewriter	Pot	Plant	Flower	Aloe	Cactus	Average
DnCNN [61]	30.32	34.65	33.24	36.58	32.37	32.95	31.87	30.87	30.83	33.04	28.49	28.04	31.94
MAXIM [62]	30.98	35.72	34.14	37.42	34.19	34.02	32.41	31.24	31.27	33.55	29.27	28.56	32.73
DKP [28]	29.67	24.40	33.97	34.43	37.06	29.76	24.79	27.09	23.77	**34.63**	28.90	28.04	29.71
DJFR [40]	28.55	33.27	31.79	34.65	31.74	31.58	29.90	30.02	28.84	31.30	28.05	27.66	30.61
MuGIF [4]	26.93	32.02	30.94	31.78	29.72	30.54	28.57	28.97	27.45	29.93	28.97	27.81	29.22
CU-Net [9]	31.08	35.91	34.20	37.23	34.11	34.22	32.40	32.88	31.16	33.60	29.17	30.69	33.05
DKN [41]	27.82	33.54	31.54	30.27	30.28	31.23	29.27	30.99	28.01	31.11	22.48	28.42	29.58
UMGF [63]	29.33	34.22	32.49	35.71	34.80	32.59	30.27	32.43	29.46	31.82	27.40	31.20	31.81
MM-CNN [64]	31.32	36.07	34.36	37.42	34.41	34.41	32.70	33.06	31.48	33.81	29.34	30.88	33.28
MN [5]	31.00	36.22	34.09	37.35	35.47	34.46	32.33	33.47	31.26	33.56	29.17	31.69	33.34
FGDNet [65]	29.30	35.56	32.88	36.73	34.83	33.29	30.71	33.25	29.27	31.93	27.32	30.67	32.15
RIDFnF [6]	31.29	36.45	34.48	37.34	35.51	34.82	32.42	33.79	31.07	33.69	29.12	31.80	33.48
DeepM^{2}CDL [42]	32.28	37.45	35.53	38.17	37.33	36.23	33.09	34.93	31.78	34.51	29.98	31.84	34.43
Ours	**32.93**	**38.09**	**36.31**	**38.75**	**38.79**	**36.62**	**33.35**	**35.30**	**31.93**	34.55	**30.45**	**33.12**	**35.01**
σ=75	Minion	Towel	Elmo	Pendant	Book	Tampax	Typewriter	Pot	Plant	Flower	Aloe	Cactus	Average
DnCNN [61]	28.33	32.75	31.16	34.82	31.83	30.74	29.97	29.21	28.46	30.83	26.25	26.55	30.08
MAXIM [62]	28.93	34.01	32.28	35.65	32.82	31.96	30.76	29.40	29.19	31.62	26.99	26.69	30.86
DKP [28]	30.27	21.10	26.56	34.92	33.74	26.97	21.13	32.61	20.27	27.57	25.38	24.47	27.08
DJFR [40]	26.65	31.77	30.63	32.78	29.54	29.86	28.85	28.33	27.54	29.25	25.56	26.26	28.92
MuGIF [4]	24.89	30.30	29.82	29.70	28.40	28.54	26.46	27.44	25.82	27.84	24.49	26.27	27.50
CU-Net [9]	29.12	34.25	32.55	35.12	32.93	32.19	30.55	31.66	29.17	31.73	27.03	29.30	31.30
DKN [41]	26.55	32.60	30.56	29.54	29.20	29.93	27.69	30.16	26.22	29.50	22.03	27.61	28.46
UMGF [63]	27.90	32.70	31.30	34.03	34.05	31.27	28.72	31.35	27.88	30.28	25.59	30.13	30.43
MM-CNN [64]	29.39	34.48	32.75	35.44	33.19	32.44	30.89	31.85	29.47	32.03	27.24	29.56	31.56
MN [5]	29.20	34.98	32.65	35.83	34.52	32.81	30.81	32.19	29.25	31.86	27.27	30.59	31.83
FGDNet [65]	27.93	34.56	31.81	35.00	34.01	31.97	29.19	32.07	27.49	30.37	25.60	29.69	30.81
RIDFnF [6]	29.71	35.17	33.07	35.52	33.50	33.06	30.96	32.27	29.30	32.00	27.30	31.13	31.92
DeepM^{2}CDL [42]	30.39	36.31	33.81	36.97	**37.05**	34.40	31.40	33.49	29.81	32.69	27.92	31.13	32.95
Ours	**31.66**	**37.34**	**35.23**	**37.49**	36.79	**35.01**	**31.84**	**33.90**	**30.07**	**32.82**	**28.59**	**31.98**	**33.56**

considers three key features of the image, i.e., luminance, contrast and structure. Compared to PSNR, the SSIM metric is more consistent with the human visual perception, which is computed as follows:

$$\mathrm{SSIM}(\boldsymbol{I}_{\mathrm{d}}, \boldsymbol{I}_{\mathrm{gt}}) = \frac{(2\mu_{\mathrm{d}}\mu_{\mathrm{gt}}+\mathrm{C}_1)(2\sigma_{\mathrm{d,gt}}+\mathrm{C}_2)}{(\mu_{\mathrm{d}}^2+\mu_{\mathrm{gt}}^2+\mathrm{C}_1)(\sigma_{\mathrm{d}}^2+\sigma_{\mathrm{gt}}^2+\mathrm{C}_2)}, \quad (14)$$

where μ_d and μ_{gt} are the mean values of $\boldsymbol{I}_d$ and $\boldsymbol{I}_{gt}$ respectively, σ_d and σ_{gt} represent the standard deviations of $\boldsymbol{I}_d$ and $\boldsymbol{I}_{gt}$. $\sigma_{d,gt}$ denotes the covariance value and C_1, C_2 are positive constant values.

B. Denoising Performance Against SOTA Methods

In this section, we evaluate both quantitative and qualitative performance of our proposed network on FAID, MID and DPD datasets. Table III presents the PSNR performance of our network and the comparison methods on the FAID dataset with Gaussian noise level as $\sigma = 25$, 50 and 75. As can be seen, our network outperforms other comparison methods on almost all testing images at different noise levels. Specifically, our method achieves on average 0.56dB, 0.58dB and 0.61dB PSNR higher than the second best method DeepM^{2}CDL [42] in $\sigma = 25$, 50 and 75 noise levels, respectively. Similar improvements can be seen on the MID dataset in Table IV, in which our method achieves 1.08dB, 0.90dB, 0.94dB PSNR higher than the second best method on the three noise levels. For the SSIM metric, our method also outperforms all comparison methods. Furthermore, Table V presents the PSNR and SSIM performance on the DPD dataset. As can be seen, our LGCNet achieves on average 0.49dB, 0.49dB and 0.82dB PSNR higher than the second best method with $\sigma = 25$, 50 and 75 noise levels, respectively. These results validate that comparing with other CNN based methods, our method achieves better denoising results for different datasets at different noise levels.

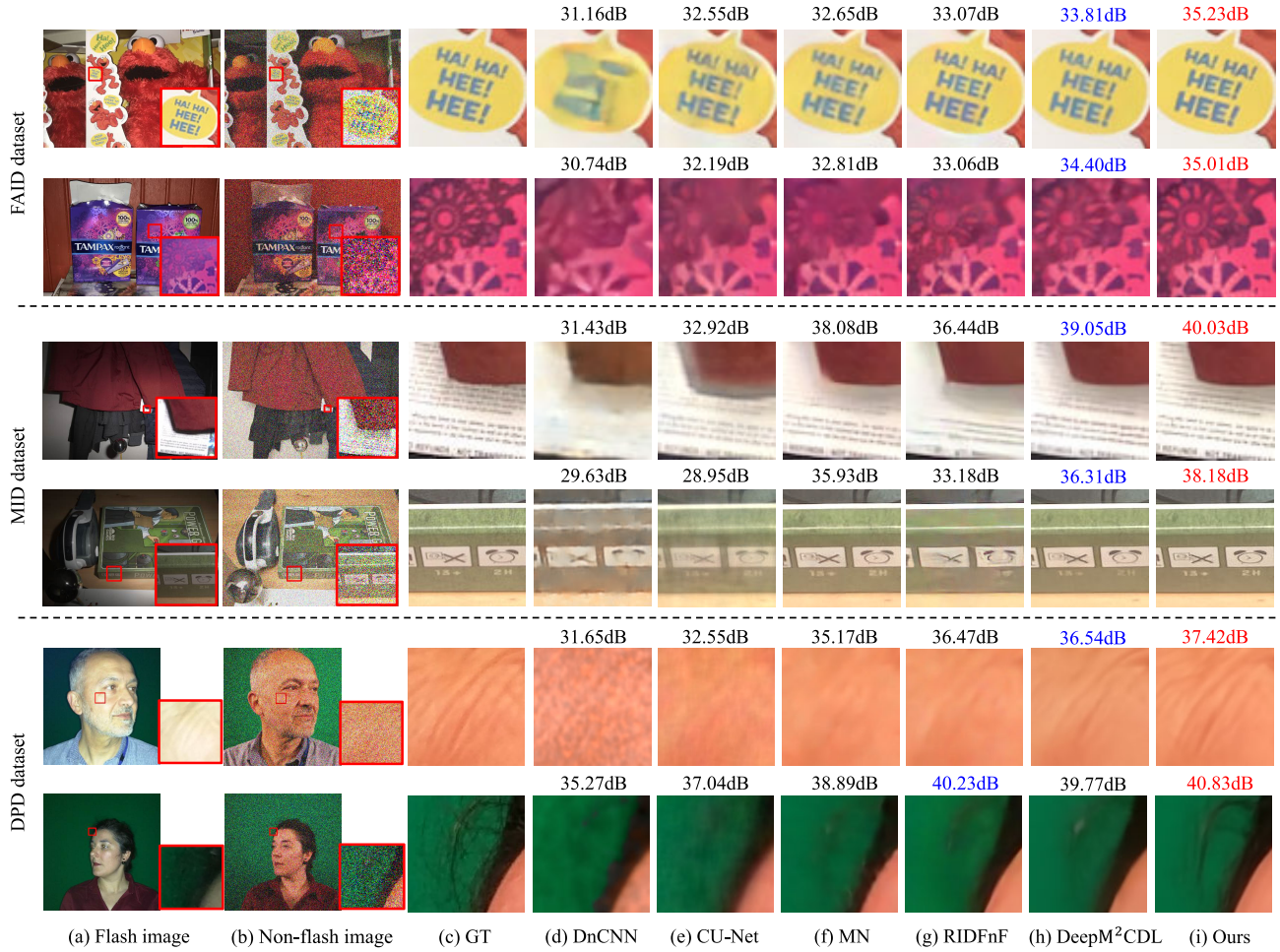


Fig. 4. Visualization of the denoised non-flash images by different methods on FAID, MID and DPD datasets, with noise level as $\sigma = 75$. The best PSNR results are marked in red and the second bests are marked in blue. For each image set, we have (a) the clean flash image, (b) the noisy non-flash image, (c) the ground-truth image, (d) DnCNN [61], (e) CU-Net [9], (f) MN [5], (g) RIDFnF [6], (h) DeepM^{2}CDL [42], and (i) our LGCNet.

TABLE IV

THE NUMERICAL RESULTS FOR FLASH GUIDED NON-FLASH IMAGE DENOISING IN TERMS OF PSNR (DB) ON MID DATASET, WITH THE BEST RESULTS IN BOLD AND SECOND BESTS UNDERLINED

Methods	σ=25		σ=50		σ=75	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
DnCNN [61]	34.57	0.9160	32.69	0.8821	31.26	0.8613
DJFR [40]	37.03	0.9363	32.96	0.8422	31.84	0.8555
CUNet [9]	34.61	0.9125	32.39	0.8917	31.18	0.8788
UMGF [63]	38.18	0.9528	35.84	0.9359	34.30	0.9226
FRL [66]	38.40	0.9520	35.82	0.9354	34.24	0.9237
MN [5]	39.51	0.9600	37.01	0.9449	35.50	0.9350
FGDNet [65]	38.38	0.9510	35.88	0.9346	34.39	0.9243
RIDFnF [6]	38.31	0.9428	35.33	0.9109	33.74	0.8919
DeepM^{2}CDL [42]	39.67	0.9624	37.61	0.9492	36.28	0.9401
Ours	**40.75**	**0.9685**	**38.51**	**0.9580**	**37.22**	**0.9516**

TABLE V

THE NUMERICAL RESULTS FOR FLASH GUIDED NON-FLASH IMAGE DENOISING IN TERMS OF PSNR (DB) ON DPD DATASET, WITH THE BEST RESULTS IN BOLD AND SECOND BESTS UNDERLINED

Methods	σ=25		σ=50		σ=75	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
DnCNN [61]	35.09	0.9539	34.26	0.9433	31.26	0.8613
DJFR [40]	37.04	0.8748	32.72	0.7148	30.21	0.5827
CUNet [9]	38.70	0.9593	36.10	0.9459	34.68	0.9328
UMGF [63]	40.57	0.9679	37.76	0.9525	36.14	0.9424
FRL [66]	40.86	0.9690	38.07	0.9559	36.49	0.9475
MN [5]	41.79	0.9736	38.80	0.9607	36.96	0.9516
FGDNet [65]	40.89	0.9692	38.13	0.9566	36.20	0.9449
RIDFnF [6]	42.06	0.9747	39.40	0.9633	37.81	0.9561
DeepM^{2}CDL [42]	41.57	0.9758	39.19	0.9670	37.77	0.9603
Ours	**42.55**	**0.9768**	**39.89**	**0.9675**	**38.63**	**0.9624**

Fig. 4 visualizes the denoised images of different methods with $\sigma = 75$ on the three datasets. As can be seen from this figure, our method is able to remove the noise effectively to restore sufficient image details of the non-flash images. In contrast, other approaches either lead to blurred edges [5], [6], [9], [42] or deficient image details [61]. Specifically, as shown in the fifth row of Fig. 4, our LGCNet is able to restore a clean non-flash image with sufficient wrinkle details on the face, while other methods struggle to do so. Fig. 5 visualizes the denoised images from different methods on two real noisy flash and non-flash datasets: the RIDFnF dataset and our proposed dataset. As shown, other approaches either suffer from noticeable artifacts [9], [61] or tend to produce overly smoothed results [5], [6], [42]. In contrast, our proposed LGCNet retains more text, color and texture details, producing visually pleasing results in real noisy scenarios. All these quantitative and qualitative results demonstrate the outstanding denoising performance of our LGCNet, by leveraging cross-modality gradient consistency prior.

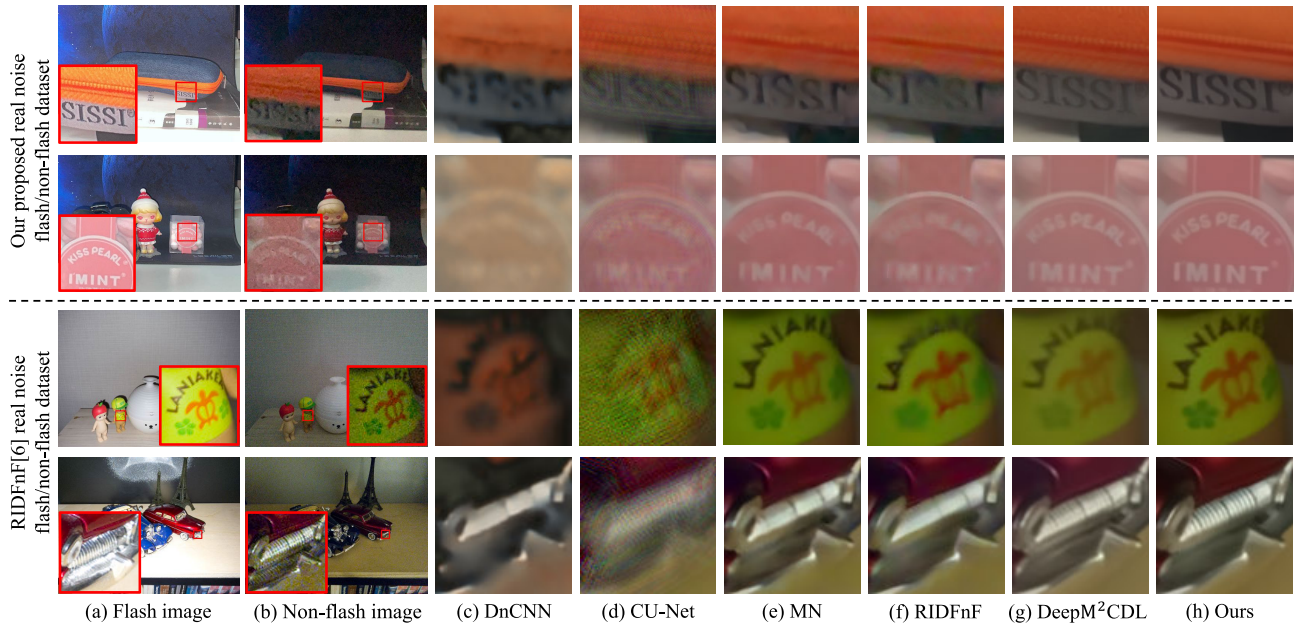


Fig. 5. Visualization of the denoised non-flash images by different methods on real noisy datasets, test on model pre-trained with the FAID dataset at a noise level of $\sigma = 75$. For each image set, we have (a) the clean flash image, (b) the noisy non-flash image, (c) DnCNN [61], (d) CU-Net [9], (e) MN [5], (f) RIDFnF [6], (g) DeepM^{2}CDL [42], and (h) our LGCNet.

C. Effectiveness of the LGC Model

The Laplacian Gradient Consistency (LGC) model in Eq. (3) is the foundation of our proposed LGCNet. Thus, it is necessary to analyze the effectiveness of the LGC model in terms of its convergence speed and denoising performance. To achieve this goal, we firstly specify the constraint $g(\boldsymbol{z})$ in Eq. (3) as the commonly used total variation function [68], resulting in the following optimization problem,

$$\underset{z}{\operatorname{argmin}} \frac{\lambda_1}{2}\|\boldsymbol{z}-\boldsymbol{x}\|_2^2 + \rho\|\boldsymbol{s}*\boldsymbol{z}-\boldsymbol{s}*\boldsymbol{y}\|_1^1 + \gamma\|\boldsymbol{z}\|_{TV}. \quad (15)$$

For comparison, we also solve the optimization problem by removing the gradient constraint, as follows,

$$\underset{z}{\operatorname{argmin}} \frac{\lambda_1}{2}\|\boldsymbol{z}-\boldsymbol{x}\|_2^2 + \rho\|\boldsymbol{z}-\boldsymbol{y}\|_1^1 + \gamma\|\boldsymbol{z}\|_{TV}. \quad (16)$$

The ADMM algorithm [11] is used to solve these two problems, with their convergence performance shown in Fig. 6. Note that the parameter values of λ_1, ρ and γ are chosen to achieve optimal results of Eq. (15) and Eq. (16), respectively. As can be seen, the convergence speed of Eq. (15) and Eq. (16) are similar, which can both converge within 15 iterations. However, the denoising performance of Eq. (15) with the LGC model is much better than the other one, in terms of both PSNR and SSIM. Table VI presents the denoising results obtained by these two approaches on FAID dataset. As can be seen, the solution of Eq. (15) with the LGC model can improve on average 1.38dB, 1.19dB and 1.10dB in PSNR with σ=25, 50, 75, respectively. These results demonstrate the effectiveness of our proposed LGC model on guided image denoising.

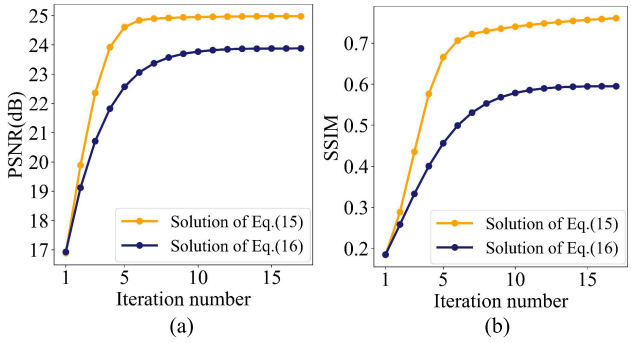


Fig. 6. Comparison of the convergence and denoising performance of Eq. (16) and Eq. (15) with iterations, in terms of (a) PSNR and (b) SSIM metrics.

D. Visualization of Intermediate Features

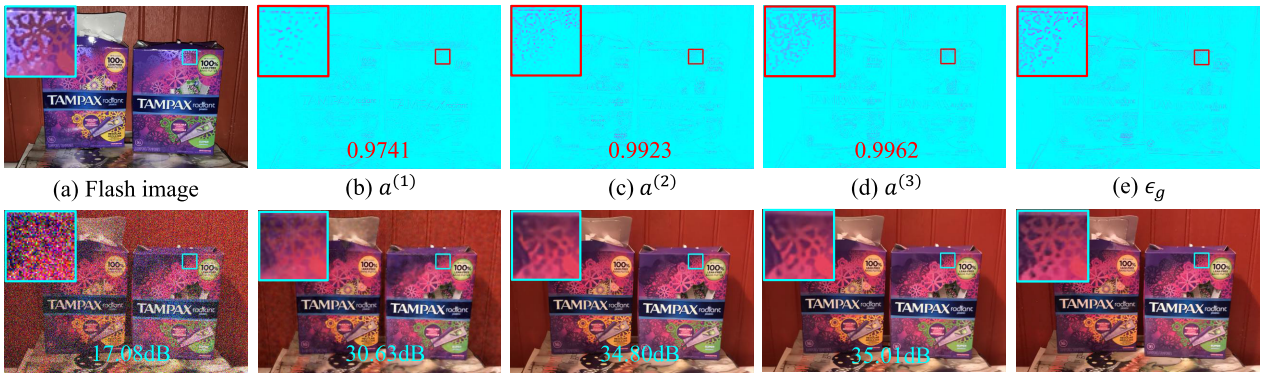
Since the proposed LGCNet is designed with the Laplacian gradient consistency prior, it is necessary to visualize the gradient modality gap $\boldsymbol{a} = \boldsymbol{s}*\boldsymbol{z} - \boldsymbol{s}*\boldsymbol{y}$ in the LGCNet. Recall in Fig. 2, $\boldsymbol{a}$ is generated by the AVU unit in each GDM module. Thus, in the first row of Fig. 7, we visualize the gradient modality gap $\boldsymbol{a}$ obtained by AVU unit in the first, second and third GDM modules, respectively. As can be seen, the initial $\boldsymbol{a}^{(1)}$ from the first GDM module is quite different from the ground-truth $\boldsymbol{\epsilon_g}$, with a histogram correlation [69] of 0.9741. In contrast, $\boldsymbol{a}^{(3)}$ from the third GDM module shows a much closer alignment to the ground truth, increasing the histogram correlation by 0.0221. These results indicate the effectiveness of the Laplacian gradient consistency in our LGCNet. In addition to the visualization of gradient difference, we also visualize the denoised non-flash images $\boldsymbol{z}$ in each GDM module. As shown in the second row of Fig. 7, the quality of denoised non-flash image gradually increases with more GDM modules, which demonstrates the effectiveness of our LGCNet.

E. Ablation Study

In this subsection, we analyze the effects of some important parameters on the proposed LGCNet, including the number J

TABLE VI
THE COMPARISON OF DENOISING RESULTS IN TERMS OF PSNR (DB) BY SOLVING EQ. (16) AND EQ. (15) ON FAID DATASET, WITH THE BEST RESULTS IN BOLD

σ=25	Minion	Towel	Elmo	Pendant	Book	Tampax	Typewriter	Pot	Plant	Flower	Aloe	Cactus	Average
Solution of Eq. (16)	28.57	31.51	31.46	33.26	33.06	30.91	29.87	31.39	30.11	30.96	26.46	28.73	30.52
Solution of Eq. (15)	**30.41**	**32.82**	**32.12**	**34.50**	**33.18**	**31.86**	**31.29**	**32.39**	**30.79**	**32.17**	**29.35**	**31.87**	**31.90**
σ=50	Minion	Towel	Elmo	Pendant	Book	Tampax	Typewriter	Pot	Plant	Flower	Aloe	Cactus	Average
Solution of Eq. (16)	25.60	28.00	27.94	27.78	28.95	27.83	26.06	27.67	26.04	26.64	23.21	25.30	26.75
Solution of Eq. (15)	**26.33**	**28.90**	**28.33**	**29.42**	**29.08**	**28.18**	**26.86**	**28.88**	**27.23**	**27.71**	**25.44**	**28.89**	**27.94**
σ=75	Minion	Towel	Elmo	Pendant	Book	Tampax	Typewriter	Pot	Plant	Flower	Aloe	Cactus	Average
Solution of Eq. (16)	**23.41**	24.81	24.63	24.28	**25.30**	24.82	23.19	24.86	23.34	23.52	21.29	23.19	23.88
Solution of Eq. (15)	23.38	**26.02**	**25.60**	**25.61**	25.01	**25.72**	**23.29**	**26.01**	**24.62**	**24.50**	**23.04**	**26.90**	**24.98**



(a) Flash image (b) $a^{(1)}$ (c) $a^{(2)}$ (d) $a^{(3)}$ (e) ϵ_g
(f) Non-flash image (g) $z^{(1)}$ (h) $z^{(2)}$ (i) $z^{(3)}$ (j) GT

Fig. 7. Visualization of the intermediate features in our LGCNet. (a) and (f) are the input flash image and noisy non-flash image. (b), (c) and (d) show the gradient modality gap $\boldsymbol{a}$ from the first to third GDM modules. (e) is the ground-truth gradient modality gap $\boldsymbol{\epsilon_g}$ between the two modalities. (g), (h) and (i) show the obtained non-flash image $\boldsymbol{z}$ from the first to third GDM modules. (j) is the ground-truth clean non-flash image.

of GDM modules, the Laplacian operator and the loss function. All the ablation studies are performed on FAID dataset.

1) Effect of GDM Module Number: As shown in Fig. 2, our LGCNet is composed of several GDM modules, with each module corresponding to one iteration in the LGC model. Typically, with more iterations, we can restore more accurate non-flash image. Thus, in this ablation study, we explore how the number J of GDM modules affects the denoising performance. Fig. 8 (a) shows the PSNR and SSIM results when J is increased from 2 to 4 with different noise levels. As shown in this figure, with the increased number of GDM modules, we can obtain higher PSNR and SSIM values, indicating that better denoising results are yielded. This is consistent with our aforementioned analysis.

2) Effect of Laplacian Operator: The gradient operator $\boldsymbol{s}$ is important for our LGCNet, and we adopt the Laplacian operator as $\boldsymbol{s}$ in this paper. For comprehensive analysis, it is necessary to explore the denoising performance of our LGCNet with other gradient operators. To this end, we replace the Laplacian operator with Sobel and Scharr operators and retrain the network. Fig. 8 (b) shows the PSNR and SSIM results with different gradient operators. As can be seen, our network can obtain good denoising results with all these three operators, indicating the effectiveness of our finding in Section III-A. Among the three operators, the Laplacian operator performs better than Sobel and Scharr operators, since it considers all pixels in the receptive field. That is the reason why Laplacian operator is adopted in this paper.

3) Effect of DPU Unit: An U-Net style architecture is utilized in our proposed DPU unit due to its effectiveness in restoration tasks. To evaluate the performance of the proposed DPU structure, we replace the DPU unit with a DenseBlock-based network structure, and retrain the network to compare denoising performance. As shown in Table VII, the U-Net style architecture in the DPU unit achieves a 1.2dB improvement in PSNR over the DenseBlock-based architecture. This result aligns with our aforementioned analysis.

4) Effect of Variables Update Order: Our LGCNet updates the target non-flash image $\boldsymbol{z}$, the auxiliary variables $\boldsymbol{m}$ and $\boldsymbol{a}$, as well as the corresponding Lagrangian multipliers $\boldsymbol{u}$ and $\boldsymbol{w}$ iteratively through cascaded GDM modules. In this ablation study, we investigate how the update order of these five variables affects the denoising performance. Table IX shows the PSNR and SSIM results when six different variable update orders are applied and the network is retrained and tested on FAID dataset with a noise level of $\sigma = 25$. As shown in this table, the update order '$\boldsymbol{zawmu}$' yields the highest RSNR and SSIM values compared to other five update orders. Consequently, we adopt the update order '$\boldsymbol{zawmu}$' in our LGCNet.

5) Effect of the Loss Function: The loss function plays an important role in the network training process. In this paper, we employ a MAE loss the train the network, and it is essential to evaluate the network performance with other loss functions employed. Fig. 8 (c) shows the performance with other two widely used loss functions: MSE loss and SSIM loss. As illustrated in this figure, the MAE loss achieves the highest PSNR value across all noise levels compared to other two loss functions. In terms of SSIM value, the MAE loss achieves a higher SSIM value than the MSE loss across different noise

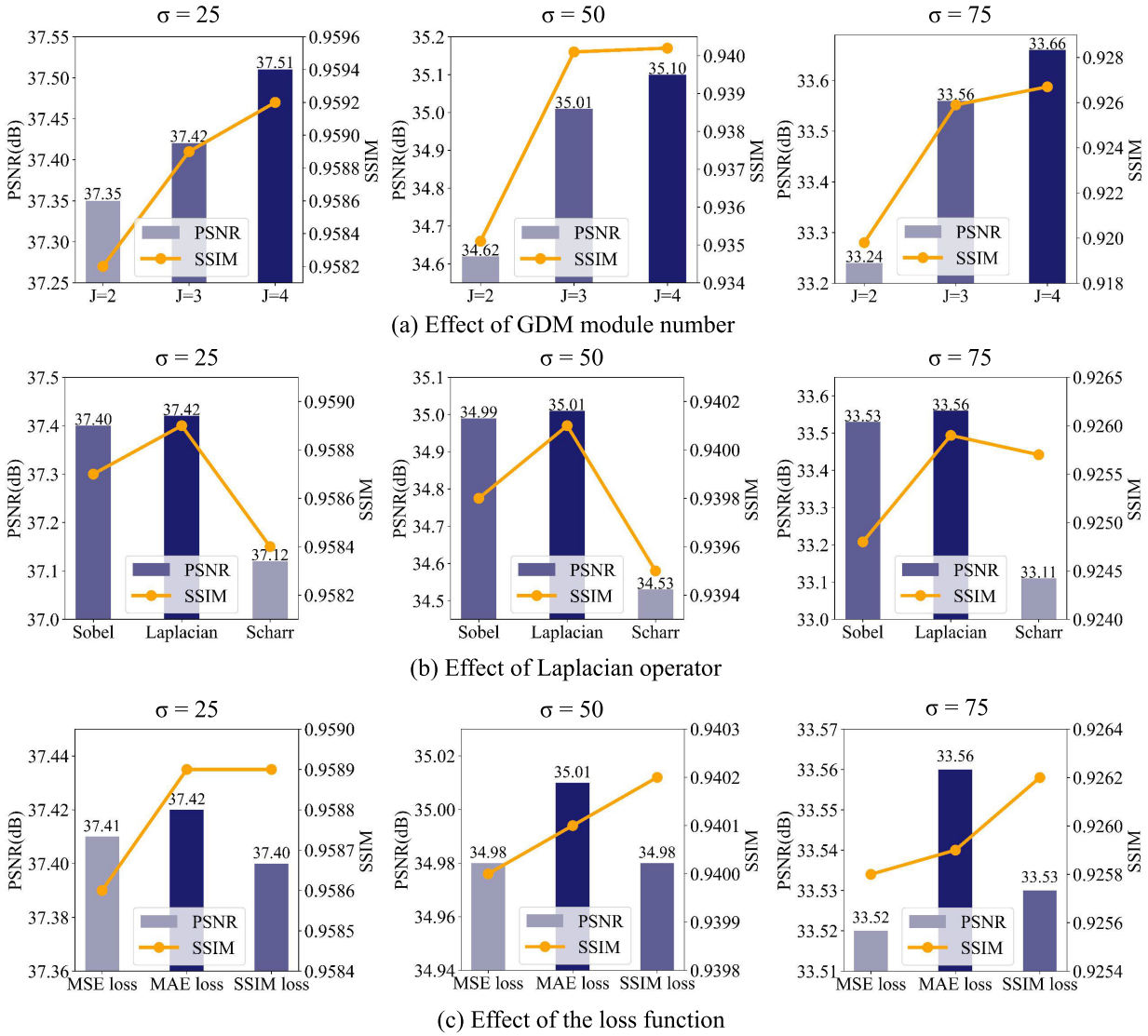


Fig. 8. The effects of (a) the GDM module number, (b) the Laplacian operator, and (c) the loss function on the performance of flash guided non-flash image denoising task with different noise levels.

TABLE VII

THE EFFECT OF THE U-NET STYLE ARCHITECTURE IN DPU UNIT. THE NUMERICAL RESULTS FOR FLASH GUIDED NON-FLASH IMAGE DENOISING IN TERMS OF PSNR (DB) ON FAID DATASET WITH A NOISE LEVEL OF $\sigma = 25$. THE BEST RESULTS ARE IN BOLD

σ=25	FLOPs	Minion	Towel	Elmo	Pendant	Book	Tampax	Typewriter	Pot	Plant	Flower	Aloe	Cactus	Average
DenseBlock-based network	465.03G	34.63	38.54	37.21	40.28	37.52	37.33	35.19	35.79	34.48	36.66	32.99	34.01	36.22
Our DPU unit	110.39G	**35.67**	**39.41**	**38.57**	**41.32**	**40.50**	**39.01**	**35.87**	**37.45**	**35.10**	**37.36**	**33.85**	**34.97**	**37.42**

TABLE VIII

THE EFFECT OF THE INCLUDING INTERMEDIATE CONSTRAINTS IN LOSS FUNCTION. THE NUMERICAL RESULTS FOR FLASH GUIDED NON-FLASH IMAGE DENOISING IN TERMS OF PSNR (DB) ON FAID DATASET WITH A NOISE LEVEL OF $\sigma = 25$. THE BEST RESULTS ARE IN BOLD

σ=25	Minion	Towel	Elmo	Pendant	Book	Tampax	Typewriter	Pot	Plant	Flower	Aloe	Cactus	Average
without intermediate constraints	35.54	39.32	38.55	41.18	40.38	38.93	35.86	37.42	35.08	37.32	33.82	34.96	37.36
with intermediate constraints(ours)	**35.67**	**39.41**	**38.57**	**41.32**	**40.50**	**39.01**	**35.87**	**37.45**	**35.10**	**37.36**	**33.85**	**34.97**	**37.42**

levels. Compared to the SSIM loss, the MAE loss can achieve the same SSIM value at noise level σ=25 and is only slightly lower at σ=50 and σ=75. By comprehensively considering the impact of loss function on PSNR and SSIM values, MAE loss is chosen for training in this paper.

In addition, it is essential to validate the effectiveness of incorporating constraints between the intermediate images $\{z^{(j)}\}_{j=1}^{J-1}$ and the ground-truth image z_{gt} within the loss function, as described in Eq.(11). Table VIII presents a comparison of the denoising performance of the loss function with and without the constraint on intermediate restored images. The results demonstrate that integrating the intermediate constraint into the loss function increases the average PSNR by 0.06 dB on the FAID dataset at a noise level of $\sigma = 25$. This improvement proves that enhancing the quality of intermediate images contributes to the overall quality of the final

TABLE IX
THE EFFECT OF THE UPDATING ORDER OF FIVE VARIABLES IN OUR NETWORK ON FAID DATASET WITH NOISE LEVEL $\sigma = 25$. THE BEST RESULTS ARE IN BOLD

Updating order	**awmuz**	**muzaw**	**zmuaw**	**muawz**	**awzmu**	**zawmu** (ours)
PSNR↑	37.34	37.36	37.04	37.37	36.93	**37.42**
SSIM↑	0.9580	0.9583	**0.9589**	0.9588	0.9588	**0.9589**

TABLE X
THE AVERAGE RUNNING SPEED (IN MILLISECONDS) OF OUR NETWORK WITH DIFFERENT INPUT IMAGE SIZES

Dataset	Image size	INM	RPP	NFR	AVU	DPU	Overall
FAID	600×900	1.1	0.5	1.7	0.3	5.9	28.4
MID	1000×1500	1.1	0.5	1.8	0.4	8.3	35.8
DPD	2000×2000	1.1	0.5	1.8	0.4	26.8	94.9

denoising output, demonstrating the effectiveness of including intermediate constraints in training our LGCNet.

F. Running Speed

For real-time applications, the network running speed is an important element to be considered. Thus, it is necessary to test the running speed of each unit in our LGCNet, especially the NFR unit which involves matrix computations. Table X presents the the running speed of INM module, RPP, NFR, AVU, DPU units and the whole network on different datasets. The time is recorded on a PC with an NVIDIA Corporation GV100GL GPU. As can be seen, thanks to the closed-form solution, the matrix computation involved NFR unit is with very fast running speed. For the whole network, the running speed increases with the input image size, which are 28.4ms, 35.8ms and 94.9ms for images with size 600×900, 1000×1500, and 2000×2000, respectively. This is also acceptable for most practical applications.

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed a novel deep network for flash guide non-flash image denoising task, with the Laplacian gradient consistency prior. Different from existing methods that model the pixel-level consistency, we find that the modality gap between flash and non-flash images conforms to Laplacian distribution in gradient domain. On the basis of this observation, we established a Laplacian gradient consistency (LGC) model, and derived a novel network architecture named LGCNet from the LGC model. The LGCNet has good network interpretability, as each component of the network strictly matches the iterative solution of the LGC model. The experimental results demonstrate that our LGCNet achieves state-of-the-art denoising performance both quantitatively and qualitatively, on FAID, MID and DPD datasets.

In the future, it is interesting to explore the consistency between unpaired or misaligned flash and non-flash images. Besides, the current work only focuses on guided image denoising. A promising future direction is to investigate cross-frame and cross-modality consistency to design interpretable networks for guided video denoising.

REFERENCES

[1] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 664–672, Aug. 2004.

[2] E. Eisemann and F. Durand, "Flash photography enhancement via intrinsic relighting," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 673–678, Aug. 2004.

[3] X. Shen, Q. Yan, L. Xu, L. Ma, and J. Jia, "Multispectral joint image restoration via optimizing a scale map," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 12, pp. 2518–2530, Dec. 2015.

[4] X. Guo, Y. Li, J. Ma, and H. Ling, "Mutually guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 3, pp. 694–707, Mar. 2020.

[5] S. Xu et al., "A model-driven network for guided image denoising," *Inf. Fusion*, vol. 85, pp. 60–71, Sep. 2022.

[6] G. Oh, J. Back, J.-P. Heo, and B. Moon, "Robust image denoising of no-flash images guided by consistent flash images," in *Proc. AAAI Conf. Artif. Intell.*, 2023, vol. 37, no. 2, pp. 1993–2001.

[7] X. Shen, C. Zhou, L. Xu, and J. Jia, "Mutual-structure for joint filtering," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3406–3414.

[8] P. Li, J. Liang, M. Zhang, W. Fan, and G. Yu, "Joint image denoising with gradient direction and edge-preserving regularization," *Pattern Recognit.*, vol. 125, May 2022, Art. no. 108506.

[9] 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.

[10] Z. Xia, M. Gharbi, F. Perazzi, K. Sunkavalli, and A. Chakrabarti, "Deep denoising of flash and no-flash pairs for photography in low-light environments," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 2063–2072.

[11] S. Boyd et al., "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[12] R. A. Haddad and A. N. Akansu, "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.

[13] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.

[14] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. 6th Int. Conf. Comput. Vis.*, Jan. 1998, pp. 839–846.

[15] 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.

[16] 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.

[17] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

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

[19] F. Gao, X. Deng, M. Xu, J. Xu, and P. L. Dragotti, "Multi-modal convolutional dictionary learning," *IEEE Trans. Image Process.*, vol. 31, pp. 1325–1339, 2022.

[20] 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.*, Jun. 2014, pp. 2862–2869.

[21] Q. Guo, C. Zhang, Y. Zhang, and H. Liu, "An efficient SVD-based method for image denoising," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 5, pp. 868–880, May 2016.

[22] T. Xie, S. Li, and B. Sun, "Hyperspectral images denoising via non-convex regularized low-rank and sparse matrix decomposition," *IEEE Trans. Image Process.*, vol. 29, pp. 44–56, 2020.

[23] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, Jan. 1999, pp. 1033–1038.

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

[25] 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.

[26] 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.

[27] 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.

[28] Z. Yang et al., "A dynamic kernel prior model for unsupervised blind image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 26046–26056.

[29] J. Xia et al., "Blind super-resolution via meta-learning and Markov chain Monte Carlo simulation," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–18, May 2024.

[30] J.-Y. Xia, S. Li, J.-J. Huang, Z. Yang, I. M. Jaimoukha, and D. Gündüz, "Metalearning-based alternating minimization algorithm for nonconvex optimization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 9, pp. 5366–5380, Sep. 2022.

[31] Z. Yang et al., "Meta-learning based blind image super-resolution approach to different degradations," *Neural Netw.*, vol. 178, Oct. 2024, Art. no. 106429.

[32] D. Simon and M. Elad, "Rethinking the CSC model for natural images," in *Proc. 33rd Conf. Neural Inf. Process. (NeurIPS)*, vol. 32, 2019.

[33] B. Lecouat, J. Ponce, and J. Mairal, "Fully trainable and interpretable non-local sparse models for image restoration," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 238–254.

[34] J. Xu, X. Deng, and M. Xu, "Revisiting convolutional sparse coding for image denoising: From a multi-scale perspective," *IEEE Signal Process. Lett.*, vol. 29, pp. 1202–1206, 2022.

[35] C. Ren, Y. Pan, and J. Huang, "Enhanced latent space blind model for real image denoising via alternative optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 38386–38399.

[36] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, Jun. 2010, pp. 399–406.

[37] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, Jun. 2012.

[38] Q. Yan et al., "Cross-field joint image restoration via scale map," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1537–1544.

[39] Y. Li, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep joint image filtering," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 154–169.

[40] Y. Li, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Joint image filtering with deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1909–1923, Aug. 2019.

[41] 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, Feb. 2021.

[42] X. Deng, J. Xu, F. Gao, X. Sun, and M. Xu, "DeepM2CDL: Deep multi-scale multi-modal convolutional dictionary learning network," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 5, pp. 2770–2787, May 2024.

[43] J. Wang, T. Xue, J. T. Barron, and J. Chen, "Stereoscopic dark flash for low-light photography," in *Proc. IEEE Int. Conf. Comput. Photography (ICCP)*, May 2019, pp. 1–10.

[44] X. Lin, Y. Zhou, Y. Liu, and C. Zhu, "Level line guided interest point detection," *IEEE Signal Process. Lett.*, vol. 30, pp. 863–867, 2023.

[45] X. Lin, Y. Zhou, Y. Liu, and C. Zhu, "A comprehensive review of image line segment detection and description: Taxonomies, comparisons, and challenges," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–20, May 2024.

[46] Y. Aksoy et al., "A dataset of flash and ambient illumination pairs from the crowd," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 634–649.

[47] L. Murmann, M. Gharbi, M. Aittala, and F. Durand, "A dataset of multi-illumination images in the wild," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4079–4088.

[48] N. Capece, F. Banterle, P. Cignoni, F. Ganovelli, R. Scopigno, and U. Erra, "DeepFlash: Turning a flash selfie into a studio portrait," *Signal Process., Image Commun.*, vol. 77, pp. 28–39, Sep. 2019.

[49] I. Sobel et al., "A 3x3 isotropic gradient operator for image processing," in *A Talk at the Stanford Artificial Project in*, vol. 1968. Stanford, CA, USA: Stanford Artificial Intelligence Project (SAIL), 1968, pp. 271–272.

[50] H. Scharr, "Optimal operators in digital image processing," Ph.D. dissertation, Dept. Comput. Sci., Heidelberg Univ., Heidelberg, Germany, 2000.

[51] R. Jain et al., *Machine Vision*, vol. 5. New York, NY, USA: McGraw-Hill, 1995.

[52] K. An, "Sulla determinazione empirica di una legge didistribuzione," *Giorn Dell'inst Ital Degli Att*, vol. 4, pp. 89–91, Jan. 1933.

[53] E. Demidenko, "Kolmogorov-smirnov test for image comparison," in *Proc. Int. Conf. Comput. Sci. Appl.*, Assisi, Italy. Cham, Switzerland: Springer, 2004, pp. 933–939.

[54] H. Bristow, A. Eriksson, and S. Lucey, "Fast convolutional sparse coding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Portland, OR, USA, Jun. 2013, pp. 391–398.

[55] J. Sherman and W. J. Morrison, "Adjustment of an inverse matrix corresponding to a change in one element of a given matrix," *Ann. Math. Statist.*, vol. 21, no. 1, pp. 124–127, Mar. 1950.

[56] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, Nov. 2014.

[57] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, vol. 9351. Cham, Switzerland: Springer, 2015, pp. 234–241.

[58] R. Timofte et al., "NTIRE 2017 challenge on single image super-resolution: Methods and results," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jul. 2017, pp. 114–125.

[59] K. Zhang, L. Van Gool, and R. Timofte, "Deep unfolding network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3217–3226.

[60] M. Qiao et al., "HyperSOR: Context-aware graph hypernetwork for salient object ranking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 9, pp. 5873–5889, Sep. 2024.

[61] 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.

[62] Z. Tu et al., "MAXIM: Multi-axis MLP for image processing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 5769–5780.

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

[64] I. Marivani, E. Tsiligianni, B. Cornelis, and N. Deligiannis, "Designing CNNs for multimodal image restoration and fusion via unfolding the method of multipliers," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 5830–5845, Sep. 2022.

[65] Z. Sheng, X. Liu, S.-Y. Cao, H.-L. Shen, and H. Zhang, "Frequency-domain deep guided image denoising," *IEEE Trans. Multimedia*, vol. 25, pp. 6767–6781, 2022.

[66] X. Liu, Z. Sheng, and H.-L. Shen, "Frequency-relevant residual learning for multi-modal image denoising," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2022, pp. 86–90.

[67] W. Zhou, 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, pp. 600–612, 2004.

[68] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, Nonlinear Phenomena*, vol. 60, nos. 1–4, pp. 259–268, 1992.

[69] N. I. Radwan, N. M. Salem, and M. I. El Adawy, "Histogram correlation for video scene change detection," in *Proc. 2nd Int. Conf. Comput. Sci., Eng. Appl. (ICCSEA)*, vol. 1, New Delhi, India. Cham, Switzerland: Springer, 2012, pp. 765–773.