External Prior Guided Internal Prior Learning for Real Noisy Image Denoising

Anonymous CVPR submission

Paper ID 1047

Abstract

Most of existing image denoising methods use some statistical models such as additive white Gaussian noise (AWGN) to model the noise, and learn image priors from either external data or the noisy image itself to remove noise. However, the noise in real-world noisy images is much more complex than AWGN, and it is hard to be modeled by simple analytical distributions. Therefore, many state-of-the-art denoising methods in literature become much less effective when applied to real noisy images. In this paper, we develop a robust denoiser for real noisy image denoising without explicit assumption on noise models. Specifically, we first learn external priors from a set of clean natural images, and then use the learned external priors to guide the learning of internal latent priors from the given noisy image. The proposed method is simple yet highly effective. Experiments on real noisy images demonstrate that it achieves much better denoising performance than state-of-the-art denoising methods, including those designed for real noisy images.

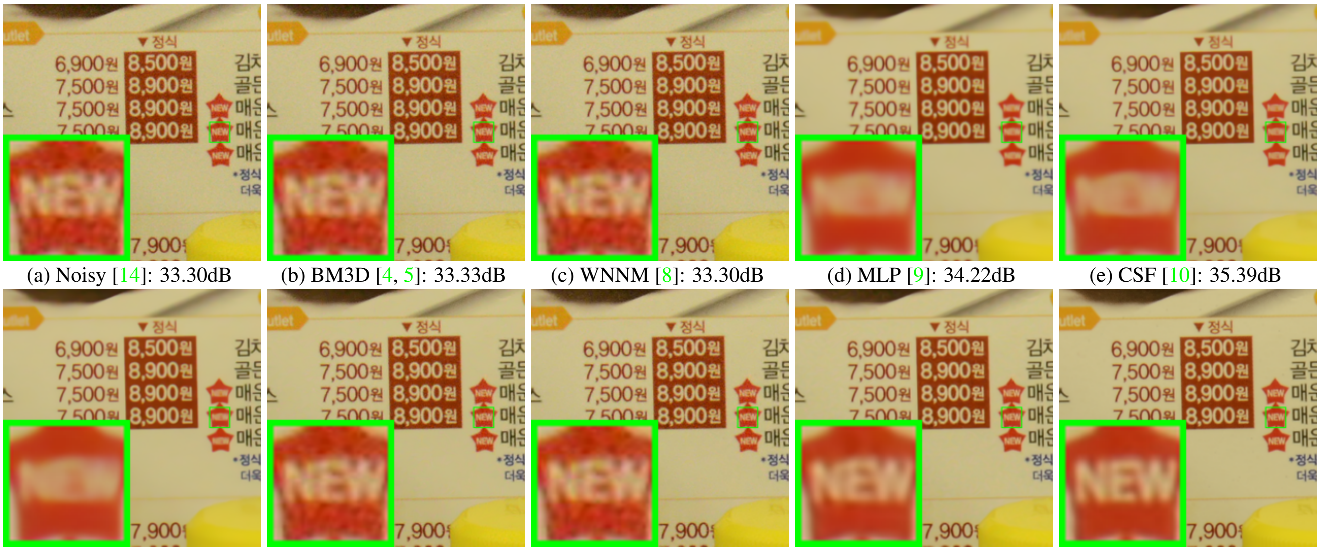
1. Introduction

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, noise is more easily to be corrupted and hence denoising is becoming increasingly important for high resolution imaging. In literature of image denoising, the observed noisy image is usually modeled as $\mathbf{y} = \mathbf{x} + \mathbf{n}$, where $\mathbf{x}$ is the latent clean image and $\mathbf{n}$ is the corrupted noise. Numerous image denoising methods [1, 2, 6, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13] have been proposed in the past decades, including sparse representation and dictionary learning based methods [1, 2, 6], nonlocal self-similarity based methods [3, 4, 5, 6, 7], low-rank based methods [8], neural network based methods [9], and discriminative learning based methods [10, 11].

Most of the existing denoising methods [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13] mentioned above assume noise n to be additive white Gaussian noise (AWGN). Unfortunately, this assumption is too ideal to be true for real-world noisy images, where the noise is much more complex than AWGN [14, 15] and varies by different cameras and camera settings (ISO, shutter speed, and aperture, etc.). According to [15], the noise corrupted in the imaging process [is signal dependent and comes from five main sources: photon shot, fixed pattern, dark current, readout, and quantization noise. As a result, many advanced denoising methods in literature becomes much 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 [5], WNNM [8], MLP [9], CSF [10], and TRD [11], to a real noisy image (captured by a Nikon D800 camera with ISO is 3200) provided in [14]. One can see that these methods either remain the noise or over-smooth the image details on this real noisy image.

There have been a few methods [16, 17, 18, 14, 19, 20, 21] developed for real noisy image denoising. Almost all of these methods follow a two-stage framework: first estimate the parameters of the assumed noise model (usually Gaussian or mixture of Gaussians (MoG)), and then perform denoising with the estimated noise model. Again, the noise in real noisy images is very complex and hard to be modeled by explicit distributions such as Gaussian and MoG. Fig. 1 also shows the denoised results of two state-of-the-art real noisy image denoising methods, Noise Clinic [19, 20] and Neat Image [21]. One can see that these two methods do not perform well on this noisy image either.

This work aims to develop a robust solution for real noisy image denoising without explicitly assuming certain noise models. 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 latent priors from the given noisy image. The flowchart of the proposed method is illustrated in Fig. 3. We first extract millions of patch groups from a set of high quality natural images, with which a Gaussian Mixture Model (GMM) is learned as the external prior. The learned GMM prior model is used to cluster the patch groups extracted from the given noisy image, and then a hybrid orthogonal dictionary (HOD) is learned as the internal prior for image denoising. Our proposed denoising method is simple and ef-



(a) Noisy [14]: 33.30dB (b) BM3D [4, 5]: 33.33dB (c) WNNM [8]: 33.30dB (d) MLP [9]: 34.22dB (e) CSF [10]: 35.39dB

(f) TRD [11]: 35.97dB (g) NC [19]: 35.33dB (h) NI [21]: 34.39dB (i) Ours: **37.02**dB (j) Mean Image [14]

Figure 1. Denoised images of the real noisy image "Nikon D800 ISO 3200 A3" from [14] by different methods. The images are better viewed by zooming in on screen.

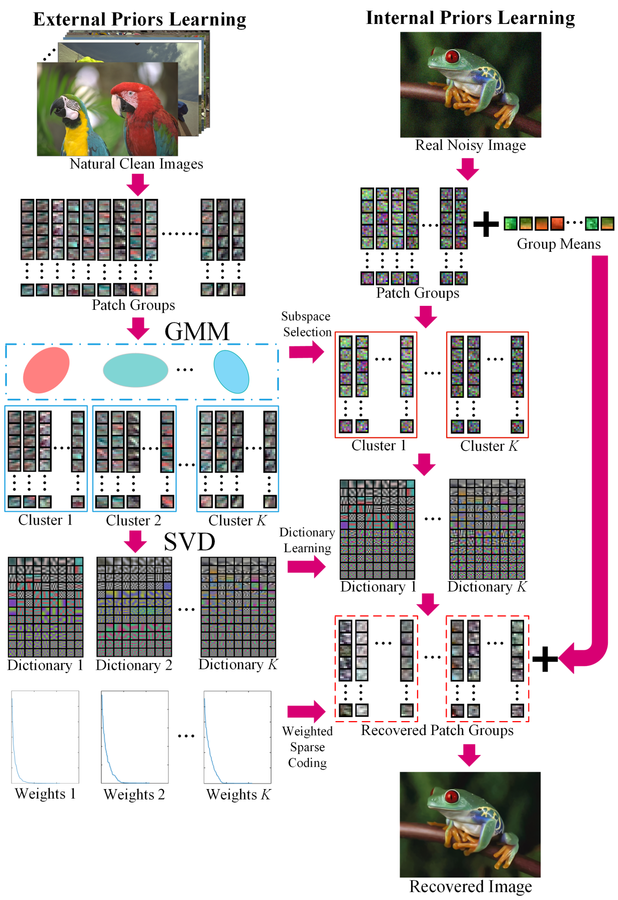


Figure 2. Flowchart of the proposed external prior guided internal prior learning and real noisy image denoising framework.

ficient, yet our extensive experiments on real noisy images clearly demonstrate its better denoising performance than the current state-of-the-arts.

2. Related Work

2.1. Internal *vs.* External Prior Learning

Image priors are playing a key role in image denoising [7, 13, 1, 23, 6, 22]. There are mainly two categories of prior learning methods. 1) External prior learning methods [12, 7, 13] 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 noisy images. 2) Internal prior learning methods [1, 6, 23, 22] directly learn priors from the given noisy image, and the image denoising is often done simultaneously with the prior learning process. It has been demonstrated [7, 13] that the external priors learned from natural clean images are effective and efficient for image denoising problem, but they are not adaptive to the given noisy image so that some fine-scale image structures may not be well recovered. By contrast, the internal priors are adaptive to content of the given image, but the learning processing are usually slow. In addition, most of the internal prior learning methods [1, 6, 23, 22] assume AWGN noise, making the learned priors less robust for real 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 effective to denoise real noisy images.

2.2. Real Noisy Image Denoising

In the last decade, there are many methods [16, 17, 19, 20, 18, 14] for blind image denoising problem. These methods can be applied to real noisy image denoising directly. Liu *et al*. [16] proposed to use "noise level function" to estimate the noise and then use Gaussian conditional random field to obtain the latent clean image. Gong et al. [17] m odels the noise by mixed ℓ_1 and ℓ_2 norms and remove the noise by sparsity prior in the wavelet transform domain. Recently, Zhu et al. proposed a Bayesian model [18] which approximates and removes the noise via low-rank mixture of Gaussians. The method of "Noise Clinic" [19, 20] and the software of Neat Image [21] are developed specifically for real noisy image denoising. "Noise Clinic" [19, 20] generalizes the NL-Bayes model [24] to deal with blind noise and achieves state-of-the-art performance. However, these methods largely depends on the modeling of noise in real noisy images which is hard to be modeled by explicit distributions. Besides, the parametric estimation of the Gaussian or MoG distribution is often time consuming.

3. External Prior Guided Internal Prior Learning

In this section, we first describe the learning of external prior, and then describe in detail the guided internal prior learning. Finally, the denoising algorithm with the learned priors is presented.

3.1. Learn External Patch Group Prior

In this section, we formulate the Patch Group prior learned on natural color images. Similar to [7], the patch group (PG) is defined as a group of similar patches to the local patch. The patch group mean is subtracted, and hence different groups of patches can share similar PGs. In this way, the space natural image patches to be modeled is largely reduced.

In this work, each local patch extracted from RGB images is of size $p \times p \times 3$. Then we search the M most similar patches $\{\mathbf{x}_m\}_{m=1}^M$ around each local patch through Euclidean distance, in a local window of size $W \times W$. The $\mathbf{x}_m \in \mathbb{R}^{3p^2\times 1}$ is a patch vector formed by combining the 3 patch vectors (of size $p^2 \times 1$) in R, G, B channels. 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 we have extracted N PGs from a set of external natural images, and the n-th PG is defined as $\overline{\mathbf{X}}_n \triangleq \{\overline{\mathbf{x}}_{n,m}\}_{m=1}^M, n = 1, ..., N$. We employ the Gaussian Mixture Model (GMM) to learn the external patch group based NSS prior. The overall objective log-likelihood function of GMM is

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

The learning process is similar to the learning stage of the PGPD method [7]. Please refer to [7] for more details. After the learning stage, we finally obtain a GMM model with K Gaussian components. The learned parameters include mixture weights $\{\pi_k\}_{k=1}^K$, mean vectors $\{\boldsymbol{\mu}_k\}_{k=1}^K$, and covariance matrices $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$. Similar to [7], the mean vector of each cluster is natural zeros, i.e., $\boldsymbol{\mu}_k = \mathbf{0}$.

Now, we have clustered the PGs extracted from external clean images into K Gaussians or subspaces. To better characterize each subspace, we perform singular value decomposition (SVD) on the covariance matrix:

$$\boldsymbol{\Sigma}_k = \mathbf{U}_k \mathbf{S}_k \mathbf{U}_k^\top. \quad (2)$$

The singular vector matrices $\{\mathbf{U}_k\}_{k=1}^K$ are employed as the external orthogonal dictionary to guide the internal dictionary learning. In Figure 4 (a) and (b), we illustrate an external clean image and one orthogonal dictionary learned via GMM on PGs of the external clean image. To better illustrate the dictionary The singular values in the diagonal of $\mathbf{S}_k$ reflect the significance of the singular vectors in $\mathbf{U}_k$ and utilized as prior weights for weighted sparse coding which will be discussed in next section.

3.2. External Prior Guided Internal Prior Learning

After the external patch group (PG) prior is learned, we can employ it to guide the internal PG prior learning for the given testing (real noisy) image. The guidance mainly comes from two aspects. One aspect is that the external prior can guide the internal noisy PGs to be assigned to most suitable Gaussians or subspaces. And for each subspace, the other aspect is to guide the orthogonal dictionary learning of internal noisy PGs.

3.2.1 Internal Subspace Selection

Given a real noisy image, assume we can totally extract N local patches from it. Similar to the external prior learning stage, for the n-th local patch ($n = 1, ..., N$), we extract its M most similar patches around it to form a noisy PG denoted by $\mathbf{Y}_n = \{\mathbf{y}_{n,1}, ..., \mathbf{y}_{n,M}\}$. Then the group mean of $\mathbf{Y}_n$, denoted by $\boldsymbol{\mu}_n$, is calculated and subtracted from each patch by $\overline{\mathbf{y}}_{n,m} = \mathbf{y}_{n,m} - \boldsymbol{\mu}_n$, leading to the mean subtracted PG $\overline{\mathbf{Y}}_n \triangleq \{\overline{\mathbf{y}}_{n,m}\}_{m=1}^M$. For adaptivity, we project the PG $\overline{\mathbf{Y}}_n$ into its most suitable Gaussian component (subspace) of the GMM learned on external PGs. The subspace most suitable for $\overline{\mathbf{Y}}_n$ is selected by firstly calculating the posterior probability of "$\overline{\mathbf{Y}}_n$ belonging to the k-th Gaussian component":

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

for $k = 1, ..., K$, and then choosing the component with the maximum A-posteriori (MAP) probability $\max_k P(k|\overline{\mathbf{Y}}_n)$.

3.2.2 Internal Orthogonal Dictionary Learning

Assume we have assigned all internal noisy PGs $\{\overline{\mathbf{Y}}_n\}_{n=1}^N$ to their corresponding most suitable Gaussians or subspaces in $\{\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_k)\}_{k=1}^K$. For the k-th subspace, assume the noisy PGs assigned to it are $\{\overline{\mathbf{Y}}_{k_n}\}_{n=1}^{N_k}$ such that $\overline{\mathbf{Y}}_{k_n} = [\overline{\mathbf{y}}_{k_n,1}, ..., \overline{\mathbf{y}}_{k_n,M}]$ and $\sum_{k=1}^{K} N_k = N$. We utilize the external orthogonal dictionary $\mathbf{U}_k$ (Eqn. (2)) to guide the learning of an orthogonal dictionary for adaptively characterizing the internal PGs in the k-th subspace. The reasons we aim to learn orthogonal dictionaries are two-fold: firstly, given an orthonormal dictionary $\mathbf{D}$, its quality measure *mutual incoherence* $\mu(\mathbf{D}) = \max_{i=j} \frac{|\mathbf{d}_i^T \mathbf{d}_j|}{\|\mathbf{d}_i\|_2 \|\mathbf{d}_j\|_2}$ is naturally 0 and therefore better than other redundant dictionaries; secondly, the orthogonality of dictioanry can guarantee the dictionary learning stage has closed-form solutions (please refer to Eqn. (9)) and the closed-form solutions will make our proposed method efficient, which will be discussed in experimental section.

The learned dictionary $\mathbf{D}_k \triangleq [\mathbf{D}_{k,e}\ \mathbf{D}_{k,i}] \in \mathbb{R}^{3p^2 \times 3p^2}$ has two parts: the external part $\mathbf{D}_{k,e} = \mathbf{U}_k(:, 1 : 3p^2 - r) \in \mathbb{R}^{3p^2 \times (3p^2 - r)}$ is directly obtained from the external dictionary $\mathbf{U}_k$, and the internal part $\mathbf{D}_{k,i}$ is consisted of dictionary atoms adaptively learned from the internal noisy PGs $\{\overline{\mathbf{Y}}_{k_n}\}_{n=1}^{N_k}$. For notation simplicity, we ignore the subspace index k and denote the noisy PGs assigned to each subspace as $\mathbf{Y} \triangleq \{\overline{\mathbf{Y}}_n\}_{n=1}^{N} = [\overline{\mathbf{y}}_{1,1}, ..., \overline{\mathbf{y}}_{1,M}, ..., \overline{\mathbf{y}}_{N,1}, ..., \overline{\mathbf{y}}_{N,M}]$. The learning is performed under the weighted sparse coding framework proposed as follows:

$$\begin{aligned} &\min_{\mathbf{D}_i, \{\boldsymbol{\alpha}_{n,m}\}} \sum_{n=1}^{N} \sum_{m=1}^{M} (\|\overline{\mathbf{y}}_{n,m} - \mathbf{D}\boldsymbol{\alpha}_{n,m}\|_2^2 + \sum_{j=1}^{3p^2} \lambda_j |\boldsymbol{\alpha}_{n,m,j}|) \\ &s.t. \quad \mathbf{D} = [\mathbf{D}_e\ \mathbf{D}_i],\ \mathbf{D}_i^\top \mathbf{D}_i = \mathbf{I}_r,\ \mathbf{D}_e^\top \mathbf{D}_i = \mathbf{0}, \end{aligned} \tag{4}$$

where $\boldsymbol{\alpha}_{n,m}$ is the sparse coefficient vector of the m-th patch $\overline{\mathbf{y}}_{n,m}$ in the n-th PG $\overline{\mathbf{Y}}_n$ 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

$$\lambda_j = \lambda / (\sqrt{\mathbf{S}_j} + \varepsilon). \tag{5}$$

We employ square roots of the singular values in $\mathbf{S}$ (please refer to Eqn. (2)) as external prior weights and add a small positive number ε to avoid zero denominator. Noted that $\mathbf{D}_e = \emptyset$ if $r = 3p^2$ and $\mathbf{D}_e = \mathbf{U}_k$ if $r = 0$. The dictionary $\mathbf{D} = [\mathbf{D}_e\ \mathbf{D}_i]$ is orthogonal by checking that:

$$\mathbf{D}^\top \mathbf{D} = \begin{bmatrix} \mathbf{D}_e^\top \\ \mathbf{D}_i^\top \end{bmatrix} [\mathbf{D}_e\ \mathbf{D}_i] = \begin{bmatrix} \mathbf{D}_e^\top \mathbf{D}_e & \mathbf{D}_e^\top \mathbf{D}_i \\ \mathbf{D}_i^\top \mathbf{D}_e & \mathbf{D}_i^\top \mathbf{D}_i \end{bmatrix} = \mathbf{I} \tag{6}$$

Similar to K-SVD [1], we employ an alternating iterative framework to solve the optimization problem (5). Specifically, we initialize the orthogonal dictionary as $\mathbf{D}_{(0)} = \mathbf{U}_k$ and for $t = 0, 1, ..., T-1$, alternatively do:

Updating Sparse Coefficient: given the orthogonal dictionary $\mathbf{D}_{(t)}$, we update the sparse coefficient vector of the m-th patch $\overline{\mathbf{y}}_{n,m}$ in the n-th PG $\overline{\mathbf{Y}}_n$ via solving

$$\begin{aligned} &\boldsymbol{\alpha}_{n,m}^{(t)} := \arg\min_{\boldsymbol{\alpha}_{n,m}} \|\overline{\mathbf{y}}_{n,m} - \mathbf{D}_{(t)}\boldsymbol{\alpha}_{n,m}\|_2^2 + \sum_{j=1}^{3p^2} \lambda_j |\boldsymbol{\alpha}_{n,m,j}| \\ &s.t. \quad \mathbf{D}_{(t)} = [\mathbf{D}_e\ \mathbf{D}_i],\ \mathbf{D}_i^\top \mathbf{D}_i = \mathbf{I}_r,\ \mathbf{D}_e^\top \mathbf{D}_i = \mathbf{0}, \end{aligned} \tag{7}$$

Since dictionary $\mathbf{D}_{(t)} = [\mathbf{D}_e\ \mathbf{D}_i^{(t)}]$ is orthogonal, the problems (7) has a closed-form solution [7]

$$\boldsymbol{\alpha}_{n,m}^{(t)} = \text{sgn}(\mathbf{D}_{(t)}^\top \overline{\mathbf{y}}_{n,m}) \odot \max(|\mathbf{D}_{(t)}^\top \overline{\mathbf{y}}_{n,m}| - \boldsymbol{\Lambda}, \mathbf{0}), \tag{8}$$

where $\boldsymbol{\Lambda} = [\lambda_1, \lambda_2, ..., \lambda_{3p^2}]$ is the vector of regularization parameter and sgn($\bullet$) is the sign function, $\odot$ means element-wise multiplication.

Updating Internal Dictionary: given the sparse coefficient vectors $\mathbf{A}^{(t)} = [\boldsymbol{\alpha}_{1,1}^{(t)}, ..., \boldsymbol{\alpha}_{1,M}^{(t)}, ..., \boldsymbol{\alpha}_{N,1}^{(t)}, ..., \boldsymbol{\alpha}_{N,M}^{(t)}]$, we update the internal orthogonal dictionary via solving

$$\begin{aligned} \mathbf{D}_i^{(t+1)} &:= \arg\min_{\mathbf{D}_i} \sum_{n=1}^{N} \sum_{m=1}^{M} (\|\overline{\mathbf{y}}_{n,m} - \mathbf{D}\boldsymbol{\alpha}_{n,m}^{(t)}\|_2^2) \\ &= \arg\min_{\mathbf{D}_i} \|\mathbf{Y} - \mathbf{D}\mathbf{A}^{(t)}\|_F^2 \\ s.t. \quad &\mathbf{D} = [\mathbf{D}_e\ \mathbf{D}_i],\ \mathbf{D}_i^\top \mathbf{D}_i = \mathbf{I}_r,\ \mathbf{D}_e^\top \mathbf{D}_i = \mathbf{0}, \end{aligned} \tag{9}$$

The sparse coefficient matrix $\mathbf{A}^{(t)} = [(\mathbf{A}_e^{(t)})^\top\ (\mathbf{A}_i^{(t)})^\top]^\top$ also has two parts: the external part $\mathbf{A}_e^{(t)} \in \mathbb{R}^{(3p^2 - r) \times NM}$ and the internal part $\mathbf{A}_i^{(t)} \in \mathbb{R}^{r \times NM}$ denote the coefficients over external dictionary $\mathbf{D}_e$ and internal dictionary $\mathbf{D}_i^{(t)}$, respectively. According to the Theorem 4 in [25], the problem (9) has a closed-form solution $\mathbf{D}_i^{(t+1)} = \mathbf{U}_i \mathbf{V}_i^\top$, where $\mathbf{U}_i \in \mathbb{R}^{3p^2 \times r}$ and $\mathbf{V}_i \in \mathbb{R}^{r \times r}$ are the orthogonal matrices obtained by the following SVD

$$(\mathbf{I} - \mathbf{D}_e \mathbf{D}_e^\top)\mathbf{Y}(\mathbf{A}_i^{(t)})^\top = \mathbf{U}_i \mathbf{S}_i \mathbf{V}_i^\top. \tag{10}$$

The orthogonality of internal dictionary $\mathbf{D}_i^{(t+1)}$ can be checked by $(\mathbf{D}_i^{(t+1)})^\top (\mathbf{D}_i^{(t+1)}) = \mathbf{V}_i \mathbf{U}_i^\top \mathbf{U}_i \mathbf{V}_i^\top = \mathbf{I}_r$. In Figure 4 (c) and (d), we illustrate a denoised image by our proposed method and one internal orthogonal dictionary learned from PGs of the given noisy image.

3.3. The Denoising Algorithm

We evaluate the performance of the proposed framework on denoising real noisy images. The denoising is simultaneously done with the guided internal dictionary learning process. We ignore the index $k \in \{1, ..., K\}$ of subspace for notation simplicity. In the denoising stage, for each subspace, the group mean vectors $\{\boldsymbol{\mu}_n\}_{n=1}^{N}$ of corresponding mean subtracted noisy PGs $\{\overline{\mathbf{Y}}_n\}_{n=1}^{N}$ are saved for reconstruction. Until now, we obtain the solutions of sparse coefficient vectors $\{\hat{\boldsymbol{\alpha}}_{n,m}^{(T-1)}\}$ in Eqn. (8) for $n = 1, ..., N; m = 1, ..., M$ and the orthogonal dictionary $\mathbf{D}_{(T)} = [\mathbf{D}_e\ \mathbf{D}_i^{(T)}]$ in Eqn. (9). Then the m-th latent clean patch $\hat{\mathbf{y}}_{n,m}$ in the n-th PG $\mathbf{Y}_n$ is recovered by

$$\hat{\mathbf{y}}_{n,m} = \mathbf{D}_{(T)} \hat{\boldsymbol{\alpha}}_{n,m} + \boldsymbol{\mu}_n, \tag{11}$$

Alg. 1: External Prior Guided Internal Prior Learning for Real Noisy Image Denoising

---

Input: Noisy image $\mathbf{y}$, external PG prior GMM model
Output: The denoised image $\hat{\mathbf{x}}$.
Initialization: $\hat{\mathbf{x}}^{(0)} = \mathbf{y}$;
for $Ite = 1 : IteNum$ **do**
1. Extracting internal PGs from $\hat{\mathbf{x}}^{(Ite-1)}$;
   for each PG $\mathbf{Y}_n$ **do**
2. Calculate group mean vector $\boldsymbol{\mu}_n$ and form mean subtracted PG $\overline{\mathbf{Y}}_n$;
3. Subspace selection via Eqn. (3);
   end for
   for the PGs in each Subspace **do**
4. External PG prior Guided Internal Orthogonal Dictionary Learning by solving (4);
5. Recover each patch in all PGs via Eqn. (11);
   end for
6. Aggregate the recovered PGs of all subspaces to form the recovered image $\hat{\mathbf{x}}^{(Ite)}$;

end for

---

where $n = 1, ..., N; m = 1, ..., M$. The latent clean image $\hat{\mathbf{x}}$ is reconstructed by aggregating all the estimated PGs. Similar to [7], we perform the above denoising procedures for several iterations for better denoising outputs. The proposed denoising algorithm is summarized in Alg. 1.

4. Experiments

In this section, we evaluate the performance of the proposed algorithm on real image denoising. To evaluation the effectiveness of the proposed framework of external prior guided internal prior learning, we compare it with the methods with only external prior or only internal prior (Section 4.3). We also compare the proposed algorithm with other state-of-the-art denoising methods [4, 5, 9, 8, 10, 11, 14, 19, 20, 21] (Section 4.4).

4.1. The Testing Datasets

The comparisons are performed on two standard datasets in which the images were captured under indoor or outdoor lighting conditions by different types of cameras and camera settings. The first dataset provided in [20] includes 20 real noisy images collected under uncontrolled outdoor environment. This dataset does not have "ground truth" images and hence the objective measurements can not be performed. In order to evaluate the compared methods on quantitative measures, we perform experiments on the second dataset provided in [14]. It includes 17 real noisy images and corresponding mean images. The noisy images were collected under controlled indoor environment. Some samples can be found in [14]. For each image, the same 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 can be computed. Since the 17 images are too large (of size about $7000\times5000\times3$) and share repetitive contents, the authors in [14] performed comparison on 15 cropped images (of size $512 \times 521 \times 3$). To evaluate the compared methods on more samples, we cropped the 17 large images from [14] into 60 smaller images (of size $500 \times 500 \times 3$) including different contents. Some samples are shown in Figure 5. Note that the noise in our cropped 60 images used in [14] are different from the noise in the 15 images cropped by the authors of [14] since they are taken in different shots.

4.2. Implementation Details

Our proposed method contains two stages, the external prior learning stage and the external prior guided internal learning stage. In the first stage, we set $p = 6$ (so the patch size is $6 \times 6 \times 3$), $M = 10$ (the number of patches in a patch group (PG)), $W = 31$ (so the window size for PG searching is 31×31, and $K = 32$ (the number of Gaussians in Gaussian Mixture Model (GMM)). We learn the external prior via GMM on about 3.6 million PGs extracted from the Kodak PhotoCD Dataset (http://r0k.us/graphics/kodak/), which includes 24 high quality color images. In the second stage, we set $r = 54$ (the number of internal atoms in the learned dictionaries), $\lambda = 0.001$ (the sparse regularization parameter), $T = 2$ (the number of iterations for solving problem (4)), and $IteNum = 4$ (the number of iterations for Alg. 1). All experiments are performed under the Matlab2014b environment on a machine with Intel(R) Core(TM) i7-5930K CPU of 3.5GHz and 32GB RAM.

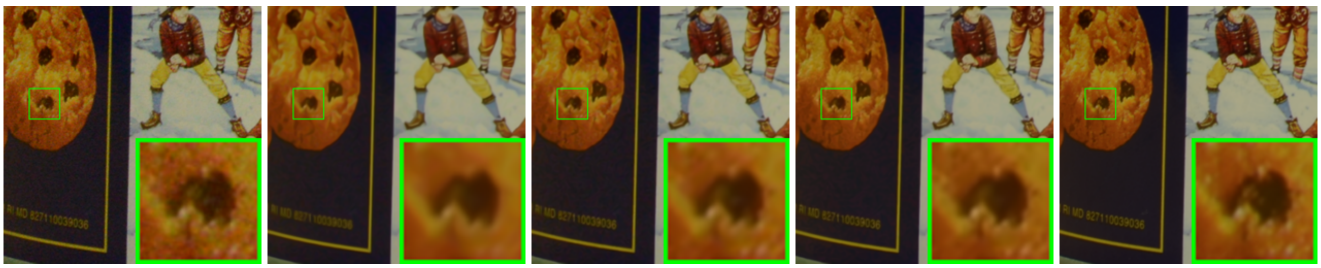
4.3. Comparison among external, internal and external guided internal priors

In this section, we compare our proposed method on real image denoising with external prior based method (denoted as "External") and internal prior based method (denoted as "Internal"). For the "External" method, we utilize the external dictionaries (i.e., $r = 0$ in Eqn. (5)) for denoising. For the given noisy image, we extract the PGs and then do internal subspace selection via Eqn. 3. The denoising is performed via the weighted sparse coding framework proposed in [7]. For the "Internal" method, the overall framework is similar to the method of [6]. We employ the GMM model (also with $K = 32$ Gaussians) to cluster the noisy PGs extracted from given noisy image into multiple subspaces, and for each subspace, we utilize the internal orthogonal dictionary obtained via Eqn. (2) by weighted sparse coding framework in [7]. All parameters of the three methods are tuned to achieve best performance.

We compare the above mentioned methods on the 60 cropped images (of size $500 \times 500 \times 3$) from [14]. The

(a) Noisy [14]: 35.89dB (b) External: 39.05dB (c) Internal: 38.75dB (d) Ours: **39.39**dB (e) Mean Image [14]

Figure 3. Denoised images of the 96-th cropped image from "Nikon D600 ISO 3200 C1" [14] by different methods. The images are better to be zoomed in on screen.



(a) Noisy [14]: 34.55dB (b) External: 36.09dB (c) Internal: 37.11dB (d) Ours: **37.39**dB (e) Mean Image [14]

Figure 4. Denoised images of the 94-th cropped image from "Nikon D600 ISO 3200 C1" [14] by different methods. The images are better to be zoomed in on screen.

Figure 5. Some samples cropped from real noisy images of [14].

Table 1. Average PSNR (dB) results and Run Time (seconds) of the External, the Internal, and our proposed methods on 60 real noisy images (of size $500 \times 500 \times 3$) cropped from [14].

	Noisy	External	Internal	Ours
PSNR	34.51	38.21	38.07	**38.75**
Time	—	**39.57**	667.36	41.89

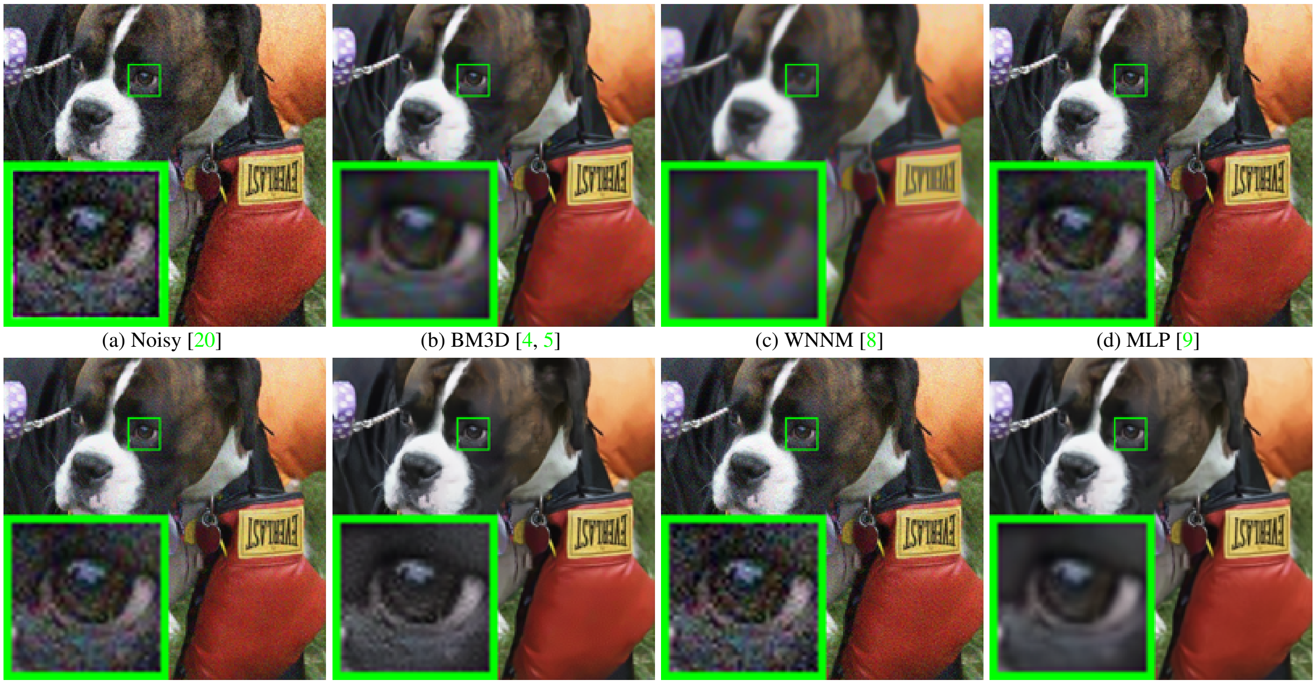
average PSNR and speed of these methods are listed in Table 1. It can be seen that our proposed method achieves better PSNR results than the methods of "External" and "Internal". The speed of our proposed method is much faster than the "Internal" method while only a little slower than the "External" method. We also compare the visual quality of the denoised images by these methods. From the results listed in Figure 3 and Figure 4, we can see that the "External" method is good at recovering structures (Figure 3) while the "Internal" method is good at recovering internal complex textures (Figure 4). And by utilizing both the external and internal priors, our proposed method can recover well both the structures and textures. Noted that the noisy images in Figures 3 and 4 are cropped from the same image captured by Nikon D600 at ISO = 3200 in [14]. Hence, the differences on PSNR and visual quality among these methods only depends on the contents of the cropped images.

4.4. Comparison with Other Denoising Methods

In this section, we compare the proposed method with other state-of-the-art image denoising methods such as BM3D [4], WNNM [8], MLP [9], CSF [10], TRD [11], Noise Clinic (NC) [19], Cross-Channel (CC) [14], and Neat Image (NI) [21]. The methods of BM3D [4], WNNM [8], MLP [9], CSF [10], and TRD [11] are designed for removing Gaussian noise. For BM3D and WNNM, the level σ of Gaussian noise is very important and is estimated by the method [26]. The other parameters are set as default. For the methods of MLP, CSF, and TRD, we employ their default parameters settings. Since these methods are designed for grayscale images, we utilize them to denoise the R, G, B channels separately for color noisy images. The Noise Clinic (NC) [19] is a blind image denoising method which does not need any noise prior. We also compare with Neat Image (NI), a commercial software for image denoising. Due to its excellent performance, Neat Image (NI) is embedded into Photoshop and Corel PaintShop [21]. The comparisons are performed on the real noisy images from [20] and [14].

4.4.1 Comparison on the First Dataset [20]

The real noisy images in the dataset [20] do not have "ground truth" images. On this dataset, we compare the proposed method with the methods of BM3D [4], WNNM [8], MLP [9], TRD [11], Noise Clinic (NC) [19], and Neat Image (NI) [21]. We only compare the visual quality of the denoised images. Figure 6 shows the denoised images



(a) Noisy [20] (b) BM3D [4, 5] (c) WNNM [8] (d) MLP [9]

(e) TRD [11] (f) NC [19] (g) NI [21] (h) Ours

Figure 6. Denoised images of the image "Dog" by different methods. The images are better to be zoomed in on screen.

of "Dog" by the competing methods. More visual comparisons can be found in the supplementary file. It can be seen that the methods of BM3D, WNNM tend to globally over-smooth the image while locally remain some noise, while the methods of MLP, TRD are likely to remain noise in the whole image. This demonstrates that the methods designed for Gaussian noise are not effective for removing the complex noise in real noisy images. Though Noise Clinic and Neat Image are specifically developed for removing complex noise, they would sometimes fail to recover real noisy images. However, our proposed method recoveries more faithfully the structures and textures (such as the eye area) than the other competing methods.

4.4.2 Comparison on the Second Dataset [14]

The real noisy images in the second dataset [14] have corresponding "ground truth" images. On this dataset, we firstly perform comparison on the 15 cropped images used in [14]. The compared method are BM3D [4], WNNM [8], MLP [9], CSF [10], TRD [11], Noise Clinic (NC) [19], and Cross-Channel (CC) [14]. The PSNR values are listed in Table 2. As we can see, on most (9 out of the 15) images captured by different cameras and camera settings, our proposed method obtains better PSNR values than the other methods. Noted that, though in [14] a specific model is trained for each camera and camera setting, our proposed general method still gains 0.28dB improvements on PNSR over [14]. We also compare the visual quality of the denoised images by the competing methods. Figure 7 shows the denoised images of a scene captured by Canon 5D Mark 3 at ISO $= 3200$ by the competing methods. More visual comparisons can be found in the supplementary file. We can see that BM3D, WNNM, NC, NI, and CC would either remain noise or generate artifacts, while MLP, TRD are likely to over-smooth the image. By combining the external and internal priors, our proposed method preserves edges and textures better than other methods.

To evaluate the compared methods on more samples, we then perform denoising experiments on the 60 smaller images cropped from the 17 images provided in [14]. The average PSNR results are listed in Table 3 (the code of [14] is not available so that it is not compared). The numbers in red color and blue color are the best and second best results, respectively. It can be seen that our proposed method achieves much better PSNR results than the other methods. The improvement of our method over the second best method (TRD) is 1dB. Due to the spacial limitations, the visual comparisions are provided in the supplementary file.

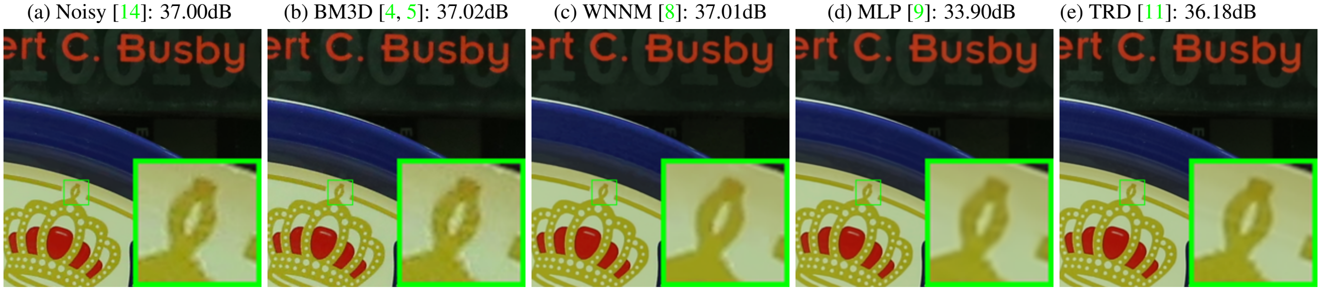
5. Conclusion and Future Work

Image priors are important for solving image denoising problems. The external priors learned from external clean images are generally effective to most images, while the internal priors learned directly from the noisy image are adaptive to the given image but would be biased by the complex noise in real noisy images. In this paper, we demon-

Table 2. Average PSNR(dB) results of different methods on 15 cropped real noisy images used in [14].

Camera Settings	**Noisy**	**BM3D**	**WNNM**	**MLP**	**CSF**	**TRD**	**NI**	**NC**	**CC**	**Ours**
Canon 5D Mark III ISO = 3200	37.00	37.08	37.09	33.92	35.68	36.20	37.68	38.76	38.37	40.50
	33.88	33.94	33.93	33.24	34.03	34.35	34.87	35.69	35.37	37.05
	33.83	33.88	33.90	32.37	32.63	33.10	34.77	35.54	34.91	36.11
Nikon D600 ISO = 3200	33.28	33.33	33.34	31.93	31.78	32.28	34.12	35.57	34.98	34.88
	33.77	33.85	33.79	34.15	35.16	35.34	35.36	36.70	35.95	36.31
	34.93	35.02	34.95	37.89	39.98	40.51	38.68	39.28	41.15	39.23
Nikon D800 ISO = 1600	35.47	35.54	35.57	33.77	34.84	35.09	37.34	38.01	37.99	38.40
	35.71	35.79	35.77	35.89	38.42	38.65	38.57	39.05	40.36	40.92
	34.81	34.92	34.95	34.25	35.79	35.85	37.87	38.20	38.30	38.97
Nikon D800 ISO = 3200	33.26	33.34	33.31	37.42	38.36	38.56	36.95	38.07	39.01	38.66
	32.89	32.95	32.96	34.88	35.53	35.76	35.09	35.72	36.75	37.07
	32.91	32.98	32.96	38.54	40.05	40.59	36.91	36.76	39.06	38.52
Nikon D800 ISO = 6400	29.63	29.66	29.71	33.59	34.08	34.25	31.28	33.49	34.61	33.76
	29.97	30.01	29.98	31.55	32.13	32.38	31.38	32.79	33.21	33.43
	29.87	29.90	29.95	31.42	31.52	31.76	31.40	32.86	33.22	33.58
Average	33.41	33.48	33.48	34.32	35.33	35.65	35.49	36.43	36.88	37.16

(a) Noisy [14]: 37.00dB (b) BM3D [4, 5]: 37.02dB (c) WNNM [8]: 37.01dB (d) MLP [9]: 33.90dB (e) TRD [11]: 36.18dB



(f) NC [19]: 38.76dB (g) NI [21]: 37.68dB (h) CC [14]: 38.37dB (i) Ours: **40.50**dB (j) Mean Image [14]

Figure 7. Denoised images of the image "Canon 5D Mark 3 ISO 3200 1" by different methods. The images are better to be zoomed in on screen.

Table 3. Average PSNR(dB) results of different methods on 60 real noisy images cropped from [14].

Methods	**BM3D**	**WNNM**	**MLP**	**CSF**
PSNR	34.58	34.52	36.19	37.40
Methods	**TRD**	**NI**	**NC**	**Ours**
PSNR	37.75	36.53	37.57	38.75

strates that, once unifying both the priors in external clean images and internal noisy images, we can achieve much better while still efficient performance on real image denoising problem. Specifically, the external patch group (PG) priors learned on natural clean images can be used to guide the subspace selection and orthogonal dictionary learning of internal noisy PGs from given noisy images. The experiments on real image denoising problem have demonstrated the powerful ability of the proposed method. In the future, we will speed up the proposed algorithm and evaluate the proposed method on other computer vision tasks such as image super-resolution.

References

[1] 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, 4

[2] 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

[3] 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

[4] 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, 2, 5, 6, 7, 8

[5] 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, 5, 7, 8

[6] 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, 5

[7] 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, 5, 6

[8] 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, 5, 6, 7, 8

[9] 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, 5, 6, 7, 8

[10] 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, 5, 6, 7

[11] 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, 5, 6, 7, 8

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

[13] 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

[14] 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, 5, 6, 7, 8

[15] 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

[16] 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

[17] 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

[18] 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

[19] 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, 5, 6, 7, 8

[20] 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, 5, 6, 7

[21] Neatlab ABSoft. Neat Image. `https://ni.neatvideo.com/home`. 1, 2, 3, 5, 6, 7, 8

[22] M. Zontak and M. Irani. Internal statistics of a single natural image. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 977–984, 2011. 2

[23] G. Yu, G. Sapiro, and S. Mallat. Solving inverse problems with piecewise linear estimators: From Gaussian mixture models to structured sparsity. *IEEE Transactions on Image Processing*, 21(5):2481–2499, 2012. 2

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

[25] H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286, 2006. 4

[26] 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. 6