

A Comparison of Image Denoising Methods

Zhaoming Kong, Fangxi Deng, Haomin Zhuang, Xiaowei Yang,
Jun Yu, and Lifang He, *Member, IEEE*

Abstract—The advancement of imaging devices and countless images generated everyday pose an increasingly high demand on image denoising, which still remains a challenging task in terms of both effectiveness and efficiency. To improve denoising quality, numerous denoising techniques and approaches have been proposed in the past decades, including different transforms, regularization terms, algebraic representations and especially advanced deep neural network (DNN) architectures. Despite their sophistication, many methods may fail to achieve desirable results for simultaneous noise removal and fine detail preservation. In this paper, to investigate the applicability of existing denoising techniques, we compare a variety of denoising methods on both synthetic and real-world datasets for different applications. We also introduce a new dataset for benchmarking, and the evaluations are performed from four different perspectives including quantitative metrics, visual effects, human ratings and computational cost. Our experiments demonstrate: (i) the effectiveness and efficiency of representative traditional denoisers for various denoising tasks, (ii) a simple matrix-based algorithm may be able to produce similar results compared with its tensor counterparts, and (iii) the notable achievements of DNN models, which exhibit impressive generalization ability and show state-of-the-art performance on various datasets. In spite of the progress in recent years, we discuss shortcomings and possible extensions of existing techniques. Datasets, code and results are made publicly available and will be continuously updated at <https://github.com/ZhaomingKong/Denoising-Comparison>.

Index Terms—Image denoising, nonlocal self-similarity, block-matching filters, deep neural network, real-world images.

1 INTRODUCTION

Image denoising enjoys a long history and pioneering works [1], [2] may date back for decades. The primary goal of denoising is to enhance image quality by estimating underlying clean images from noisy observations. As a simple and special form of inverse problems [3], it has attracted much attention from both academia and industry. In real-world applications, it can be used as a pre-processing step for subsequent tasks, such as visual tracking [4], segmentation [5] and classification [6].

These years, the rapid development of modern imaging systems and technologies has largely enriched the information preserved and presented by an image, which can deliver more faithful representation for real scenes. For example, the rise and spread of advanced mobile phones facilitates the production of high-quality images and videos. In practice, noise removal has become a necessity for various imaging sensors and techniques such as multispectral/hyperspectral imaging (MSI/HSI) [7], [8], [9], magnetic resonance imaging (MRI) [10], computed topography (CT) [11] and ultrasound imaging [12]. Meanwhile, the increase of image size and dimension also puts forward higher requirements for denoising in terms of both effectiveness and efficiency. Therefore, the interest in the realm of denoising grows consistently with a large quantity of approaches [2], which may be roughly divided into two categories, namely *traditional denoisers* and *DNN methods*, depending on whether neural network architectures are utilized.

- Z. Kong, H. Zhuang and X. Yang are with the School of Software Engineering, South China University of Technology, Guangzhou, China (e-mail: kong.zm@mail.scut.edu.cn; semzm@mail.scut.edu.cn; xwyang@scut.edu.cn). Corresponding author: Zhaoming Kong.
- F. Deng is with Tencent Technology (email: fangxideng@outlook.com).
- J. Yu and L. He are with the Department of Computer Science and Engineering, Lehigh University, PA, USA (email: juy220@lehigh.edu; lih319@lehigh.edu).

Briefly, traditional denoisers normally filter out noise based solely on the input noisy observation by taking advantage of different regularization terms and image priors [13], [14]. The recent success and surge of related algorithms is largely credited to the famous block-matching 3D (BM3D) framework [15], which integrates the nonlocal self-similarity (NLSS) characteristic of natural images [16], sparse representation [17] and transform domain techniques [18] into a subtle paradigm. Since the birth of BM3D, there is no shortage of extensions originating from different disciplines. To name a few, Dabov et al. [19] improve BM3D by exploiting shape-adaptive image patches and principal component analysis (PCA). Maggioni et al. [20] and Rajwade et al. [21] introduce 3D cubes of voxels for high-dimensional data. Zhang et al. [22] replace the sparsity constraint with the low-rank assumption. Xu et al. [23] employ the Maximum A-Posterior (MAP) estimation technique [24] and propose a trilateral weighted sparse coding scheme.

Despite the steady improvements brought by classical denoisers, they suffer from several drawbacks [25] such as the need for solving complex optimization problems in the test phase, manual setting parameters and failure to exploit auxiliary information. To address these issues, DNN methods have been given an exceptionally large attention and shown promising results in image denoising [26]. The arrival of the deep learning era has significantly broadened the scope of denoising and infused new insights into the design of effective denoisers. For example, Zhang et al. [27] incorporate batch normalization (BN) [28], rectified linear unit (ReLU) [29] and residual learning [30] into the convolutional neural network (CNN) model. Chen et al. [31] introduce generative adversarial networks (GANs) [32] to resolve the problem of unpaired noisy images. Lefkimiatis et al. [33] and Davy [34] et al. combine NLSS and CNN to efficiently remove noise.

Accompanying the significant and inspiring progress of denoising algorithms, concerns may arise about their practical applicability, as a large proportion of approaches are verified on a limited number (often less than three) of datasets. With a considerable amount of existing methods [2], [10], [16], [18], [26], [35], [36], [37], [38], [39], [40], there still lacks a study on their performance for different image denoising tasks and applications. In this paper, we intend to narrow the gap by collecting and comparing various denoisers to investigate their effectiveness, efficiency, applicability and generalization ability with both synthetic experiments and real-world scenarios.

The main contributions of the paper are as follows.

(1) We construct a real-world dataset for image and video denoising tasks with a variety of digital cameras and smartphones. The dataset is composed of images and video sequences of both indoor and outdoor scenes under different lighting conditions, which serves as a good complement to current benchmark datasets.

(2) We compare over 60 denoising methods and perform extensive experiments in both synthetic and real-world cases for different denoising tasks and applications, including images, video sequences, 3D MRI volumes and MSI/HSI data. We adopt both subjective and objective metrics and evaluate the denoising quality of compared methods with quantitative results and human visual ratings.

(3) We make several interesting observations based on experimental results. For traditional denoisers, the BM3D family [15], [20], [41], [42], [43] still demonstrate very competitive performance for multiple denoising tasks and applications in terms of both effectiveness and efficiency. Besides, we argue that a modified singular value decomposition (M-SVD) method is able to produce similar results with many tensor-based approaches in image denoising. For DNN methods, many network architectures show significant improvements over traditional denoisers when fine-tuned with the training/validation samples, but the trained and predefined models may not generalize well to other datasets. Nevertheless, we identify related works that exhibit impressive generalizability. For example, FCCF [44], DRUNet [45] and PNGAN [46] produce state-of-the-art results on a number of real-world image datasets. FastDVDNet [47] and FloRNN [48] show impressive performance in the video denoising task in terms of both effectiveness and efficiency. Interestingly, many of these effective models such as FCCF, DRUNet and FastDVDNet take advantage of Gaussian noise modeling and denoisers.

The rest of this paper is organized as follows. Section 2 introduces background knowledge. Section 3 gives a brief review on related denoising techniques and datasets. Section 4 provides experimental results and discussions. Section 5 concludes the paper.

2 BACKGROUND

2.1 Symbols and Notations

In this paper, we mainly adopt the mathematical notations and preliminaries of tensors from [49] for image representation. Vectors and matrices are first- and second- order tensors which are denoted by boldface lowercase letters \mathbf{a} and capital letters \mathbf{A} , respectively. A higher order tensor

(the tensor of order three or above) is denoted by calligraphic letters, e.g., \mathcal{A} . An N th-order tensor is denoted as $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$. The n -mode product of a tensor \mathcal{A} by a matrix $\mathbf{U} \in \mathbb{R}^{P_n \times I_n}$, denoted by $\mathcal{A} \times_n \mathbf{U} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_{n-1} P_n I_{n+1} \times \dots \times I_N}$ is also a tensor. The mode- n matricization or unfolding of \mathcal{A} , denoted by $\mathbf{A}_{(n)}$, maps tensor elements (i_1, i_2, \dots, i_N) to matrix element (i_n, j) where $j = 1 + \sum_{k=1, k \neq n}^N (i_k - 1) J_k$, with $J_k = \prod_{m=1, m \neq n}^{k-1} I_m$. The Frobenius norm of a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ is defined as $\|\mathcal{A}\|_F = \sqrt{\sum_{i_1=1}^{I_1} \dots \sum_{i_N=1}^{I_N} \mathcal{A}_{i_1 \dots i_N}^2}$.

2.2 Noise Modeling

Let us consider a noisy observation \mathcal{Y} and its underlying clean image \mathcal{X} , a general assumption of noise distribution is additive white Gaussian noise (AGWN) with variance σ^2 represented by $\mathcal{N}(0, \sigma^2)$, and the degradation process is then given as

$$\mathcal{Y} = \mathcal{X} + \mathcal{N} \quad (1)$$

Indeed, noise modeling can be more complex and challenging in that noise in real-world images may be multiplicative and signal dependent [50]. Therefore, there are a plenty of non i.i.d Gaussian models tailored for the need of different applications, such as the mixed Gaussian impulse noise of grayscale and color images [51], [52], sparse random corruptions of video data [53], mixture noise removal of MSI/HSI [54], [55], and Rician noise reconstruction of MRI [20], [56]. In this paper, our synthetic experiments and analysis are mainly based on the AGWN model because: (i) the majority of existing methods are able to handle Gaussian noise, (ii) certain types of noise can be transformed to Gaussian distribution, and (iii) Romano et al. [57] recently pointed out that the removal of AGWN from an image is largely a solved problem, which may help explain the effectiveness of the simplified noise modeling in Eq. (1).

2.3 Nonlocal Self-similarity

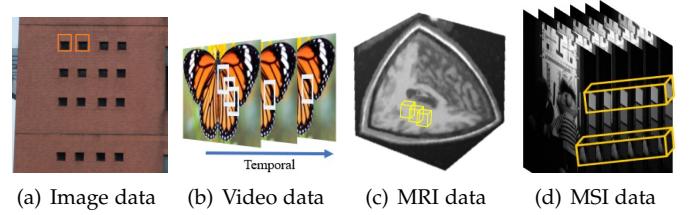


Fig. 1. The nonlocal self-similarity (NLSS) prior and patch representation of different imaging sensors, techniques and applications.

The nonlocal self-similarity (NLSS) property of natural images is probably the most important prior adopted by many different denoising methods. Briefly, NLSS refers to the fact that a local image patch often has many nonlocal similar patches to it across the image [58]. Usually, the similarity between two patches \mathcal{P}_A and \mathcal{P}_B is measured by their Euclidean distance $d_{AB} = \|\mathcal{P}_A - \mathcal{P}_B\|$. In practice, to save some time, the search for similar patches is restricted to a local window Ω_{SR} with pre-defined size. As illustrated in Fig. 1, the patch representation and the rule of similar patch search (SPS) may vary for different types of images. For example, SPS of grayscale/color images can be conducted only on

the single/luminance channel; for video sequences, SPS is performed along both temporal and spatial dimensions; and for MRI and MSI/HSI data, a patch could be represented by a 3D cube or a square tube with multiple spectral bands.

3 RELATED WORKS

In this section, we briefly introduce related denoising methods and datasets of different applications, which are summarized in Table 1, Table 2 and Table 3. More details can be found in the supplementary material and previous works [2], [10], [16], [18], [26], [36], [37], [38], [39], [59].

3.1 Traditional Denoisers

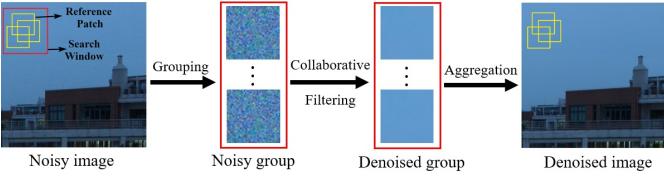


Fig. 2. Illustration of the grouping-collaborative filtering-aggregation framework for traditional denoisers.

For traditional denoisers, learning and denoising are usually accomplished only with the noisy image by leveraging the NLSS property. To achieve this goal, the most popular and successful framework is attributed to BM3D [15], which mainly follows three consecutive stages: grouping, collaborative filtering and aggregation. The flowchart of this effective three-stage paradigm is illustrated in Fig. 2.

3.1.1 Grouping

For every d -dimensional noisy image patch \mathcal{P}_n , based on certain patch matching criteria [64], [82], [84], [110], [111], [112], the grouping step stacks K similar (overlapping) patches located within a local window Ω_{SR} into a $d + 1$ -dimensional group. For example, consider a 3D patch $\mathcal{P}_n \in \mathbb{R}^{H \times W \times N}$, where H , W and N represents height, width and the number of channels or spectral bands, respectively, the 4D group of K patches can be directly represented by a fourth-order tensor $\mathcal{G}_n \in \mathbb{R}^{H \times W \times N \times K}$, or a 2D matrix $\mathbf{G}_n \in \mathbb{R}^{HWN \times K}$ if every patch \mathcal{P}_n is reshaped into a long vector $\mathbf{p}_n \in \mathbb{R}^{HWN}$.

3.1.2 Collaborative Filtering

Collaborative filters operate on the noisy patch group \mathcal{G}_n to estimate the corresponding underlying clean group \mathcal{G}_c via

$$\hat{\mathcal{G}}_c = \arg \min_{\mathcal{G}_c} \|\mathcal{G}_n - \mathcal{G}_c\|_F^2 + \rho \cdot \Psi(\mathcal{G}_c) \quad (2)$$

or in the matrix form

$$\hat{\mathbf{G}}_c = \arg \min_{\mathbf{G}_c} \|\mathbf{G}_n - \mathbf{G}_c\|_F^2 + \rho \cdot \Psi(\mathbf{G}_c) \quad (3)$$

where $\|\mathcal{G}_n - \mathcal{G}_c\|_F^2$ or $\|\mathbf{G}_n - \mathbf{G}_c\|_F^2$ indicates the conformity between the clean and noisy groups, and $\Psi(\cdot)$ is a regularization term for certain priors. For example, to model the nonlocal redundancies, the low-rank prior is adopted in [74], [79], [96], [98] with $\Psi(\mathbf{G}_c) = \|\mathbf{G}_c\|_*$ for matrix and $\Psi(\mathcal{G}_c) = \sum_{n=1}^N a_{(n)} \|\mathcal{G}_{c(n)}\|_*$ for tensor [113]. In addition,

the dictionary learning model with over-complete representations [17], [23], [67] is utilized to reconstruct \mathbf{G}_c with a dictionary \mathbf{D} and sparse coding coefficients \mathbf{C} via

$$\hat{\mathbf{C}} = \arg \min_{\mathbf{C}} \|\mathbf{G}_n - \mathbf{DC}\|_F^2 + \lambda \|\mathbf{C}\|_1 \quad (4)$$

where $\lambda \|\cdot\|$ is the regularization term that enforces sparsity constraint on \mathbf{C} . Once $\hat{\mathbf{C}}$ is computed, the latent clean patch group $\hat{\mathcal{G}}_c$ can be estimated as $\hat{\mathcal{G}}_c = \mathbf{DC}$. In [7] and [94], Eq. (4) is extended to tensor for MSI/HSI denoising with higher-order SVD (HOSVD) [114], [115] and tensor-SVD (t-SVD) [116], [117] transforms, respectively. A simple and effective method is to model the sparsity with certain thresholding techniques [60], [118] to attenuate the noise. For example, the hard-thresholding technique is adopted by the BM3D family and some state-of-the-art methods [21], [100], which attempts to shrink the coefficients $\mathcal{T}(\mathcal{G}_n)$ in the transform-domain [18] under a threshold τ via

$$\mathcal{G}_t = \begin{cases} \mathcal{T}(\mathcal{G}_n), & |\mathcal{T}(\mathcal{G}_n)| \geq \tau \\ 0, & |\mathcal{T}(\mathcal{G}_n)| < \tau \end{cases} \quad (5)$$

where \mathcal{T} represents a pre-defined or learned transform. The estimated clean group $\hat{\mathcal{G}}_c$ is obtained by inverting the transform via

$$\hat{\mathcal{G}}_c = \mathcal{T}^{-1}(\mathcal{G}_t) \quad (6)$$

It is noticed that SVD-based transforms are widely adopted among traditional denoisers, and such popularity largely results from the invertible orthogonal bases of SVD.

3.1.3 Aggregation

To further smooth out noise, the estimated clean patches of $\hat{\mathcal{G}}_c$ are averagely written back to their original location. More specifically, at the pixel level, every pixel \hat{p}_i of the denoised image is the (weighted) average of all pixels at the same position of filtered group $\hat{\mathcal{G}}_c$, which can be formulated as

$$\hat{p}_i = \sum_{\hat{p}_{i_k} \in \hat{\mathcal{G}}_c} w_{i_k} \hat{p}_{i_k} \quad (7)$$

where w_{i_k} and \hat{p}_{i_k} are weight and local pixel, respectively.

3.1.4 Discussion

The major difference among various traditional patch-based denoisers mainly lies in the collaborative filtering step, which often varies according to the choice of regularization terms, transforms and algebraic representation. Intuitively, reshaping the 4D group \mathcal{G} of Eq. (2) into the 2D matrix \mathbf{G} of Eq. (3) may break the internal structure of natural images. Therefore, a typical assumption [7], [21], [92], [94], [98], [100], [105], [119] is that tensor representation and decomposition techniques can help preserve more structure information, based on the fact that images can be naturally represented by multi-dimensional array. However, the conventional and widely-used tensor model may fall into the unbalance trap [120], [121], leading to unsatisfactory image restoration performance. In this paper, we further show that with slight modifications, a simple modified SVD (M-SVD) implementation may be able to produce competitive results compared with several tensor-based methods. The M-SVD approach is described in Algorithm 1 with more details given in the supplementary material.

TABLE 1
Representative traditional denoisers with different noise modeling techniques and representation.

| Category | Representation | Methods | year | Noise modeling | Key words |
|-----------------------|----------------|--------------------------|------|---------------------------|---|
| Traditional denoisers | Matrix | AWT [60], [61], [62] | 1995 | AWGN | Adaptive wavelet thresholding methods |
| | | DCT [63], [64] | 1999 | AWGN | Image denoising with discrete cosine transform (DCT) |
| | | NLM [16], [65] | 2005 | AWGN | nonlocal means (NLM) algorithm with Gaussian kernels |
| | | AMF [66] | 2005 | Impulse | Impulsive noise removal by adaptive median filter |
| | | K-SVD [17], [67], [68] | 2006 | AWGN | Over-complete dictionary learning and sparse coding |
| | | ONLM [69] | 2008 | AWGN and Rician | An optimized blockwise NLM Filter |
| | | ROLMMSE [70] | 2010 | Rician | A novel linear minimum mean square error estimator |
| | | AONLM [71] | 2010 | AWGN and Rician | Adaptive NLM with spatially varying noise levels |
| | | TCVD [72] | 2010 | AWGN | Combination of robust optical flow with NLM |
| | | NLPCA [73], [74], [75] | 2010 | AWGN | Applications of SVD transform |
| | | EPLL [14], [76] | 2011 | AWGN | A Bayesian method for whole image restoration |
| | | PRI-NLM [77] | 2012 | AWGN and Rician | A rotationally invariant version of the NLM filter |
| | | ODCT [77] | 2012 | AWGN and Rician | A 3D DCT implementation |
| | | LRMR [78] | 2013 | AWGN | A low-rank matrix decomposition model |
| | | MCWNNM [22], [79] | 2014 | AWGN | Extension of WNNM to color images |
| | | PRI-NLPCA [80] | 2015 | AWGN and Rician | A two stage filter based on NLPCA and PRI-NLM |
| | | NoiseClinic [81] | 2015 | AWGN and Realistic | A multiscale blind Bayes denoising algorithm |
| | | SPTWO [82] | 2016 | AWGN | NLPCA with optical flow estimation |
| | | LSCD [83] | 2016 | AWGN | Spectral component decomposition with line feature |
| | | Global-Search [84] | 2017 | AWGN | Efficient approximate global patch search |
| | | LSM-NLR [52] | 2017 | AWGN, Poisson and Impulse | Low-rank with laplacian scale mixture modeling |
| | | NMoG [54] | 2017 | AWGN, Stripe and Impulse | Low-rank matrix model with non i.i.d Gaussian noise |
| | | FastHyde [85] | 2018 | AWGN and Poisson | Low-rank and sparse representation |
| | | TWSC [23] | 2018 | AWGN | A trilateral weighted sparse coding scheme |
| | | VNLB [86] | 2018 | AWGN | A patch-based Bayesian model for video denoising |
| | | GID [87] | 2018 | AWGN | External data guided and internal prior learning |
| | | NLH [88] | 2020 | AWGN | Nonlocal pixel similarity with Haar transform |
| | | Bitonic [89] | 2022 | AWGN and Realistic | A non-learning bitonic filter with locally adaptive masks |
| | Tensor | CBM3D [41] | 2007 | AWGN | BM3D with opponent and YUV color mode transforms |
| | | LRTA [90] | 2008 | AWGN | A low Tucker rank tensor decomposition model |
| | | PARAFAC [91] | 2012 | AWGN | A rank-one candecomp/parafac (CP) model |
| | | BM4D [20] | 2012 | AWGN | An extention of BM3D to MSI and MRI using 3D patch |
| | | 4DHOSVD [21], [92], [93] | 2012 | AWGN | Applications of 4DHOSVD transform |
| | | TDL [7] | 2014 | AWGN | A Tucker based tensor dictionary learning algorithm |
| | | KTSDV [94] | 2015 | AWGN | A t-SVD based tensor dictionary learning algorithm |
| | | LRTV [95] | 2015 | AWGN | Low-rank model with total variation (TV) regularization |
| | | NLTA-LSM [96] | 2015 | AWGN and Poisson | Low Tucker rank with laplacian scale mixture modeling |
| | | ITSReg [97] | 2016 | AWGN | Intrinsic Tensor Sparsity Regularization |
| | | LLRT [98] | 2017 | AWGN | low-rank tensor with hyper-laplacian regularization |
| | | MSI-SVD [99], [100] | 2017 | AWGN | An efficient one-step t-SVD implementation |
| | | LLRGTV [101] | 2018 | AWGN | Low Tucker rank decomposition with total variation |
| | | WTR1 [102] | 2019 | AWGN | A weighted rank-one CP decomposition model |
| | | ILR-HOSVD [103] | 2019 | AWGN | A recursive low Tucker rank model with rank estimation |
| | | NCMeet [104] | 2019 | AWGN | Low-rank tensor model with iterative regularization |
| | | LTDL [105] | 2020 | AWGN | A low-rank tensor dictionary learning method |
| | | OLRT [106] | 2020 | AWGN and Stripe | A low-rank tensor model with sparse error component |
| | | GLF [107] | 2021 | AWGN | A low-rank tensor model with global and local factorization |
| | | HLTA [108] | 2022 | AWGN, Stripe and Impulse | A low-rank tensor model for denoising and completion |
| | | TRSTR [109] | 2023 | AWGN, Stripe and Impulse | A low-rank tensor-ring decomposition model |

Algorithm 1 Modified SVD (M-SVD)

Input: Noisy image \mathcal{A} , patch size ps , number of similar patches K and search window size Ω_{SR} .

Output: Estimated clean image $\hat{\mathcal{A}}_c$.

Step 1 (Grouping): For every reference patch of \mathcal{A} , stack K similar patches in a group \mathcal{G} within Ω_{SR} .

Step 2 (Collaborative filtering):

(1) Obtain the group and patch level transform \mathbf{U} and \mathbf{V} by performing SVD on $\mathbf{G}_{opp(3)}$ and $\mathbf{G}_{(4)}^T$, respectively.

(2) Apply the hard-threshold technique to $\mathbf{C} = \mathbf{U}^T \mathbf{G}_{(4)} \mathbf{V}$ via $\mathbf{C}_{trun} = \text{hard-threshold}(\mathbf{C})$.

(3) Take the inverse transform to obtain estimated clean group via $\hat{\mathbf{G}}_c = \mathbf{U} \mathbf{C}_{trun} \mathbf{V}^T$.

Step 3 (Aggregation): Averagely write back all image patches in $\hat{\mathbf{G}}_c$ to their original locations.

3.2 DNN Methods

3.2.1 Overview

The most recent development of image processing stems largely from the applications of deep learning techniques, which demonstrate outstanding performance in a wide vari-

ety of tasks [175]. Image denoising is not an exception. From the early plain networks [122], [176] to recently proposed generative and diffusion models [46], [173], numerous network architectures and frameworks have been developed with different training strategies, including supervised, self-supervised and unsupervised learning¹.

3.2.2 Supervised Methods

Different from traditional denoisers that use only internal information of the noisy observation, the supervised training strategy of DNN methods is often guided by external priors and data. The goal is to minimize the distance \mathcal{L} between predicted and clean images via

$$\min_{\theta} \sum_i \mathcal{L}(\mathcal{F}_{\theta}(\mathcal{Y}_i), \mathcal{X}_i) + \rho \cdot \Psi(\mathcal{F}_{\theta}(\mathcal{Y}_i)) \quad (8)$$

where \mathcal{L} can be measured by different loss functions [2], \mathcal{X}_i and \mathcal{Y}_i are clean-noisy image (patch) pairs, \mathcal{F}_{θ} with parameter θ is a nonlinear function that maps noisy patches onto predicted clean patches, and $\Psi(\cdot)$ represents certain

¹It is noticed that the terms *self-supervised*, *unsupervised* and *blind* denoising are often used interchangeably in the literature [40].

TABLE 2
Related DNN denoising methods with different noise modeling techniques and training strategy.

| Category | Training strategy | Methods | Year | Noise modeling | Key words |
|----------------------------------|-------------------|-------------------------|------|--------------------|--|
| DNN Methods | Supervised | MLP [122] | 2012 | AWGN | A multilayer perceptron model |
| | | TNRD [123] | 2016 | AWGN | A trainable nonlinear diffusion model |
| | | DnCNN [27] | 2017 | AWGN | A CNN model with BN and residual learning |
| | | IRCNN [124] | 2017 | AWGN | A CNN model with BN and dilated filters |
| | | NLNet [33] | 2017 | AWGN | A nonlocal CNN model for grayscale/color image denoising |
| | | UDNet [125] | 2018 | AWGN | A robust and flexible CNN model using UNet |
| | | HSID-CNN [126] | 2018 | AWGN | A 2D and 3D combined CNN model |
| | | VNLNet [34] | 2018 | AWGN | The first nonlocal CNN model for video denoising |
| | | FFDNet [127] | 2018 | AWGN | A flexible CNN model with tunable input noise levels |
| | | HSI-DeNet [128] | 2018 | AWGN and Strip | A CNN model for mixed noise removal |
| | | PRI-PB-CNN [129] | 2018 | AWGN and Rician | Combination of sliding window scheme and 3D CNN |
| | | GCBD [31] | 2018 | AWGN and Realistic | A GAN-based blind denoiser |
| | | DnGAN [130] | 2018 | AWGN and Blur | A GAN model with maximum a posteriori (MAP) framework |
| | | DBF [131] | 2019 | AWGN and Realistic | Integration of CNN into a boosting algorithm |
| | | FCCF [44] | 2019 | AWGN and Realistic | A deep fusion scheme of collaborative filtering (CBM3D) and CNN |
| | | DIDN [132] | 2019 | AWGN and Realistic | A deep iterative down-up CNN model |
| | | CBDNet [133] | 2019 | Realistic | A deep iterative blind denoising network |
| | | ViDeNN [134] | 2019 | AWGN | Combination of spatial and temporal filtering with CNN |
| | | RIDNet [135] | 2019 | AWGN and Realistic | A single stage model with feature attention |
| | | MIFCN [136] | 2019 | Realistic | A fully convolutional network model |
| | | DRDN [137] | 2019 | Realistic | A dynamic residual dense network model |
| | | SGN [138] | 2019 | Realistic | A self-guided network with top-down architecture |
| | | ADGAN [139] | 2019 | Realistic | An attentive GAN model with noise domain adaptation |
| | | QRNN3D [140] | 2020 | AWGN and Stripe | A deep recurrent neural network model with 3D convolution |
| | | CycleISP [141] | 2020 | Realistic | A GAN based framework modeling the camera pipeline |
| | | GCDN [142] | 2020 | AWGN and Realistic | A graph convolution network based denoising model |
| | | FastDVDNet [47] | 2020 | AWGN | A real-time video denoising network without flow estimation |
| Self-supervised/ Unsupervised | | DANet [143] | 2020 | Realistic | A Bayesian framework for noise removal and generation |
| | | LIDIA [144] | 2020 | AWGN | A lightweight model with instance adaptation |
| | | SADNet [145] | 2020 | AWGN | A CNN model with residual spatial-adaptive blocks |
| | | AIINDNet [146] | 2020 | AWGN and Realistic | A CNN model based on transfer learning scheme |
| | | MIRNet [147] | 2020 | Realistic | A multi-scale model with parallel convolution streams |
| | | PD-denoising [148] | 2020 | AWGN and Realistic | A CNN model with pixel-shuffle down-sampling adaptation |
| | | ADRNet [149] | 2020 | AWGN | A deep residual network model with channel attention scheme |
| | | MPRNNet [150] | 2021 | Realistic | A multi-stage network with cross-stage feature fusion and attention modules |
| | | DRUNet [151] | 2021 | AWGN | A plug-and-play method with deep denoiser prior |
| | | DudeNet [152] | 2021 | AWGN and Realistic | A dual network with four different blocks and sparse mechanism |
| | | InvDN [153] | 2021 | Realistic | An invertible denoising network with wavelet transformation |
| | | Restormer [153] | 2021 | AWGN and Realistic | An efficient Transformer model with multi-head attention networks |
| | | DCDiCL [154] | 2021 | AWGN | A deep convolution dictionary learning denoising network |
| | | NBNNet [155] | 2021 | AWGN and Realistic | A UNet based model that learns subspace basis and image projection |
| | | DeamNet [156] | 2021 | AWGN and Realistic | A CNN model with adaptive consistency prior and self-attention mechanism |
| | | PNGAN [46] | 2021 | AWGN and Realistic | A GAN based model for real image synthesis with domain alignment |
| | | SwinIR [157] | 2021 | AWGN | A shifted window (Swin) Transformer model with self-attention mechanism |
| | | NAFNet [158] | 2022 | Realistic | A UNet based model without nonlinear activation layers |
| | | MalleNet [159] | 2022 | AWGN and Realistic | A CNN with spatially-varying convolution kernels |
| | | Masked [160] | 2023 | AWGN and Realistic | A novel training approach that randomly masks pixels of the input image |
| | | Noise2Noise [161] | 2018 | AWGN and Poisson | A model trained on noisy image pairs of the same scene |
| | | SCGAN [162] | 2019 | AWGN | Unsupervised modeling with self-consistent GAN |
| | | Noise2Void [163] | 2019 | AWGN | A blind-spot masking strategy that excludes central pixels of the input |
| | | Self2Self [164] | 2020 | AWGN and Realistic | A dropout based strategy that generates noisy pairs with the Bernoulli sampler |
| | | C2N [165] | 2021 | Realistic | A GAN based model that generates real noisy images from arbitrary clean ones |
| | | UDVD [166] | 2021 | AWGN | An unsupervised video denoising network based on blind-spot strategy |
| | | R2R [167] | 2021 | AWGN and Realistic | A model trained on recorrupted noisy pairs |
| | | Neighbor2Neighbor [168] | 2021 | AWGN and Realistic | A model trained on noisy pairs generated by the downsampling strategy |
| | | IDR [169] | 2022 | AWGN and Realistic | A model trained on noisy pairs with an iterative refinement strategy |
| | | CVF-SID [170] | 2022 | Realistic | A CNN model with a self-supervised cycle |
| | | Blind2Unblind [171] | 2022 | AWGN and Realistic | A blind-spot network with a global-aware mask mapper |
| | | AP-BSN [172] | 2022 | Realistic | A blind-spot network with an asymmetric pixel-shuffle downsampling strategy |
| | | DDS2M [173] | 2023 | AWGN | A deep diffusion model with a variational spatio-spectral module |
| | | SASL [174] | 2023 | Realistic | A blind-spot network with spatially adaptive supervision |

regularizers [33], [177]. Early methods [178], [179] work with the known shift blur function and weighting factors. Burger et al. [122] show that a simple multi-layer perceptron (MLP) network is able to compete with representative traditional denoisers at certain noise levels. To extract latent features and exploit the self-similarity property of images, a widely used model is CNN [180], [181] with flexible size of convolution filters and local receptive fields. Fig. 3 illustrates a simple CNN denoising framework with three convolutional layers. Due to the effectiveness of CNN, its variations are quite extensive. To name a few, GCDN [142] utilizes graph convolution networks (GCNs) to capture self-similar information. SADNet [145] introduces a residual spatial-adaptive block and context block to sample related features and obtain multi-scale information. QRNN3D [140] exploits 3D convolutions to extract structural spatio-spectral correlation of MSI/HSI data.

3.2.3 Self-supervised and Unsupervised Methods

Collecting a large-scale matched noisy/clean image pairs for training is expensive and impractical, especially in the medical imaging sector [182]. Such limitation of supervised denoising prompts the development of self-supervised and unsupervised denoising networks.

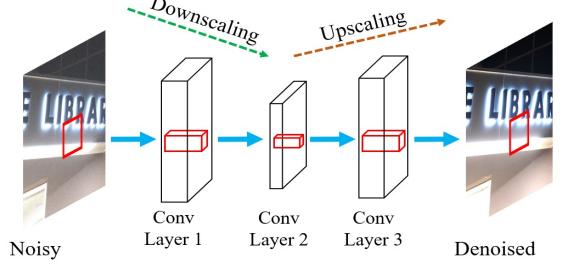


Fig. 3. A plain CNN denoising framework with three convolutional layers.

To get rid of the prerequisite on training data and leverage the power of DNNs, pioneering methods such as DIP [177] trains a network on a single image to fit itself by early stopping. SURE-based methods [183], [184] impose regularizations on DNNs to avoid overfitting. An interesting alternative strategy is to use noisy-noisy pairs for training [161]. For example, Noise2Self [185] and Noise2Void [163] introduce blind-spot learning by masking pixels of the noisy input. Self2Self [164] adopts a dropout-based scheme with noisy pairs generated by the Bernoulli sampler. Noise-As-Clean [186] and R2R [167] obtain noisy pairs by corrupting

the input with certain noise distributions and noise levels. Neighbor2Neighbor [168] creates noisy pairs via subsampling and pixel-wise independent noise assumption.

A recent appealing tool and new trend for self-supervised denoising is the generative model such as GANs [165], [182], [187] and diffusion models [173], [188], [189]. Briefly, thanks to the powerful image synthesis capability of GAN, it is able to learn a variety of complex noise distributions and thus generate more accurate realistic noisy images. The diffusion model can go beyond image synthesis [173], which intents to approximate the score function [2], [190] and then adopts an iterative algorithm.

3.2.4 Discussion

The penetration of deep learning in image denoising has beyond doubt pushed forward the frontiers of denoising. Despite the effectiveness of DNN methods, they may be double-edged swords, which enjoy three major advantages and also face the same challenges. First, by utilizing external information to guide the training process, DNN methods are not confined to the theoretical and practical bound of traditional denoisers [191]. However, the high quality training datasets and certain prior information such as ISO, shutter speed and camera brands are not always available in practice. Second, with the aid of advanced GPU devices for acceleration, real-time denoising [47], [192] is achievable for certain tasks, yet the expensive computational resources may not be accessible to ordinary users and researchers. Last but not least, the deep, powerful and sophisticated architectures are capable of capturing latent features underlying noisy images. But compared with benchmark traditional denoisers which only store several small pre-defined transform matrices, the complex networks with millions of parameters may drastically increase the storage cost.

3.3 Datasets

In this section, we provide a short overview of popular datasets for various denoising applications included in this paper and briefly introduce the proposed dataset. More information is available in the supplementary material and [44], [59], [193], [194], [195], [196], [197], [198], [199], [200], [201], [202], [203].

3.3.1 A Brief Overview

The statistics of popular datasets for different denoising applications is summarized in Table 3. Some examples are illustrated in Fig. 4. Typically, datasets used for synthetic experiments consist of noise-free (ground-truth) images acquired under ideal conditions with sufficient light and careful camera settings, while the corresponding noisy images are generated by manually adding noise of different levels and distributions to the noise-free ones. In many real-world applications, images are inevitably contaminated by noise to various degrees, often decided by the environments and imaging devices. In such cases, the ‘image averaging’ strategy is often adopted to generate the ‘ground-truth’ data by averaging a series of images captured based on the same, static scene. It is noteworthy that compared to grayscale/color images and videos, collecting MSI/HSI and MRI data is of greater difficulty and also more expensive.

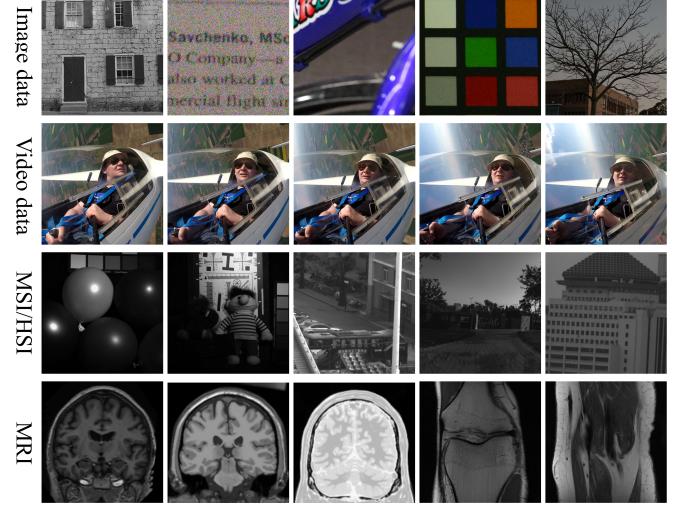


Fig. 4. Illustration of datasets for different applications. From the first row to the fourth row: grayscale/color image, video, MSI/HSI and MRI data.

TABLE 3

Popular datasets for synthetic and real-world experiments. ‘GT’: ground-truth, ‘✓’: Available, ‘-’: Not available, ‘F’: number of frames.

| Applications | Name | Experiments | GT | Image size | # Images |
|-----------------------|--------------------|-------------|----|-----------------------|----------|
| Grayscale/Color Image | Standard set [15] | Synthetic | ✓ | 512 × 512 / 256 × 256 | 11 |
| | BSD [204] | Synthetic | ✓ | 481 × 321 × 3 | 500 |
| | Kodak [205] | Synthetic | ✓ | 512 × 512 × 3 | 24 |
| | RENOIR [193] | Real-world | ✓ | 3684 × 2760 × 3 | 120 |
| | Nam-CC15 [194] | Real-world | ✓ | 512 × 512 × 3 | 15 |
| | Nam-CC60 [194] | Real-world | ✓ | 500 × 500 × 3 | 60 |
| | PolyU [195] | Real-world | ✓ | 512 × 512 × 3 | 100 |
| | DnD [206] | Real-world | - | 512 × 512 × 3 | 1000 |
| | SIDD [196] | Real-world | - | 256 × 256 × 3 | 1280 |
| | HighISO [44] | Real-world | ✓ | 512 × 512 × 3 | 100 |
| Video | IOCI | Real-world | ✓ | 1024 × 1024 × 3 | 848 |
| | Set8 [207] | Synthetic | ✓ | 960 × 540 × 3 × F | 8 |
| | DAVIS [197] | Synthetic | ✓ | 854 × 480 × 3 × F | 30 |
| | CRVD [198] | Real-world | ✓ | 1920 × 1080 × 3 × F | 61 |
| | PVDD [208] | Real-world | - | - | 200 |
| MSI/HSI | IOCV | Real-world | ✓ | 512 × 512 × 3 × F | 39 |
| | CAVE [199] | Synthetic | ✓ | 512 × 512 × 31 | 32 |
| | ICVL [200] | Synthetic | ✓ | 1392 × 1300 × 31 | 201 |
| | Indian Pines [209] | Synthetic | ✓ | 145 × 145 × 224 | 1 |
| | Urban [209] | Real-world | - | 307 × 307 × 210 | 1 |
| MRI | HHD [201] | Real-world | - | 1392 × 1040 × 31 | 77 |
| | Real-HSI [210] | Real-world | ✓ | 696 × 520 × 34 | 59 |
| | BrainWeb [202] | Synthetic | ✓ | 181 × 217 × 181 | 3 |
| | fastMRI [211] | Synthetic | ✓ | 320 × 320 × 40 | 50 |
| OASIS [203] | OASIS | Real-world | - | 256 × 256 × 128 | 2 |

3.3.2 The Proposed Dataset

In this subsection, we briefly introduce the motivation and details regarding the setup and protocol followed by our indoor-outdoor color image (IOCI) and video (IOCV) dataset, some examples are illustrated in Fig. 5.

IOCI. To capture images of various scenes, 13 different camera devices are used to collect data in both indoor and outdoor environments. To reduce human interference, we mostly utilize the cameras’ *auto mode* instead of predefined camera settings such as ISO, shutter speed and aperture [44], [195], [196]. Captured images that display obvious misalignment and illumination differences are discarded.

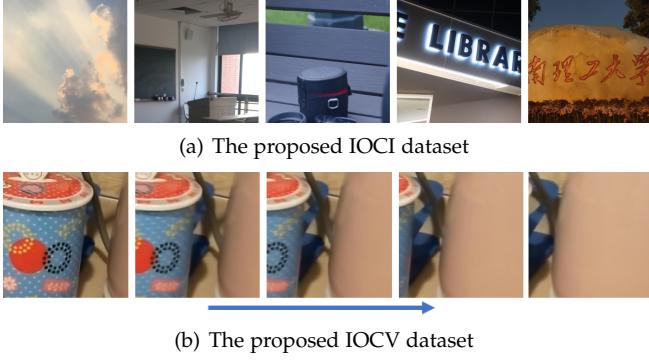


Fig. 5. Illustration of the proposed IOCI and IOCV dataset.

IOCV. To obtain reference videos for benchmarking, we adopt a video-by-video strategy. Instead of manually controlling static objects [198], we propose to move cameras automatically. The procedure of generating mean videos as ground-truth is illustrated in Fig. 6. We fix the cameras onto a rotatable tripod ball-head placed on top of a motorized slider. The slider and the tripod ball-head can be set to repeatedly move and rotate at different speeds, which simulate the movement of observed objects in more than one directions. Both the slider and cameras are controlled remotely to reduce human interference.

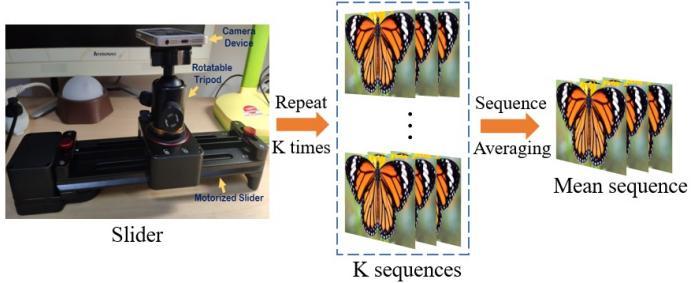


Fig. 6. The procedure of generating mean video sequences with a motorized slider. The camera is fixed to a rotatable tripod ball-head placed on top of the slider.

4 EXPERIMENTS

In this section, we report the results of different methods/models for the denoising task of images, videos, MSI/HSI and MRI data. All implementations and source codes are provided by the authors or downloaded from the authors' personal websites. For methods that require GPU devices, we resort to Google ColabPro's computing resources. All other experiments are performed on a computer equipped with Core(TM) i5-7500 @ 3.4 GHz and 16GB RAM.

4.1 Results for Real-World Image Datasets

4.1.1 Experimental Settings

We compare the performance of over 40 methods or models with real-world image datasets, including DnD, SIDD, Nam (CC15 [79], [194] and CC60 [87], [194]), PolyU, High-ISO and our IOCI, where SIDD dataset provides training/validation samples. The clean-noisy image pairs of DnD and SIDD are not available, and the test results can be obtained by uploading denoised images online. In this paper, we focus

on the sRGB color space since the raw data [212], [213] are not always accessible.

Typically, the decisive parameters for traditional denoisers are input noise level, patch size and the number of local similar patches, etc, whereas the number of layers, learning rate and weight size are essential for DNN methods. However, many existing methods are designed for synthetic experiments, while ideally, in real-world cases, the parameters of all compared methods should be fine-tuned to obtain the best possible performance. But this can be computationally expensive and the clear reference images for training and validation are always not available. Therefore, a more practical way is to selectively adopt effective pretrained models or predefined parameter settings. Since many DNN methods offer two to six pretrained models with different parameter settings for testing, for fair comparison, we select the input noise level for each Gaussian denoiser from four different values, which may be regarded as the equivalence of four pretrained models corresponding to *low*, *medium-low*, *medium-high* and *high* denoising modes. We report the best average values of all compared methods on each dataset. Peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [214] are employed for objective evaluations. Normally, the higher the PSNR and SSIM values, the better the quality of denoised images.

4.1.2 Objective Results

Detailed denoising results are presented in Table 4. Briefly, it is observed that many DNN methods show outstanding performance on both DnD and SIDD datasets, but they do not always demonstrate evident advantages over traditional denoisers on other datasets when training or validation data are not available.

More specifically, for traditional denoisers, three effective transform-domain approaches namely CBM3D, CMSt-SVD, and NLHCC show similar denoising performance on almost all datasets. Besides, the effectiveness of CBM3D1 and CMSt-SVD indicates that a one-step implementation with a small number of local patches and the hard-thresholding technique is able to produce very competitive results. In addition, it is noticed that the matrix-based methods such as MCWNNM and the proposed M-SVD present comparable denoising capability with several tensor-based methods such as LLRT and 4DHOSVD, which indicates that the use of tensor representation may not significantly boost structural information retrieval in realistic cases.

For DNN methods, we can see that with the aid of training or validation process, models targeting real-world denoising such as AINDNet, DIDN, NAFNet and Restormer produce much better performance on SIDD and DnD datasets. For example, compared to CBM3D, Restormer provides the PSNR improvements of 1.99 dB on SIDD and 4.79 dB on DnD, respectively. However, once the testing data is no longer compatible with the training conditions, they can exhibit poor generalization and lead to overfitting or degrading performance. As a result, none of the pretrained models can significantly advance benchmark traditional denoisers on other real-world datasets. Nevertheless, we also notice that several network frameworks such as DBF, DIDN, DRUNet, FCCF and PNGAN show competitive and robust performance in the absence of training data. Furthermore, it

TABLE 4

Average PSNR, SSIM values on real-world color image datasets. The average time is calculated based on the PolyU dataset. '-' means the results are not available. 'C: CANON', 'F: FUJIFILM', 'H: HUAWEI', 'I: IPHONE', 'N: NIKON', 'O: OPPO', 'S: SONY', 'X: XIAOMI'.

| Methods/Models | DnD | SIDD | CC15 | CC60 | PolyU | HighISO | IOCI | | | | | | | | | | | | | | Time (s) |
|-----------------------|----------|-------|-------|-------|-------|---------|--------|--------|-----------|---------|-----------|-------------|-------|-------|-------|---------|--------|---------|-------|---------|----------|
| | | | | | | | C-100D | C-600D | C-5DMark4 | F-X100T | H-Honor6X | H-Mate40Pro | I-5S | I-6S | I-13 | N-D5300 | O-R11s | S-A6500 | X-M1s | | |
| - | # Images | 1000 | 1280 | 15 | 60 | 100 | 100 | 55 | 25 | 83 | 71 | 30 | 126 | 36 | 67 | 174 | 56 | 39 | 36 | 50 | |
| LSCD [83] | - | - | 36.20 | 37.69 | 37.97 | 38.64 | 40.65 | 41.45 | - | - | 39.22 | - | 39.84 | 39.53 | - | 40.83 | - | 44.34 | 35.86 | 3.19 | |
| GID [87] | - | - | 0.942 | 0.963 | 0.964 | 0.962 | 0.973 | 0.979 | - | - | 0.963 | - | 0.963 | 0.963 | - | 0.973 | - | 0.987 | 0.957 | | |
| Bitonic [89] | 34.32 | 27.16 | 37.17 | 38.41 | 38.37 | 39.63 | 40.86 | 41.60 | 44.11 | 42.31 | 39.52 | 38.66 | 40.12 | 40.16 | 41.50 | 41.09 | 40.50 | 44.96 | 36.11 | 49.99 | |
| LLRT [98] | 0.817 | 0.634 | 0.946 | 0.963 | 0.967 | 0.967 | 0.974 | 0.979 | 0.989 | 0.975 | 0.965 | 0.972 | 0.964 | 0.967 | 0.979 | 0.973 | 0.973 | 0.989 | 0.960 | 2.57 | |
| MCWNNM [79] | 37.85 | 36.67 | 35.22 | 35.98 | 36.64 | 37.37 | 39.38 | 39.63 | 40.76 | 41.05 | 37.71 | 37.41 | 38.98 | 38.04 | 39.09 | 39.22 | 38.87 | 43.25 | 34.92 | | |
| TWSC [23] | 0.936 | 0.933 | 0.925 | 0.931 | 0.940 | 0.943 | 0.959 | 0.952 | 0.965 | 0.964 | 0.940 | 0.956 | 0.947 | 0.939 | 0.952 | 0.954 | 0.959 | 0.979 | 0.941 | | |
| Traditional denoisers | 37.85 | 36.67 | 35.22 | 35.98 | 36.64 | 37.37 | 39.38 | 39.63 | 40.76 | 41.05 | 37.71 | 37.41 | 38.98 | 38.04 | 39.09 | 39.22 | 38.87 | 43.25 | 34.92 | | |
| WTR1 [102] | - | - | 37.80 | 37.32 | 38.20 | 38.19 | - | - | - | - | - | - | - | - | - | - | - | - | - | 600.33 | |
| M-SVD | 37.45 | 33.51 | 37.63 | 39.36 | 38.57 | 40.27 | 41.81 | 42.55 | 44.39 | 42.68 | 39.99 | 38.93 | 40.78 | 40.52 | 41.72 | 42.18 | 40.73 | 45.71 | 36.37 | | |
| 4DHOSDV [21] | 0.924 | 0.867 | 0.954 | 0.971 | 0.967 | 0.971 | 0.979 | 0.983 | 0.989 | 0.976 | 0.967 | 0.973 | 0.969 | 0.969 | 0.977 | 0.977 | 0.974 | 0.990 | 0.961 | 104.24 | |
| CBM3D1 [41] | 37.58 | 34.49 | 37.52 | 39.15 | 38.54 | 40.27 | 41.40 | 42.19 | 44.49 | 42.60 | 39.82 | 38.88 | 40.59 | 40.37 | 41.75 | 41.82 | 40.71 | 45.58 | 36.27 | | |
| CBM3D2 [41] | 0.929 | 0.911 | 0.956 | 0.973 | 0.968 | 0.973 | 0.977 | 0.982 | 0.990 | 0.976 | 0.966 | 0.973 | 0.965 | 0.967 | 0.980 | 0.977 | 0.974 | 0.990 | 0.961 | 123.06 | |
| CMSt-SVD [100] | 38.04 | 35.00 | 37.58 | 39.21 | 38.60 | 40.11 | 41.58 | 42.40 | 44.43 | 42.53 | 39.93 | 38.84 | 40.56 | 40.41 | 41.93 | 42.29 | 40.62 | 45.58 | 36.24 | 1.02 | |
| NLH-CC [88] | 37.73 | 34.74 | 37.70 | 39.41 | 38.69 | 40.35 | 41.69 | 42.54 | 44.74 | 42.65 | 39.97 | 38.97 | 40.77 | 40.55 | 42.03 | 42.20 | 40.75 | 45.72 | 36.38 | | |
| AINDNet [146] | 0.940 | 0.900 | 0.959 | 0.976 | 0.971 | 0.974 | 0.979 | 0.984 | 0.991 | 0.977 | 0.967 | 0.973 | 0.967 | 0.967 | 0.982 | 0.980 | 0.974 | 0.991 | 0.962 | | |
| AP-BSN [172] | 38.85 | - | 38.49 | 39.86 | 38.36 | 40.29 | 41.77 | 42.72 | 43.66 | 42.80 | 38.84 | 38.31 | 40.44 | 39.94 | 41.13 | 43.25 | - | 46.02 | 35.73 | 40.32 | |
| Blind2Unblind [171] | 0.953 | - | 0.965 | 0.976 | 0.965 | 0.971 | 0.979 | 0.974 | 0.984 | 0.986 | 0.978 | 0.959 | 0.968 | 0.963 | 0.976 | 0.981 | 0.991 | 0.955 | | | |
| BRDNet [215] | 33.80 | - | 37.27 | 39.16 | 38.04 | 39.64 | 39.85 | 40.95 | 43.54 | 40.27 | 39.48 | 38.66 | 38.19 | 40.07 | 41.92 | 41.34 | 39.66 | 42.76 | 36.08 | 14.66 | |
| C2N [165] | 37.28 | - | 37.02 | - | 37.69 | 38.86 | 41.12 | 41.95 | 42.61 | 41.33 | 38.73 | 38.15 | 38.95 | 38.67 | 40.50 | 40.78 | 40.05 | 44.64 | 35.53 | | |
| CBDNet [133] | 38.06 | 33.26 | 36.20 | 37.67 | 37.81 | 38.18 | 41.43 | 42.41 | 42.55 | 41.88 | 38.35 | 38.17 | 39.80 | 39.07 | 40.63 | 40.92 | 39.54 | 44.38 | 35.54 | 0.04 | |
| CFV-SID [170] | 36.31 | 34.43 | 29.13 | 31.02 | 33.08 | 33.35 | 36.68 | 39.08 | 39.73 | 36.40 | 33.13 | 33.16 | 32.07 | 33.33 | 36.92 | 38.40 | 35.22 | 38.56 | 29.89 | 0.12 | |
| CycleISP [141] | 39.57 | 39.42 | 35.40 | 36.87 | 37.61 | 37.70 | 41.26 | 41.84 | 42.64 | 41.59 | 38.47 | 38.01 | 38.79 | 38.67 | 40.61 | 40.03 | 39.86 | 44.75 | 35.54 | 0.26 | |
| DANet [143] | 39.55 | 39.43 | 37.17 | 37.63 | 38.63 | 41.49 | 38.92 | 41.59 | 42.59 | 42.83 | 41.70 | 37.91 | 38.17 | 39.23 | 38.74 | 41.58 | 41.54 | 39.58 | 44.03 | 36.21 | 0.02 |
| DBF [131] | 0.953 | 0.955 | 0.953 | 0.969 | 0.977 | 0.960 | 0.979 | 0.981 | 0.990 | 0.970 | 0.967 | 0.971 | 0.962 | 0.958 | 0.976 | 0.970 | 0.971 | 0.986 | 0.956 | | |
| DCDiC-L [154] | - | - | 0.960 | 0.981 | 0.970 | 0.974 | 0.965 | 0.974 | 0.977 | 0.990 | 0.986 | 0.988 | 0.966 | 0.975 | 0.956 | 0.945 | 0.977 | 0.964 | 0.970 | 0.987 | 0.953 |
| DeamNet [156] | 39.63 | 39.35 | 36.63 | - | 37.70 | 36.93 | 40.90 | 40.86 | 41.22 | 39.72 | 33.67 | 36.40 | 37.87 | 35.07 | 40.25 | 38.92 | 39.56 | 40.52 | 34.61 | 0.18 | |
| DIDN [132] | 39.64 | 39.78 | 36.06 | - | 37.36 | 38.24 | 40.74 | 41.68 | 42.35 | 41.17 | 38.15 | 37.87 | 38.27 | 38.58 | 40.03 | 40.32 | 39.73 | 44.07 | 35.28 | 7.25 | |
| DnCNN [27] | 37.90 | 37.73 | 37.47 | 39.32 | 38.51 | 40.01 | 40.81 | 41.91 | 44.16 | 41.57 | 39.92 | 38.72 | 39.34 | 40.22 | 42.03 | 42.12 | 40.26 | 43.97 | 36.30 | | |
| DRUNet [45] | - | - | 38.30 | 40.33 | 38.93 | 40.77 | 41.94 | 42.62 | 45.36 | 42.78 | 40.35 | 40.48 | 40.80 | 40.26 | 42.86 | 40.88 | 45.76 | 36.61 | 0.17 | | |
| FFDNet [127] | 37.61 | 38.27 | 37.67 | 39.73 | 38.76 | 40.28 | 41.67 | 42.55 | 44.77 | 42.44 | 38.05 | 37.80 | 38.60 | 38.55 | 40.28 | 40.46 | 39.73 | 43.66 | 35.42 | | |
| IDR [169] | 0.942 | 0.948 | 0.956 | 0.977 | 0.970 | 0.973 | 0.977 | 0.982 | 0.992 | 0.990 | 0.976 | 0.964 | 0.960 | 0.961 | 0.948 | 0.966 | 0.976 | 0.989 | 0.961 | | |
| InvDN [153] | 39.57 | 39.28 | 34.55 | 36.33 | 35.94 | 38.01 | 38.95 | 40.09 | 42.03 | 40.02 | 33.32 | 37.00 | 36.64 | 37.66 | 40.06 | 40.98 | 39.10 | 40.74 | 34.15 | 0.74 | |
| IRCNN [124] | 0.952 | 0.955 | 0.937 | 0.953 | 0.947 | 0.952 | 0.965 | 0.973 | 0.978 | 0.960 | 0.953 | 0.965 | 0.959 | 0.955 | 0.966 | 0.972 | 0.974 | 0.981 | 0.953 | | |
| MIRNet [147] | 39.88 | 39.55 | 36.06 | 37.25 | 37.49 | 38.10 | 40.72 | 41.68 | 42.76 | 41.05 | 38.28 | 37.80 | 38.60 | 38.55 | 40.28 | 40.46 | 39.73 | 43.66 | 35.42 | 0.71 | |
| MPRNet [150] | 39.82 | 39.57 | 35.94 | - | 37.50 | 38.04 | 40.59 | 41.30 | 42.59 | 41.23 | 38.35 | 37.92 | 38.95 | 38.62 | 40.21 | 40.20 | 39.77 | 44.05 | 35.49 | | |
| NAFNet [158] | 38.36 | 40.15 | 34.39 | - | 36.38 | 37.88 | 40.12 | 40.23 | 40.66 | 40.01 | 36.13 | 35.87 | 36.66 | 36.79 | 36.53 | 39.93 | 39.32 | 40.45 | 34.82 | 0.35 | |
| NBNNet [155] | 0.955 | 0.958 | 0.935 | - | 0.957 | 0.955 | 0.977 | 0.980 | 0.982 | 0.971 | 0.955 | 0.967 | 0.956 | 0.955 | 0.972 | 0.983 | 0.977 | 0.984 | 0.953 | | |
| Noise2Noise [161] | - | - | 34.47 | 35.65 | 37.10 | 36.59 | 40.22 | 40.41 | 41.29 | 41.24 | 38.13 | 37.94 | 39.71 | 38.50 | 39.63 | 38.05 | 39.17 | 44.49 | 35.30 | | |
| Neigh2Neigh [168] | - | - | 0.883 | 0.907 | 0.942 | 0.908 | 0.967 | 0.960 | 0.966 | 0.958 | 0.940 | 0.961 | 0.956 | 0.936 | 0.952 | 0.919 | 0.958 | 0.964 | 0.944 | | |
| NLNet [33] | - | - | 37.56 | 39.70 | 38.58 | 40.17 | 41.43 | 42.18 | 44.53 | 42.19 | 39.96 | 38.89 | 40.04 | 40.61 | 42.14 | 42.16 | 40.63 | 44.87 | 36.34 | | |
| PD-Denoising [148] | 38.40 | 33.99 | 35.85 | 37.13 | 37.13 | 37.12 | 40.85 | 41.38 | 41.13 | 41.05 | 37.72 | 37.57 | 39.03 | 38.01 | 39.18 | 39.64 | 38.80 | 43.73 | 35.05 | 0.61 | |
| PNGAN [46] | 39.38 | 39.81 | 38.44 | - | 39.57 | 40.51 | 41.79 | 43.13 | 44.73 | 42.55 | 39.84 | 38.78 | 40.63 | 40.20 | 41.85 | 43.15 | 45.40 | 45.32 | 36.40 | | |
| R2R [167] | - | - | 34.78 | 36.32 | 38.11 | 38.57 | 38.71 | - | - | - | - | - | - | - | - | - | - | - | - | 4481.98 | |
| Restormer [216] | - | - | 0.844 | 0.937 | 0.961 | 0.970 | 0.958 | - | - | - | - | - | - | - | - | - | - | - | - | | |
| RIDNet [135] | 40.03 | 40.02 | 36.33 | - | 37.66 | 38.29 | 41.10 | 41.84 | 42.49 | 41.47 | 38.42 | 38.08 | 39.05 | 38.77 | 40.13 | 40.53 | 39.56 | 44.19 | 35.65 | 0.77 | |
| SADNet [145] | 0.956 | 0.960 | 0.941 | - | 0.956 | 0.948 | 0.977 | 0.979 | 0.976 | 0.973 | 0.963 | 0.966 | 0.949 | 0.963 | 0.962 | 0.963 | 0.968 | 0.953 | 0.953 | | |
| SASL [174] | 38.00 | 37.41 | 34.93 | - | 37.13 | 38.24 | 40.04 | 41.86 | 43.07 | 40.29 | 37.22 | 37.38 | 36.45 | 37.64 | 40.85 | 41.64 | 39.10 | 42.89 | 34.25 | 0.06 | |
| Self2Self [164] | - | - | 36.26 | 38.23 | - | 39.49 | - | - | - | - | - | - | - | - | - | - | - | - | - | 4100.76 | |
| UDNet [217] | - | - | 0.947 | 0.969 | - | 0.963 | - | - | - | - | - | - | - | - | - | - | - | - | - | | |
| VDIR [218] | 39.63 | 39.26 | 34.90 | 36.51 | 37.01 | 36.84 | 40.84 | 41.40 | 41.29 | 40.81 | 37.65 | 37.84 | 39.09 | 38.17 | 39.20 | 39.41 | 38.98 | 44.40 | 35.04 | 2.03 | |
| VDNet [219] | - | - | 0.945 | - | 0.957 | 0.955 | 0.976 | 0.964 | 0.982 | 0.987 | | | | | | | | | | | |

is noteworthy that DBF, DRUNet and FCCF utilize Gaussian noise modeling and denoisers, which supports the effectiveness of incorporating Gaussian noise modeling.

From the perspective of denoising speed, DNN methods are normally much faster than traditional denoisers in the test phase thanks to the power of advanced GPU devices. Specifically, efficient DNN models are able to handle images of size $512 \times 512 \times 3$ within 0.2 seconds. We also notice that representative self-supervised DNN methods such as Self2Self and R2R bear high computational burden, since for each noisy input, the training process involves thousands of iterations. For traditional denoisers, the time complexity lies mainly in the iterative local patch search and learning. For example, M-SVD spends 26 and 42 seconds on grouping and performing local SVD transforms, respectively. But it is slightly faster than its tensor counterpart 4DHOSVD, because it avoids folding and unfolding operations of high dimensional data along different modes. Among all the traditional denoisers, the state-of-the-art CBM3D is the most efficient because its grouping step is performed only on the luminance channel, and it does not need to train local transforms or solve optimization problems.

4.1.3 Visual Evaluation

Denoised results of compared methods on the DND and SIDD datasets are illustrated in Fig. 7. The drawback of traditional patch-based denoisers is obvious when dealing with severely corrupted images, since the patch search and local transform learning steps are adversely affected by the presence of noise. Consequently, it is not difficult to see the distortion, artifacts, loss of true color and details. By comparison, fine-tuned DNN models show clear advantages in terms of both noise removal and detail recovery, which demonstrates their powerful feature learning and extraction capability. Interestingly, the difference among the sophisticated and well-trained networks is barely noticeable in many cases of the two benchmark datasets. Therefore, we present visual evaluations of the PolyU and the IOCI datasets in Fig. 8 and Fig. 9, respectively. When facing different and unseen noise patterns, the pretrained DNN models may also leave unwanted artifacts and produce over-smooth effects to varying degrees, while traditional denoisers show their strengths by exploiting nonlocal information of the noisy image, which renders some robustness and adaptability. Nevertheless, we notice that several DNN models such as DRUNet, FCCF and PNGAN show impressive generalizability and achieve good balance between noise removal and detail preservation. The observation is consistent with results reported in Table 4.

4.2 Results for Video Datasets

4.2.1 Experimental Settings

Compared to images, videos are more engaging and informative by recording and displaying dynamic objects. Four benchmark datasets are included for video denoising evaluation. Briefly, DAVIS-2017-test-dev-480p and Set8 are used for synthetic experiments with Gaussian noise, and the entire CRVD and IOCV for real-world experiments. Our evaluations mainly focus on video denoising in the sRGB space, and the parameters and models of compared methods are carefully chosen in the same way as section

4.1.1. Similar to [47], the PSNR and SSIM of a sequence are computed as their average values of each frame.

4.2.2 Objective Results

Table 5 lists the average PSNR and SSIM results of compared methods. For traditional denoisers, it is noticed that CVMSt-SVD and VBM4D are able to achieve state-of-the-art results on the CRVD and IOCV datasets, respectively. This demonstrates the effectiveness of patch-based paradigm on capturing NLSS features among different frames. For DNN methods, representative models such as DVDNet, FastDVDNet, FloRNN, MMNet and VNLNet show dominating performance, especially when dealing with high Gaussian noise levels, which manifests their ability to extract deep features and exploit spatio-temporal information. When it comes to real-world experiments, although DNN methods are no longer significantly superior in the absence of corresponding training data, certain Gaussian-based models exhibit impressive generalization ability. For example, benefiting from the recurrent modules, FloRNN provides the SSIM improvements of 0.0142 on the CRVD dataset. Besides, by integrating the NLSS prior into CNN models, VNLNet achieves state-of-the-art results on the IOCV dataset.

In terms of denoising efficiency, FastDVDNet achieves almost real-time video denoising by getting rid of the time-consuming patch search and explicit flow estimation steps. Specifically, it takes FastDVDNet less than 0.1 second to process a single video frame of size $960 \times 540 \times 3$, which is 8 times faster than FloRNN, and at least 100 times faster than the benchmark traditional denoisers such as VBM4D and CVMSt-SVD. The denoising speed of FastDVDNet is remarkable considering its competitive performance in different cases, making it an exceedingly appealing denoising algorithm in practice.

4.2.3 Visual Evaluation

Visual comparison is presented in Fig. 10. From the results of synthetic experiments, we can observe that Gaussian-based DNN models such as FloRNN and VNLNet produce impressive and pleasant visual effects by suppressing noise and restoring true colors and details, while traditional denoisers CVMSt-SVD and VBM4D struggle to remove Gaussian noise and therefore generate unwanted color artifacts. Interestingly, from the results of CRVD data, it can be seen that the powerful DNN models may lead to more obvious over-smooth effects in real-world experiments. In this case, we notice that the background is static, the toy dog on the wheels is dynamic and moves fast in more than one directions, thus some details and textures are present only in certain frames. Therefore, traditional denoisers are able to benefit from their NLSS framework to leverage spatial similarity and preserve more structural information.

4.2.4 Human Ratings

Due to the limitations of hardware equipments and environment, the videos acquired by the motorized slider also inevitably exhibit some noise, flickering, staircase effects, motion blur and misalignment that will undermine the accuracy of objective evaluations. In addition, the quality of videos may not be assessed frame-by-frame. Therefore, we conduct additional qualitative evaluations by collecting

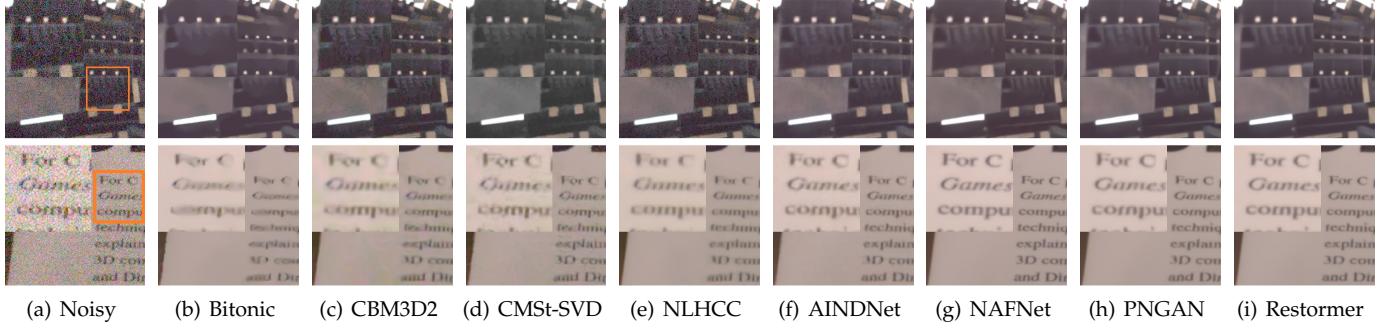


Fig. 7. Visual evaluation of compared methods on real-world datasets. First row: DND, second row: SIDD.

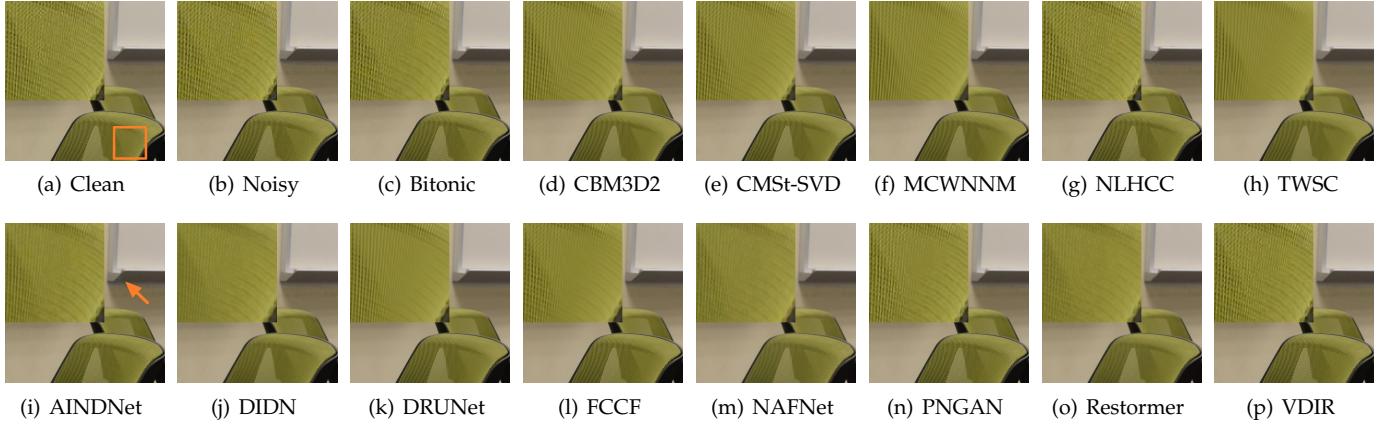


Fig. 8. Visual evaluation of compared methods on the real-world PolyU dataset. The camera device is NIKON D800.

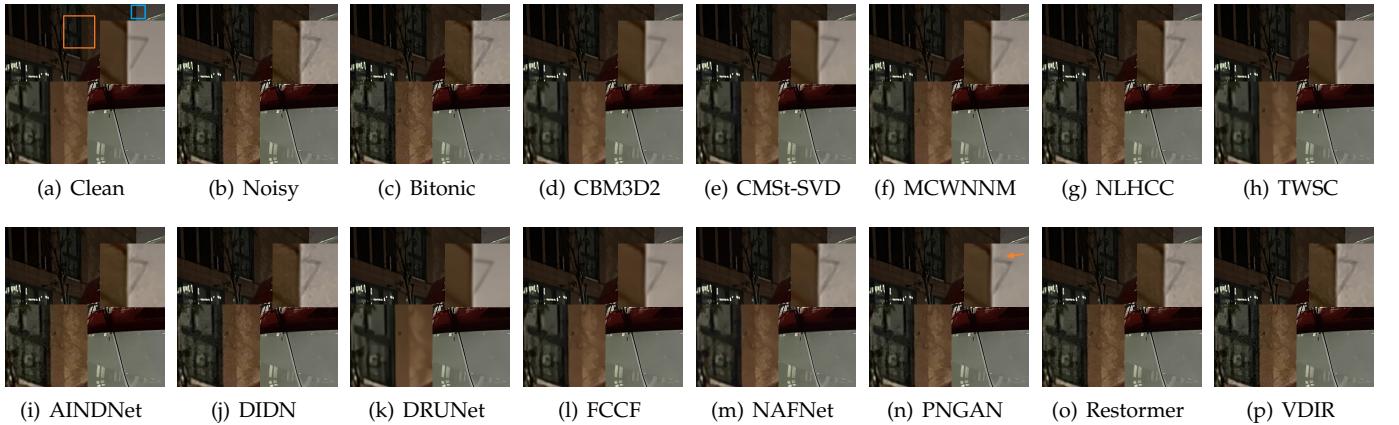


Fig. 9. Visual evaluation of compared methods on the real-world IOCI dataset. The camera device is IPHONE 13.



Fig. 10. Visual evaluation of compared methods on synthetic and real-world video data. First row: Set8 ($\sigma = 40$), second row: CRVD (ISO = 6400).

TABLE 5

Average PSNR and SSIM values of compared methods on four video datasets (sRGB space). The average computational time (s) is calculated based on the Set8 dataset with noise level $\sigma = 50$. '-' means the results are not available due to high computational cost or other issues.

| Dataset | σ | Traditional_denoisers | | | | | | DNN methods | | | | | | | |
|----------------|----------|-----------------------|---------------|-------------|---------------|---------------|--------------|-----------------|---------------|-----------------|-------------|-----------|------------|--------------|---------------|
| | | CVMSt-SVD [100] | RTA-LSM [220] | VBM4D1 [43] | VBM4D2 [43] | VIDOSAT [221] | DVDNet [222] | FastDVDNet [47] | FloRNN [48] | MAP-VDNet [223] | MMNet [224] | RFR [225] | UDVD [166] | ViDeNN [134] | VNLNet [226] |
| Set8 | 10 | 36.25 | 36.36 | 35.69 | 35.96 | 34.75 | 36.04 | 36.38 | 37.55 | - | 37.28 | 29.73 | 32.30 | 34.80 | 37.28 |
| | | 0.9453 | 0.9474 | 0.9342 | 0.9405 | 0.9244 | 0.9487 | 0.9513 | 0.9617 | - | 0.9595 | 0.7395 | 0.9081 | 0.9114 | 0.9584 |
| | 20 | 32.50 | 32.87 | 31.56 | 32.10 | 31.12 | 33.42 | 33.35 | 34.65 | - | 34.11 | 30.95 | 32.24 | 28.34 | 34.02 |
| | | 0.8891 | 0.8977 | 0.8485 | 0.8731 | 0.8571 | 0.9151 | 0.9159 | 0.9347 | - | 0.9270 | 0.8321 | 0.8937 | 0.7619 | 0.9243 |
| | 30 | 30.38 | 30.84 | 29.20 | 29.90 | - | 31.70 | 31.59 | 32.94 | - | 32.27 | 30.18 | 31.02 | - | - |
| | | 0.8369 | 0.8542 | 0.7626 | 0.8081 | - | 0.8825 | 0.8848 | 0.9101 | - | 0.8973 | 0.8299 | 0.8653 | - | - |
| | 40 | 28.92 | 29.41 | 27.56 | 28.37 | 27.67 | 30.45 | 30.36 | 31.71 | - | 30.98 | 28.89 | 29.46 | - | 30.72 |
| | | 0.7871 | 0.8170 | 0.6820 | 0.7482 | 0.7526 | 0.8528 | 0.8569 | 0.8869 | - | 0.8702 | 0.7891 | 0.8221 | - | 0.8596 |
| | 50 | 27.80 | 28.34 | 26.29 | 27.20 | - | 29.45 | 29.42 | 30.75 | 26.48 | 29.99 | 26.57 | 27.89 | - | - |
| | | 0.7392 | 0.7860 | 0.6086 | 0.6935 | - | 0.8264 | 0.8317 | 0.8653 | 0.7220 | 0.8452 | 0.6461 | 0.7654 | - | - |
| DAVIS | 10 | 38.49 | - | 37.57 | 37.97 | - | 37.70 | 38.33 | 38.92 | - | 38.25 | 31.28 | 33.65 | - | 38.68 |
| | | 0.9611 | - | 0.9475 | 0.9555 | - | 0.9600 | 0.9631 | 0.9680 | - | 0.9639 | 0.7685 | 0.9204 | - | 0.9653 |
| | 20 | 34.73 | - | 33.33 | 34.03 | - | 35.34 | 35.52 | 36.43 | - | 35.63 | 33.09 | 33.95 | - | 35.84 |
| | | 0.9147 | - | 0.8709 | 0.9000 | - | 0.9361 | 0.9364 | 0.9493 | - | 0.9384 | 0.8765 | 0.9154 | - | 0.9398 |
| | 30 | 32.53 | - | 30.82 | 31.71 | - | 33.84 | 33.88 | 35.06 | - | 34.06 | 31.93 | 32.85 | - | - |
| | | 0.8665 | - | 0.7866 | 0.8415 | - | 0.9140 | 0.9134 | 0.9341 | - | 0.9166 | 0.8564 | 0.8918 | - | - |
| | 40 | 30.96 | - | 29.02 | 30.06 | - | 32.70 | 32.71 | 34.03 | - | 32.93 | 30.22 | 31.29 | - | 33.01 |
| | | 0.8182 | - | 0.7036 | 0.7838 | - | 0.8931 | 0.8926 | 0.9195 | - | 0.8969 | 0.7997 | 0.8545 | - | 0.8959 |
| | 50 | 29.72 | - | 27.61 | 28.78 | - | 31.76 | 31.79 | 33.18 | 28.35 | 32.04 | 27.22 | 29.65 | - | - |
| | | 0.7701 | - | 0.6270 | 0.7287 | - | 0.8730 | 0.8735 | 0.9054 | 0.7911 | 0.8788 | 0.6350 | 0.8033 | - | - |
| CRVD | - | 36.66 | - | 34.14 | 34.66 | 34.16 | 34.50 | 35.84 | 36.66 | - | - | 31.30 | - | 32.31 | 36.11 |
| IOCV | - | 0.9463 | - | 0.9079 | 0.9224 | 0.9384 | 0.9493 | 0.9306 | 0.9605 | - | - | 0.7785 | - | 0.8449 | 0.9449 |
| Implementation | - | 38.22 | - | 38.65 | 38.76 | - | 38.53 | 37.57 | 38.64 | 35.52 | - | 31.46 | 35.02 | 36.13 | 38.76 |
| Implementation | - | 0.9736 | - | 0.9763 | 0.9765 | - | 0.9754 | 0.9699 | 0.9743 | 0.9313 | - | 0.8346 | 0.9660 | 0.9506 | 0.9765 |
| Time (s) | - | 888.0 | 30803.5 | 920.9 | 2076.0 | 4313.7 | 818.8 | 6.6 | 53.9 | 77.2 | 30.1 | 37.8 | 163.4 | 19.0 | 217.1 |

human opinions [227]. Specifically, we randomly select 10 videos from our IOCV dataset and invite 10 volunteers to rate the mean, noisy and denoised sequences of four compared methods. The invited volunteers have very little background knowledge of denoising, and they are not aware of how the presented video sequences are processed. For each of the 10 videos, the volunteers are asked to choose at least 2 best sequences, which then earn 1 point for the corresponding methods. The detailed human rating results are reported in Table 6. First, we observe that the mean video obtains the highest score on 9 of 10 videos, which suggests that our video averaging strategy may provide an alternative to the approach of generating reference videos frame-by-frame [198]. Second, the human rating results show that DNN methods produce videos of higher quality than the benchmark traditional denoisers on our IOCV dataset, since both CVMSt-SVD and VBM4D leave behind medium-to-low-frequency noise, resulting in noticeable flickering. Last but not least, for the comparison of two traditional denoisers, VBM4D looks more visually pleasant than CVMSt-SVD because CVMSt-SVD presents more temporally decorrelated low-frequency noise in flat areas, which will appear as particularly bothersome for the viewers.

4.3 Results for MSI/HSI Datasets

4.3.1 Experimental Settings

MSI/HSI play an important role in a variety of remote sensing applications [187], [228]. In this subsection, we evaluate the performance of various MSI/HSI denoising methods on synthetic and real noisy data. For synthetic experiments, due to the high computational cost, we mainly use the CAVE dataset for comparison. We assume that entries in all slices of noisy data are corrupted by zero-mean i.i.d Gaussian

TABLE 6
Human rating results of sequences generated by different methods based on our IOCV dataset. The top two results are bolded.

| # Image | Ratings | | | | | |
|---------|-------------|-------|-----------|----------|------------|-------------|
| | Mean | Noisy | CVMSt-SVD | VBM4D1 | FastDVDNet | VNLNet |
| 1 | 9 | 0 | 1 | 2 | 5 | 3 |
| 2 | 8 | 1 | 0 | 4 | 7 | 6 |
| 3 | 10 | 0 | 0 | 8 | 3 | 5 |
| 4 | 8 | 0 | 0 | 4 | 7 | 8 |
| 5 | 10 | 0 | 0 | 4 | 4 | 7 |
| 6 | 10 | 0 | 0 | 5 | 6 | 7 |
| 7 | 10 | 0 | 0 | 5 | 8 | 5 |
| 8 | 10 | 2 | 2 | 8 | 5 | 6 |
| 9 | 9 | 0 | 2 | 4 | 2 | 7 |
| 10 | 4 | 4 | 4 | 4 | 5 | 8 |
| Average | 8.80 | 0.70 | 0.90 | 4.80 | 5.20 | 6.20 |

noise. In addition to the classical spatial-based quality indices PSNR and SSIM, we adopt two widely used spectral-based quality indicators for MSI/HSI data, namely spectral angle mapper (SAM) [229] and relative dimensionless global error in synthesis (ERGAS) [230]. Different from PSNR and SSIM, recovered data with lower SAM and ERGAS are considered of better quality.

4.3.2 Objective Results

Since the ground-truth data of HHD dataset is not available, we report the objective results of compared methods on CAVE and Real-HSI datasets in Table 7.

For traditional denoisers, tensor-based methods such as NGMeet, LLRT, LTDL and OLRT demonstrate outstanding performance by exploiting both spatial and spectral correlation, which consistently outperform other compared

TABLE 7

Comparison of quantitative results on CAVE (with i.i.d Gaussian noise $\sigma = \{10, 30, 50, 100\}$) and Real-HSI datasets. The average denoising time (minutes) is calculated based on the CAVE dataset when $\sigma = 100$.

| Datasets | σ | Metrics | Traditional denoisers | | | | | | | | | | | | | | | | DNN methods | | | | | | | | | | | |
|----------|----------|---------|-----------------------|-----------|------------|------------|---------------|-------------|--------------|---------------|-----------|-----------|------------|--------------|--------------|--------------|---------------|---------------|--------------|---------------|-----------|------------|------------|-----------|--------------|--------------|-----------|------------|-----|------------|
| | | | 4DHOSVD [21] | ANLM [71] | BM4D1 [20] | BM4D2 [20] | FastHyDe [85] | ISTReg [97] | LLRGTV [101] | LLRT [98] | LRMR [78] | LRTA [90] | LTDL [105] | LRTDGS [231] | MSt-SVD [99] | MTSNMF [232] | NGMeet [104] | OLRT [106] | PARAFAC [91] | RTA-LSM [220] | SDS [233] | SSTV [234] | TDL [7] | GRN [128] | QRNN3D [140] | sDeCNN [236] | SST [237] | | | |
| CAVE | 10 | PSNR | 45.43 | 41.52 | 43.04 | 44.61 | 42.87 | 45.14 | 39.62 | 46.60 | 39.47 | 41.36 | 45.90 | 35.99 | 45.20 | 43.04 | 46.59 | 47.07 | 35.43 | 46.43 | 39.69 | 38.91 | 44.39 | 30.98 | 37.01 | 40.36 | 41.40 | 39.75 | | |
| | | SSIM | 0.9812 | 0.9576 | 0.9700 | 0.9784 | 0.9763 | 0.9792 | 0.9274 | 0.9868 | 0.9125 | 0.9499 | 0.9855 | 0.8976 | 0.9814 | 0.9706 | 0.9880 | 0.9877 | 0.8762 | 0.9857 | 0.9483 | 0.9153 | 0.9797 | 0.8622 | 0.8936 | 0.9672 | 0.9624 | 0.9730 | | |
| | | SAM | 0.1084 | 0.2183 | 0.1844 | 0.1289 | 0.1398 | 0.1311 | 0.2676 | 0.0842 | 0.3282 | 0.1718 | 0.0923 | 0.1764 | 0.1064 | 0.1342 | 0.0853 | 0.0840 | 0.2407 | 0.0926 | 0.2159 | 0.2669 | 0.1055 | 0.4971 | 0.3236 | 0.1926 | 0.1524 | 0.1632 | | |
| | | EGRAS | 30.83 | 47.79 | 40.07 | 33.32 | 42.90 | 31.71 | 63.65 | 26.75 | 63.52 | 49.53 | 29.06 | 103.69 | 32.05 | 40.29 | 27.26 | 25.53 | 108.34 | 27.15 | 61.88 | 65.86 | 34.39 | 209.61 | 85.92 | 65.31 | 52.04 | 66.63 | | |
| CAVE | 30 | PSNR | 39.78 | 34.70 | 37.05 | 38.80 | 37.22 | 40.37 | 32.26 | 41.49 | 31.43 | 36.06 | 41.21 | 35.12 | 40.23 | 37.02 | 41.60 | 41.79 | 33.46 | 41.32 | 32.07 | 31.35 | 39.06 | 30.94 | 30.83 | 38.48 | 37.58 | 37.82 | | |
| | | SSIM | 0.9336 | 0.8068 | 0.9236 | 0.9283 | 0.9151 | 0.9462 | 0.7150 | 0.9681 | 0.6517 | 0.8775 | 0.9656 | 0.8795 | 0.9530 | 0.8758 | 0.9693 | 0.9680 | 0.8255 | 0.9598 | 0.6723 | 0.6078 | 0.9493 | 0.8605 | 0.6467 | 0.9496 | 0.9247 | 0.9564 | | |
| | | SAM | 0.2272 | 0.4302 | 0.2898 | 0.2579 | 0.3123 | 0.2254 | 0.5538 | 0.1221 | 0.5899 | 0.2446 | 0.1299 | 0.2104 | 0.1737 | 0.2267 | 0.1238 | 0.1180 | 0.3362 | 0.1556 | 0.4959 | 0.5190 | 0.1496 | 0.4970 | 0.5105 | 0.2320 | 0.1959 | 0.2143 | | |
| | | EGRAS | 59.12 | 103.96 | 83.28 | 65.23 | 81.44 | 54.82 | 141.67 | 48.50 | 154.15 | 90.50 | 49.68 | 109.98 | 56.26 | 79.30 | 48.15 | 46.65 | 126.99 | 49.42 | 143.67 | 155.14 | 63.24 | 209.90 | 177.76 | 76.21 | 78.28 | 80.26 | | |
| CAVE | 50 | PSNR | 36.82 | 30.70 | 34.51 | 35.90 | 34.56 | 37.88 | 28.53 | 38.65 | 27.44 | 33.52 | 38.62 | 33.84 | 37.73 | 33.50 | 38.87 | 39.13 | 31.25 | 38.60 | 26.09 | 25.48 | 36.44 | 30.82 | - | 37.02 | 35.49 | 36.43 | | |
| | | SSIM | 0.8722 | 0.6064 | 0.8823 | 0.8685 | 0.8471 | 0.9168 | 0.5368 | 0.9482 | 0.4410 | 0.8201 | 0.9464 | 0.8470 | 0.9285 | 0.7585 | 0.9507 | 0.9481 | 0.7307 | 0.9301 | 0.3529 | 0.3019 | 0.9171 | 0.8564 | - | 0.9314 | 0.8947 | 0.9408 | | |
| | | SAM | 0.3385 | 0.5801 | 0.3540 | 0.3557 | 0.4395 | 0.2737 | 0.7113 | 0.1551 | 0.7273 | 0.2897 | 0.1630 | 0.2674 | 0.2231 | 0.2956 | 0.1617 | 0.1436 | 0.4541 | 0.2199 | 0.7170 | 0.7473 | 0.2007 | 0.4966 | - | 0.2741 | 0.2141 | 0.2524 | | |
| | | EGRAS | 83.36 | 164.17 | 110.14 | 91.19 | 108.74 | 73.32 | 211.99 | 67.56 | 241.87 | 121.15 | 66.73 | 122.70 | 75.03 | 118.36 | 65.72 | 63.41 | 158.53 | 67.27 | 290.93 | 306.21 | 85.24 | 210.86 | - | 87.15 | 98.54 | 92.86 | | |
| Real-HSI | 100 | PSNR | 32.66 | 24.92 | 31.21 | 31.84 | 30.62 | 31.85 | 23.36 | 35.39 | 21.91 | 30.06 | 34.86 | 30.34 | 34.20 | 28.04 | 34.98 | 35.18 | 26.77 | 34.66 | 18.48 | 16.75 | 32.89 | 29.45 | - | 30.33 | 32.51 | 33.96 | | |
| | | SSIM | 0.7307 | 0.2828 | 0.7854 | 0.7197 | 0.6717 | 0.8176 | 0.2859 | 0.9154 | 0.1914 | 0.7192 | 0.7138 | 0.9002 | 0.6852 | 0.4988 | 0.9073 | 0.8944 | 0.4481 | 0.8418 | 0.1067 | 0.0720 | 0.8283 | 0.7814 | - | 0.7128 | 0.8309 | 0.9016 | | |
| | | SAM | 0.5600 | 0.7968 | 0.4638 | 0.5160 | 0.6097 | 0.3605 | 0.9229 | 0.1962 | 0.9175 | 0.3649 | 0.2349 | 0.4707 | 0.3142 | 0.4180 | 0.2504 | 0.2100 | 0.6942 | 0.3714 | 0.9868 | 1.0569 | 0.3138 | 0.4991 | - | 0.5161 | 0.2748 | 0.3225 | | |
| | | EGRAS | 134.34 | 324.49 | 158.47 | 144.91 | 168.92 | 148.29 | 376.10 | 99.37 | 456.12 | 180.03 | 102.61 | 174.87 | 115.24 | 221.84 | 101.86 | 99.93 | 260.85 | 105.99 | 718.40 | 834.58 | 128.22 | 225.29 | - | 172.68 | 136.17 | 120.40 | | |
| Real-HSI | - | PSNR | 25.86 | - | 25.84 | 25.88 | 23.99 | 25.91 | 24.63 | 25.90 | 24.81 | 24.27 | 25.80 | 25.12 | 25.86 | 25.51 | 25.87 | 25.94 | 25.08 | - | 25.58 | 25.31 | 25.33 | 24.91 | 25.63 | 25.82 | 25.70 | - | | |
| | | SSIM | 0.8664 | - | 0.8574 | 0.8653 | 0.5484 | 0.8688 | 0.7344 | 0.8610 | 0.7753 | 0.5876 | 0.8413 | 0.8181 | 0.8665 | 0.7819 | 0.8659 | 0.8695 | 0.7668 | - | 0.8126 | 0.7923 | 0.7634 | 0.8006 | 0.8534 | 0.8691 | 0.8597 | - | | |
| | | SAM | 0.0632 | - | 0.0698 | 0.0660 | 0.2275 | 0.0551 | 0.1057 | 0.0601 | 0.1542 | 0.2198 | 0.0742 | 0.0669 | 0.0635 | 0.1020 | 0.0509 | 0.0538 | 0.1084 | - | 0.0891 | 0.0816 | 0.1232 | 0.1635 | 0.0921 | 0.0638 | 0.0928 | - | | |
| | | EGRAS | 222.73 | - | 224.51 | 222.67 | 252.24 | 222.18 | 243.68 | 224.05 | 246.39 | 248.57 | 223.32 | 234.02 | 222.64 | 226.58 | 222.69 | 221.94 | 233.43 | - | 226.23 | 229.18 | 229.32 | 248.41 | 232.73 | 225.32 | 227.88 | - | | |
| Time (m) | | | - | - | 2.5 | 1.7 | 1.5 | 4.1 | 0.1 | 41.9 | 2.9 | 16.5 | 4.9 | 0.1 | 35.0 | 0.6 | 2.8 | 0.5 | 4.1 | 24.3 | 1.5 | 12.5 | 0.1 | 3.0 | 0.9 | 0.1 | 0.8 | 0.1 | 0.7 | 0.1 |

approaches at all noise levels by more than 1 dB on the CAVE dataset. However, their iterative denoising strategy with a large number of local similar patches significantly increases computational burden. By comparison, BM4D and MSt-SVD are more efficient, in that BM4D does not need to learn local patch transforms, and MSt-SVD is a one-step approach that utilizes global patch representation. In addition, they also produce very competitive performance on the Real-HSI dataset.

For DNN methods, we can observe that their results on the CAVE dataset are not satisfactory, since the networks are trained with predefined and a limited range of noise levels, distributions and bandwidths on other datasets such as ICVL. Nevertheless, it is noticed that QRNN3D and sDeCNN are able to gain state-of-the-art performance on the Real-HSI dataset, which indicates the success of CNN-based frameworks in capturing spatial and spectral correlations in real-world cases. Besides, they are potentially effective and efficient tools for handling large data in practice due to much faster denoising speed.

4.3.3 Visual Evaluation

First, we notice that all compared methods can effectively remove to certain extents, but their performance are far from satisfactory. Specifically, BM4D and MStSVD tend to produce unexpected artifacts at high noise levels since the pre-defined transforms may not fully exploit the correlation among all the spectral bands. Besides, although NGMeet, LLRT and LTDL produce outstanding quantitative results, they fail to preserve high-frequency components such as edges and textures, as can be seen from Fig. 11 and Fig. 12. This observation indicates that increasing the number of iterations and local similar patches may not help preserve fine details and structure of MSI/HSI data. In addition, the CNN-based method QRNN3D shows impressive noise removal ability at the cost of over-smooth and blurry effects on the edges and details. To conclude, all state-of-the-art

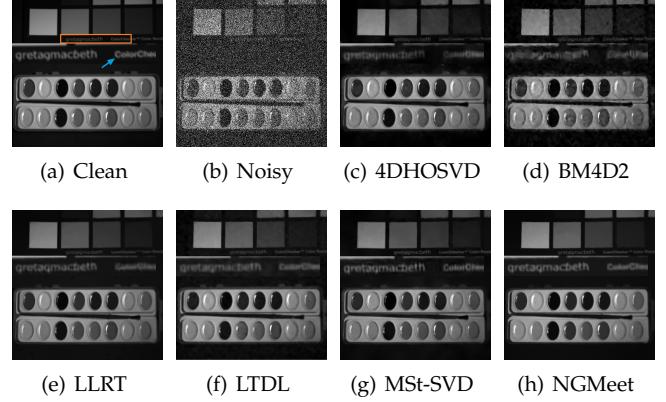


Fig. 11. Visual evaluation of the CAVE dataset with noise level $\sigma = 100$.

methods struggle to adapt to different local image contents to balance smoothness and sharp details. For MSI/HSI data, each spectral band signal has unique spectroscopic characteristics [187], applying a same set of parameters or transforms to multiple spectral bands may result in similar denoising patterns across the whole data. Therefore, apart from the high computational burden, another challenge of filtering MSI/HSI data lies in dealing with the spatial and spectral variation of noise levels and distributions.

4.4 Results for MRI Datasets

4.4.1 Experimental Settings

MRI is a non-invasive imaging technology that produces three dimensional detailed anatomical images, which has been shown to be helpful in disease detection, diagnosis, and treatment monitoring. Different from the AWGN noise modeling, MRI data are often corrupted by Rician noise [56]. Specifically, Let \mathcal{X} be the original noise-free signal, the noisy

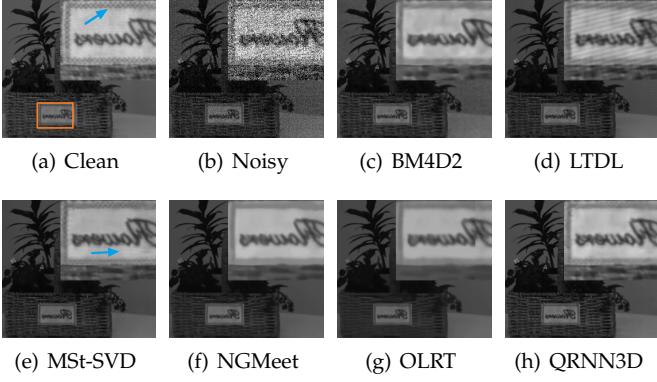


Fig. 12. Visual evaluation of the real-world Real-HSI dataset.

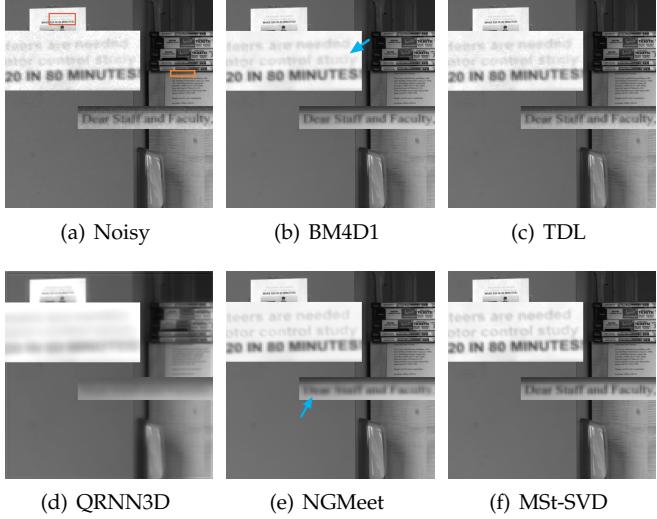


Fig. 13. Visual evaluation of the real-world HHD dataset.

Rician MRI data \mathcal{Y} is defined by

$$\mathcal{Y} = \sqrt{(\mathcal{X} + \sigma\eta_r)^2 + (\sigma\eta_i)^2} \quad (9)$$

where $\eta_r, \eta_i \sim N(0, 1)$ are i.i.d. random vectors following the standard normal distribution, σ is the standard deviation in both real and imaginary channels of the noisy 3D MR images. To handle Rician noise, the technique of forward and inverse variance stabilizing transform (VST) [238] is often adopted by Gaussian denoisers via

$$\hat{\mathcal{X}} = VST^{-1}(denoise(VST(\mathcal{Y}, \sigma), \sigma_{VST}), \sigma) \quad (10)$$

where VST^{-1} denotes the inverse of VST, σ_{VST} is the stabilized standard deviation after VST, and σ is the standard deviation of the noise in Eq. (9). According to Eq. (10), the noisy data \mathcal{Y} is first stabilized by the VST and then filtered by certain Gaussian denoisers using a constant noise level σ_{VST} , and the final estimate is obtained by applying the inverse VST to the output of the denoising result [20].

In our experiments, we use PSNR and SSIM as the objective metrics, and similar to [20], [77], the PSNR value is computed on the foreground defined by $\mathcal{X}_f = \{x \in \mathcal{X} : x > 10 \cdot D/255\}$, where D is the peak of clean data \mathcal{X} .

4.4.2 Synthetic Experiments

The volume data of Brainweb and fastMRI are added with varying levels of stationary Rician noise from 1% to 19% of the maximum intensity with an increase of 2%. In real-world applications, the noise level is usually lower than 19%, but we are also interested in the denoising capability of compared methods under extreme conditions. Table 8 lists detailed quantitative results, and Fig. 14 compares the denoising performance at high noise levels when $\sigma \geq 11\%$. We notice that at lower noise levels, PRI-NLPCA is able to take advantage of the high-quality initial estimate of NLPCA, and thus shows outstanding performance when $\sigma \leq 9\%$. As noise level increases, the tensor-based method ILR-HOSVD demonstrates advantages of extracting latent features, which produces state-of-the-art results on the fastMRI dataset when $\sigma \geq 9\%$. However, from Fig. 14(a) we observe that the iterative learning strategy of ILR-HOSVD is also subject to the presence of severe noise and may not generalize well to other datasets, while BM4D benefits from its pre-defined transforms and outperforms other methods on the Brainweb dataset when $\sigma \geq 17\%$.

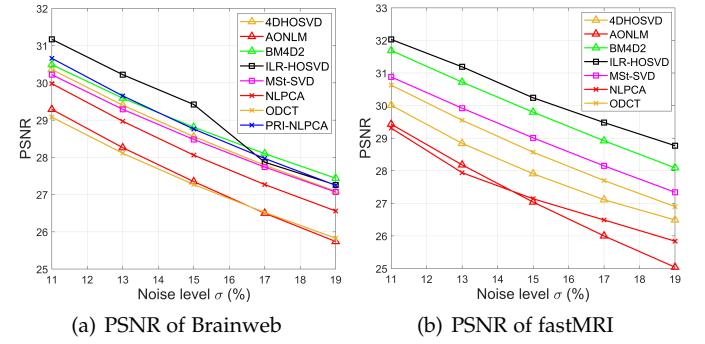


Fig. 14. Average PSNR values of compared methods on the Brainweb and fastMRI data at high noise levels ($\sigma \geq 11\%$).

Visual evaluation of compare methods on T1w data ($\sigma = 19\%$) is illustrated in Fig. 15. BM4D is successful in both noise suppression and detail preservation. Compared with ILR-HOSVD, another tensor-based filter MSt-SVD removes noise to a greater extent but blurs the edges and textures. The state-of-the-art NLM methods, namely PRINLM and PRI-NLPCA exhibit pleasant visual effects in homogeneous areas at the cost of slight over-smoothness along the edges. Furthermore, PRI-NLPCA does not remove noise in the background sufficiently since the local PCA transforms are severely degraded.

4.4.3 Real-world MRI Dataset

To evaluate the performance of compared methods on real-world 3D MRI data, we carry out experiments on T1w MR images of the OASIS dataset [203]. The Rician noise levels of two selected T1w data, namely OAS1_0112 and OAS1_0092 are estimated to be 3% and 4.5% of the maximum intensity, respectively [92]. The filtered images of different methods are compared in Fig. 16 and Fig. 17, respectively.

According to synthetic experiments in Table 8, we notice that in most cases all methods can achieve promising denoising performance at relatively low noise levels, as can also be seen from the visual effects in Fig. 16. Therefore,

TABLE 8

Average PSNR-SSIM and computational time (s) of different methods on Brainweb and fastMRI data corrupted by Rician noise. The standard deviations σ of the noise is expressed as percentage relative to the maximum intensity value of the noise-free data [20].

| Datasets | Noise level | 4DHOSVD [21] | ANLM [71] | BM4D1 [20] | BM4D2 [20] | ILR-HOSVD [103] | OLMMSE [70] | ROLMMSE [70] | MStSVD [100] | NLPCA [80] | ODCT [77] | PRINLM [77] | PRI-NLPCA [80] |
|----------|-------------|--------------|-------------|-------------|---------------------|--------------------|-------------|--------------|--------------|-------------|-------------|-------------|---------------------|
| Brainweb | 1 | 43.86/0.993 | 41.87/0.991 | 43.18/0.992 | 43.43/0.992 | 44.55/0.994 | 42.37/0.991 | 42.39/0.991 | 43.99/0.993 | 44.13/0.994 | 43.25/0.992 | 43.33/0.993 | 44.28/0.994 |
| | 3 | 37.49/0.976 | 36.52/0.971 | 36.59/0.971 | 37.21/0.975 | 38.22/0.981 | 35.99/0.968 | 35.98/0.969 | 37.46/0.975 | 37.74/0.977 | 36.43/0.970 | 36.87/0.975 | 38.12/0.981 |
| | 5 | 34.64/0.959 | 33.78/0.947 | 33.71/0.950 | 34.56/0.959 | 35.46/0.966 | 33.37/0.949 | 33.36/0.949 | 34.65/0.958 | 33.84/0.940 | 33.57/0.949 | 33.97/0.955 | 35.29/ 0.967 |
| | 7 | 32.85/0.941 | 31.91/0.921 | 31.80/0.926 | 32.84/0.943 | 33.62/0.951 | 31.58/0.925 | 31.61/0.928 | 32.83/0.940 | 31.87/0.912 | 31.66/0.927 | 32.01/0.934 | 33.36/0.951 |
| | 9 | 31.49/0.922 | 30.47/0.894 | 30.35/0.901 | 31.54/0.928 | 32.31/0.938 | 30.16/0.898 | 30.23/0.904 | 31.31/0.924 | 31.15/0.908 | 30.23/0.905 | 30.49/0.911 | 31.86/0.934 |
| | 11 | 30.36/0.903 | 29.29/0.865 | 29.15/0.873 | 30.49/0.912 | 31.17/0.922 | 28.93/0.868 | 29.09/0.880 | 30.22/0.906 | 29.98/0.883 | 29.08/0.882 | 29.28/0.888 | 30.66/0.916 |
| | 13 | 29.40/0.883 | 28.26/0.836 | 28.14/0.844 | 29.59/0.895 | 30.22/0.909 | 27.82/0.834 | 28.08/0.856 | 29.29/0.888 | 28.97/0.855 | 28.11/0.861 | 28.24/0.864 | 29.65/0.898 |
| | 15 | 28.56/0.862 | 27.35/0.807 | 27.26/0.814 | 28.81/0.879 | 29.42/0.894 | 26.82/0.799 | 27.26/0.835 | 28.48/0.870 | 28.06/0.826 | 27.27/0.841 | 27.36/0.841 | 28.76/0.879 |
| | 17 | 27.79/0.842 | 26.50/0.775 | 26.46/0.785 | 28.10/0.862 | 27.87/0.845 | 25.91/0.762 | 26.54/0.815 | 27.74/0.853 | 27.27/0.798 | 26.52/0.819 | 26.59/0.817 | 27.97/0.860 |
| | 19 | 27.09/0.822 | 25.74/0.744 | 25.73/0.755 | 27.44/0.845 | 27.26/0.828 | 25.05/0.725 | 25.86/0.797 | 27.07/0.835 | 26.56/0.772 | 25.83/0.799 | 25.92/0.797 | 27.25/0.841 |
| | 1 | 41.50/0.975 | 39.36/0.956 | 42.07/0.977 | 42.53/0.979 | 42.30/0.977 | - | - | 42.09/0.977 | 41.69/0.976 | 40.96/0.968 | - | - |
| | 3 | 36.58/0.931 | 36.12/0.921 | 36.68/0.933 | 37.07/ 0.937 | 37.09/0.933 | - | - | 36.74/0.935 | 37.01/0.932 | 36.68/0.930 | - | - |
| | 5 | 34.13/0.901 | 34.01/0.884 | 34.57/0.901 | 35.12/0.911 | 35.34/0.911 | - | - | 34.61/0.906 | 34.15/0.902 | 34.62/0.899 | - | - |
| | 7 | 32.58/0.871 | 32.29/0.842 | 33.01/0.867 | 33.82/0.889 | 34.01/0.888 | - | - | 33.10/0.878 | 32.08/0.866 | 33.11/0.869 | - | - |
| | 9 | 31.27/0.845 | 30.79/0.793 | 31.61/0.829 | 32.71/0.866 | 33.04/0.868 | - | - | 31.92/0.854 | 30.73/0.838 | 31.81/0.836 | - | - |
| | 11 | 30.01/0.814 | 29.43/0.741 | 30.29/0.785 | 31.69/0.841 | 32.03/0.844 | - | - | 30.88/0.827 | 29.31/0.804 | 30.63/0.801 | - | - |
| | 13 | 28.84/0.785 | 28.18/0.683 | 29.05/0.737 | 30.72/0.814 | 31.19/0.822 | - | - | 29.92/0.799 | 27.94/0.768 | 29.55/0.764 | - | - |
| | 15 | 27.91/0.754 | 27.04/0.626 | 27.87/0.686 | 29.80/0.785 | 30.24/0.794 | - | - | 29.01/0.770 | 27.12/0.711 | 28.57/0.727 | - | - |
| | 17 | 27.11/0.708 | 26.01/0.571 | 26.77/0.634 | 28.92/0.755 | 29.48/0.769 | - | - | 28.15/0.741 | 26.49/0.668 | 27.70/0.693 | - | - |
| | 19 | 26.49/0.671 | 25.04/0.519 | 25.76/0.583 | 28.09/0.725 | 28.77/0.744 | - | - | 27.34/0.711 | 25.84/0.606 | 26.90/0.660 | - | - |
| Time (s) | / | 614.1 | 426.7 | 57.1 | 171.2 | 1552.9 | > 1h | > 1h | 141.4 | 517.8 | 4.5 | 11.0 | 986.9 |

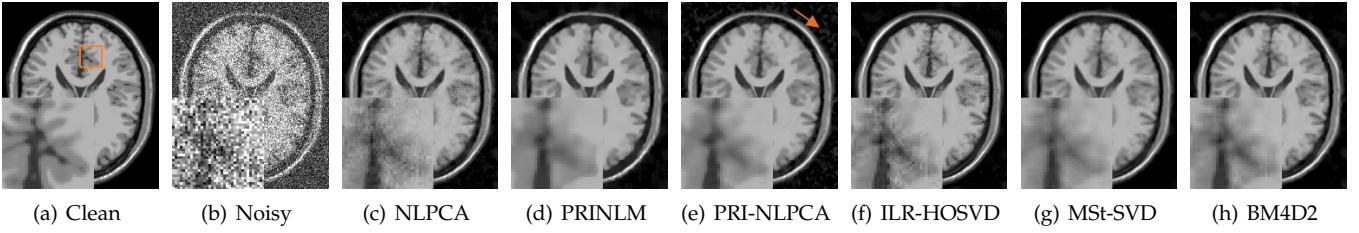


Fig. 15. Visual evaluation of compared methods on the Brainweb T1w data with noise level $\sigma = 19\%$.

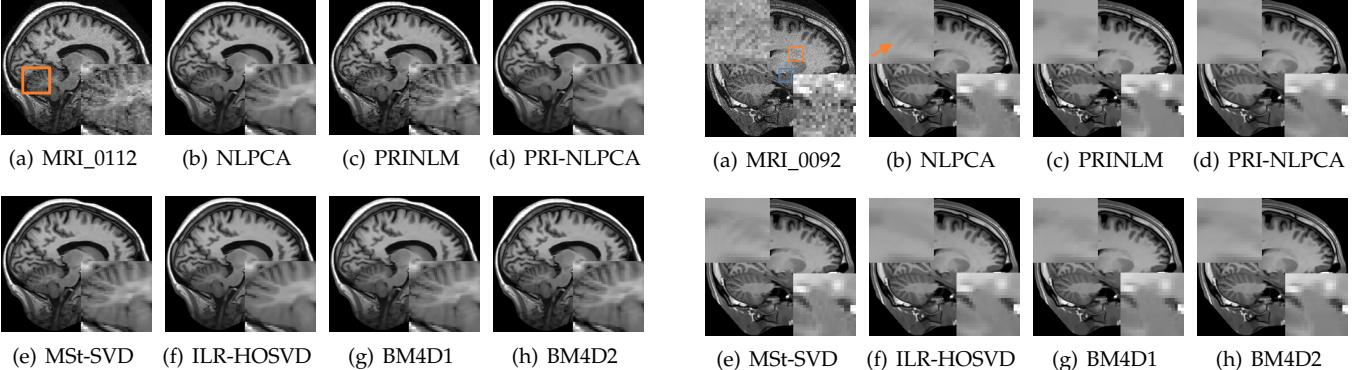


Fig. 16. Visual evaluation of compared methods on the real OAS1_0092 T1w data with estimated noise level $\sigma = 3\%$.

from the perspective of real-world denoising, BM4D1 and PRINLM are more competitive with low computational cost. Despite the similar denoising results of compared methods, we observe that for certain slices of OAS1_0092, NLPCA restores more details and textures, while other methods suffer from over-smoothness to varying degrees, as illustrated in Fig. 17. This interesting observation is another vivid example to show that the use of tensor representation

and transforms do not always help preserve more structural information of the underlying clean images.

4.5 Discussion

4.5.1 Understanding the Denoising Performance

The theoretical bound of compared image denoising methods is hard to obtain, but it is interesting to investigate the denoising capability of compared methods under the

Fig. 17. Visual evaluation of compared methods on the real OAS1_0092 T1w data with estimated noise level $\sigma = 4.5\%$.

challenging practical cases when prior information, such as training/validation datasets and camera settings are unavailable. We use our IOCI's FUJIFILM X100T camera dataset, and for each scene, we generate three new mean images by averaging 3, 5 and 10 noisy images, and they are named Mean_3, Mean_5 and Mean_10, respectively. They can be regarded as images captured via different continuous shooting modes with *high*, *medium* and *low* noise levels. Fig. 18 illustrates the average PSNR differences of six implementations compared with Mean_3 on FUJI dataset. Interestingly, Fig. 18 shows that state-of-the-art denoising methods only produce marginal improvements compared with Mean_3, which indicates that their denoising performance is similar to obtaining the mean image by averaging 3 consecutive noisy observations.

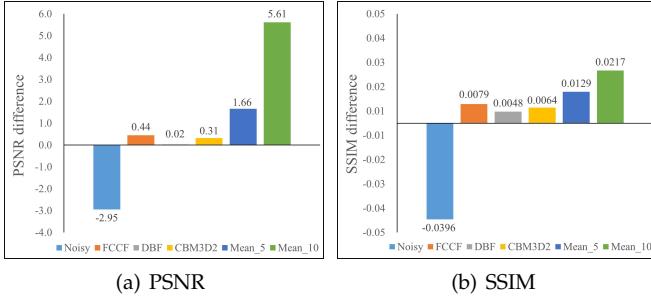


Fig. 18. PSNR and SSIM difference of six implementations compared with 'Mean_3' on the FUJI dataset.

4.5.2 The Cost of Pre-Training

The strength of many DNN methods lies in their amazing capability of exploiting extra information to capture noise patterns and customize camera-aware noise models. Despite all the good sides of DNN denoising frameworks, the training process can be time-consuming. To investigate the cost of pre-training, we use DANet trained with the PolyU dataset as an example. Specifically, for each camera of the PolyU dataset, we randomly select 80% of clean/noisy pairs for training, and the rest 20% for testing. The training dataset is augmented via random flipping and rotation. We use the code provided by the authors to train the DANet model for up to 60 epochs, and each epoch costs about 15 minutes run on the GPU device. From Fig. 19, it is observed that DANet slightly outperforms CBM3D2 on the test set after 30 training epochs.

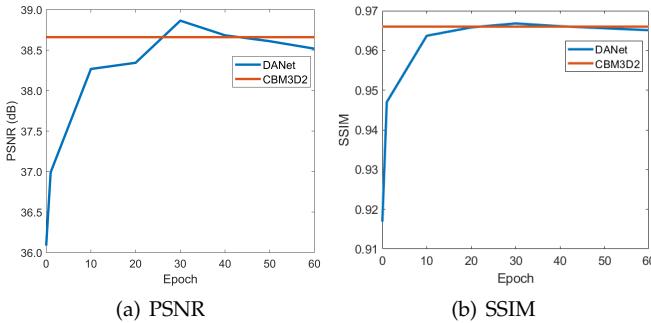


Fig. 19. PSNR and SSIM values of DANet on the PolyU test set w.r.t different training epochs.

In real-world applications, it may be impractical to train

a model for each camera setting due to insufficient training data and high computational burdens. Recently, some few-shot domain transfer schemes [131] are proposed to enable model reuse and also reduce the load of re-training, thus it is interesting to further investigate their effectiveness on more camera settings.

5 CONCLUSION

Image denoising is an appealing research topic with potentially important applications to different fields. Over the past several decades, the rapid development of imaging systems and techniques has fostered the emergence of numerous denoising approaches, varying from traditional Gaussian denoisers to advanced DNN methods. Driven by the curiosity about the applicability, effectiveness and generalization ability of many pioneering and representative methods, we carry out experiments in both synthetic and real-world scenarios. A new dataset is also introduced to enrich benchmarking. Throughout objective and subjective evaluations, we observe the astonishing success brought by DNN methods, and also recognize the competitive performance of traditional denoisers.

Nowadays, image denoising also serves as a perfect test-bed for assessing new ideas and techniques [2]. Many recent studies focus on the extention of image denoising methods to other computer vision tasks such as dehazing [239], demoisacking [240] and classification [241]. It is therefore interesting to further explore the potential of related works to satisfy needs beyond denoising.

ACKNOWLEDGMENT

The authors are grateful for researchers who make their code, software packages and datasets publicly available. We also appreciate the generosity of our friends and volunteers for sharing their camera devices and participating in the image quality assessment task.

REFERENCES

- [1] N. Wiener, *Extrapolation, interpolation, and smoothing of stationary time series: with engineering applications*. MIT press, 1950.
- [2] M. Elad, B. Kawar, and G. Vaksman, "Image denoising: The deep learning revolution and beyond—a survey paper—," *arXiv preprint arXiv:2301.03362*, 2023.
- [3] T. Tirer and R. Giryes, "Back-projection based fidelity term for ill-posed linear inverse problems," *IEEE Trans. Image Process.*, vol. 29, pp. 6164–6179, 2020.
- [4] M. Isard and A. Blake, "Condensation conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, 1998.
- [5] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognit.*, vol. 26, no. 9, pp. 1277–1294, 1993.
- [6] M. K. Raczkowska, P. Kozioł, S. Urbaniak-Wasik, C. Palusziewicz, W. M. Kwiatek, and T. P. Wrobel, "Influence of denoising on classification results in the context of hyperspectral data: High definition ft-ir imaging," *Anal. Chim. Acta*, vol. 1085, pp. 39–47, 2019.
- [7] Y. Peng, D. Meng, Z. Xu, C. Gao, Y. Yang, and B. Zhang, "De-composable nonlocal tensor dictionary learning for multispectral image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2949–2956.
- [8] B. Rasti, P. Scheunders, P. Ghamisi, G. Licciardi, and J. Chanussot, "Noise reduction in hyperspectral imagery: Overview and application," *Remote Sens.*, vol. 10, no. 3, p. 482, 2018.

- [9] A. Signoroni, M. Savardi, A. Baronio, and S. Benini, "Deep learning meets hyperspectral image analysis: A multidisciplinary review," *J. Imaging*, vol. 5, no. 5, p. 52, 2019.
- [10] J. Mohan, V. Krishnaveni, and Y. Guo, "A survey on the magnetic resonance image denoising methods," *Biomed. Signal Process. Control.*, vol. 9, pp. 56–69, 2014.
- [11] M. Diwakar and M. Kumar, "A review on ct image noise and its denoising," *Biomed. Signal Process. Control.*, vol. 42, pp. 73–88, 2018.
- [12] Y. Lan and X. Zhang, "Real-time ultrasound image despeckling using mixed-attention mechanism based residual unet," *IEEE Access*, vol. 8, pp. 195 327–195 340, 2020.
- [13] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola, "From local kernel to nonlocal multiple-model image denoising," *Int. J. Comput. Vis.*, vol. 86, no. 1, pp. 1–32, 2010.
- [14] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 479–486.
- [15] 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, 2007.
- [16] A. Buades, B. Coll, and J.-M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490–530, 2005.
- [17] 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, 2006.
- [18] L. P. Yaroslavsky, K. O. Egiazarian, and J. T. Astola, "Transform domain image restoration methods: review, comparison, and interpretation," in *Proc. Nonlinear Image Process. Pattern Anal. XII*, vol. 4304, 2001, pp. 155–169.
- [19] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Bm3d image denoising with shape-adaptive principal component analysis," in *Proc. Workshop on SPARS*, 2009, pp. 1–6.
- [20] M. Maggioni, V. Katkovnik, K. Egiazarian, and A. Foi, "Nonlocal transform-domain filter for volumetric data denoising and reconstruction," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 119–133, 2012.
- [21] A. Rajwade, A. Rangarajan, and A. Banerjee, "Image denoising using the higher order singular value decomposition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 4, pp. 849–862, 2012.
- [22] 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.
- [23] 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.
- [24] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.
- [25] A. Lucas, M. Iliadis, R. Molina, and A. K. Katsaggelos, "Using deep neural networks for inverse problems in imaging: beyond analytical methods," *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 20–36, 2018.
- [26] C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo, and C.-W. Lin, "Deep learning on image denoising: An overview," *Neural Netw.*, vol. 131, pp. 251–275, 2020.
- [27] 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, 2017.
- [28] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
- [29] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 807–814.
- [30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [31] J. Chen, J. Chen, H. Chao, and M. Yang, "Image blind denoising with generative adversarial network based noise modeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3155–3164.
- [32] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.
- [33] S. Lefkimmias, "Non-local color image denoising with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3587–3596.
- [34] A. Davy, T. Ehret, J.-M. Morel, P. Arias, and G. Facciolo, "Video denoising by combining patch search and cnns," *J. Math. Imag. Vis.*, pp. 1–16, 2020.
- [35] B. Wen, "Reproducible denoising methods," [Online]. Available: <https://github.com/wenbihan/reproducible-image-denoising-state-of-the-art>.
- [36] P. Milanfar, "A tour of modern image filtering: New insights and methods, both practical and theoretical," *IEEE Sig. Process. Mag.*, vol. 30, no. 1, pp. 106–128, 2012.
- [37] D. Thanh, P. Surya *et al.*, "A review on ct and x-ray images denoising methods," *Informatica*, vol. 43, no. 2, 2019.
- [38] L. Shao, R. Yan, X. Li, and Y. Liu, "From heuristic optimization to dictionary learning: A review and comprehensive comparison of image denoising algorithms," *IEEE Trans. Cybern.*, vol. 44, no. 7, pp. 1001–1013, 2013.
- [39] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.
- [40] S. Izadi, D. Sutton, and G. Hamarneh, "Image denoising in the deep learning era," *Artificial Intelligence Review*, pp. 1–46, 2022.
- [41] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Color image denoising via sparse 3d collaborative filtering with grouping constraint in luminance-chrominance space," in *Proc. IEEE Int. Conf. Image Process.*, 2007, pp. 313–316.
- [42] D. Kostadin, F. Alessandro, and E. Karen, "Video denoising by sparse 3d transform-domain collaborative filtering," in *Proc. 15th Eur. Signal Process. Conf.*, 2007, pp. 145–149.
- [43] M. Maggioni, G. Boracchi, A. Foi, and K. Egiazarian, "Video denoising, deblocking, and enhancement through separable 4-d nonlocal spatiotemporal transforms," *IEEE Trans. Image Process.*, vol. 21, no. 9, pp. 3952–3966, 2012.
- [44] H. Yue, J. Liu, J. Yang, T. Q. Nguyen, and F. Wu, "High iso jpeg image denoising by deep fusion of collaborative and convolutional filtering," *IEEE Trans. Image Process.*, vol. 28, no. 9, pp. 4339–4353, 2019.
- [45] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, and R. Timofte, "Plug-and-play image restoration with deep denoiser prior," *arXiv preprint arXiv:2008.13751*, 2020.
- [46] Y. Cai, X. Hu, H. Wang, Y. Zhang, H. Pfister, and D. Wei, "Learning to generate realistic noisy images via pixel-level noise-aware adversarial training," *Proc. Advances Neural Inf. Process. Syst.*, vol. 34, pp. 3259–3270, 2021.
- [47] M. Tassano, J. Delon, and T. Veit, "Fastdvdnet: Towards real-time deep video denoising without flow estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, June 2020.
- [48] J. Li, X. Wu, Z. Niu, and W. Zuo, "Unidirectional video denoising by mimicking backward recurrent modules with look-ahead forward ones," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2022, pp. 592–609.
- [49] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.
- [50] R. Ramanath, W. E. Snyder, Y. Yoo, and M. S. Drew, "Color image processing pipeline," *IEEE Sig. Process. Mag.*, vol. 22, no. 1, pp. 34–43, 2005.
- [51] M. Yan, "Restoration of images corrupted by impulse noise and mixed gaussian impulse noise using blind inpainting," *SIAM J. Imag. Sci.*, vol. 6, no. 3, pp. 1227–1245, 2013.
- [52] T. Huang, W. Dong, X. Xie, G. Shi, and X. Bai, "Mixed noise removal via laplacian scale mixture modeling and nonlocal low-rank approximation," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3171–3186, 2017.
- [53] Z. Zhang, G. Ely, S. Aeron, N. Hao, and M. Kilmer, "Novel methods for multilinear data completion and de-noising based on tensor-svd," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3842–3849.
- [54] Y. Chen, X. Cao, Q. Zhao, D. Meng, and Z. Xu, "Denoising hyperspectral image with non-iid noise structure," *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 1054–1066, 2017.
- [55] S. Xu, X. Cao, J. Peng, Q. Ke, C. Ma, and D. Meng, "Hyperspectral image denoising by asymmetric noise modeling," *IEEE Trans. Geosci. Remote Sens.*, 2022.
- [56] S. P. Awate and R. T. Whitaker, "Feature-preserving mri denoising: a nonparametric empirical bayes approach," *IEEE Trans. Med. Imag.*, vol. 26, no. 9, pp. 1242–1255, 2007.

- [57] Y. Romano, M. Elad, and P. Milanfar, "The little engine that could: Regularization by denoising (red)," *SIAM J. Imag. Sci.*, vol. 10, no. 4, pp. 1804–1844, 2017.
- [58] J. Xu, L. Zhang, W. Zuo, D. Zhang, and X. Feng, "Patch group based nonlocal self-similarity prior learning for image denoising," in *Proc. IEEE Conf. Comput. Vis.*, 2015, pp. 244–252.
- [59] Z. Kong and X. Yang, "A brief review of real-world color image denoising," *arXiv preprint arXiv:1809.03298*, 2018.
- [60] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, 1995.
- [61] K.-q. Huang, Z.-y. Wu, G. S. Fung, and F. H. Chan, "Color image denoising with wavelet thresholding based on human visual system model," *Signal Process.: Image Commun.*, vol. 20, no. 2, pp. 115–127, 2005.
- [62] H. Othman and S.-E. Qian, "Noise reduction of hyperspectral imagery using hybrid spatial-spectral derivative-domain wavelet shrinkage," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 2, pp. 397–408, 2006.
- [63] L. P. Yaroslavsky, "Local adaptive image restoration and enhancement with the use of dft and dct in a running window," in *Proc. SPIE*, vol. 2825, 1996, pp. 2–13.
- [64] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images," *IEEE Trans. Image Process.*, vol. 16, no. 5, pp. 1395–1411, 2007.
- [65] J. Dai, O. C. Au, L. Fang, C. Pang, F. Zou, and J. Li, "Multichannel nonlocal means fusion for color image denoising," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 11, pp. 1873–1886, 2013.
- [66] R. H. Chan, C.-W. Ho, and M. Nikolova, "Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization," *IEEE Trans. Image Process.*, vol. 14, no. 10, pp. 1479–1485, 2005.
- [67] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. Image Process.*, vol. 17, no. 1, pp. 53–69, 2007.
- [68] Y. Fu, A. Lam, I. Sato, and Y. Sato, "Adaptive spatial-spectral dictionary learning for hyperspectral image denoising," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 343–351.
- [69] P. Coupé, P. Yger, S. Prima, P. Hellier, C. Kervrann, and C. Barillot, "An optimized blockwise nonlocal means denoising filter for 3-d magnetic resonance images," *IEEE Trans. Med. Imag.*, vol. 27, no. 4, pp. 425–441, 2008.
- [70] S. Aja-Fernández, C. Alberola-López, and C.-F. Westin, "Noise and signal estimation in magnitude mri and rician distributed images: a lmmse approach," *IEEE Trans. Image Process.*, vol. 17, no. 8, pp. 1383–1398, 2008.
- [71] J. V. Manjón, P. Coupé, L. Martí-Bonmatí, D. L. Collins, and M. Robles, "Adaptive non-local means denoising of mr images with spatially varying noise levels," *J. Magn. Reson. Imag.*, vol. 31, no. 1, pp. 192–203, 2010.
- [72] C. Liu and W. T. Freeman, "A high-quality video denoising algorithm based on reliable motion estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 706–719.
- [73] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principal component analysis with local pixel grouping," *Pattern Recognit.*, vol. 43, no. 4, pp. 1531–1549, 2010.
- [74] 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, 2012.
- [75] A. Phophalia and S. K. Mitra, "3d mr image denoising using rough set and kernel pca method," *Magn. Reson. Imaging*, vol. 36, pp. 135–145, 2017.
- [76] S. Hurault, T. Ehret, and P. Arias, "Eppl: an image denoising method using a gaussian mixture model learned on a large set of patches," *Image Processing On Line*, vol. 8, pp. 465–489, 2018.
- [77] J. V. Manjón, P. Coupé, A. Buades, D. L. Collins, and M. Robles, "New methods for mri denoising based on sparseness and self-similarity," *Med. Image Anal.*, vol. 16, no. 1, pp. 18–27, 2012.
- [78] H. Zhang, W. He, L. Zhang, H. Shen, and Q. Yuan, "Hyperspectral image restoration using low-rank matrix recovery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 4729–4743, 2013.
- [79] J. Xu, L. Zhang, D. Zhang, and X. Feng, "Multi-channel weighted nuclear norm minimization for real color image denoising," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1096–1104.
- [80] J. V. Manjón, P. Coupé, and A. Buades, "Mri noise estimation and denoising using non-local pca," *Med. Image Anal.*, vol. 22, no. 1, pp. 35–47, 2015.
- [81] M. Lebrun, M. Colom, and J.-M. Morel, "Multiscale image blind denoising," *IEEE Trans. Image Process.*, vol. 24, no. 10, pp. 3149–3161, 2015.
- [82] A. Buades, J.-L. Lisani, and M. Miladinović, "Patch-based video denoising with optical flow estimation," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2573–2586, 2016.
- [83] M. Rizkinia, T. Baba, K. Shirai, and M. Okuda, "Local spectral component decomposition for multi-channel image denoising," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3208–3218, 2016.
- [84] T. Ehret, P. Arias, and J.-M. Morel, "Global patch search boosts video denoising," in *Proc. Int. Conf. Comput. Vis. Theory Appl.*, vol. 5, 2017, pp. 124–134.
- [85] L. Zhuang and J. M. Bioucas-Dias, "Fast hyperspectral image denoising and inpainting based on low-rank and sparse representations," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 3, pp. 730–742, 2018.
- [86] P. Arias and J.-M. Morel, "Video denoising via empirical bayesian estimation of space-time patches," *J. Math. Imag. Vis.*, vol. 60, no. 1, pp. 70–93, 2018.
- [87] J. Xu, L. Zhang, and D. Zhang, "External prior guided internal prior learning for real-world noisy image denoising," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2996–3010, 2018.
- [88] Y. Hou, J. Xu, M. Liu, G. Liu, L. Liu, F. Zhu, and L. Shao, "Nlh: A blind pixel-level non-local method for real-world image denoising," *IEEE Trans. Image Process.*, vol. 29, pp. 5121–5135, 2020.
- [89] G. Treece, "Real image denoising with a locally-adaptive bitonic filter," *IEEE Trans. Image Process.*, vol. 31, pp. 3151–3165, 2022.
- [90] N. Renard, S. Bourennane, and J. Blanc-Talon, "Denoising and dimensionality reduction using multilinear tools for hyperspectral images," *IEEE Geosci. Remote. Sens. Lett.*, vol. 5, no. 2, pp. 138–142, 2008.
- [91] X. Liu, S. Bourennane, and C. Fossati, "Denoising of hyperspectral images using the parafac model and statistical performance analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 10, pp. 3717–3724, 2012.
- [92] X. Zhang, Z. Xu, N. Jia, W. Yang, Q. Feng, W. Chen, and Y. Feng, "Denoising of 3d magnetic resonance images by using higher-order singular value decomposition," *Med. Image Anal.*, vol. 19, no. 1, pp. 75–86, 2015.
- [93] X. Zhang, J. Peng, M. Xu, W. Yang, Z. Zhang, H. Guo, W. Chen, Q. Feng, E. X. Wu, and Y. Feng, "Denoise diffusion-weighted images using higher-order singular value decomposition," *Neuroimage*, vol. 156, pp. 128–145, 2017.
- [94] Z. Zhang and S. Aeron, "Denoising and completion of 3d data via multidimensional dictionary learning," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, 2016, pp. 2371–2377.
- [95] W. He, H. Zhang, L. Zhang, and H. Shen, "Total-variation-regularized low-rank matrix factorization for hyperspectral image restoration," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 178–188, 2015.
- [96] W. Dong, G. Li, G. Shi, X. Li, and Y. Ma, "Low-rank tensor approximation with laplacian scale mixture modeling for multiframe image denoising," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 442–449.
- [97] Q. Xie, Q. Zhao, D. Meng, Z. Xu, S. Gu, W. Zuo, and L. Zhang, "Multispectral images denoising by intrinsic tensor sparsity regularization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1692–1700.
- [98] Y. Chang, L. Yan, and S. Zhong, "Hyper-laplacian regularized unidirectional low-rank tensor recovery for multispectral image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4260–4268.
- [99] Z. Kong, L. Han, X. Liu, and X. Yang, "A new 4-d nonlocal transform-domain filter for 3-d magnetic resonance images denoising," *IEEE Trans. Med. Imag.*, vol. 37, no. 4, pp. 941–954, 2017.
- [100] Z. Kong and X. Yang, "Color image and multispectral image denoising using block diagonal representation," *IEEE Trans. Image Process.*, vol. 28, no. 9, pp. 4247–4259, 2019.
- [101] W. He, H. Zhang, H. Shen, and L. Zhang, "Hyperspectral image denoising using local low-rank matrix recovery and global spatiotemporal total variation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 3, pp. 713–729, 2018.
- [102] Y. Wu, L. Fang, and S. Li, "Weighted tensor rank-1 decomposition for nonlocal image denoising," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2719–2730, 2018.

- [103] H. Lv and R. Wang, "Denoising 3d magnetic resonance images based on low-rank tensor approximation with adaptive multi-rank estimation," *IEEE Access*, vol. 7, pp. 85 995–86 003, 2019.
- [104] W. He, Q. Yao, C. Li, N. Yokoya, and Q. Zhao, "Non-local meets global: An integrated paradigm for hyperspectral denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6861–6870.
- [105] X. Gong, W. Chen, and J. Chen, "A low-rank tensor dictionary learning method for hyperspectral image denoising," *IEEE Trans. Signal Process.*, vol. 68, pp. 1168–1180, 2020.
- [106] Y. Chang, L. Yan, B. Chen, S. Zhong, and Y. Tian, "Hyperspectral image restoration: Where does the low-rank property exist," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6869–6884, 2020.
- [107] L. Zhuang, X. Fu, M. K. Ng, and J. M. Bioucas-Dias, "Hyperspectral image denoising based on global and nonlocal low-rank factorizations," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 12, pp. 10 438–10 454, 2021.
- [108] M. Xie, X. Liu, and X. Yang, "Novel hybrid low-rank tensor approximation for hyperspectral image mixed denoising based on global-guided-nonlocal prior mechanism," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–17, 2022.
- [109] H. Zhang, T.-Z. Huang, X.-L. Zhao, W. He, J. K. Choi, and Y.-B. Zheng, "Hyperspectral image denoising: Reconciling sparse and low-tensor-ring-rank priors in the transformed domain," *IEEE Trans. Geosci. Remote Sens.*, 2023.
- [110] A. S. Rubel, V. V. Lukin, and K. O. Egiazarian, "Metric performance in similar blocks search and their use in collaborative 3d filtering of grayscale images," in *Image Processing: Algorithms and Systems XII*, vol. 9019, 2014.
- [111] A. Foi and G. Boracchi, "Foveated nonlocal self-similarity," *Int. J. Comput. Vis.*, vol. 120, no. 1, pp. 78–110, 2016.
- [112] Y. Mkinen, L. Azzari, and A. Foi, "Collaborative filtering of correlated noise: Exact transform-domain variance for improved shrinkage and patch matching," *IEEE Trans. Image Process.*, 2020.
- [113] J. Liu, P. Musalski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 208–220, 2012.
- [114] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.
- [115] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000.
- [116] M. E. Kilmer and C. D. Martin, "Factorization strategies for third-order tensors," *Linear Algebra Appl.*, vol. 435, no. 3, pp. 641–658, 2011.
- [117] M. E. Kilmer, K. Braman, N. Hao, and R. C. Hoover, "Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging," *SIAM J. Matrix Anal. Appl.*, vol. 34, no. 1, pp. 148–172, 2013.
- [118] D. L. Donoho and J. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *biometrika*, vol. 81, no. 3, pp. 425–455, 1994.
- [119] D. Muti, S. Bourennane, and J. Marot, "Lower-rank tensor approximation and multiway filtering," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1172–1204, 2008.
- [120] I. V. Oseledets, "Tensor-train decomposition," *SIAM J. Sci. Comput.*, vol. 33, no. 5, pp. 2295–2317, 2011.
- [121] J. A. Bengua, H. N. Phien, H. D. Tuan, and M. N. Do, "Efficient tensor completion for color image and video recovery: Low-rank tensor train," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2466–2479, 2017.
- [122] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with bm3d?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2392–2399.
- [123] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1256–1272, 2016.
- [124] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3929–3938.
- [125] S. Lefkimiatis, "Universal denoising networks : A novel cnn architecture for image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3204–3213.
- [126] Q. Yuan, Q. Zhang, J. Li, H. Shen, and L. Zhang, "Hyperspectral image denoising employing a spatial-spectral deep residual convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1205–1218, 2018.
- [127] K. Zhang, W. Zuo, and L. Zhang, "Ffdnet: Toward a fast and flexible solution for cnn-based image denoising," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4608–4622, 2018.
- [128] Y. Chang, L. Yan, H. Fang, S. Zhong, and W. Liao, "Hsi-denet: Hyperspectral image restoration via convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 667–682, 2018.
- [129] J. V. Manjón and P. Coupé, "Mri denoising using deep learning," in *Patch-Based Techniques in Medical Imaging*, 2018, pp. 12–19.
- [130] R. A. Yeh, T. Y. Lim, C. Chen, A. G. Schwing, M. Hasegawa-Johnson, and M. Do, "Image restoration with deep generative models," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 6772–6776.
- [131] C. Chen, Z. Xiong, X. Tian, Z.-J. Zha, and F. Wu, "Real-world image denoising with deep boosting," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.
- [132] S. Yu, B. Park, and J. Jeong, "Deep iterative down-up cnn for image denoising," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019.
- [133] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, "Toward convolutional blind denoising of real photographs," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1712–1722.
- [134] M. Claus and J. Van Gemert, "Videnn: Deep blind video denoising," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019, pp. 0–0.
- [135] S. Anwar and N. Barnes, "Real image denoising with feature attention," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 3155–3164.
- [136] A. Abbasi, A. Monadjemi, L. Fang, H. Rabbani, and Y. Zhang, "Three-dimensional optical coherence tomography image denoising through multi-input fully-convolutional networks," *Comput. Biol. Med.*, vol. 108, pp. 1–8, 2019.
- [137] Y. Song, Y. Zhu, and X. Du, "Dynamic residual dense network for image denoising," *Sensors*, vol. 19, no. 17, p. 3809, 2019.
- [138] S. Gu, Y. Li, L. V. Gool, and R. Timofte, "Self-guided network for fast image denoising," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 2511–2520.
- [139] K. Lin, T. H. Li, S. Liu, and G. Li, "Real photographs denoising with noise domain adaptation and attentive generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, June 2019.
- [140] K. Wei, Y. Fu, and H. Huang, "3-d quasi-recurrent neural network for hyperspectral image denoising," *IEEE Trans. Neural Netw. Learn. Syst.*, 2020.
- [141] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, "Cycleisp: Real image restoration via improved data synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2696–2705.
- [142] D. Valsesia, G. Fracastoro, and E. Magli, "Deep graph-convolutional image denoising," *IEEE Trans. Image Process.*, vol. 29, pp. 8226–8237, 2020.
- [143] Z. Yue, Q. Zhao, L. Zhang, and D. Meng, "Dual adversarial network: Toward real-world noise removal and noise generation," in *Proc. Eur. Conf. Comput. Vis.*, 2020.
- [144] G. Vaksman, M. Elad, and P. Milanfar, "Lidia: Lightweight learned image denoising with instance adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, June 2020.
- [145] M. Chang, Q. Li, H. Feng, and Z. Xu, "Spatial-adaptive network for single image denoising," *arXiv preprint arXiv:2001.10291*, 2020.
- [146] Y. Kim, J. W. Soh, G. Y. Park, and N. I. Cho, "Transfer learning from synthetic to real-noise denoising with adaptive instance normalization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3482–3492.
- [147] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, "Learning enriched features for real image restoration and enhancement," in *Proc. Eur. Conf. Comput. Vis.*, 2020.
- [148] Y. Zhou, J. Jiao, H. Huang, Y. Wang, J. Wang, H. Shi, and T. Huang, "When awgn-based denoiser meets real noises," in *Proc. AAAI*, vol. 34, no. 07, 2020, pp. 13 074–13 081.
- [149] Y. Zhao, D. Zhai, J. Jiang, and X. Liu, "Adrn: Attention-based deep residual network for hyperspectral image denoising," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 2668–2672.
- [150] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, "Multi-stage progressive image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14 821–14 831.

- [151] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, and R. Timofte, "Plug-and-play image restoration with deep denoiser prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 6360–6376, 2021.
- [152] C. Tian, Y. Xu, W. Zuo, B. Du, C.-W. Lin, and D. Zhang, "Designing and training of a dual cnn for image denoising," *Knowledge-Based Systems*, vol. 226, p. 106949, 2021.
- [153] Y. Liu, Z. Qin, S. Anwar, P. Ji, D. Kim, S. Caldwell, and T. Gedeon, "Invertible denoising network: A light solution for real noise removal," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 13 365–13 374.
- [154] H. Zheng, H. Yong, and L. Zhang, "Deep convolutional dictionary learning for image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 630–641.
- [155] S. Cheng, Y. Wang, H. Huang, D. Liu, H. Fan, and S. Liu, "Nbnet: Noise basis learning for image denoising with subspace projection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4896–4906.
- [156] C. Ren, X. He, C. Wang, and Z. Zhao, "Adaptive consistency prior based deep network for image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8596–8606.
- [157] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "Swinir: Image restoration using swin transformer," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 1833–1844.
- [158] L. Chen, X. Chu, X. Zhang, and J. Sun, "Simple baselines for image restoration," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2022, pp. 17–33.
- [159] Y. Jiang, B. Wronski, B. Mildenhall, J. T. Barron, Z. Wang, and T. Xue, "Fast and high quality image denoising via malleable convolution," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2022, pp. 429–446.
- [160] H. Chen, J. Gu, Y. Liu, S. A. Magid, C. Dong, Q. Wang, H. Pfister, and L. Zhu, "Masked image training for generalizable deep image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023.
- [161] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2noise: Learning image restoration without clean data," *arXiv preprint arXiv:1803.04189*, 2018.
- [162] H. Yan, X. Chen, V. Y. Tan, W. Yang, J. Wu, and J. Feng, "Unsupervised image noise modeling with self-consistent gan," *arXiv preprint arXiv:1906.05762*, 2019.
- [163] 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.
- [164] Y. Quan, M. Chen, T. Pang, and H. Ji, "Self2self with dropout: Learning self-supervised denoising from single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1890–1898.
- [165] G. Jang, W. Lee, S. Son, and K. M. Lee, "C2n: Practical generative noise modeling for real-world denoising," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 2350–2359.
- [166] D. Y. Sheth, S. Mohan, J. L. Vincent, R. Manzorro, P. A. Crozier, M. M. Khapra, E. P. Simoncelli, and C. Fernandez-Granda, "Unsupervised deep video denoising," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 1759–1768.
- [167] T. Pang, H. Zheng, Y. Quan, and H. Ji, "Recorrupted-to-recorrupted: unsupervised deep learning for image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2043–2052.
- [168] T. Huang, S. Li, X. Jia, H. Lu, and J. Liu, "Neighbor2neighbor: Self-supervised denoising from single noisy images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14 781–14 790.
- [169] Y. Zhang, D. Li, K. L. Law, X. Wang, H. Qin, and H. Li, "Idr: Self-supervised image denoising via iterative data refinement," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 2098–2107.
- [170] R. Neshatavar, M. Yavartanoo, S. Son, and K. M. Lee, "Cvf-sid: Cyclic multi-variate function for self-supervised image denoising by disentangling noise from image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17 583–17 591.
- [171] Z. Wang, J. Liu, G. Li, and H. Han, "Blind2unblind: Self-supervised image denoising with visible blind spots," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 2027–2036.
- [172] 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 Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17 725–17 734.
- [173] Y. Miao, L. Zhang, L. Zhang, and D. Tao, "Dds2m: Self-supervised denoising diffusion spatio-spectral model for hyperspectral image restoration," *arXiv preprint arXiv:2303.06682*, 2023.
- [174] J. Li, Z. Zhang, X. Liu, C. Feng, X. Wang, L. Lei, and W. Zuo, "Spatially adaptive self-supervised learning for real-world image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023.
- [175] A. Abdelhamed, R. Timofte, M. S. Brown, S. Yu, B. Park, J. Jeong, S. W. Jung, D. W. Kim, J. R. Chung, J. Liu *et al.*, "Ntire 2019 challenge on real image denoising: Methods and results," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019, pp. 2197–2210.
- [176] V. Jain and S. Seung, "Natural image denoising with convolutional networks," *Proc. Advances Neural Inf. Process. Syst.*, vol. 21, 2008.
- [177] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 9446–9454.
- [178] Y. Zhou, R. Chellappa, and B. Jenkins, "A novel approach to image restoration based on a neural network," in *Proceedings of the International Conference on Neural Networks, San Diego, California*, 1987.
- [179] Y.-W. Chiang and B. Sullivan, "Multi-frame image restoration using a neural network," in *Proc. Midwest Symp. Circuits Syst.* IEEE, 1989, pp. 744–747.
- [180] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [181] V. Jain and S. Seung, "Natural image denoising with convolutional networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2009, pp. 769–776.
- [182] A. Guo, L. Fang, M. Qi, and S. Li, "Unsupervised denoising of optical coherence tomography images with nonlocal-generative adversarial network," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–12, 2020.
- [183] C. A. Metzler, A. Mousavi, R. Heckel, and R. G. Baraniuk, "Unsupervised learning with stein's unbiased risk estimator," *arXiv preprint arXiv:1805.10531*, 2018.
- [184] S. Soltanayev and S. Y. Chun, "Training deep learning based denoisers without ground truth data," *Proc. Advances Neural Inf. Process. Syst.*, vol. 31, 2018.
- [185] J. Batson and L. Royer, "Noise2self: Blind denoising by self-supervision," in *Proc. 36th Int. Conf. Mach. Learn.* PMLR, 2019, pp. 524–533.
- [186] J. Xu, Y. Huang, M.-M. Cheng, L. Liu, F. Zhu, Z. Xu, and L. Shao, "Noisy-as-clean: Learning self-supervised denoising from corrupted image," *IEEE Trans. Image Process.*, vol. 29, pp. 9316–9329, 2020.
- [187] J. Song, J.-H. Jeong, D.-S. Park, H.-H. Kim, D.-C. Seo, and J. C. Ye, "Unsupervised denoising for satellite imagery using wavelet directional cyclegan," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6823–6839, 2020.
- [188] B. Kawar, M. Elad, S. Ermon, and J. Song, "Denoising diffusion restoration models," *arXiv preprint arXiv:2201.11793*, 2022.
- [189] S. Fadnavis, J. Batson, and E. Garyfallidis, "Patch2self: Denoising diffusion mri with self-supervised learning," *Proc. Advances Neural Inf. Process. Syst.*, vol. 33, pp. 16 293–16 303, 2020.
- [190] Y. Song and S. Ermon, "Improved techniques for training score-based generative models," *Proc. Advances Neural Inf. Process. Syst.*, vol. 33, pp. 12 438–12 448, 2020.
- [191] P. Chatterjee and P. Milanfar, "Is denoising dead?" *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 895–911, 2009.
- [192] X. Zhang and R. Wu, "Fast depth image denoising and enhancement using a deep convolutional network," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2016, pp. 2499–2503.
- [193] J. Anaya and A. Barbu, "Renoir—a dataset for real low-light image noise reduction," *J. Vis. Commun. Image Represent.*, vol. 51, pp. 144–154, 2018.
- [194] S. Nam, Y. Hwang, Y. Matsushita, and S. Joo Kim, "A holistic approach to cross-channel image noise modeling and its application to image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1683–1691.
- [195] J. Xu, H. Li, Z. Liang, D. Zhang, and L. Zhang, "Real-world noisy image denoising: A new benchmark," *arXiv preprint arXiv:1804.02603*, 2018.

- [196] A. Abdelhamed, S. Lin, and M. S. Brown, "A high-quality denoising dataset for smartphone cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1692–1700.
- [197] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 724–732.
- [198] H. Yue, C. Cao, L. Liao, R. Chu, and J. Yang, "Supervised raw video denoising with a benchmark dataset on dynamic scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2301–2310.
- [199] F. Yasuma, T. Mitsunaga, D. Iso, and S. Nayar, "Generalized Assorted Pixel Camera: Post-Capture Control of Resolution, Dynamic Range and Spectrum," Tech. Rep., Nov 2008.
- [200] B. Arad and O. Ben-Shahar, "Sparse recovery of hyperspectral signal from natural rgb images," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 19–34.
- [201] A. Chakrabarti and T. Zickler, "Statistics of real-world hyperspectral images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 193–200.
- [202] C. A. Cocosco, V. Kollokian, R. K.-S. Kwan, G. B. Pike, and A. C. Evans, "Brainweb: Online interface to a 3d mri simulated brain database," in *NeuroImage*, 1997.
- [203] D. S. Marcus, T. H. Wang, J. Parker, J. G. Csernansky, J. C. Morris, and R. L. Buckner, "Open access series of imaging studies (oasis): cross-sectional mri data in young, middle aged, nondemented, and demented older adults," *J. Cogn. Neurosci.*, vol. 19, no. 9, pp. 1498–1507, 2007.
- [204] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, July 2001, pp. 416–423.
- [205] Kodak, "Kodak gallery dataset," [Online]. Available: <http://r0k.us/graphics/kodak>.
- [206] T. Plotz and S. Roth, "Benchmarking denoising algorithms with real photographs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1586–1595.
- [207] Derf, "Derfs test media collection," [Online]. Available: <https://media.xiph.org/video/derf>.
- [208] X. Xu, Y. Yu, N. Jiang, J. Lu, B. Yu, and J. Jia, "Pvdd: A practical video denoising dataset with real-world dynamic scenes," *arXiv preprint arXiv:2207.01356*, 2022.
- [209] M. F. Baumgardner, L. L. Biehl, and D. A. Landgrebe, "220 band aviris hyperspectral image data set: June 12, 1992 indian pine test site 3," Sep 2015. [Online]. Available: <https://purr.purdue.edu/publications/1947/1>
- [210] T. Zhang, Y. Fu, and C. Li, "Hyperspectral image denoising with realistic data," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 2248–2257.
- [211] J. Zbontar, F. Knoll, A. Sriram, T. Murrell, Z. Huang, M. J. Muckley, A. Defazio, R. Stern, P. Johnson, M. Bruno *et al.*, "fastmri: An open dataset and benchmarks for accelerated mri," *arXiv preprint arXiv:1811.08839*, 2018.
- [212] K. Wei, Y. Fu, J. Yang, and H. Huang, "A physics-based noise formation model for extreme low-light raw denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2758–2767.
- [213] Y. Wang, H. Huang, Q. Xu, J. Liu, Y. Liu, and J. Wang, "Practical deep raw image denoising on mobile devices," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 1–16.
- [214] 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, 2004.
- [215] C. Tian, Y. Xu, and W. Zuo, "Image denoising using deep cnn with batch renormalization," *Neural Networks*, vol. 121, pp. 461–473, 2020.
- [216] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, and M.-H. Yang, "Restormer: Efficient transformer for high-resolution image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 5728–5739.
- [217] S. Lefkimiatis, "Universal denoising networks: a novel cnn architecture for image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3204–3213.
- [218] J. W. Soh and N. I. Cho, "Variational deep image restoration," *IEEE Trans. Image Process.*, vol. 31, pp. 4363–4376, 2022.
- [219] Z. Yue, H. Yong, Q. Zhao, D. Meng, and L. Zhang, "Variational denoising network: Toward blind noise modeling and removal," in *Proc. Advances Neural Inf. Process. Syst.*, 2019, pp. 1690–1701.
- [220] W. Dong, T. Huang, G. Shi, Y. Ma, and X. Li, "Robust tensor approximation with laplacian scale mixture modeling for multi-frame image and video denoising," *IEEE J. Sel. Top. Signal Process.*, vol. 12, no. 6, pp. 1435–1448, 2018.
- [221] B. Wen, S. Ravishankar, and Y. Bresler, "Vidosat: High-dimensional sparsifying transform learning for online video denoising," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1691–1704, 2018.
- [222] M. Tassano, J. Delon, and T. Veit, "Dvdnet: A fast network for deep video denoising," in *Proc. IEEE Conf. Int. Image Process.*, 2019, pp. 1805–1809.
- [223] L. Sun, W. Dong, X. Li, J. Wu, L. Li, and G. Shi, "Deep maximum a posterior estimator for video denoising," *Int. J. Comput. Vis.*, vol. 129, pp. 2827–2845, 2021.
- [224] H. Chen, Y. Jin, K. Xu, Y. Chen, and C. Zhu, "Multiframe-to-multiframe network for video denoising," *IEEE Trans. Multimed.*, vol. 24, pp. 2164–2178, 2021.
- [225] S. Lee, D. Cho, J. Kim, and T. H. Kim, "Restore from restored: Video restoration with pseudo clean video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 3537–3546.
- [226] A. Davy, T. Ehret, J.-M. Morel, P. Arias, and G. Facciolo, "A non-local cnn for video denoising," in *Proc. IEEE Conf. Int. Image Process.*, 2019, pp. 2409–2413.
- [227] Y. Fang, H. Zhu, K. Ma, Z. Wang, and S. Li, "Perceptual evaluation for multi-exposure image fusion of dynamic scenes," *IEEE Trans. Image Process.*, vol. 29, pp. 1127–1138, 2019.
- [228] S. Voigt, T. Kemper, T. Riedlinger, R. Kiefl, K. Scholte, and H. Mehl, "Satellite image analysis for disaster and crisis-management support," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 6, pp. 1520–1528, 2007.
- [229] R. H. Yuhas, J. W. Boardman, and A. F. Goetz, "Determination of semi-arid landscape endmembers and seasonal trends using convex geometry spectral unmixing techniques," *ratio*, vol. 4, p. 22, 1990.
- [230] L. Wald, *Data fusion: definitions and architectures: fusion of images of different spatial resolutions*. Les Presses de l'Ecole des Mines, 2002.
- [231] Y. Chen, W. He, N. Yokoya, and T.-Z. Huang, "Hyperspectral image restoration using weighted group sparsity-regularized low-rank tensor decomposition," *IEEE Trans. Cybern.*, vol. 50, no. 8, pp. 3556–3570, 2019.
- [232] M. Ye, Y. Qian, and J. Zhou, "Multitask sparse nonnegative matrix factorization for joint spectral-spatial hyperspectral imagery denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2621–2639, 2014.
- [233] A. Lam, I. Sato, and Y. Sato, "Denoising hyperspectral images using spectral domain statistics," in *Proc. Int. Conf. Pattern Recognit.* IEEE, 2012, pp. 477–480.
- [234] H. Zeng, X. Xie, and J. Ning, "Hyperspectral image denoising via global spatial-spectral total variation regularized nonconvex local low-rank tensor approximation," *Signal Process.*, vol. 178, p. 107805, 2021.
- [235] X. Cao, X. Fu, C. Xu, and D. Meng, "Deep spatial-spectral global reasoning network for hyperspectral image denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2021.
- [236] A. Maffei, J. M. Haut, M. E. Paoletti, J. Plaza, L. Bruzzone, and A. Plaza, "A single model cnn for hyperspectral image denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2516–2529, 2019.
- [237] M. Li, Y. Fu, and Y. Zhang, "Spatial-spectral transformer for hyperspectral image denoising," *arXiv preprint arXiv:2211.14090*, 2022.
- [238] A. Foi, "Noise estimation and removal in mr imaging: The variance-stabilization approach," in *Proc. IEEE Int. Symp. Biomed. Imag., Nano Macro*, 2011, pp. 1809–1814.
- [239] Q. Wu, W. Ren, and X. Cao, "Learning interleaved cascade of shrinkage fields for joint image dehazing and denoising," *IEEE Trans. Image Process.*, vol. 29, pp. 1788–1801, 2019.
- [240] L. Liu, X. Jia, J. Liu, and Q. Tian, "Joint demosaicing and denoising with self guidance," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2240–2249.
- [241] D. Liu, B. Wen, J. Jiao, X. Liu, Z. Wang, and T. S. Huang, "Connecting image denoising and high-level vision tasks via deep learning," *IEEE Trans. Image Process.*, vol. 29, pp. 3695–3706, 2020.