External Prior Guided Internal Prior Learning
for Real-World Noisy Image Denoising

Jun Xu[1], *Student Member, IEEE*, Lei Zhang[1,*], *Fellow, IEEE*, and David Zhang[1,2], *Fellow, IEEE*

[1]Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR, China

[2]School of Science and Engineering, The Chinese University of Hong Kong (Shenzhen), Shenzhen, China

Abstract: Most of existing image denoising methods learn image priors from either external data or the noisy image itself to remove noise. However, priors learned from external data may not be adaptive to the image to be denoised, while priors learned from the given noisy image may not be accurate due to the interference of corrupted noise. Meanwhile, the noise in real-world noisy images is very complex, which is hard to be described by simple distributions such as Gaussian distribution, making real-world noisy image denoising a very challenging problem. We propose to exploit the information in both external data and the given noisy image, and develop an external prior guided internal prior learning method for real-world noisy image denoising. We first learn external priors from an independent set of clean natural images. With the aid of learned external priors, we then learn internal priors from the given noisy image to refine the prior model. The external and internal priors are formulated as a set of orthogonal dictionaries to efficiently reconstruct the desired image. Extensive experiments are performed on several real-world noisy image datasets. The proposed method demonstrates highly competitive denoising performance, outperforming state-of-the-art denoising methods including those designed for real-world noisy images.

Index Terms—Image Denoising, Real-World Noisy Image, Image Prior Learning, Guided Dictionary Learning.

I. INTRODUCTION

IMAGE denoising is a crucial and indispensable step to improve image quality in digital imaging systems. In particular, with the decrease of size of CMOS/CCD sensors, image is more easily to be corrupted by noise and hence denoising is becoming increasingly important for high resolution imaging. The problem of image denoising has been extensively studied in literature and numerous image denoising methods [1]–[44] have been proposed in the past decades. Most of existing denoising methods focus on the scenario of additive white Gaussian noise (AWGN) [1]–[25], where the observed noisy image $\mathbf{y}$ is modeled as the addition of clean image $\mathbf{x}$ and AWGN $\mathbf{n}$, i.e., $\mathbf{y} = \mathbf{x} + \mathbf{n}$. There are also methods proposed for removing Poisson noise [26], [27], mixed Poisson and Gaussian noise [28]–[31], mixed Gaussian and impulse noise [32]–[34], and realistic noise in real photography [35]–[44].

Natural images have many properties, such as sparsity and nonlocal self-similarity, which can be employed as useful priors for designing image denoising methods. Based on the facts that natural images will be sparsely distributed in some transformed domain, wavelet [1] and curvelet [2] transforms have been widely adopted for image denoising. The sparse representation based methods [3]–[8] encode image patches over a dictionary by using ℓ_1-norm minimization to enforce the sparsity. The well-known bilateral filters [9] employ the prior information that image pixels exhibit similarity in both spatial domain and intensity domain. Other image priors such as multiscale self-similarity [10] and nonlocal self-similarity [11], [12], or the combination of multiple image priors [13], [14] have also been successfully used in image denoising. For example, by using low-rank minimization to characterize

the image nonlocal self-similarity, the WNNM [13] method achieves state-of-the-art performance for AWGN denoising.

Instead of using predefined image priors, methods have also been proposed to learn priors from natural images for denoising. The generative image prior learning methods usually learn prior models from a set of external clean images and apply the learned prior models to the given noisy image [14]–[19], or learn priors from the given noisy image to perform denoising [3]. Recently, the discriminative image prior learning methods [20]–[25], which learn denoising models from pairs of clean and noisy images, have been becoming popular. The representative methods include the neural network based methods [20]–[22], random fields based methods [23], [24], and reaction diffusion based methods [25].

Most of the above mentioned methods focus on AWGN removal, however, the assumption of AWGN is too ideal to be true for real-world noisy images, where the noise is much more complex and varies with different scenes, cameras and camera settings (ISO, shutter speed, and aperture, etc.) [42], [45]. As a result, many denoising methods in literature, including those learning based methods, become less effective when applied to real-world noisy images. Fig. 1 shows an example, where we apply some representative and state-of-the-art denoising methods, including CBM3D [7], WNNM [13], DnCNN [22], CSF [24], and TNRD [25] to a real-world noisy image (captured by a Nikon D800 camera with ISO is 3200) provided in [42]. One can see that these methods either remain much the noise or over-smooth the image details.

There have been a few methods [35]–[43] and software toolboxes [44] developed for real-world noisy image denoising. Almost all of these methods follow a two-stage framework: first estimate the parameters of the noise model (usually assumed to be Gaussian or mixture of Gaussians (MoG)), and then perform denoising with the estimated noise model. However, the noise in real-world noisy images is very complex
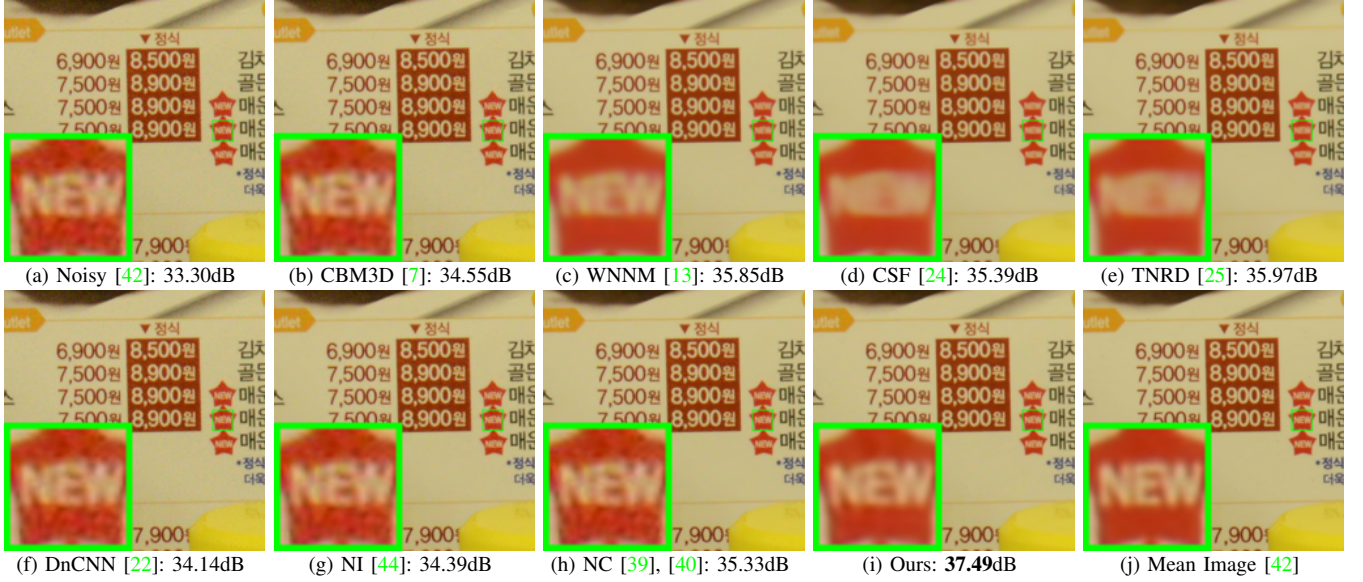
Fig. 1: Denoised images of a region cropped from the real-world noisy image "Nikon D800 ISO 3200 A3" [42] by different methods. The scene was shot 500 times with the same camera and camera setting. The mean image of the 500 shots is roughly taken as the "ground truth", with which the PSNR can be computed. The images are better viewed by zooming in on screen.

and is hard to be modeled by explicit distributions such as Gaussian and MoG. According to [45], the noise corrupted in the in-camera imaging process [42], [46]–[48] is signal dependent and comes from five main sources: photon shot, fixed pattern, dark current, readout, and quantization noise. The existing methods [35]–[42], [44] mentioned above may not perform well on real-world noisy image denoising tasks. Fig. 1 also shows the denoising results of two real-world noisy image denoising methods, Noise Clinic [39], [40] and Neat Image [44]. One can see that these two methods still generate much noise caused artifacts.

This work aims to develop a new paradigm for real-world noisy image denoising. Different from existing real-world noisy image denoising methods [35]–[42] which focus on noise modeling, we focus on image prior learning. We argue that with a strong and adaptive prior learning scheme, robust denoising performance on real-world noisy images can still be obtained. To achieve this goal, we propose to first learn image priors from external clean images, and then employ the learned external priors to guide the learning of internal priors from the given noisy image. The flowchart of the proposed method is illustrated in Fig. 2. We first extract millions of patch groups (PGs) from a set of high quality natural images, with which a Gaussian Mixture Model (GMM) is learned as the external image prior. The learned GMM prior model is used to assign each PG extracted from the given noisy image into its most suitable cluster via maximum a-posterior, and then an external-internal hybrid orthogonal dictionary is learned as the final prior for each cluster, with which the denoising can be readily performed by weighted sparse coding with closed form solution. The external priors learned from clean images preserve fine-scale image structural information, which is hard to be reproduced from noisy images. Therefore, external dictionary can serve as a good supplement to the internal dictionary. Our proposed denoising method is simple and efficient, yet our extensive experiments on real-world

noisy images demonstrate its better denoising performance than the current state-of-the-arts.

II. RELATED WORK

A. Internal and External Prior Learning

Learning natural image priors plays a key role in image denoising [3]–[5], [8], [10], [14]–[25]. There are mainly four categories of prior learning based methods. 1) External prior learning methods [14]–[16] learn priors (e.g., dictionaries) from a set of external clean images, and the learned priors are used to recover the latent clean image from the given noisy image. 2) Internal prior learning methods [3]–[5], [8], [10] directly learn priors from a given noisy image, and image denoising is often done simultaneously with the prior learning process. 3) Discriminative prior learning methods [20]–[25] learn discriminative models or mapping functions from clean and noisy image pairs, and the learned models or mapping functions are applied to a noisy image for denoising. 4) Hybrid methods [17]–[19] combine the external and internal priors to denoise the given input image.

It has been shown [14]–[16] that the external priors learned from natural clean images are effective and efficient for universal image denoising problems, whereas they are not adaptive to the given noisy image and some fine-scale image structures may not be well recovered. By contrast, the internal priors learned from the given noisy image are adaptive to image content, but the learned priors can be much affected by noise and the learning processing is usually slow [3]–[5], [8], [10]. Besides, most of the internal prior learning methods [3]–[5], [8], [10] assume additive white Gaussian noise (AWGN), making the learned priors less robust for real-world noisy images. In this paper, we use external priors to guide the internal prior learning. Our method is not only much faster than the traditional internal learning methods, but also very robust to denoise real-world noisy images.

In [17], the authors employed external clean patches to denoise noisy patches with high individual Signal-to-Noise-
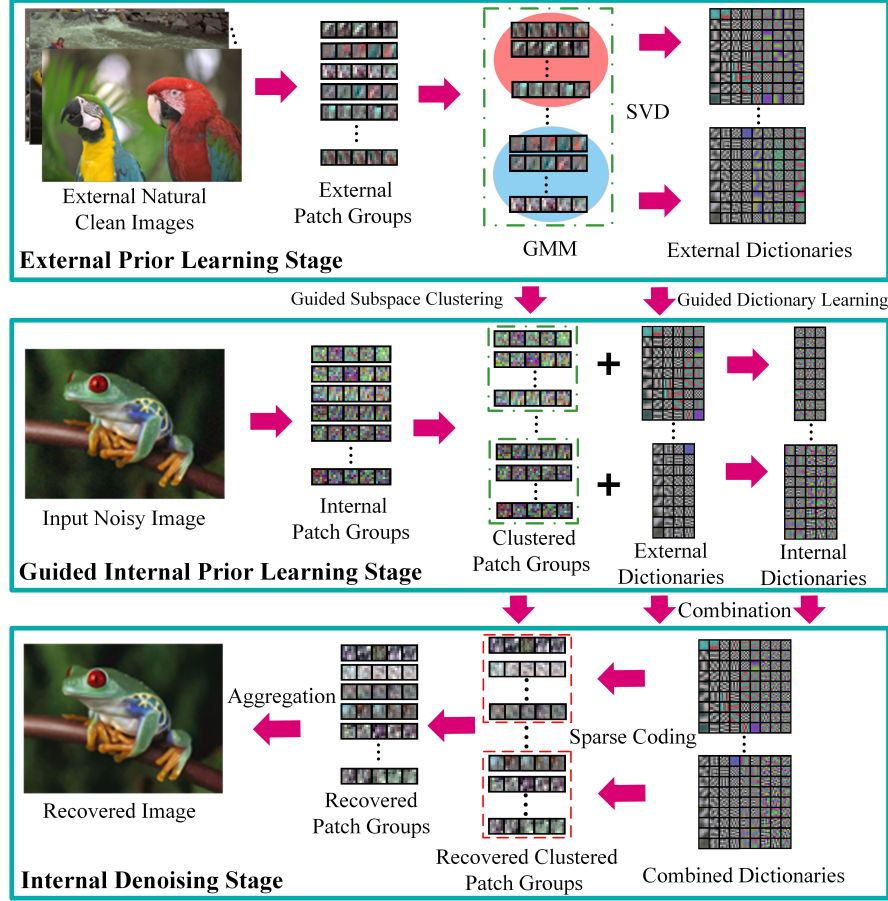
Fig. 2: Flowchart of the proposed external prior guided internal prior learning and denoising framework.

Ratio (PatchSNR), and employed internal noisy patches to denoise noisy patches with low PatchSNR. This is essentially different from our work which employs the external patch group based prior to guide the clustering and dictionary learning of the internal noisy patch groups. In [18], the external priors are only used to guide the internal patch clustering for image denoising, while in our work, the learned external priors are employed to guide not only the internal clustering, but also the internal dictionary learning. Besides, the method of [18] follows a patch based framework for AWGN removal, while in our work we employ a patch group based framework for real-world noisy image denoising. In addition, some technical details are also different. For example, method in [18] utilizes low-rank minimization for denoising, while we use dictionary learning and sparse coding for denoising. In the Targeted Image Denoising (TID) method [19], targeted images are selected from a large dataset for each patch in the input noisy image for denoising, which is computationally expensive.

B. Real-World Noisy Image Denoising

Most of the denoising methods in literature [1]–[16], [20]–[25] assume AWGN noise and use simulated noisy images for algorithm design and evaluation. Recently, several denoising methods have been proposed to remove unknown noise from real-world noisy images [35]–[42]. Portilla [35] employed a correlated Gaussian model to estimate the noise of each wavelet subband. Rabie [36] modeled the noisy pixels as outliers and performed denoising via Lorentzian robust estimator. Liu et al. [37] proposed the "noise level function" to estimate the noise and performed denoising by learning a Gaussian conditional random field. Gong et al. [38] proposed to model the data fitting term via weighted sum of ℓ_1 and ℓ_2 norms and performed denoising by a simple sparsity regularization term in the wavelet transform domain. The "Noise Clinic" [39], [40] estimates the noise distribution by using a multivariate Gaussian model and removes the noise by using a generalized version of nonlocal Bayesian model [12]. Zhu et al. [41] proposed a Bayesian method to approximate and remove the noise via a low-rank mixture of Gaussians (MoG) model. The method in [42] models the cross-channel noise in real-world noisy image as a multivariate Gaussian and the noise is removed by the Bayesian nonlocal means filter [49]. The commercial software Neat Image [44] estimates the noise parameters from a flat region of the given noisy image and filters the noise correspondingly.

The methods [35]–[42] emphasize much on the noise modeling, and they use Gaussian or MoG to model the noise in real-world noisy images. Nonetheless, the noise in real-world noisy images is very complex and hard to be modeled by explicit distributions [45]. These works ignore the importance of learning image priors, which actually can be easier to model compared with modeling the complex realistic noise. In this paper, we propose a simple yet effective image prior learning method for real-world noisy image denoising. Due to its strong prior modeling ability, the proposed method simply models the noise as locally Gaussian, and it achieves highly competitive performance on real-world noisy image denoising.

III. EXTERNAL PRIOR GUIDED INTERNAL PRIOR LEARNING FOR IMAGE DENOISING

In this section, we first describe the learning of external prior, and then describe in detail the guided internal prior learning method, followed by the denoising algorithm.

A. Learn External Patch Group Priors

The nonlocal self-similarity based patch group (PG) prior learning [14] has proved to be very effective for image denosing. In this work, we extract PGs from natural clean images to learn external priors. A PG is a group of similar patches to a local patch. In our method, each local patch is extracted from a RGB image with patch size $p \times p \times 3$. We search the M most similar (i.e., smallest Euclidean distance) patches to this local patch (including the local patch itself) in a $W \times W$ region around it. Each patch is stretched to a patch vector $\mathbf{x}_m \in \mathbb{R}^{3p^2 \times 1}$ to form the PG, denoted by $\{\mathbf{x}_m\}_{m=1}^M$. The mean vector of this PG is $\boldsymbol{\mu} = \frac{1}{M} \sum_{m=1}^M \mathbf{x}_m$, and the group mean subtracted PG is defined as $\overline{\mathbf{X}} \triangleq \{\overline{\mathbf{x}}_m = \mathbf{x}_m - \boldsymbol{\mu}\}_{m=1}^M$.

Assume that a number of L PGs are extracted from a set of external natural images, and the l-th PG is $\overline{\mathbf{X}}_l \triangleq \{\overline{\mathbf{x}}_{l,m}\}_{m=1}^M, l = 1, ..., L$. A Gaussian Mixture Model (GMM) is learned to model the PG prior. The overall log-likelihood function is

$$\ln \mathcal{L} = \sum_{l=1}^L \ln(\sum_{k=1}^K \pi_k \prod_{m=1}^M \mathcal{N}(\overline{\mathbf{x}}_{l,m} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)). \tag{1}$$

The learning process is similar to the GMM learning in [14], [16], [50]. Finally, a GMM model with K Gaussian components is learned, and the learned parameters include mixture weights $\{\pi_k\}_{k=1}^K$, mean vectors $\{\boldsymbol{\mu}_k\}_{k=1}^K$, and covariance matrices $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$. Note that the mean vector of each cluster is naturally zero, i.e., $\boldsymbol{\mu}_k = \mathbf{0}$.

To better describe the subspace of each Gaussian component, we perform singular value decomposition (SVD) [51] on the covariance matrix:

$$\boldsymbol{\Sigma}_k = \boldsymbol{U}_k \boldsymbol{S}_k \boldsymbol{U}_k^\top. \tag{2}$$

The eigenvector matrices $\{\boldsymbol{U}_k\}_{k=1}^K$ will be employed as the external orthogonal dictionary to guide the internal sub-dictionary learning in next sub-section. The singular values in $\boldsymbol{S}_k$ reflect the significance of the singular vectors in $\boldsymbol{U}_k$. They will also be utilized as prior weights for weighted sparse coding in our denoising algorithm.

B. Guided Internal Prior Learning

After the external PG prior model is learned from external natural clean images, we employ it to guide the internal PG prior learning for a given real-world noisy image. The guidance lies in two aspects. First, the external prior will guide the subspace clustering [52], [53] of internal noisy PGs. Second, the external prior will guide the orthogonal dictionary learning of internal noisy PGs.

1) Internal Subspace Clustering

Given a real-world noisy image $\mathbf{y}$, we extract N (over-lapped) local patches from it. Similar to the external prior learning stage, for the n-th ($n = 1, ..., N$) local patch we search its M most similar (by Euclidean distance)

patches around it to form a noisy PG, denoted by $\boldsymbol{Y}_n = \{\mathbf{y}_{n,1}, ..., \mathbf{y}_{n,M}\}$. Then the group mean of $\boldsymbol{Y}_n$, denoted by $\boldsymbol{\mu}_n$, is subtracted from each patch by $\overline{\mathbf{y}}_{n,m} \triangleq \mathbf{y}_{n,m} - \boldsymbol{\mu}_n$, leading to the mean subtracted noisy PG $\overline{\boldsymbol{Y}}_n \triangleq \{\overline{\mathbf{y}}_{n,m}\}_{m=1}^M$.

The external GMM prior models $\{\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_k)\}_{k=1}^K$ basically characterize the subspaces of natural high quality PGs. Therefore, we project each noisy PG $\overline{\boldsymbol{Y}}_n$ into the subspaces of $\{\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_k)\}_{k=1}^K$ and assign it to the most suitable subspace based on the posterior probability:

$$P(k|\overline{\boldsymbol{Y}}_n) = \frac{\prod_{m=1}^M \mathcal{N}(\overline{\mathbf{y}}_{n,m} | \mathbf{0}, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^K \prod_{m=1}^M \mathcal{N}(\overline{\mathbf{y}}_{n,m} | \mathbf{0}, \boldsymbol{\Sigma}_l)} \tag{3}$$

for $k = 1, ..., K$. Then $\overline{\boldsymbol{Y}}_n$ is assigned to the subspace with the maximum *a*-posteriori (MAP) probability $\max_k P(k|\overline{\boldsymbol{Y}}_n)$.

2) Guided Orthogonal Dictionary Learning

Assume that we have assigned all the internal noisy PGs $\{\overline{\boldsymbol{Y}}_n\}_{n=1}^N$ to their corresponding most suitable subspaces in $\{\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_k)\}_{k=1}^K$. For the k-th subspace, the noisy PGs assigned to it are $\{\overline{\boldsymbol{Y}}_{k_n}\}_{n=1}^{N_k}$, where $\overline{\boldsymbol{Y}}_{k_n} = [\overline{\mathbf{y}}_{k_n,1}, ..., \overline{\mathbf{y}}_{k_n,M}]$ and $\sum_{k=1}^K N_k = N$. We propose to learn an orthogonal dictionary $\boldsymbol{D}_k$ from each set of PGs $\overline{\boldsymbol{Y}}_{k_n}$ to characterize the internal PG prior with the guidance of the corresponding external orthogonal dictionary $\boldsymbol{U}_k$ (Eq. (2)). The reasons that we learn orthogonal dictionaries are two-fold. Firstly, the PGs $\{\overline{\boldsymbol{Y}}_{k_n}\}_{n=1}^{N_k}$ are in a subspace of the whole space of all PGs; therefore, there is no necessary to learn a redundant over-complete dictionary to characterize it, while an orthonormal dictionary has naturally zero *mutual incoherence* [54]. Secondly, the orthogonality of dictionary can make the patch encoding in the testing stage very efficient, leading to an efficient denoising algorithm (please refer to sub-section III-C for more details).

We let the orthogonal dictionary $\boldsymbol{D}_k$ be

$$\boldsymbol{D}_k \triangleq [\boldsymbol{D}_{k,\mathrm{E}} \ \boldsymbol{D}_{k,\mathrm{I}}] \in \mathbb{R}^{3p^2 \times 3p^2}, \tag{4}$$

where $\boldsymbol{D}_{k,\mathrm{E}} = \boldsymbol{U}_k(:,1:r) \in \mathbb{R}^{3p^2 \times r}$ is the external sub-dictionary and it includes the first r most important eigenvectors of $\boldsymbol{U}_k$, and the internal sub-dictionary $\boldsymbol{D}_{k,\mathrm{I}} \in \mathbb{R}^{3p^2 \times (3p^2 - r)}$ is to be adaptively learned from the noisy PGs $\{\overline{\boldsymbol{Y}}_{k_n}\}_{n=1}^{N_k}$. The rationale to design $\boldsymbol{D}_k$ as a hybrid dictionary is as follows. The external sub-dictionary $\boldsymbol{D}_{k,\mathrm{E}}$ is pre-trained from external clean data, and it represents the k-th latent subspace of natural images, which is helpful to reconstruct the common latent structures of images. However, $\boldsymbol{D}_{k,\mathrm{E}}$ is general to all images but not adaptive to the given noisy image. Some fine-scale details specific to the given image may not be well characterized by $\boldsymbol{D}_{k,\mathrm{E}}$. Therefore, we learn an internal sub-dictionary $\boldsymbol{D}_{k,\mathrm{I}}$ to supplement $\boldsymbol{D}_{k,\mathrm{E}}$. In other words, $\boldsymbol{D}_{k,\mathrm{I}}$ is to reveal the latent subspace adaptive to the input noisy image, which cannot be effectively represented by $\boldsymbol{D}_{k,\mathrm{E}}$.

For notation simplicity, in the following development we ignore the subspace index k for $\overline{\boldsymbol{Y}}_{k_n}$ and $\boldsymbol{D}_k$, etc. The learning of hybrid orthogonal dictionary $\boldsymbol{D}$ is performed under the following weighted sparse coding framework:

$$\min_{\boldsymbol{D}_\mathrm{I}, \{\boldsymbol{\alpha}_{n,m}\}} \sum_{n=1}^N \sum_{m=1}^M (\|\overline{\mathbf{y}}_{n,m} - \boldsymbol{D}\boldsymbol{\alpha}_{n,m}\|_2^2 + \sum_{j=1}^{3p^2} \lambda_j |\boldsymbol{\alpha}_{n,m,j}|)$$
$$\text{s.t.} \quad \boldsymbol{D} = [\boldsymbol{D}_\mathrm{E} \ \boldsymbol{D}_\mathrm{I}], \ \boldsymbol{D}^\top \boldsymbol{D} = \boldsymbol{I}, \tag{5}$$

where $\boldsymbol{I}$ is the $3p^2$ dimensional identity matrix, $\boldsymbol{\alpha}_{n,m}$ is the sparse coding vector of the m-th patch $\overline{\boldsymbol{y}}_{n,m}$ in the n-th PG $\overline{\boldsymbol{Y}}_n$ and $\boldsymbol{\alpha}_{n,m,j}$ is the j-th element of $\boldsymbol{\alpha}_{n,m}$. λ_j is the j-th regularization parameter defined as

$$\lambda_j = \lambda/(\sqrt{\boldsymbol{S}_k(j)} + \varepsilon), \tag{6}$$

where $\boldsymbol{S}_k(j)$ is the j-th singular value of diagonal singular value matrix $\boldsymbol{S}_k$ (please refer to Eq. (2)) and ε is a small positive number to avoid zero denominator. Note that $\boldsymbol{D}_{\mathrm{E}} = \boldsymbol{U}_k$ if $r = 3p^2$ and $\boldsymbol{D}_{\mathrm{E}} = \varnothing$ if $r = 0$.

In the dictionary learning model (5), we use the ℓ_2 norm to model the representation residual of PGs. This is because the patches in those PGs have similar content, and we assume that the noise therein will have similar statistics, which can be roughly modeled as locally Gaussian. On the other hand, this will make the dictionary learning much easier to solve. We employ an alternating iterative approach to solve the optimization problem (5). Specifically, we initialize the orthogonal dictionary as $\boldsymbol{D}^{(0)} = \boldsymbol{U}_k$ and for $t = 0, 1, ..., T-1$, and alternatively update $\boldsymbol{\alpha}_{n,m}$ and $\boldsymbol{D}_{\mathrm{I}}$ as follows.

Updating Sparse Coding Coefficients: Given the orthogonal dictionary $\boldsymbol{D}^{(t)}$, we update each sparse coding vector $\boldsymbol{\alpha}_{n,m}$ by solving

$$\boldsymbol{\alpha}_{n,m}^{(t+1)} := \arg\min_{\boldsymbol{\alpha}_{n,m}} \|\overline{\boldsymbol{y}}_{n,m} - \boldsymbol{D}^{(t)}\boldsymbol{\alpha}_{n,m}\|_2^2 + \sum_{j=1}^{3p^2} \lambda_j |\boldsymbol{\alpha}_{n,m,j}|. \tag{7}$$

Since dictionary $\boldsymbol{D}^{(t)}$ is orthogonal, the problems (7) has a closed-form solution

$$\boldsymbol{\alpha}_{n,m}^{(t+1)} = \mathrm{sgn}((\boldsymbol{D}^{(t)})^\top \overline{\boldsymbol{y}}_{n,m}) \odot \max(|(\boldsymbol{D}^{(t)})^\top \overline{\boldsymbol{y}}_{n,m}| - \boldsymbol{\lambda}, \boldsymbol{0}), \tag{8}$$

where $\boldsymbol{\lambda} = \frac{1}{2}[\lambda_1, \lambda_2, ..., \lambda_{3p^2}]^\top$ is the vector of regularization parameter, $\mathrm{sgn}(\bullet)$ is the sign function and $\odot$ means element-wise multiplication. The detailed derivation of Eq. (8) can be found in Appendix A.

Updating Internal Sub-dictionary: Given the sparse coding vectors $\{\boldsymbol{\alpha}_{n,m}^{(t+1)}\}$, we update the internal sub-dictionary by solving

$$\begin{aligned} \boldsymbol{D}_{\mathrm{I}}^{(t+1)} &:= \arg\min_{\boldsymbol{D}_{\mathrm{I}}} \sum_{n=1}^{N} \sum_{m=1}^{M} \|\overline{\boldsymbol{y}}_{n,m} - \boldsymbol{D}\boldsymbol{\alpha}_{n,m}^{(t+1)}\|_2^2 \\ &= \arg\min_{\boldsymbol{D}_{\mathrm{I}}} \|\overline{\boldsymbol{Y}}_n - \boldsymbol{D}\boldsymbol{A}^{(t+1)}\|_F^2 \end{aligned} \tag{9}$$

$$\text{s.t.} \quad \boldsymbol{D} = [\boldsymbol{D}_{\mathrm{E}} \ \boldsymbol{D}_{\mathrm{I}}], \ \boldsymbol{D}_{\mathrm{I}}^\top \boldsymbol{D}_{\mathrm{I}} = \boldsymbol{I}_{(3p^2-r)}, \ \boldsymbol{D}_{\mathrm{E}}^\top \boldsymbol{D}_{\mathrm{I}} = \boldsymbol{0},$$

where $\boldsymbol{A}^{(t+1)} = [\boldsymbol{\alpha}_{1,1}^{(t+1)}, ..., \boldsymbol{\alpha}_{1,M}^{(t)}, ..., \boldsymbol{\alpha}_{N,1}^{(t+1)}, ..., \boldsymbol{\alpha}_{N,M}^{(t+1)}]$ and $\boldsymbol{I}_{(3p^2-r)}$ is the $(3p^2 - r)$ dimensional identity matrix. The sparse coefficients matrix can be written as $\boldsymbol{A}^{(t+1)} = [(\boldsymbol{A}_{\mathrm{E}}^{(t+1)})^\top \ (\boldsymbol{A}_{\mathrm{I}}^{(t+1)})^\top]^\top$ where the external part $\boldsymbol{A}_{\mathrm{E}}^{(t+1)} \in \mathbb{R}^{r \times NM}$ and the internal part $\boldsymbol{A}_{\mathrm{I}}^{(t+1)} \in \mathbb{R}^{(3p^2-r) \times NM}$ represent the coding coefficients of $\boldsymbol{Y}$ over external sub-dictionary $\boldsymbol{D}_{\mathrm{E}}$ and internal sub-dictionary $\boldsymbol{D}_{\mathrm{I}}^{(t)}$, respectively. According to the following Theorem 1, by setting $\mathcal{Y} = \overline{\boldsymbol{Y}}_n - \boldsymbol{D}_{\mathrm{E}}\boldsymbol{A}_{\mathrm{E}}^{(t+1)}, \mathcal{E} = \boldsymbol{D}_{\mathrm{E}}, \mathcal{D} = \boldsymbol{D}_{\mathrm{I}}, \mathcal{A} = \boldsymbol{A}_{\mathrm{I}}$, the problem (9) has a closed-form solution $\boldsymbol{D}_{\mathrm{I}}^{(t+1)} = \boldsymbol{U}_{\mathrm{I}}\boldsymbol{V}_{\mathrm{I}}^\top$, where $\boldsymbol{U}_{\mathrm{I}} \in \mathbb{R}^{3p^2 \times (3p^2-r)}$ and $\boldsymbol{V}_{\mathrm{I}} \in \mathbb{R}^{(3p^2-r) \times (3p^2-r)}$ are the orthogonal matrices obtained by the following SVD [51]

Algorithm 1: External Prior Guided Internal Prior Learning

Input: Matrices $\overline{\boldsymbol{Y}}_n$, external sub-dictionary $\boldsymbol{D}_{\mathrm{E}}$, parameter vector $\boldsymbol{\lambda}$

Initialization: initialize $\boldsymbol{D}^{(0)} = \boldsymbol{U}_k$ by Eq. (2);

for $t = 0, 1, ..., T-1$ **do**

1. Update $\boldsymbol{\alpha}_{n,m}^{(t+1)}$ by Eq. (7);
2. Update $\boldsymbol{D}_{\mathrm{I}}^{(t+1)}$ by Eq. (9);

end for

Output: Internal orthogonal dictionary $\boldsymbol{D}_{\mathrm{I}}^{(T)}$ and sparse codes $\mathbf{A}^{(T)}$.

$$(\boldsymbol{I} - \boldsymbol{D}_{\mathrm{E}}\boldsymbol{D}_{\mathrm{E}}^\top)\boldsymbol{Y}(\boldsymbol{A}_{\mathrm{I}}^{(t+1)})^\top = \boldsymbol{U}_{\mathrm{I}}\boldsymbol{S}_{\mathrm{I}}\boldsymbol{V}_{\mathrm{I}}^\top. \tag{10}$$

The orthogonality of internal sub-dictionary $\boldsymbol{D}_{\mathrm{I}}^{(t+1)}$ can be checked by $(\boldsymbol{D}_{\mathrm{I}}^{(t+1)})^\top (\boldsymbol{D}_{\mathrm{I}}^{(t+1)}) = \boldsymbol{V}_{\mathrm{I}}\boldsymbol{U}_{\mathrm{I}}^\top \boldsymbol{U}_{\mathrm{I}}\boldsymbol{V}_{\mathrm{I}}^\top = \boldsymbol{I}_{(3p^2-r)}$. In fact, the Theorem 1 provides a sufficient and necessary condition to guarantee the existence of the closed-form solution for the internal sub-dictionary of the problem (9).

Theorem 1. *Let $\mathcal{A} \in \mathbb{R}^{(3p^2-r) \times M}$, $\mathcal{Y} \in \mathbb{R}^{3p^2 \times M}$ be two given data matrices. $\mathcal{E} \in \mathbb{R}^{3p^2 \times r}$ is a given matrix satisfying $\mathcal{E}^\top \mathcal{E} = \boldsymbol{I}_{r \times r}$, then $\hat{\mathcal{D}} = \mathcal{U}\mathcal{V}^\top$ is the necessary condition of*

$$\hat{\mathcal{D}} = \arg\min_{\mathcal{D}} \|\mathcal{Y} - \mathcal{D}\mathcal{A}\|_F^2$$

$$s.t. \quad \mathcal{D}^\top \mathcal{D} = \boldsymbol{I}_{(3p^2-r) \times (3p^2-r)}, \mathcal{E}^\top \mathcal{D} = \boldsymbol{0}_{r \times (3p^2-r)}, \tag{11}$$

where $\mathcal{U} \in \mathbb{R}^{3p^2 \times (3p^2-r)}$ and $\mathcal{V} \in \mathbb{R}^{(3p^2-r) \times (3p^2-r)}$ are the orthogonal matrices obtained by performing economy (a.k.a. reduced) SVD [51]:

$$(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top = \mathcal{U}\Sigma\mathcal{V}^\top \tag{12}$$

Besides, if $rank(\Sigma) = 3p^2 - r$, $\hat{\mathcal{D}} = \mathcal{U}\mathcal{V}^\top$ is also the sufficient condition of problem (11).

The proof of the Theorem 1 can be found in Appendix B. Though the problem (9) has a closed-form solution by SVD [51], the uniqueness of solution cannot be guaranteed since the matrices $(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top$ as well as $\mathcal{U}$ and $\mathcal{V}$ may be reduced to matrices of lower rank. Hence, we also analyze the uniqueness of the solution $\hat{\mathcal{D}}$ by the following Theorem 2, whose proof can be found in Appendix C.

Theorem 2. *(a) If $(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top \in \mathbb{R}^{3p^2 \times (3p^2-r)}$ is nonsingular, i.e., $rank(\Sigma) = 3p^2 - r$, then the solution of $\hat{\mathcal{D}} = \mathcal{U}\mathcal{V}^\top$ is unique; (b) If $(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top$ is singular, i.e., $0 \leq rank(\Sigma) < 3p^2 - r$, then the number of possible solutions of $\hat{\mathcal{D}}$ is $2^{3p^2-r-rank(\Sigma)}$ for fixed $\mathcal{U}$ and $\mathcal{V}$.*

The above alternative updating steps are repeated until the number of iterations exceeds a preset threshold. In each step, the energy value of the objective function (5) is decreased and we empirically found that the proposed model usually converges in 10 iterations. We summarize the procedures in Algorithm 1.

C. The Denoising Algorithm

The denoising of the given noisy image $\mathbf{y}$ can be simultaneously done with the guided internal sub-dictionary learning process. Once we obtain the solutions of sparse coding vectors $\{\hat{\boldsymbol{\alpha}}_{n,m}^{(T)}\}$ in Eq. (8) and the orthogonal dictionary $\boldsymbol{D}^{(T)} = [\boldsymbol{D}_{\mathrm{E}} \ \boldsymbol{D}_{\mathrm{I}}^{(T)}]$ in Eq. (9), the latent clean patch $\hat{\mathbf{y}}_{n,m}$ of the m-th noisy patch in PG $\boldsymbol{Y}_n$ is reconstructed as

---

Algorithm 2: External Prior Guided Internal Prior Learning for Real-World Noisy Image Denoising

---

Input: Noisy image $\mathbf{y}$, external PG prior GMM model
Initialization: $\hat{\mathbf{x}}^{(0)} = \mathbf{y}$;
for $Ite = 1 : IteNum$ **do**
  1. Extracting internal PGs $\{\mathbf{Y}_n\}_{n=1}^N$ from $\hat{\mathbf{x}}^{(Ite-1)}$;
  Guided Internal Subspace Clustering:
    for each PG $\mathbf{Y}_n$ **do**
  2.   Calculate group mean $\boldsymbol{\mu}_n$ and form mean subtracted PG $\overline{\mathbf{Y}}_n$;
  3.   Subspace clustering via Eq. (3);
    end for
  Guided Internal Orthogonal Dictionary Learning:
    for the PGs in each subspace **do**
  4.   External PG prior guided internal orthogonal dictionary learning by solving Eq. (5);
  5.   Recover each patch in all PGs via Eq. (13);
    end for
  6. Aggregate the recovered PGs of all subspaces to form the recovered image $\hat{\mathbf{x}}^{(Ite)}$;
end for
Output: The denoised image $\hat{\mathbf{x}}$.

$$\hat{\mathbf{y}}_{n,m} = \mathbf{D}^{(T)}\hat{\boldsymbol{\alpha}}_{n,m}^{(T)} + \boldsymbol{\mu}_n, \tag{13}$$

where $\boldsymbol{\mu}_n$ is the group mean of $\mathbf{Y}_n$. The latent clean image is then reconstructed by aggregating all the reconstructed patches in all PGs. We perform the above denoising procedures for several iterations for better denoising outputs. The proposed denoising algorithm is summarized in Algorithm 2.

IV. EXPERIMENTS

A. Implementation Details

The noise in real-world images is very complex due to the many factors such as sensors, lighting conditions and camera settings. It is difficult to evaluate one algorithm by tuning its parameters for all these different settings. In this work, we fix the parameters of our algorithm and apply it to all the testing datasets, though they were captured by different types of sensors and under different camera settings. The parameters of our method include the patch size p, the number of similar patches M in a patch group (PG), the window size W for PG searching, the number of Gaussian components K in GMM, the number of atoms r in the external sub-dictionaries, the sparse regularization parameter λ, the iteration numbers T for solving problem (5) and $IteNum$ for Alg. 2.

The performance of our proposed method varies little when we set patch size between $p = 6$ and $p = 9$, and we fix the patch size as $p = 6$ to save computational cost. The search window is fixed to $W = 31$ to balance computational cost and denoising accuracy of the proposed method. The number of patches in a patch group is set as $M = 10$, while using more patches will not bring clear benefits. We learn the external GMM prior with 3.6 million PGs extracted from the Kodak PhotoCD Dataset (http://r0k.us/graphics/kodak/), which includes 24 high quality color images. The number of Gaussians in GMM is set as $K = 32$, while using more Gaussians can only bring slightly better performance but cost more computational resources. The number of atoms in the external sub-dictionaries affects little the performance when it is set between $r = 27$ and $r = 81$, and we set it as $r = 54$ to make the external and internal sub-dictionaries have the same number of atoms. We set the number of iterations as $T = 2$ for solving the problem (5), while the number of iterations for Alg. 2 is set as $IteNum = 4$.
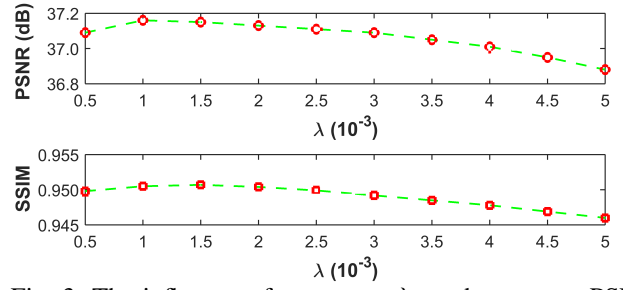

Fig. 3: The influence of parameter λ on the average PSNR (dB)/SSIM results of the proposed method on dataset [42].

One key parameter of our model is the regularization parameter λ. Fig. 3 plots the curves of PSNR/SSIM results w.r.t λ on the 15 cropped image in dataset [42]. One can see that our proposed method achieves good PSNR/SSIM performance within a certain range of λ. Similar observations can be made on other datasets. We fix $\lambda = 0.001$ in the paper, and it works well across the three datasets used in our experiments.

All the parameters of our method are fixed in all experiments, which are run under the Matlab2014b environment on a machine with Intel(R) Core(TM) i7-5930K CPU of 3.5GHz and 32GB RAM. We will release the code with the publication of this work.

B. The Testing Datasets

We evaluate the proposed method on three real-world noisy image datasets, where the images were captured under indoor or outdoor lighting conditions by different types of cameras and camera settings.

Dataset 1. The first dataset is provided in [42], which includes noisy images of 11 static scenes. The noisy images were collected under controlled indoor environment. Each scene was shot 500 times under the same camera and camera setting. The mean image of the 500 shots is roughly taken as the "ground truth", with which the PSNR and SSIM [55]can be computed.

Since the image size is very large (about 7000×5000) and the 11 scenes share repetitive contents, the authors of [42] cropped 15 smaller images (of size 512×512) to perform experiments. In order to evaluate the proposed methods more comprehensively, we cropped 60 images of size 500×500 from the dataset for experiments. Some samples are shown in Fig. 4. Note that our cropped 60 images and the 15 cropped images by the authors of [42] are from different shots.

Dataset 2 is called the Darmstadt Noise Dataset (DND) [56], which includes 50 different pairs of images of the same scenes captured by Sony A7R, Olympus E-M10, Sony RX100 IV, and Huawei Nexus 6P. The real-world noisy images are collected under higher ISO values with shorter exposure time, while the "ground truth" images are captured under lower ISO values with longer exposure times. Since the captured images are of megapixel-size, the authors cropped 20 bounding boxes of 512×512 pixels from each image in the dataset, yielding $50 \times 20 = 1000$ test crops in total. Some samples are shown in Fig. 5. Note that the "ground truth" images of this dataset have not been released yet, but one can submit the denoised

Fig. 4: Some sample images from the **Dataset 1** [42].

Fig. 6: Some sample images from our dataset (**Dataset 3**).

images to the Project Website and get the average PSNR (dB) and SSIM results.

Dataset 3. On one hand, the scenes of **Dataset 1** are mostly printed photos, and they cannot represent realistic objects and scenes with different reflectance properties. On the other hand, the **Dataset 2** contains repetitive contents in the 20 cropped images for each of the 50 scenes. To remedy the limitations of **Dataset 1** and **Dataset 2**, we construct another dataset which contains images of 10 different scenes captured by Canon 80D and Sony A7II cameras with more ISO settings and more comprehensive scenes. The ISO settings in our dataset are 800, 1600, 3200, 6400, 12800 while those of **Dataset 1** are 1600, 3200, 6400. Compared to **Dataset 2**, our new dataset is more comprehensive on scene contents. Similar to **Dataset 1**, each scene was captured 500 shots, and the mean image of these 500 shots can be used a kind of ground-truth to evaluate the denoising algorithms. Fig. 6 shows some cropped images of the scenes in our dataset. One can see that the images contain a lot of different realistic objects with varying colors, shapes, materials, etc.

Our dataset provides real-world noisy images of realistic objects with different ISO settings. It can be used to more fairly evaluate the performance of different real-world noisy image denoising methods. Consider that the image resolution is very high (about 4000×4000), for the convenience of experimental studies, we cropped 100 (10 for each scene) smaller images (of size 512×512) from it to perform experiments. The whole dataset will be made publically available with the publication of this paper.

the method of [5]. A GMM model (with $K = 32$ Gaussians) is directly learned from the PGs extracted from the given noisy image without using any external data, and then the internal orthogonal dictionaries are obtained via Eq. (2) to perform denoising. All parameters of the "External" and "Internal" methods are tuned to achieve their best performance.

We compare the three methods on the 60 cropped images from **Dataset 1** [42]. The average PSNR and run time are listed in Table I. The best results are highlighted in bold. It can be seen that "Guided Internal" method achieves better PSNR than both "External" and "Internal" methods. In addition, the "Internal" method is very slow because it involves online GMM learning, while the "Guided Internal" method is only a little slower than the "External" method. Figs. 7 and 8 show the denoised images of two noisy images by the three methods. One can see that the "External" method is good at recovering large-scale structures (see Fig. 7) while the "Internal" method is good at recovering fine-scale textures (see Fig. 8). By utilizing external priors to guide the internal prior learning, our proposed method can effectively recover both the large-scale structures and fine-scale textures.

TABLE I: Average PSNR (dB) and Run Time (seconds) of the "External", "Internal", and "Guided Internal" methods on 60 real-world noisy images (of size $500 \times 500 \times 3$) cropped from **Dataset 1** [42].

	Noisy	External	Internal	Guided Internal
PSNR	34.51	38.21	38.07	**38.75**
Time	—	**21.19**	312.67	22.26

Fig. 5: Some sample images from the **Dataset 2** [56].

C. Comparison among external, internal and guided internal priors

To demonstrate the advantages of external prior guided internal prior learning, we perform real-world noisy image denoising by using external priors only (denoted by "External"), internal priors only (denoted by "Internal"), and the proposed guided internal priors (denoted by "Guided Internal"), respectively. For the "External" method, we utilize the full external dictionaries (i.e., $r = 108$ in Eq. (5)) for denoising. For the "Internal" method, the overall framework is similar to
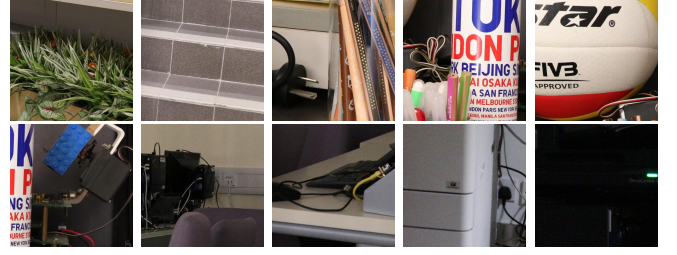
D. Comparison with State-of-the-Art Denoising Methods

Comparison methods. We compare the proposed method with state-of-the-art image denoising methods, including GAT-BM3D [30], CBM3D [7], WNNM [13], TID [19], MLP [20], DnCNN [22], CSF [24], TNRD [25], Noise Clinic (NC) [39], [40], Cross-Channel (CC) [42], and Neat Image (NI) [44]. Among these methods, GAT-BM3D [30] is a state-of-the-art Poisson noise reduction method. The method CBM3D [7] is a state-of-the-art method for color image denoising and the noise on color images is assumed to be additive white Gaussian. The methods of WNNM, MLP, DnCNN, CSF, and TNRD are state-of-the-art Gaussian noise removal methods for grayscale images, and we apply them to each channel of color images for denoising. NC is a blind image denoising method, and NI is a set of commercial software for image denoising, which has been embedded into Photoshop and Corel PaintShop. The code of CC is not released but its results on the 15 cropped images are available at [42]. Therefore, we only compare with it on the 15 cropped images in **Dataset 1** [42].

(a) Noisy [42]: 35.89dB   (b) External: 39.05dB   (c) Internal: 38.75dB   (d) Guided Internal: **39.39**dB   (e) Mean Image [42]

Fig. 7: Denoised images of a region cropped from the real-world noisy image "Nikon D600 ISO 3200 C1" [42] by different methods. The images are better to be zoomed-in on screen.

(a) Noisy [42]: 33.77dB   (b) External: 36.97dB   (c) Internal: 37.40dB   (d) Guided Internal: **38.01**dB   (e) Mean Image [42]
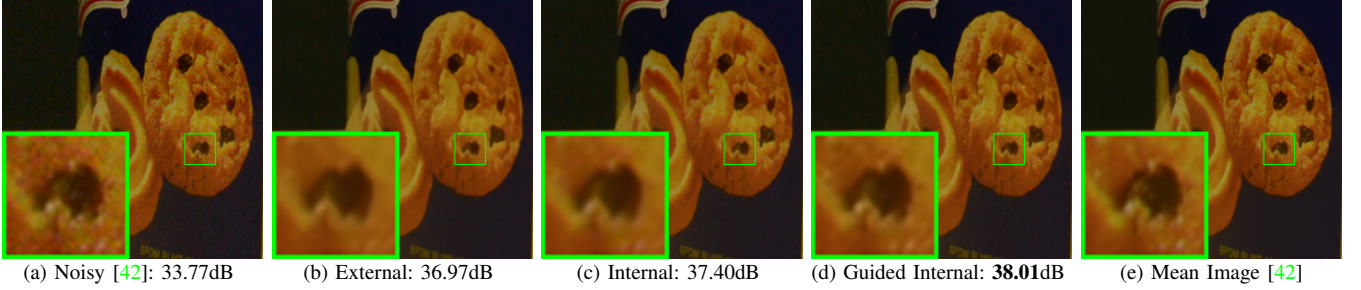
Fig. 8: Denoised images of a region cropped from the real-world noisy image "Nikon D600 ISO 3200 C1" [42] by different methods. The images are better to be zoomed-in on screen.

Noise level of comparison methods. For the CBM3D method, the standard deviation of noise on color images should be given as a parameter. For methods of WNNM, MLP, CSF, and TNRD, the noise level in each color channel should be input. For the DnCNN method, it is trained to deal with noise in a range of levels $0 \sim 55$. We retrain the models of discriminative denoising methods MLP, CSF, and TNRD (using the released codes by the authors) at different noise levels from $\sigma = 5$ to $\sigma = 50$ with a gap of 5. The denoising is performed by processing each channel with the model trained at the same (or nearest) noise level. The noise levels $(\sigma_r, \sigma_g, \sigma_b)$ in R, G ,B channels are assumed to be Gaussian and can be estimated via some noise estimation methods [57], [58]. In this paper, we employ the method [58] to estimate the noise level for each color channel.

Results on Dataset 1. As described in section 4.2, there is a mean image for each of the 11 scenes used in **Dataset 1** [42], and those mean images can be roughly taken as "ground truth" images for quantitative evaluation of denoising algorithms. We firstly perform quantitative comparison on the 15 cropped images used in [42]. The results on PSNR (dB) and speed (second) of GAT-BM3D, CBM3D, WNNM, TID, MLP, CSF, TNRD, DnCNN, NC, NI and CC are listed in Table II (The results of CC are copied from the original paper [42]). The best PSNR results of each image are highlighted in bold. One can see that on 8 out of the 15 images, our method achieves the best PSNR values. CC achieves the best PSNR on 3 of the 15 images. It should be noted that in the CC method, a specific model is trained for each camera and camera setting, while our method uses the same model for all images. On average, our proposed method has 0.27dB PSNR improvements over the second best method CC and much higher PSNR gains over other competing methods. The method GAT-BM3D does not work well on most images. This is because real world noise

is much more complex than Poisson.

Figs. 9 and 10 show the denoised images of one scene captured by Canon 5D Mark 3 at ISO = 3200 and Nikon D800 at ISO = 6400, respectively. We can see that GAT-BM3D, CBM3D, TID, DnCNN, NC, NI and CC would either remain noise or generate artifacts, while TNRD over-smooths much the image. By using the external prior guided internal priors, our proposed method preserves edges and textures better than other methods while removing the noise, leading to visually more pleasant outputs. Specifically, Fig. 10 is used to illustrate the denoising performance of our method on fine-scale textures such as hair, which is a very challenging task. Even the "ground truth" mean image cannot show very clear details of the hair. Though our method cannot reproduce clearly the details (e.g., the local direction of hair in some regions), it demonstrates the best visual results among the competing methods. More comparisons on visual quality and SSIM [55] index can be found in the supplementary file.

We then perform denoising experiments on the 60 images we cropped from [42]. The average PSNR results are listed in Table III (CC is not compared since the code is not available). Again, our proposed method achieves much better PSNR results than the other methods. The improvements of our method over the second best method (TNRD) are 0.43dB on PSNR. Fig. 11 shows the denoised images of one scene captured by Nikon D800 at ISO = 3200. We can see again that the proposed method obtain better visual quality than other competing methods. More comparisons on visual quality and SSIM can be found in the supplementary file.

Results on Dataset 2. In Table IV, we list the average PSNR (dB) results of the competing methods on the 1000 cropped images in the DND dataset [56]. We can see again that the proposed method achieves better performance than the other competing methods. Note that the "ground truth" images of

TABLE II: PSNR(dB) results and Speed (sec.) of different methods on 15 cropped real-world noisy images used in [42].

Setting	GAT-BM3D	CBM3D	WNNM	TID	MLP	CSF	TNRD	DnCNN	NI	NC	CC	Ours
Canon 5D ISO = 3200	31.23	39.76	37.51	37.22	39.00	35.68	39.51	37.26	37.68	38.76	38.37	**40.50**
	30.55	36.40	33.86	34.54	36.34	34.03	36.47	34.13	34.87	35.69	35.37	**37.05**
	27.74	36.37	31.43	34.25	36.33	32.63	**36.45**	34.09	34.77	35.54	34.91	36.11
Nikon D600 ISO = 3200	28.55	34.18	33.46	32.99	34.70	31.78	34.79	33.62	34.12	**35.57**	34.98	34.88
	32.01	35.07	36.09	34.20	36.20	35.16	36.37	34.48	35.36	**36.70**	35.95	36.31
	39.78	37.13	39.86	35.58	39.33	39.98	39.49	35.41	38.68	39.28	**41.15**	39.23
Nikon D800 ISO = 1600	32.24	36.81	36.35	34.94	37.95	34.84	38.11	35.79	37.34	38.01	37.99	**38.40**
	33.86	37.76	39.99	35.19	40.23	38.42	40.52	36.08	38.57	39.05	40.36	**40.92**
	33.90	37.51	37.15	35.26	37.94	35.79	38.17	35.48	37.87	38.20	38.30	**38.97**
Nikon D800 ISO = 3200	36.49	35.05	38.60	33.70	37.55	38.36	37.69	34.08	36.95	38.07	**39.01**	38.66
	32.91	34.07	36.04	31.04	35.91	35.53	35.90	33.70	35.09	35.72	36.75	**37.07**
	40.20	34.42	39.73	33.07	38.15	40.05	38.21	33.31	36.91	36.76	39.06	38.52
Nikon D800 ISO = 6400	29.84	31.13	33.29	29.40	32.69	34.08	32.81	29.83	31.28	33.49	**34.61**	33.76
	27.94	31.22	31.16	29.86	32.33	32.13	32.33	30.55	31.38	32.79	33.21	**33.43**
	29.15	30.97	31.98	29.21	32.29	31.52	32.29	30.09	31.40	32.86	33.22	**33.58**
Average	32.43	35.19	35.77	33.36	36.46	35.33	36.61	33.86	35.49	36.43	36.88	**37.15**
Time (s)	10.9	6.9	151.5	7353.2	16.8	19.3	5.1	79.2	**0.6**	15.3	NA	23.9

(a) Noisy [42]: 37.00dB   (b) CBM3D [7]: 39.76dB   (c) TID [19]: 37.22dB   (d) TNRD [25]: 39.51dB   (e) DnCNN [22]: 37.26dB

(f) NI [44]: 37.68dB   (g) NC [39], [40]: 38.76dB   (h) CC [42]: 38.37dB   (i) Ours: **40.50**dB   (j) Mean Image [42]
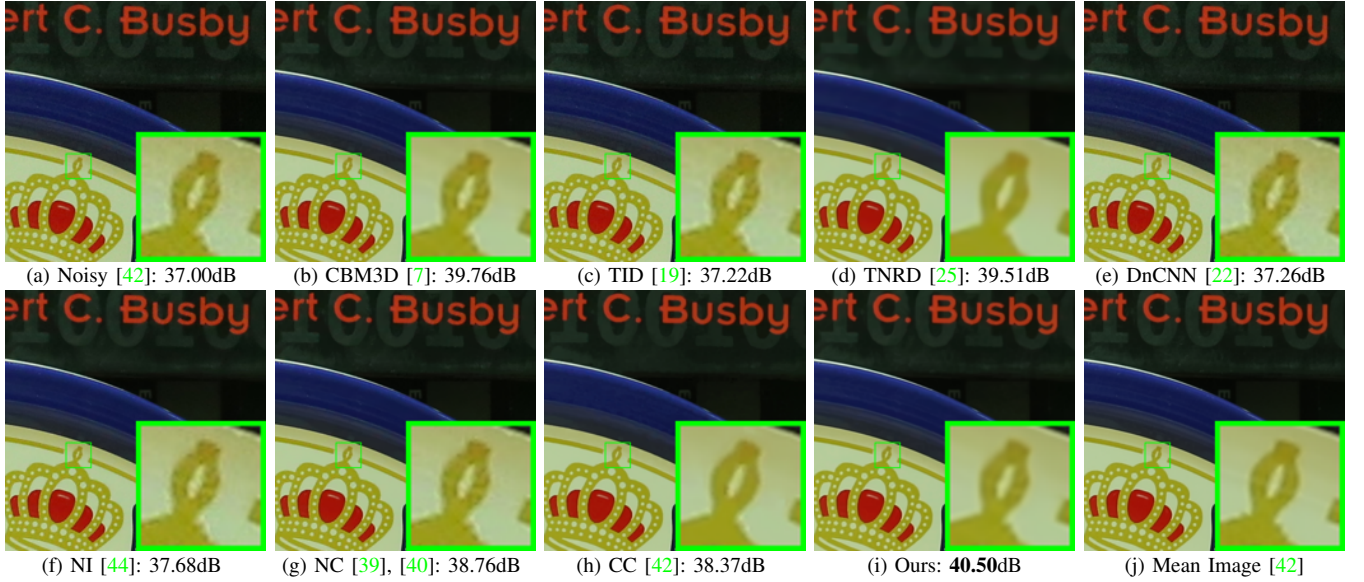
Fig. 9: Denoised images of a region cropped from the real-world noisy image "Canon 5D Mark 3 ISO 3200 1" [42] by different methods. The images are better to be zoomed-in on screen.
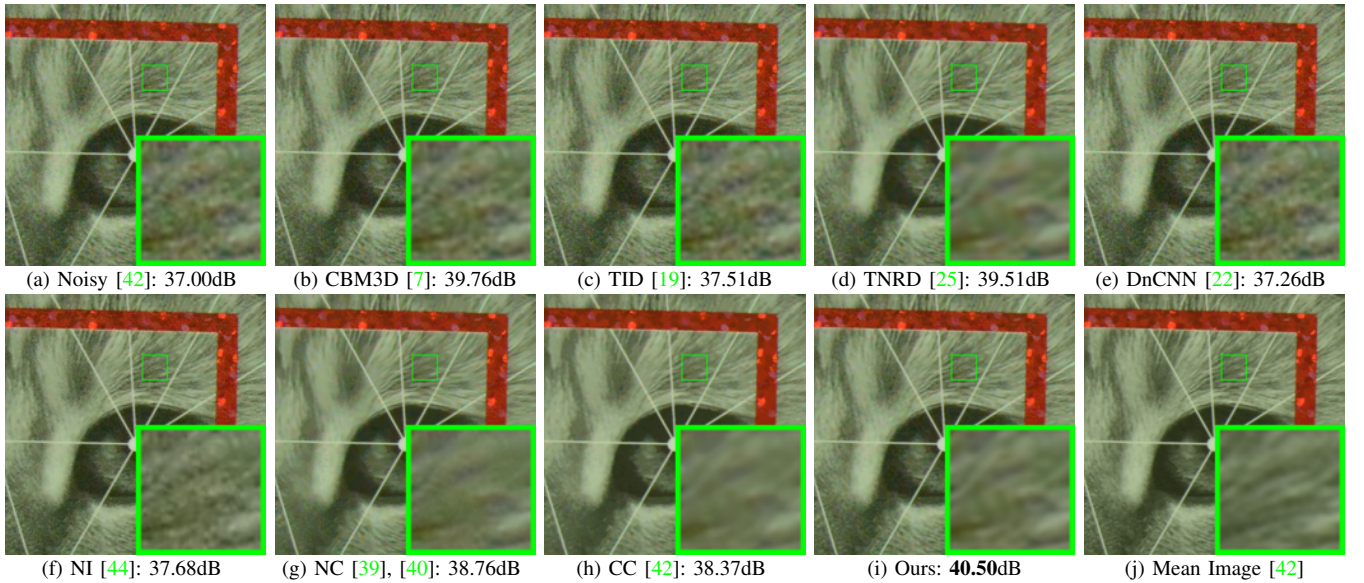

(a) Noisy [42]: 37.00dB   (b) CBM3D [7]: 39.76dB   (c) TID [19]: 37.51dB   (d) TNRD [25]: 39.51dB   (e) DnCNN [22]: 37.26dB

(f) NI [44]: 37.68dB   (g) NC [39], [40]: 38.76dB   (h) CC [42]: 38.37dB   (i) Ours: **40.50**dB   (j) Mean Image [42]

Fig. 10: Denoised images of a region cropped from the real-world noisy image "Nikon D800 ISO 6400 1" [42] by different methods. The images are better to be zoomed-in on screen.

Fig. 11: Denoised images of a region cropped from the real-world noisy image "Nikon D800 ISO 3200 A3" [42] by different methods. The images are better viewed by zooming in on screen.

(a) Noisy [42]: 33.60dB (b) CBM3D [7]: 35.23dB (c) WNNM [13]: 36.50dB (d) CSF [24]: 36.21dB (e) TNRD [25]: 37.10dB

(f) DnCNN [22]: 34.43dB (g) NI [44]: 35.02dB (h) NC [39], [40]: 36.07dB (i) Ours: **37.50**dB (j) Mean Image [42]

this dataset have not been released yet, so we are not able to calculate the PSNR and SSIM results for each noisy image in this dataset, nor compare with the "ground truth" mean image. However, one can submit the denoised images to the project website and get the average PSNR and SSIM results on the whole 1000 images. Fig. 12 shows the denoised images of a scene "0001_2" captured by a Nexus 6P phone [56]. The noise level in this image is relatively high. Hence, this image can be used to justify the performance of the proposed method on real-world noisy images with lower PSNR (around 20dB). One can see that the proposed method achieves visually more pleasing results than the other denoising methods. More comparisons on visual quality and SSIM can be found in the supplementary file.

Results on Dataset 3. Similar to **Dataset 1** [42], there is a "ground truth" image for each of the 10 scenes used in our constructed **Dataset 3**. We perform quantitative comparison on the 100 cropped images. The average PSNR results of competing methods are listed in Table IV. We can see that our proposed method achieves much better PSNR results than the other methods. The improvements of our method over the second best method (TNRD) is 0.16dB on PSNR. Fig. 13 shows the denoised images of one scene captured by Canon 80D at ISO = 12800. We can see again that the proposed method removes the noise while maintains better details (such as the vertical black shadow area) than other competing methods. More comparisons on visual quality and SSIM can be found in the supplementary file.

Comparison on speed. Efficiency is an important aspect to evaluate the efficiency of algorithms. We compare the speed of all competing methods except for CC. All experiments are run under the Matlab2014b environment on a machine with Intel(R) Core(TM) i7-5930K CPU of 3.5GHz and 32GB RAM. The average running time (second) of the compared methods on the 100 real-world noisy images is shown in Table V. The least average running time are highlighted in bold. One can easily see that the commercial software Neat Image (NI) is

the fastest method with highly optimized code. For a 512×512 image, NI costs about 0.6 second. The other methods cost from 5.2 (TNRD) to 152.2 (WNNM) seconds, while the proposed method costs about 24.1 seconds. It should be noted that GAT-BM3D, CBM3D, TNRD, and NC are implemented with compiled C++ mex-function and with parallelization, while WNNM, TID, MLP, CSF, DnCNN, and the proposed method are implemented purely in Matlab.

V. CONCLUSION

We proposed a new prior learning method for the real-world noisy image denoising problem by exploiting the useful information in both external and internal data. We first learned Gaussian Mixture Models (GMMs) from a set of clean external images as general image prior, and then employed the learned GMM model to guide the learning of adaptive internal prior from the given noisy image. Finally, a set of orthogonal dictionaries were output as the external-internal hybrid prior models for image denoising. Extensive experiments on three real-world noisy image datasets, including a new dataset constructed by us by different types of cameras and camera settings, demonstrated that our proposed method achieves much better performance than state-of-the-art image denoising methods in terms of both quantitative measure and visual perceptual quality.

APPENDIX A
CLOSED-FORM SOLUTION OF THE WEIGHTED SPARSE CODING PROBLEM (7)

For notation simplicity, we ignore the indices n, m, t in problem (7). It turns into the following weighted sparse coding problem:

$$\min_{\boldsymbol{\alpha}} \|\mathbf{y} - \boldsymbol{D\alpha}\|_2^2 + \sum_{j=1}^{3p^2} \lambda_j |\boldsymbol{\alpha}_j|. \quad (14)$$

Since $\boldsymbol{D}$ is an orthogonal matrix, problem (14) is equivalent to:

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{D}^T \mathbf{y} - \boldsymbol{\alpha}\|_2^2 + \sum_{j=1}^{3p^2} \lambda_j |\boldsymbol{\alpha}_j|. \quad (15)$$

(a) Noisy [40] (b) CBM3D [7] (c) WNNM [13] (d) MLP [20] (e) CSF [24]

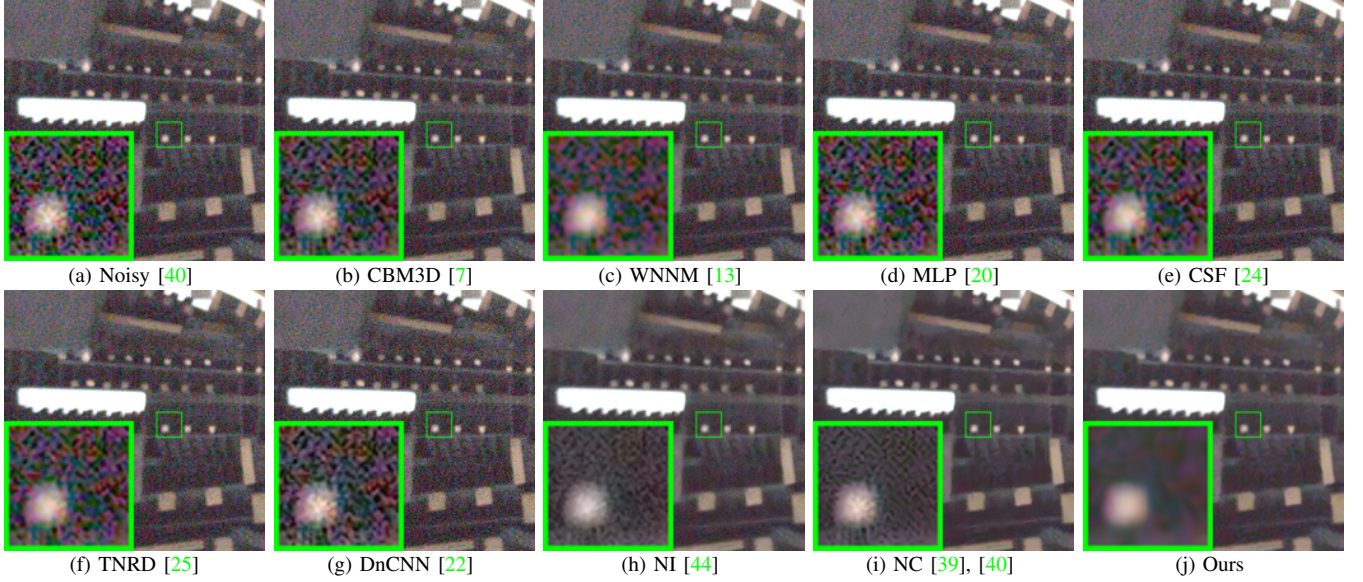(f) TNRD [25] (g) DnCNN [22] (h) NI [44] (i) NC [39], [40] (j) Ours

Fig. 12: Denoised images by different methods of the real-world noisy image "0001_2" captured by a Huawei Nexus 6P phone [56]. Note that the ground-truth clean image of the noisy input is not publicly released yet.

TABLE III: Average PSNR(dB) results of different methods on 60 real-world noisy images cropped from [42].

Methods	GAT-BM3D	CBM3D	WNNM	MLP	CSF	TNRD	DnCNN	NI	NC	Ours
PSNR	34.33	36.34	37.67	38.13	37.40	38.32	34.99	36.53	37.57	**38.75**

TABLE IV: Average PSNR(dB) results of different methods on the 1000 real-world noisy images from the DND dataset [56].

Methods	GAT-BM3D	CBM3D	WNNM	MLP	CSF	TNRD	DnCNN	NI	NC	Ours
PSNR	30.07	32.14	33.28	34.02	33.87	34.15	32.41	35.11	36.07	**36.41**

TABLE V: Average PSNR(dB) results of different methods on 100 real-world noisy images cropped from our new dataset.

Methods	GAT-BM3D	CBM3D	WNNM	MLP	CSF	TNRD	DnCNN	NI	NC	Ours
PSNR	33.54	37.14	35.18	37.34	37.07	37.48	34.74	35.70	36.76	**37.64**

(a) Noisy [42]: 36.51dB (b) CBM3D [7]: 37.91dB (c) WNNM [13]: 38.23dB (d) CSF [24]: 39.02dB (e) TNRD [25]: 39.26dB

(f) DnCNN [22]: 36.52dB (g) NI [44]: 37.52dB (h) NC [39], [40]: 37.53dB (i) Ours: **39.41**dB (j) Mean Image
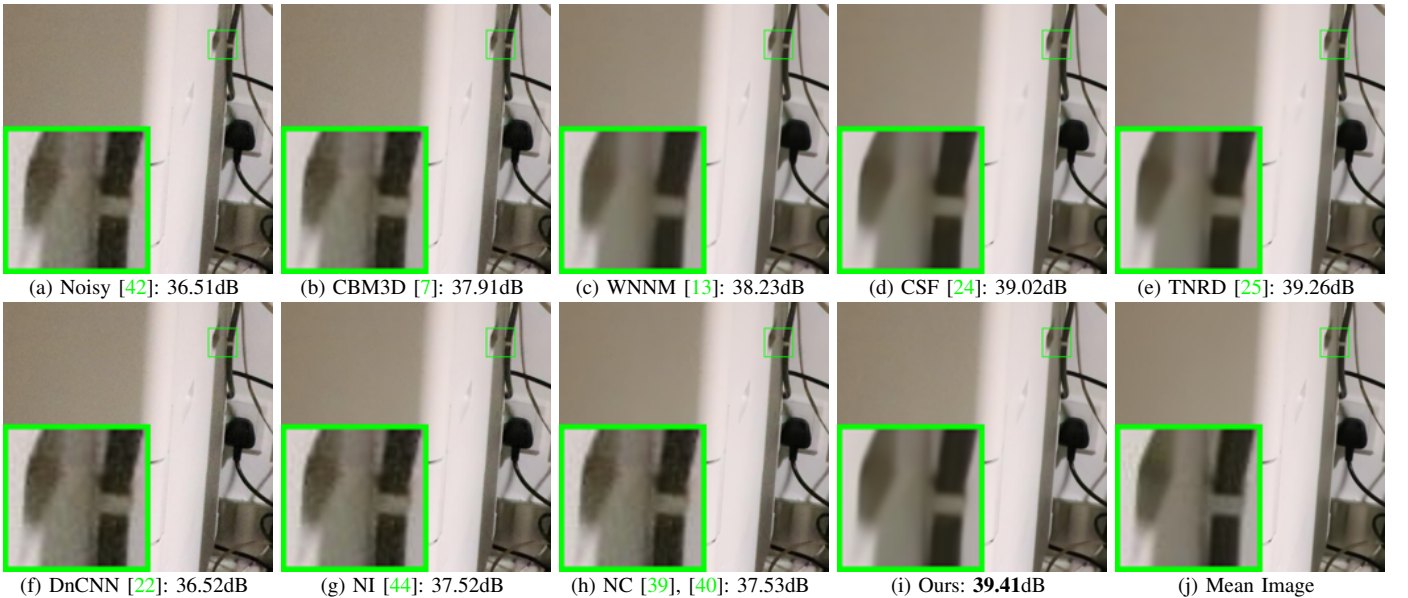
Fig. 13: Denoised images of a region cropped from the real-world noisy image "Canon 80D ISO 12800 IMG 2321" in our new dataset by different methods. The images are better viewed by zooming in on screen.

For simplicity, we denote $\mathbf{z} = \mathbf{D^T y}$. Here we have $\lambda_j > 0$, $j = 1, ..., 3p^2$, then problem (15) can be written as:

$$\min_{\boldsymbol{\alpha}} \sum_{j=1}^{3p^2} ((\mathbf{z}_j - \boldsymbol{\alpha}_j)^2 + \lambda_j |\boldsymbol{\alpha}_j|). \quad (16)$$

The problem (16) is separable w.r.t. each $\boldsymbol{\alpha}_j$ and hence can be simplified to $3p^2$ independent scalar minimization problems:

$$\min_{\boldsymbol{\alpha}_j} (\mathbf{z}_j - \boldsymbol{\alpha}_j)^2 + \lambda_j |\boldsymbol{\alpha}_j|, \quad (17)$$

where $j = 1, ..., 3p^2$. Taking derivative of $\boldsymbol{\alpha}_j$ in problem (17) and setting the derivative to be zero. There are two cases for the solution.

(a) If $\boldsymbol{\alpha}_j \geq 0$, we have $2(\boldsymbol{\alpha}_j - \mathbf{z}_j) + \lambda_j = 0$, and the solution

TABLE VI: Average Speed (sec.) results of different methods on 100 real-world noisy images cropped from our new dataset.

Methods	GAT-BM3D	CBM3D	WNNM	MLP	CSF	TNRD	DnCNN	NI	NC	Ours
Time	11.1	6.9	152.2	17.1	19.5	5.2	79.5	**0.6**	15.6	24.1

is $\hat{\boldsymbol{\alpha}}_j = \mathbf{z}_j - \frac{\lambda_j}{2} \geq 0$. So $\mathbf{z}_j \geq \frac{\lambda_j}{2} > 0$, and the solution $\hat{\boldsymbol{\alpha}}_j$ can be written as $\hat{\boldsymbol{\alpha}}_j = \text{sgn}(\mathbf{z}_j) * (|\mathbf{z}_j| - \frac{\lambda_j}{2})$, where $\text{sgn}(\bullet)$ is the sign function.

(b) If $\boldsymbol{\alpha}_j < 0$, we have $2(\boldsymbol{\alpha}_j - \mathbf{z}_j) - \lambda_j = 0$ and the solution is $\hat{\boldsymbol{\alpha}}_j = \mathbf{z}_j + \frac{\lambda_j}{2} < 0$. So $\mathbf{z}_j < -\frac{\lambda_j}{2} < 0$, and the solution $\hat{\boldsymbol{\alpha}}_j$ can be written as $\hat{\boldsymbol{\alpha}}_j = \text{sgn}(\mathbf{z}_j) * (-\mathbf{z}_j - \frac{\lambda_j}{2}) = \text{sgn}(\mathbf{z}_j) * (|\mathbf{z}_j| - \frac{\lambda_j}{2})$.

In summary, we have the final solution of the weighted sparse coding problem (14) as:

$$\hat{\boldsymbol{\alpha}} = \text{sgn}(\mathbf{D^T y}) \odot \max(|\mathbf{D^T y}| - \boldsymbol{\lambda}, 0), \qquad (18)$$

where $\boldsymbol{\lambda} = \frac{1}{2}[\lambda_1, \lambda_2, ..., \lambda_{3p^2}]^\top$ is the vector of regularization parameter and $\odot$ means element-wise multiplication.

APPENDIX B
PROOF OF THE THEOREM 1

Let $\mathcal{A} \in \mathbb{R}^{(3p^2-r) \times M}, \mathcal{Y} \in \mathbb{R}^{3p^2 \times M}$ be two given data matrices. Denote by $\mathcal{E} \in \mathbb{R}^{3p^2 \times r}$ the external subdictionary and $\mathcal{D} \in \mathbb{R}^{3p^2 \times (3p^2-r)}$ the internal subdictionary. For simplicity, we assume $3p^2 \geq M$. The problem in **Theorem 1** is as follows:

$$\hat{\mathcal{D}} = \arg\min_{\mathcal{D}} \|\mathcal{Y} - \mathcal{D}\mathcal{A}\|_F^2$$
$$\text{s.t.} \quad \mathcal{D}^\top \mathcal{D} = \boldsymbol{I}_{(3p^2-r) \times (3p^2-r)}, \mathcal{E}^\top \mathcal{D} = \boldsymbol{0}_{r \times (3p^2-r)}. \qquad (19)$$

Proof. We firstly prove the necessary condition. Since $\mathcal{D}^\top \mathcal{D} = \boldsymbol{I}_{(3p^2-r) \times (3p^2-r)}$, we have

$$\hat{\mathcal{D}} = \arg\min_{\mathcal{D}} \|\mathcal{Y} - \mathcal{D}\mathcal{A}\|_F^2 = \arg\max_{\mathcal{D}} \text{Tr}(\mathcal{A}\mathcal{Y}^\top \mathcal{D})$$
$$\text{s.t.} \quad \mathcal{D}^\top \mathcal{D} = \boldsymbol{I}_{(3p^2-r) \times (3p^2-r)}, \mathcal{E}^\top \mathcal{D} = \boldsymbol{0}_{r \times (3p^2-r)}. \qquad (20)$$

The Lagrange function is $\mathcal{L} = \text{Tr}(\mathcal{A}\mathcal{Y}^\top \mathcal{D}) - \text{Tr}(\Gamma_1(\mathcal{D}^\top \mathcal{D} - \boldsymbol{I}_{(3p^2-r) \times (3p^2-r)})) - \text{Tr}(\Gamma_2(\mathcal{D}^\top \mathcal{E}))$, where Γ_1 and Γ_2 are the Lagrange multipliers. Take the derivative of $\mathcal{L}$ w.r.t. $\mathcal{D}$ and set it to be matrix $\boldsymbol{0}$ of conformal dimensions, we can get

$$\partial \mathcal{L}/\partial \mathcal{D} = \mathcal{Y}\mathcal{A}^\top - \mathcal{D}(\Gamma_1 + \Gamma_1^\top) - \mathcal{E}\Gamma_2^\top = \boldsymbol{0}_{3p^2 \times (3p^2-r)}. \qquad (21)$$

Since $\mathcal{D}^\top \mathcal{D} = \boldsymbol{I}_{(3p^2-r) \times (3p^2-r)}$ and $\mathcal{E}^\top \mathcal{D} = \boldsymbol{0}_{3p^2 \times (3p^2-r)}$, by left multiplying both sides of the Eq. (22) by $\mathcal{E}^\top$, we have

$$\mathcal{E}^\top \mathcal{Y}\mathcal{A}^\top = \Gamma_2^\top. \qquad (22)$$

Put the Eq. (22) back into Eq. (21), we have

$$(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top = \mathcal{D}(\Gamma_1 + \Gamma_1^\top). \qquad (23)$$

Right multiplying both sides of Eq. (23) by $\mathcal{D}^\top$, we have

$$(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top \mathcal{D}^\top = \mathcal{D}(\Gamma_1 + \Gamma_1^\top)\mathcal{D}^\top. \qquad (24)$$

This shows that $(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top \mathcal{D}^\top$ is a symmetric matrix of order $3p^2 \times 3p^2$. Then we perform economy (or reduced) singular value decomposition (SVD) [51] on $(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top = \mathcal{U}\Sigma\mathcal{V}^\top$, there is

$$(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top \mathcal{D}^\top = \mathcal{U}\Sigma\mathcal{V}^\top \mathcal{D}^\top = \mathcal{D}\mathcal{V}\Sigma\mathcal{U}^\top. \qquad (25)$$

Hence, we have $\mathcal{U} = \mathcal{D}\mathcal{V}$, or equivalently $\hat{\mathcal{D}} = \mathcal{U}\mathcal{V}^\top$. The necessary condition is proved.

Now we prove the sufficient condition. If $\hat{\mathcal{D}} = \mathcal{U}\mathcal{V}^\top$, then $\hat{\mathcal{D}}^\top \hat{\mathcal{D}} = \boldsymbol{I}_{(3p^2-r) \times (3p^2-r)}$. To prove $\mathcal{E}^\top \hat{\mathcal{D}} = \boldsymbol{0}_{3p^2 \times (3p^2-r)}$, we left multiply both sides of Eq. (25) by $\mathcal{E}^\top$ and have $\boldsymbol{0}_{3p^2 \times (3p^2-r)} = \mathcal{E}^\top(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top \hat{\mathcal{D}}^\top = \mathcal{E}^\top \mathcal{U}\Sigma\mathcal{V}^\top \hat{\mathcal{D}}^\top = \mathcal{E}^\top \mathcal{U}\Sigma\mathcal{U}^\top$. It means that $\mathcal{E}^\top \mathcal{U}\Sigma\mathcal{U}^\top = \boldsymbol{0}_{3p^2 \times 3p^2}$. This only happens when $\mathcal{E}^\top \mathcal{U} = \boldsymbol{0}_{3p^2 \times (3p^2-r)}$ since $\text{rank}(\Sigma) = 3p^2 - r$ and $\mathcal{U}\Sigma\mathcal{U}^\top$ is positive definite. Then $\mathcal{E}^\top \hat{\mathcal{D}} = \mathcal{E}^\top \mathcal{U}\mathcal{V}^\top = \boldsymbol{0}_{3p^2 \times (3p^2-r)}$.

Finally we prove that $\hat{\mathcal{D}} = \mathcal{U}\mathcal{V}^\top$ is the solution of

$$\hat{\mathcal{D}} = \arg\min_{\mathcal{D}} \|\mathcal{Y} - \mathcal{D}\mathcal{A}\|_F^2 = \arg\max_{\mathcal{D}} \text{Tr}(\mathcal{Y}^\top \mathcal{D}\mathcal{A}). \qquad (26)$$

Note that by cyclic perturbation which retains the trace unchanged and due to $\mathcal{E}^\top \hat{\mathcal{D}} = \boldsymbol{0}_{3p^2 \times (3p^2-r)}$, we have $\text{Tr}(\mathcal{Y}^\top \hat{\mathcal{D}}\mathcal{A}) = \text{Tr}(\mathcal{Y}\mathcal{A}^\top \hat{\mathcal{D}}^\top) = \text{Tr}((\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top \hat{\mathcal{D}}^\top) = \text{Tr}(\mathcal{U}\Sigma\mathcal{V}^\top \mathcal{V}\mathcal{U}^\top) = \text{Tr}(\Sigma)$. For every $\mathcal{D}$ satisfying that $\mathcal{D}^\top \mathcal{D} = \boldsymbol{I}_{(3p^2-r) \times (3p^2-r)}$, $\mathcal{E}^\top \mathcal{D} = \boldsymbol{0}_{3p^2 \times (3p^2-r)}$, we have $\text{Tr}(\mathcal{Y}^\top \mathcal{D}\mathcal{A}) = \text{Tr}((\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top \mathcal{D}^\top) = \text{Tr}(\mathcal{U}\Sigma\mathcal{V}^\top \mathcal{D}^\top) = \text{Tr}(\Sigma\mathcal{V}^\top \mathcal{D}^\top \mathcal{U})$. By using a generalization version [59] of the Kristof's Theorem [60], we have $\text{Tr}(\mathcal{Y}^\top \mathcal{D}\mathcal{A}) = \text{Tr}(\Sigma\mathcal{V}^\top \mathcal{D}^\top \mathcal{U}) \leq \text{Tr}(\Sigma)$. The equality is obtained at $\mathcal{V}^\top \mathcal{D}^\top \mathcal{U} = \boldsymbol{I}_{(3p^2-r) \times (3p^2-r)}$, i.e., $\mathcal{D} = \mathcal{U}\mathcal{V}^\top = \hat{\mathcal{D}}$. This completes the proof. $\square$

APPENDIX C
PROOF OF THE THEOREM 2

Before we prove the Theorem 2, we need firstly prove the following Lemma 1.

Lemma 1: Let $\mathcal{E} \in \mathbb{R}^{3p^2 \times r}$ be an orthogonal matrix with $\mathcal{E}^\top \mathcal{E} = \boldsymbol{I}_{r \times r}$, then $\text{rank}(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top) \geq 3p^2 - r$.

Proof. Since $\text{rank}(\mathcal{E}\mathcal{E}^\top) \leq \min\{\text{rank}(\mathcal{E}), \text{rank}(\mathcal{E}^\top)\} = r$ and $\text{rank}(\mathcal{E}\mathcal{E}^\top) \geq \text{rank}(\mathcal{E}) + \text{rank}(\mathcal{E}^\top) - r = 2r - r = r$ by Sylvester's inequality, we have $\text{rank}(\mathcal{E}\mathcal{E}^\top) = r$. Then, $\text{rank}(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top) \geq \text{rank}(\boldsymbol{I}_{3p^2 \times 3p^2}) - \text{rank}(\mathcal{E}\mathcal{E}^\top) \geq 3p^2 - r$. $\square$

The $\text{rank}(\Sigma)$ (Σ is defined in Theorem 1) depends on $\text{rank}(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)$, $\text{rank}(\mathcal{Y})$ and $\text{rank}(\mathcal{A})$. Note that $\text{rank}(\mathcal{Y}) \geq M$ and $\text{rank}(\mathcal{A}) \geq \min\{3p^2, M\}$ and $\text{rank}(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top) \geq 3p^2 - r$. Hence, $\text{rank}(\Sigma) \leq \min\{3p^2 - r, M\}$.

Now we prove the Theorem 2:

Proof. a) If $(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top \in \mathbb{R}^{3p^2 \times (3p^2-r)}$ is nonsingular, i.e., $\text{rank}(\Sigma) = 3p^2 - r$, Σ may have distinct or multiple non-zero singular values. In the SVD [51] of $(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top = \mathcal{U}\Sigma\mathcal{V}^\top$, the singular vectors in $\mathcal{U}$ and $\mathcal{V}$ can be determined up to orientation. Hence, we can reformulate it as

$$(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top = \mathcal{U}^* \mathcal{K}_u \Sigma \mathcal{K}_v (\mathcal{V}^*)^\top, \qquad (27)$$

where $\mathcal{U}^* \in \mathbb{R}^{3p^2 \times (3p^2-r)}$ and $\mathcal{V}^* \in \mathbb{R}^{(3p^2-r) \times (3p^2-r)}$ are arbitrarily oriented singular vectors of $\mathcal{U}$ and $\mathcal{V}$, respectively. The $\mathcal{K}_u$ and $\mathcal{K}_v$ are diagonal matrices with $+1$ or -1 as diagonal elements in arbitrary distribution. $\Sigma \in \mathbb{R}^{(3p^2-r) \times (3p^2-r)}$ is a diagonal matrix with singular values in non-increasing order, i.e., $\Sigma_{11} \geq \Sigma_{22} \geq ... \geq \Sigma_{(3p^2-r)(3p^2-r)} \geq 0$. If we fix $\mathcal{K}_u$, then $\mathcal{K}_v$ is uniquely determined to meet the above requirements of Σ. If the orientations of the singular vectors of $\mathcal{U}^*$ are fixed, then $\mathcal{U} = \mathcal{U}^* \mathcal{K}_u$ is determined, so do the orientations of the singular vectors of $\mathcal{V}^*$ and $\mathcal{V}^\top = \mathcal{K}_v (\mathcal{V}^*)^\top$. In this case, the solution of $\hat{\mathcal{D}} = \mathcal{U}\mathcal{V}^\top = \mathcal{U}^* \mathcal{K}_u \mathcal{K}_v (\mathcal{V}^*)^\top$ is unique. When Σ has multiple singular values, the unique solution of $\hat{\mathcal{D}}$ can be proved in a similar way.

b) If $(\boldsymbol{I}_{3p^2 \times 3p^2} - \mathcal{E}\mathcal{E}^\top)\mathcal{Y}\mathcal{A}^\top$ is singular, i.e., $0 \leq \text{rank}(\Sigma) < 3p^2 - r$, and Σ has $3p^2 - r - \text{rank}(\Sigma)$ (at least one) zero singular values. The discussion in a) can still be applied to the singular vectors corresponding to the nonzero singular values, and the production of these singular vectors in $\mathcal{U}$ and $\mathcal{V}$ is still unique. However, the singular vectors corresponding to the zero singular values could be in arbitrary orientations as long as they satisfy the conditions of $\mathcal{U}^\top \mathcal{U} = \mathcal{V}^\top \mathcal{V} = \mathcal{V}\mathcal{V}^\top = \boldsymbol{I}_{(3p^2-r) \times (3p^2-r)}$. Since $\mathcal{U} \in \mathbb{R}^{3p^2 \times (3p^2-r)}$, $\mathcal{U}\mathcal{U}^\top$ no longer equals to the identity matrix of order $3p^2 \times 3p^2$. From Eq. (25), we have

$$\mathcal{U}\Sigma\mathcal{V}^\top \mathcal{D}^\top = \mathcal{D}\mathcal{V}\Sigma\mathcal{U}^\top \qquad (28)$$

Right multiplying both sides of Eq. (28) by $\mathcal{D}\mathcal{V}$ and left multiplying each side by $\mathcal{U}^\top$, we have

$$\Sigma = \mathcal{U}^\top \mathcal{D}\mathcal{V}\Sigma\mathcal{U}^\top \mathcal{D}\mathcal{V} \qquad (29)$$

Hence, $\Delta = \mathcal{U}^\top \mathcal{D}\mathcal{V} \in \mathbb{R}^{(3p^2-r) \times (3p^2-r)}$ is a diagonal matrix, the diagonal elements of which are

$$\Delta_{ii} = \begin{cases} 1 & \text{if } 1 \leq i \leq \text{rank}(\Sigma); \\ \pm 1 & \text{if } \text{rank}(\Sigma) < i \leq 3p^2 - r. \end{cases}$$

Thus, we have $\mathcal{D} = \mathcal{U}\Delta\mathcal{V}^\top$. That is, if $\text{rank}(\Sigma) < 3p^2 - r$, once we get the solution of $\hat{\mathcal{D}} = \mathcal{U}\mathcal{V}^\top$ in problem (19), $\mathcal{D} = \mathcal{U}\Delta\mathcal{V}^\top$ with suitable Δ is also the solution of problem (19). In fact, the number of solutions $\hat{\mathcal{D}}$ for problem (19) is $2^{3p^2-r-\text{rank}(\Sigma)}$ given fixed $\mathcal{U}$ and $\mathcal{V}$. $\qquad \square$

REFERENCES

[1] S. G. Chang, B. Yu, and M. Vetterli. Adaptive wavelet thresholding for image denoising and compression. *IEEE Transactions on Image Processing*, 9(9):1532–1546, 2000. 1, 3

[2] J. L. Starck, E. J. Candès, and D. L. Donoho. The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11(6):670–684, 2002. 1, 3

[3] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006. 1, 2, 3

[4] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. *IEEE International Conference on Computer Vision (ICCV)*, pages 2272–2279, 2009. 1, 2, 3

[5] W. Dong, L. Zhang, G. Shi, and X. Li. Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing*, 22(4):1620–1630, 2013. 1, 2, 3, 7

[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007. 1, 3

[7] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space. *IEEE International Conference on Image Processing (ICIP)*, pages 313–316, 2007. 1, 2, 3, 7, 9, 10, 11

[8] Mingyuan Zhou, Haojun Chen, John Paisley, Lu Ren, Lingbo Li, Zhengming Xing, David Dunson, Guillermo Sapiro, and Lawrence Carin. Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images. *IEEE Transactions on Image Processing*, 21(1):130–144, 2012. 1, 2, 3

[9] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. *IEEE International Conference on Computer Vision (ICCV)*, pages 839–846, 1998. 1, 3

[10] J. Portilla, V. Strela, M.J. Wainwright, and E.P. Simoncelli. Image denoising using scale mixtures of Gaussians in the wavelet domain. *Image Processing, IEEE Transactions on*, 12(11):1338–1351, 2003. 1, 2, 3

[11] A. Buades, B. Coll, and J. M. Morel. A non-local algorithm for image denoising. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 60–65, 2005. 1, 3

[12] M. Lebrun, A. Buades, and J. M. Morel. A nonlocal Bayesian image denoising algorithm. *SIAM Journal on Imaging Sciences*, 6(3):1665–1688, 2013. 1, 3

[13] S. Gu, L. Zhang, W. Zuo, and X. Feng. Weighted nuclear norm minimization with application to image denoising. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2862–2869, 2014. 1, 2, 3, 7, 10, 11

[14] J. Xu, L. Zhang, W. Zuo, D. Zhang, and X. Feng. Patch group based nonlocal self-similarity prior learning for image denoising. *IEEE International Conference on Computer Vision (ICCV)*, pages 244–252, 2015. 1, 2, 3, 4

[15] S. Roth and M. J. Black. Fields of experts. *International Journal of Computer Vision*, 82(2):205–229, 2009. 1, 2, 3

[16] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. *IEEE International Conference on Computer Vision (ICCV)*, pages 479–486, 2011. 1, 2, 3, 4

[17] I. Mosseri, M. Zontak, and M. Irani. Combining the power of internal and external denoising. *IEEE International Conference on Computational Photography (ICCP)*, pages 1–9, 2013. 1, 2

[18] Fei Chen, Lei Zhang, and Huimin Yu. External patch prior guided internal clustering for image denoising. *Proceedings of the IEEE international conference on computer vision*, pages 603–611, 2015. 1, 2, 3

[19] Enming Luo, Stanley H Chan, and Truong Q Nguyen. Adaptive image denoising by targeted databases. *IEEE Transactions on Image Processing*, 24(7):2167–2181, 2015. 1, 2, 3, 7, 9

[20] H. C. Burger, C. J. Schuler, and S. Harmeling. Image denoising: Can plain neural networks compete with BM3D? *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2392–2399, 2012. 1, 2, 3, 7, 11

[21] Junyuan Xie, Linli Xu, and Enhong Chen. Image denoising and inpainting with deep neural networks. *Advances in Neural Information Processing Systems*, pages 341–349, 2012. 1, 2, 3

[22] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a Gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 2017. 1, 2, 3, 7, 9, 10, 11

[23] Adrian Barbu. Training an active random field for real-time image denoising. *IEEE Transactions on Image Processing*, 18(11):2451–2462, 2009. 1, 2, 3

[24] U. Schmidt and S. Roth. Shrinkage fields for effective image restoration. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2774–2781, June 2014. 1, 2, 3, 7, 10, 11

[25] Y. Chen, W. Yu, and T. Pock. On learning optimized reaction diffusion processes for effective image restoration. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5261–5269, 2015. 1, 2, 3, 7, 9, 10, 11

[26] B. Zhang, J. M. Fadili, and J. L. Starck. Wavelets, ridgelets, and curvelets for poisson noise removal. *IEEE Transactions on Image Processing*, 17(7):1093–1108, 2008. 1

[27] Joseph Salmon, Zachary Harmany, Charles-Alban Deledalle, and Rebecca Willett. Poisson noise reduction with non-local pca. *Journal of mathematical imaging and vision*, 48(2):279–294, 2014. 1

[28] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian. Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, 2008. 1

[29] F. Luisier, T. Blu, and M. Unser. Image denoising in mixed Poisson-Gaussian noise. *IEEE Transactions on Image Processing*, 20(3):696–708, 2011. 1

[30] M. Makitalo and A. Foi. Optimal inversion of the generalized anscombe transformation for poisson-gaussian noise. *IEEE Transactions on Image Processing*, 22(1):91–103, 2013. 1, 7

[31] Y. Le Montagner, E. D. Angelini, and J. C. Olivo-Marin. An unbiased risk estimator for image denoising in the presence of mixed poisson-gaussian noise. *IEEE Transactions on Image Processing*, 23(3):1255–1268, 2014. 1

[32] Jielin Jiang, Lei Zhang, and Jian Yang. Mixed noise removal by weighted encoding with sparse nonlocal regularization. *IEEE transactions on image processing*, 23(6):2651–2662, 2014. 1

[33] Haijuan Hu, Bing Li, and Quansheng Liu. Removing mixture of gaussian and impulse noise by patch-based weighted means. *Journal of Scientific Computing*, 67(1):103–129, 2016. 1

[34] Jun Xu, Dongwei Ren, Lei Zhang, and David Zhang. Patch group based bayesian learning for blind image denoising. *Asian Conference on Computer Vision (ACCV) New Trends in Image Restoration and Enhancement Workshop*, pages 79–95, 2016. 1

[35] J. Portilla. Full blind denoising through noise covariance estimation using Gaussian scale mixtures in the wavelet domain. *IEEE International Conference on Image Processing (ICIP)*, 2:1217–1220, 2004. 1, 2, 3

[36] T. Rabie. Robust estimation approach for blind denoising. *IEEE Transactions on Image Processing*, 14(11):1755–1765, 2005. 1, 2, 3

[37] C. Liu, R. Szeliski, S. Bing Kang, C. L. Zitnick, and W. T. Freeman. Automatic estimation and removal of noise from a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):299–314, 2008. 1, 2, 3

[38] Z. Gong, Z. Shen, and K.-C. Toh. Image restoration with mixed or unknown noises. *Multiscale Modeling & Simulation*, 12(2):458–487, 2014. 1, 2, 3

[39] M. Lebrun, M. Colom, and J.-M. Morel. Multiscale image blind denoising. *IEEE Transactions on Image Processing*, 24(10):3149–3161, 2015. 1, 2, 3, 7, 9, 10, 11

[40] M. Lebrun, M. Colom, and J. M. Morel. The noise clinic: a blind image denoising algorithm. http://www.ipol.im/pub/art/2015/125/. Accessed 01 28, 2015. 1, 2, 3, 7, 9, 10, 11

[41] F. Zhu, G. Chen, and P.-A. Heng. From noise modeling to blind image denoising. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1, 2, 3

[42] S. Nam, Y. Hwang, Y. Matsushita, and S. J. Kim. A holistic approach to cross-channel image noise modeling and its application to image denoising. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1683–1691, 2016. 1, 2, 3, 6, 7, 8, 9, 10, 11

[43] Jun Xu, Lei Zhang, David Zhang, and Xiangchu Feng. Multi-channel weighted nuclear norm minimization for real color image denoising. In *ICCV*, pages 1096–1104, 2017. 1

[44] Neatlab ABSoft. Neat Image. https://ni.neatvideo.com/home. 1, 2, 3, 7, 9, 10, 11

[45] G. E Healey and R. Kondepudy. Radiometric CCD camera calibration and noise estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(3):267–276, 1994. 1, 2, 3

[46] Y. Tsin, Visvanathan Ramesh, and Takeo Kanade. Statistical calibration of ccd imaging process. *IEEE International Conference on Computer Vision (ICCV)*, 1:480–487, 2001. 2

[47] S. J. Kim, H. T. Lin, Z. Lu, S. Ssstrunk, S. Lin, and M. S. Brown. A new in-camera imaging model for color computer vision and its application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(12):2289–2302, 2012. 2

[48] H. C. Karaimer and M. S. Brown. A software platform for manipulating the camera imaging pipeline. *European Conference on Computer Vision (ECCV)*, October 2016. 2

[49] C. Kervrann, J. Boulanger, and P. Coupé. Bayesian non-local means filter, image redundancy and adaptive dictionaries for noise removal. *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 520–532, 2007. 3

[50] D. Ren, W. Zuo, D. Zhang, J. Xu, and L. Zhang. Partial deconvolution with inaccurate blur kernel. *IEEE Transactions on Image Processing*, 27(1):511–524, 2018. 4

[51] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936. 4, 5, 12

[52] R. Vidal. A tutorial on subspace clustering. *IEEE Signal Processing Magazine*, 2011. 4

[53] J. Xu, K. Xu, K. Chen, and J. Ruan. Reweighted sparse subspace clustering. *Computer Vision and Image Understanding*, 138:25–37, 2015. 4

[54] D. L. Donoho and X. Huo. Uncertainty principles and ideal atomic decomposition. *IEEE Transactions on Information Theory*, 47(7):2845–2862, 2001. 4

[55] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 6, 8

[56] T. Plötz and S. Roth. Benchmarking denoising algorithms with real photographs. In *CVPR*, 2017. 6, 7, 8, 10, 11

[57] X. Liu, M. Tanaka, and M. Okutomi. Single-image noise level estimation for blind denoising. *IEEE transactions on Image Processing*, 22(12):5226–5237, 2013. 8

[58] G. Chen, F. Zhu, and A. H. Pheng. An efficient statistical method for image noise level estimation. *IEEE International Conference on Computer Vision (ICCV)*, December 2015. 8

[59] Jos M. F. Ten Berge. A generalization of kristof's theorem on the trace of certain matrix products. *Psychometrika*, 48(4):519–523, 1983. 12

[60] Walter Kristof. A theorem on the trace of certain matrix products and some applications. *Journal of Mathematical Psychology*, 7(3):515 – 530, 1970. 12

Jun Xu received the B.Sc. degree in Pure Mathematics and the M. Sc. Degree in Information and Probability both from the School of Mathematics Science, Nankai University, China, in 2011 and 2014, respectively. He is currently pursuing the Ph.D. degree in the Department of Computing, The Hong Kong Polytechnic University. His research interests include image restoration, subspace clustering, sparse and low rank models.

Lei Zhang (M'04, SM'14, F'18) received his B.Sc. degree in 1995 from Shenyang Institute of Aeronautical Engineering, Shenyang, P.R. China, and M.Sc. and Ph.D degrees in Control Theory and Engineering from Northwestern Polytechnical University, Xi'an, P.R. China, respectively in 1998 and 2001, respectively. From 2001 to 2002, he was a research associate in the Department of Computing, The Hong Kong Polytechnic University. From January 2003 to January 2006 he worked as a Postdoctoral Fellow in the Department of Electrical and Computer Engineering, McMaster University, Canada. In 2006, he joined the Department of Computing, The Hong Kong Polytechnic University, as an Assistant Professor. Since July 2017, he has been a Chair Professor in the same department. His research interests include Computer Vision, Pattern Recognition, Image and Video Analysis, and Biometrics, etc. Prof. Zhang has published more than 200 papers in those areas. As of 2018, his publications have been cited more than 30,000 times in the literature. Prof. Zhang is an Associate Editor of *IEEE Trans. on Image Processing*, *SIAM Journal of Imaging Sciences* and *Image and Vision Computing*, etc. He is a "Clarivate Analytics Highly Cited Researcher" from 2015 to 2017. More information can be found in his homepage http://www4.comp.polyu.edu.hk/~cslzhang/.

David Zhang (F'08) received the degree in computer science from Peking University, the M.Sc. degree in 1982, and the Ph.D. degree in computer science from the Harbin Institute of Technology (HIT), in 1985, respectively. From 1986 to 1988, he was a Post-Doctoral Fellow with Tsinghua University and an Associate Professor with the Academia Sinica, Beijing. In 1994, he received the second Ph.D. degree in electrical and computer engineering from the University of Waterloo, Ontario, Canada. He is currently the Chair Professor with the Hong Kong Polytechnic University, since 2005, where he is the Founding Director of the Biometrics Research Centre (UGC/CRC) supported by the Hong Kong SAR Government in 1998. He is a Croucher Senior Research Fellow, Distinguished Speaker of the IEEE Computer Society, and a Fellow of IAPR. So far, he has published over 20 monographs, over 400 international journal papers and over 40 patents from USA/Japan/HK/China. He was selected as a Highly Cited Researcher in Engineering by Thomson Reuters in 2014, 2015, and 2016, respectively. He also serves as a Visiting Chair Professor with Tsinghua University and an Adjunct Professor with Peking University, Shanghai Jiao Tong University, HIT, and the University of Waterloo. He is Founder and Editor-in-Chief, International Journal of Image and Graphics, the Founder and the Series Editor, Springer International Series on Biometrics (KISB); Organizer, International Conference on Biometrics Authentication, an Associate Editor for over ten international journals including the IEEE Transactions and so on.