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 priors from the given noisy image for latent clean image recovery. 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, image is more easily to be corrupted by noise 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 [?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?] have been proposed in the past decades, including sparse representation and dictionary learning based methods [?, ?, ?], nonlocal self-similarity based methods [?, ?, ?, ?, ?], low-rank based methods [?], neural network based methods [?], and discriminative learning based methods [?, ?].

Most of the denoising methods [?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?] mentioned above assume noise $\mathbf{n}$ 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 [?, ?] and varies by different cameras and camera settings (ISO, shutter speed, and aperture, etc.). According to [?], 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. **??** shows an example, where we apply some representative and state-of-the-art denoising methods, including CBM3D [?], WNNM [?], MLP [?], CSF [?] and TNRD [?], to a real noisy image (captured by a Nikon D800 camera with ISO is 3200) provided in [?]. 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 [?, ?, ?, ?, ?, ?, ?] 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. **??** also shows the denoising results of two state-of-the-art real noisy image denoising methods, Noise Clinic [?, ?] and Neat Image [?]. One can see that these two methods do not perform well on this noisy image either.

This work aims to develop a robust denoiser 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 priors from the given noisy image. The flowchart of the proposed method is illustrated in Fig. **??**. 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 an external-internal hybrid orthogonal dictionary is learned as the final prior for image denois-

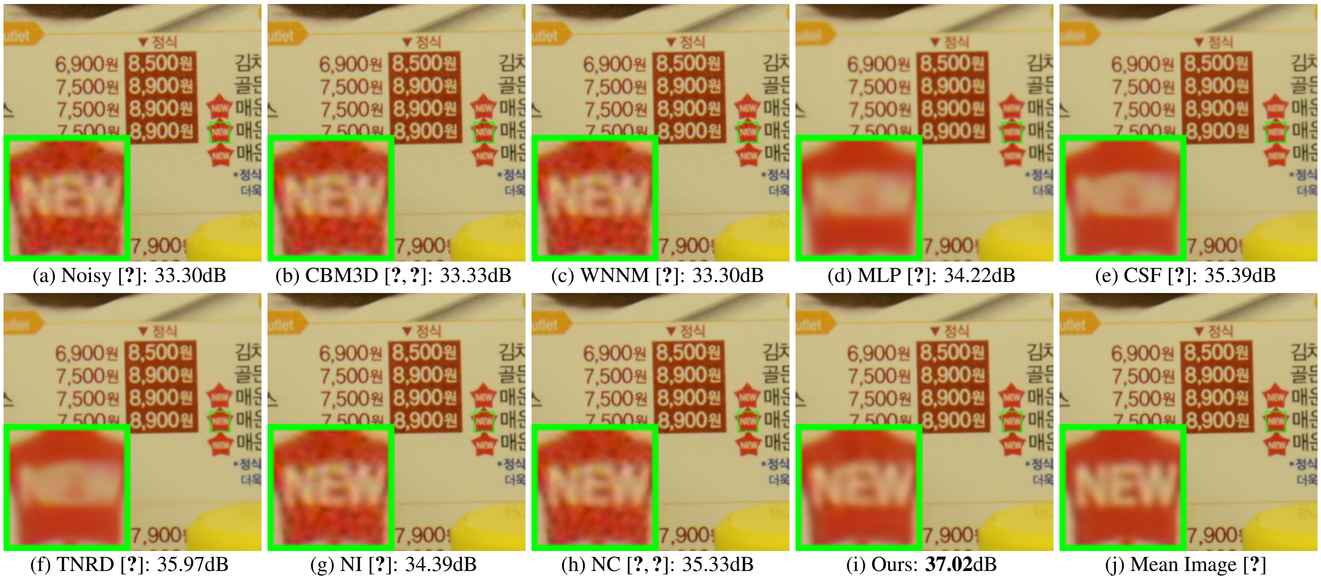

(a) Noisy [?]: 33.30dB (b) CBM3D [?, ?]: 33.33dB (c) WNNM [?]: 33.30dB (d) MLP [?]: 34.22dB (e) CSF [?]: 35.39dB
(f) TNRD [?]: 35.97dB (g) NI [?]: 34.39dB (h) NC [?, ?]: 35.33dB (i) Ours: **37.02**dB (j) Mean Image [?]

Figure 1. Denoised images of a region cropped from the real noisy image "Nikon D800 ISO 3200 A3" [?] 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 index can be computed. The images are better viewed by zooming in on screen.

ing. Our proposed denoising method is simple and efficient, 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 [?, ?, ?, ?, ?, ?, ?]. There are mainly two categories of prior learning methods. 1) External prior learning methods [?, ?, ?] 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 [?, ?, ?, ?] directly learn priors from the given noisy image, and image denoising is often done simultaneously with the prior learning process. It has been demonstrated [?, ?, ?] 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 [?,?,?,?] 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

Recently, several denoising methods have been proposed to remove unknown noise from images [?, ?, ?, ?, ?]. Among them, the "Noise Clinic" [?, ?] estimates the noise distribution by using a multivariate Gaussian model and removes the noise by using a generalized version of nonlocal Bayesian model [?]. Zhu *et al.* proposed a Bayesian method [?] to approximate and remove the noise via a low-rank mixture of Gaussians (MoG) model. There are also several methods specifically designed for real noisy image denoising [?,?]. The method in [?] models the cross-channel noise in real noisy image as a multivariate Gaussian and the noise is removed by the Bayesian nonlocal means filter [?]. The commercial software Neat Image [?] estimates the noise parameters from a flat region of the given noisy image and filters the noise correspondingly. Almost all the methods [?, ?, ?, ?, ?, ?] mentioned above use Gaussian or MoG to model the noise in real noisy images. Nonetheless, the noise in real noisy images is very complex and hard to be modeled by explicit distributions. In this paper, we propose a simple yet effective method for real noisy image denoising without explicitly assuming certain noise models.

3. External Prior Guided Internal Prior Learning and Image Denoising

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.

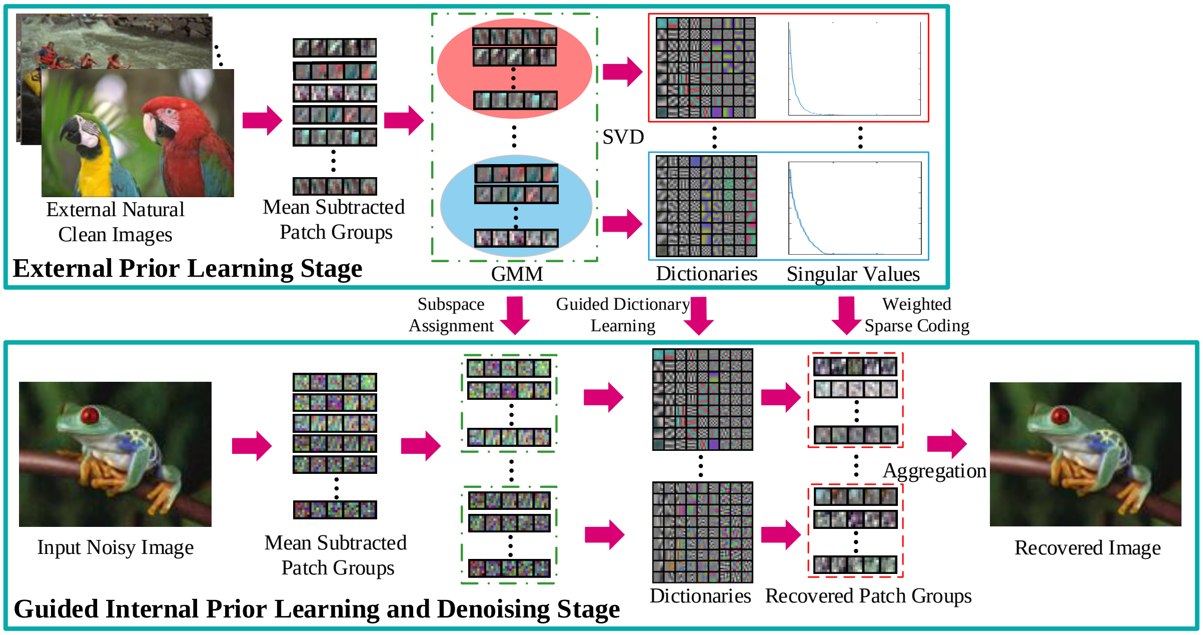

External Natural Clean Images
Mean Subtracted Patch Groups
GMM
SVD
Dictionaries
Singular Values
External Prior Learning Stage
Subspace Assignment
Guided Dictionary Learning
Weighted Sparse Coding
Input Noisy Image
Mean Subtracted Patch Groups
Dictionaries
Recovered Patch Groups
Aggregation
Recovered Image
Guided Internal Prior Learning and Denoising Stage

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

3.1. Learn External Patch Group Priors

The nonlocal self-similarity based patch group (PG) [?] has proved to be a very effective unit for image prior learning. In this work, we extract PGs from natural clean images to learn 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 patches to this local patch (including the local patch itself) in a $W \times W$ local 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 $\{\mathbf{x}_m\}_{m=1}^{M}$. 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 we have extracted a number of N PGs from a set of external natural images, and the n-th PG is $\overline{\mathbf{X}}_n \triangleq \{\overline{\mathbf{x}}_{n,m}\}_{m=1}^{M}, n = 1, ..., N$. A Gaussian Mixture Model (GMM) is learned to model the PG prior. The overall log-likelihood function is

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

The learning process is similar to the GMM learning in [?, ?]. 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) on the covariance matrix:

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

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

3.2. Guided Internal Prior Learning

After the external PG prior is learned on external natural clean images, we employ it to guide the internal PG prior learning for a given real noisy image. The guidance lies in two aspects. First, the external prior will guide the subspace assignments of internal noisy PGs. Second, the external prior will guide the orthogonal dictionary learning of internal noisy PGs.

3.2.1 Internal Subspace Assignment

Given a real noisy image $\mathbf{y}$, we extract N (overlapped) 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 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 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 $\overline{\mathbf{Y}}_n \triangleq \{\overline{\mathbf{y}}_{n,m}\}_{m=1}^{M}$.

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 the noisy PG $\overline{\mathbf{Y}}_n$ 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:

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

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

3.2.2 Guided Orthogonal Dictionary Learning

Assume that we have assigned all the internal noisy PGs $\{\overline{\mathbf{Y}}_n\}_{n=1}^{N}$ to their corresponding most suitable subspaces in $\{\mathcal{N}(\mathbf{0},\mathbf{\Sigma}_k)\}_{k=1}^{K}$. For the k-th subspace, the noisy PGs assigned to it are $\{\overline{\mathbf{Y}}_{k_n}\}_{n=1}^{N_k}$ where $\overline{\mathbf{Y}}_{k_n} = [\overline{\mathbf{y}}_{k_n,1}, ..., \overline{\mathbf{y}}_{k_n,M}]$ and $\sum_{k=1}^{K} N_k = N$. We propose to learn an orthogonal dictionary $\mathbf{D}_k$ from each set of PGs $\overline{\mathbf{Y}}_{k_n}$ to characterize the internal PG prior with the guidance of the corresponding external orthogonal dictionary $\mathbf{U}_k$ (Eq. (**??**)). The reasons that we learn orthogonal dictionaries are two-fold. Firstly, the PGs $\{\overline{\mathbf{Y}}_{k_n}\}_{n=1}^{N_k}$ 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* [**?**]. Secondly, the orthogonality of dictionary can make the encoding in the testing stage very efficient, leading to an efficient denoising algorithm (please refer to subsection 3.3 for more details).

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

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

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

where $\mathbf{I}$ is the $3p^2$ dimensional identity matrix, $\boldsymbol{\alpha}_{n,m}$ is the sparse coding 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

$$\lambda_j = \lambda/(\sqrt{\mathbf{S}_k(j)} + \varepsilon), \tag{5}$$

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

We employ an alternating iterative approach to solve the optimization problem (**??**). Specifically, we initialize the orthogonal dictionary as $\mathbf{D}^{(0)} = \mathbf{U}_k$ and for $t = 0, 1, ..., T-1$, we alternatively update $\boldsymbol{\alpha}_{n,m}$ and $\mathbf{D}_\text{I}$ as follows.

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

$$\boldsymbol{\alpha}_{n,m}^{(t)} := \arg\min_{\boldsymbol{\alpha}_{n,m}} \|\overline{\mathbf{y}}_{n,m} - \mathbf{D}^{(t)}\boldsymbol{\alpha}_{n,m}\|_2^2 + \sum_{j=1}^{3p^2}\lambda_j|\boldsymbol{\alpha}_{n,m,j}|. \tag{6}$$

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

$$\boldsymbol{\alpha}_{n,m}^{(t)} = \text{sgn}((\mathbf{D}^{(t)})^\top\overline{\mathbf{y}}_{n,m}) \odot \max(|(\mathbf{D}^{(t)})^\top\overline{\mathbf{y}}_{n,m}| - \boldsymbol{\lambda}, \mathbf{0}), \tag{7}$$

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

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

$$\begin{aligned}\mathbf{D}_\text{I}^{(t+1)} &:= \arg\min_{\mathbf{D}_\text{I}} \sum_{n=1}^{N}\sum_{m=1}^{M}\|\overline{\mathbf{y}}_{n,m} - \mathbf{D}\boldsymbol{\alpha}_{n,m}^{(t)}\|_2^2\\ &= \arg\min_{\mathbf{D}_\text{I}} \|\mathbf{Y} - \mathbf{D}\mathbf{A}^{(t)}\|_F^2\\ s.t.\quad & \mathbf{D} = [\mathbf{D}_\text{E}\ \mathbf{D}_\text{I}],\ \mathbf{D}_\text{I}^\top\mathbf{D}_\text{I} = \mathbf{I}_{(3p^2-r)},\ \mathbf{D}_\text{E}^\top\mathbf{D}_\text{I} = \mathbf{0},\end{aligned} \tag{8}$$

where $\mathbf{A}^{(t)} = [\boldsymbol{\alpha}_{1,1}^{(t)}, ..., \boldsymbol{\alpha}_{1,M}^{(t)}, ..., \boldsymbol{\alpha}_{N,1}^{(t)}, ..., \boldsymbol{\alpha}_{N,M}^{(t)}]$ and $\mathbf{I}_{(3p^2-r)}$ is the $(3p^2-r)$ dimensional identity matrix. The sparse coefficients matrix can be written as $\mathbf{A}^{(t)} = [(\mathbf{A}_\text{E}^{(t)})^\top\ (\mathbf{A}_\text{I}^{(t)})^\top]^\top$ where the external part $\mathbf{A}_\text{E}^{(t)} \in \mathbb{R}^{r\times NM}$ and the internal part $\mathbf{A}_\text{I}^{(t)} \in \mathbb{R}^{(3p^2-r)\times NM}$ represent the coding coefficients of $\mathbf{Y}$ over external sub-dictionary $\mathbf{D}_\text{E}$ and internal sub-dictionary $\mathbf{D}_\text{I}^{(t)}$, respectively. According to the Theorem 4 in [**?**], the problem (**??**) has a closed-form solution $\mathbf{D}_\text{I}^{(t+1)} = \mathbf{U}_\text{I}\mathbf{V}_\text{I}^\top$, where $\mathbf{U}_\text{I} \in \mathbb{R}^{3p^2\times(3p^2-r)}$ and $\mathbf{V}_\text{I} \in \mathbb{R}^{(3p^2-r)\times(3p^2-r)}$ are the orthogonal matrices obtained by the following SVD

$$(\mathbf{I} - \mathbf{D}_\text{E}\mathbf{D}_\text{E}^\top)\mathbf{Y}(\mathbf{A}_\text{I}^{(t)})^\top = \mathbf{U}_\text{I}\mathbf{S}_\text{I}\mathbf{V}_\text{I}^\top. \tag{9}$$

```
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 $\{\mathbf{Y}_n\}_{n=1}^{N}$ 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 assignment via Eq. (??);
   end for
   for the PGs in each subspace do
4.   External PG prior guided internal orthogonal
     dictionary learning by solving (??);
5.   Recover each patch in all PGs via Eq. (??);
   end for
6. Aggregate the recovered PGs of all subspaces to form
   the recovered image $\hat{\mathbf{x}}^{(Ite)}$;
end for
```

The orthogonality of internal sub-dictionary $\mathbf{D}_{\mathrm{I}}^{(t+1)}$ can be checked by $(\mathbf{D}_{\mathrm{I}}^{(t+1)})^{\top}(\mathbf{D}_{\mathrm{I}}^{(t+1)}) = \mathbf{V}_{\mathrm{I}}\mathbf{U}_{\mathrm{I}}^{\top}\mathbf{U}_{\mathrm{I}}\mathbf{V}_{\mathrm{I}}^{\top} = \mathbf{I}_{(3p^2-r)}$.

3.3. 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 $\{\hat{\boldsymbol{\alpha}}_{n,m}^{(T-1)}\}$ in Eq. (**??**) and the orthogonal dictionary $\mathbf{D}^{(T)} = [\mathbf{D}_{\mathrm{E}}\ \mathbf{D}_{\mathrm{I}}^{(T)}]$ in Eq. (**??**), the latent clean patch $\hat{\mathbf{y}}_{n,m}$ of the m-th noisy patch in PG $\mathbf{Y}_n$ is reconstructed as

$$\hat{\mathbf{y}}_{n,m} = \mathbf{D}^{(T)}\hat{\boldsymbol{\alpha}}_{n,m}^{(T-1)} + \boldsymbol{\mu}_n, \tag{10}$$

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 Alg. 1.

4. Experiments

We evaluate the performance of the proposed algorithm on real-world noisy images [**?**, **?**] in comparison with state-of-the-art denoising methods [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**].

4.1. Implementation Details

Our proposed method has two stages: the external prior learning stage and the external prior guided internal prior learning stage. In the first stage, we set $p = 6$ (the patch size), $M = 10$ (the number of similar patches in a PG), $W = 31$ (the window size for PG searching) and $K = 32$ (the number of Gaussian components in GMM). 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.

Figure 3. Some samples cropped from real noisy images of [**?**].

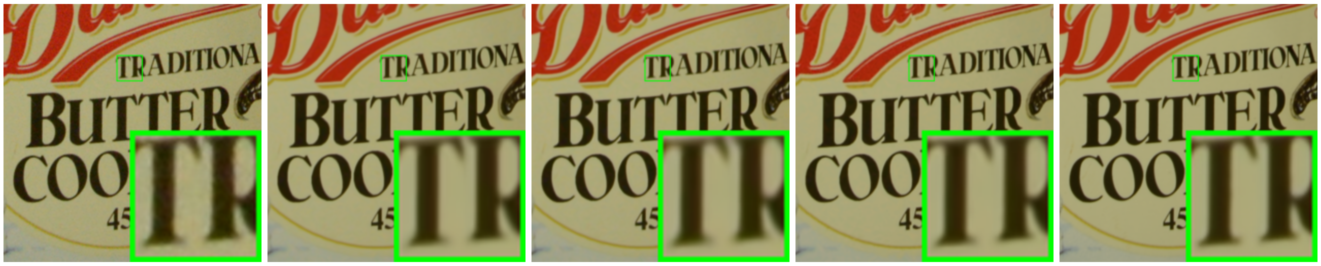
In the second stage, we set $r = 54$ (the number of atoms in the external sub-dictionaries); that is, we let the external sub-dictionaries have the same number of atoms as the internal sub-dictionaries to be learned. Our experiments show that setting r between 27 and 81 will lead to very similar results. For other parameters, we set $\lambda = 0.001$ (the sparse regularization parameter), $T = 2$ (the number of iterations for solving problem (**??**)), and $IteNum = 4$ (the number of iterations for Alg.1). All parameters of our method are fixed to 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.

4.2. The Testing Datasets

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

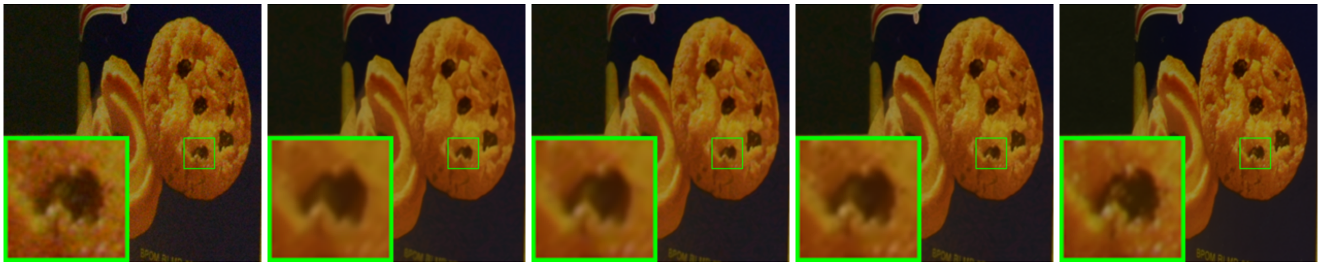
The first dataset is provided in [**?**], which includes 20 real noisy images collected under uncontrolled outdoor environment. Since there is no "ground truth" of the noisy images, the objective measures such as PSNR cannot be computed on this dataset.

The second dataset is provided in [**?**], which includes noisy images of 17 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 can be computed. Since the image size is very large (about 7000 × 5000) and the 17 scenes share repetitive contents, the authors of [**?**] cropped 15 smaller images (of size 512 × 512) to perform experiments. To more comprehensively evaluate the proposed methods, we cropped 60 images of size 500 × 500 from the dataset for experiments. Some samples are shown in Fig. **??**. Note that the noise in our cropped 60 images is different from the noise in the 15 images cropped by the authors of [**?**] since they are from different shots.



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

Figure 4. Denoised images of a region cropped from the real noisy image "Nikon D600 ISO 3200 C1" [?] by different methods. The images are better to be zoomed in on screen.



(a) Noisy [?]: 33.77dB (b) External: 36.97dB (c) Internal: 37.40dB (d) Guided Internal: **38.01**dB (e) Mean Image [?]

Figure 5. Denoised images of a region cropped from the real noisy image "Nikon D600 ISO 3200 C1" [?] by different methods. The images are better to be zoomed in on screen.

4.3. Comparison among external, internal and guided internal priors

To demonstrate the advantages of external prior guided internal prior learning, in this section we perform real noisy image denoising by methods using external priors (denoted by "External"), internal priors (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. (**??**)) for denoising. For the "Internal" method, the overall framework is similar to the method of [?]. 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. (**??**) to perform denoising. All parameters of the "External" and "Internal" methods are tuned to achieve their corresponding best performance.

We compare the three methods mentioned above on the 60 cropped images from [?]. The average PSNR and run time are listed in Table **??**. 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. Fig. **??** and Fig. **??** show the denoised images of two noisy images by the three methods. We can see that the "External" method is good at recovering large-scale structures (see Fig. **??**) while the "Internal" method is good at recovering fine-scale textures (see Fig. **??**). 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.

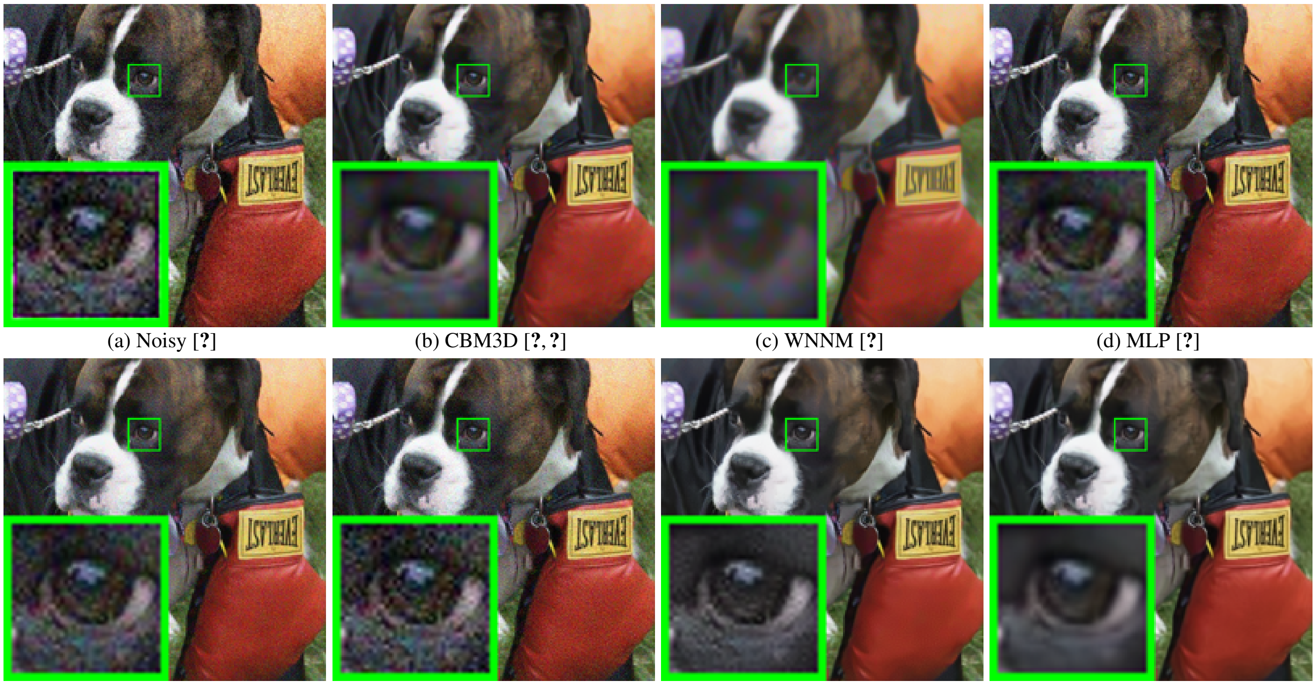
4.4. Comparison with State-of-the-Art Denoising Methods

We compare the proposed method with state-of-the-art image denoising methods, including CBM3D [?, ?], WNNM [?], MLP [?], CSF [?], TNRD [?], Noise Clinic (NC) [?, ?], Neat Image (NI) [?], and Cross-Channel (CC) [?]. Among them, CBM3D, WNNM, MLP, CSF and TNRD are designed based on Gaussian noise model, and they need to know the noise level for denoising. We use the method in [?] to estimate the noise level for them. All the other parameters in these methods are set as the default ones. Since WNNM, MLP, CSF and TNRD are designed for grayscale images, we use them to denoise the R, G, B channels separately for color noisy images.

Like our method, the NC is a blind image denoising method which does not need any noise prior. The NI is a 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 [?]. Therefore, we only compare with it on the 15 cropped images from [?].

4.4.1 Results on Dataset [?]

Since there is no "ground truth" for the real noisy images in dataset [?], we only compare the visual quality of the denoised images by different methods. (Note that method CC [?] is not compared since its code is not available.) Fig.


(a) Noisy [?] (b) CBM3D [?, ?] (c) WNNM [?] (d) MLP [?]

(e) TNRD [?] (f) NI [?] (g) NC [?, ?] (h) Ours

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

Table 1. Average PSNR (dB) and Run Time (seconds) of the "External", "Internal", and "Guided Internal" methods on 60 real noisy images (of size 500 × 500 × 3) cropped from [?].

	Noisy	External	Internal	Guided Internal
PSNR	34.51	38.21	38.07	**38.75**
Time	—	**39.57**	587.36	41.89

?? shows the denoised images of "Dog". It can be seen that CBM3D and WNNM tend to over-smooth much the image while remaining some noise caused color artifacts. MLP and TNRD are likely to remain many noise-caused color artifacts across the whole image. These results demonstrate that the methods designed with Gaussian noise model are not effective for real noise removal. Though NC and NI methods are specifically developed for real noisy images, their performance on noise removal is not very satisfactory. In comparison, our proposed method recovers much better the structures and textures (such as the eye area) than the other competing methods. More visual comparisons on this dataset [?] can be found in the supplementary file.

4.4.2 Results on Dataset [?]

As described in section 4.2, there is a mean image for each of the 17 scenes used in dataset [?], 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 [?]. The PSNR results of CBM3D [?], WNNM [?], MLP [?], CSF [?], TNRD [?], NC [?, ?], NI [?] and CC [?] are listed in Table **??** (The results of CC are copied from the original paper [?]). The best and second best PSNR results of each image are highlighted in red and blue, respectively. One can see that on 9 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 0.28dB PSNR improvements over [?] and much higher PSNR gains over other competing methods. Fig. **??** shows the denoised images of a scene captured by Canon 5D Mark 3 at ISO = 3200. We can see that CBM3D, WNNM, NC, NI and CC would either remain noise or generate artifacts, while MLP, TNRD over-smooth much the image. By using the external prior guided internal priors, our proposed method preserves edges and textures better than other methods, leading to visually pleasant outputs. More visual comparisons can be found in the supplementary file.

We then perform denoising experiments on the 60 images we cropped from [?]. The average PSNR results are listed in Table **??** (CC is not compared since the code of [?] is not available). The best result is highlighted in bold. Again, our proposed method achieves much better PSNR results than the other methods. The improvement of our method over the second best method (TNRD) is about 1dB. Due to the space limitations, the visual comparisons are pro-

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

Camera Settings	**Noisy**	**CBM3D**	**WNNM**	**MLP**	**CSF**	**TNRD**	**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

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

Methods	**CBM3D**	**WNNM**	**MLP**	**CSF**
PSNR	34.58	34.52	36.19	37.40
Methods	**TNRD**	**NI**	**NC**	**Ours**
PSNR	37.75	36.53	37.57	**38.75**

vided in the supplementary file.

5. Conclusion

The noise in real-world noisy images is much more complex than additive white Gaussian noise and hard to be modeled by simple analytical distributions such as Gaussian or mixture of Gaussians, making the real noisy image denoising problem very challenging. In this paper, we proposed a simple yet effective solution for real noisy image denoising without explicitly assuming certain noise models. Specifically, we firstly learned image priors from external clean images, and then employed the learned external priors to guide the learning of internal priors from the given noisy image. The learned hybrid priors are utilized for real noisy image denoising. Experiments on two real noisy image datasets, where the images were captured under indoor or outdoor lighting conditions by different types of cameras and camera settings, demonstrated that our proposed method achieved much better performance than state-of-the-art image denoising methods.

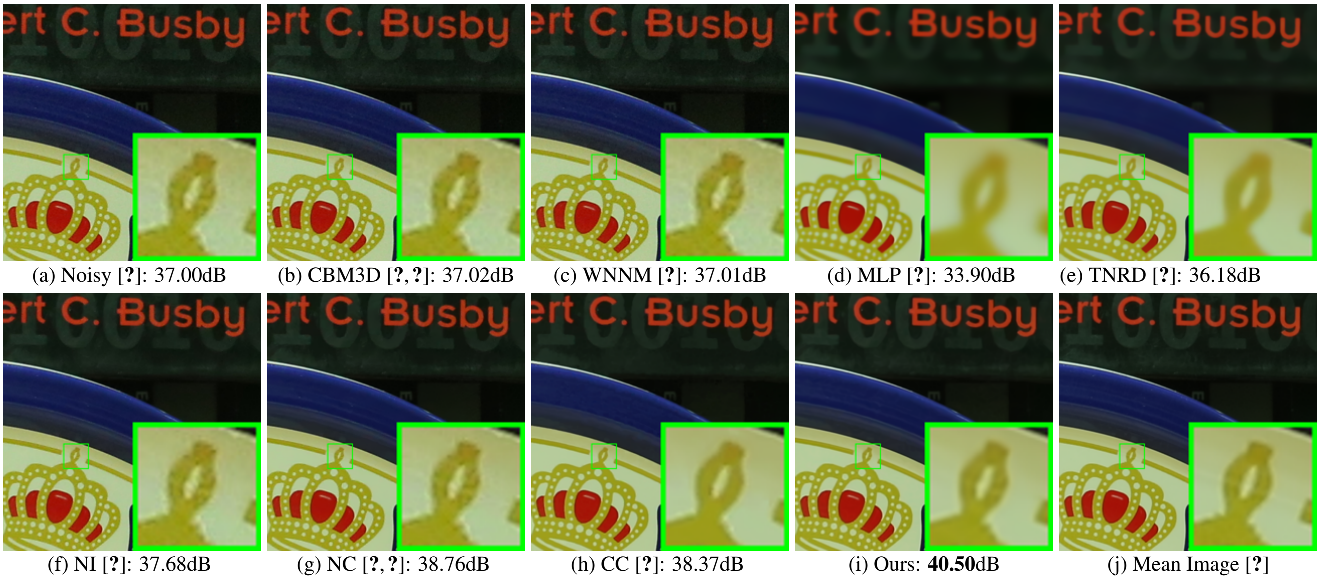


Figure 7. Denoised images of a region cropped from the real noisy image "Canon 5D Mark 3 ISO 3200 1" [?] by different methods. The images are better to be zoomed in on screen.