https://doi.org/10.1016/j.image.2021.116299

# Bayesian Sparse Hierarchical Model for Image Denoising

Jun Xiao, Rui Zhao, Kin-Man Lam*

*Department of Electronic and Information Engineering,*

*The Hong Kong Polytechnic University, Hong Kong*

## Abstract

Sparse models and their variants have been extensively investigated, and have achieved great success in image denoising. Compared with recently proposed deep-learning-based methods, sparse models have several advantages: 1). Sparse models do not require a large number of pairs of noisy images and the corresponding clean images for training. 2). The performance of sparse models is less reliant on the training data, and the learned model can be easily generalized to natural images across different noise domains. In sparse models, $\ell_0$ norm penalty makes the problem highly non-convex, which is difficult to be solved. Instead, $\ell_1$ norm penalty is commonly adopted for convex relaxation, which is considered as the Laplacian prior from the Bayesian perspective. However, many previous works have revealed that $\ell_1$ norm regularization causes a biased estimation for the sparse code, especially for high-dimensional data, e.g., images. In this paper, instead of using the $\ell_1$ norm penalty, we employ an improper prior in the sparse model and formulate a hierarchical sparse model for image denoising. Compared with other competitive methods, experiment results show that our proposed method achieves a better generalization for images with different characteristics across various domains, and achieves state-of-the-art performance for image denoising on several benchmark datasets.

*Keywords:* Image denoising, Sparse model, Bayesian hierachical prior

[1] Email: jun-gwansiu.xiao@connect.polyu.hk (J.Xiao), rick10.zhao@connect.polyu.hk (R.Zhao), enkmlam@polyu.edu.hk (K.-M.Lam)

*Preprint submitted to Journal of Signal Processing: Image CommunicationDecember 21, 2020*

## 1. Introduction

Noise is unavoidably introduced into modern imaging systems due to camera sensor settings and other hardware issues. Therefore, algorithms for noise removal are indispensable for obtaining high-quality images in an imaging system. Image denoising is one of the fundamental low-level tasks in computer vision, and has been extensively studied by researchers in the past. In the early stages, researchers focused on the synthesized noise, such as additive white Gaussian noise (AWGN) [1–7], Poisson noise [8–10], and mixed Poisson-Gaussian noise [11–15]. However, the assumption of synthesized noise is too ideal, and those proposed methods based on the synthesized noise hardly achieve satisfactory performance in realistic noisy images. Realistic noise is spatially variant and channel correlated, so it is much more challenging than synthesized noise. In recent years, many denoising methods [16–19] for real-world noisy images have been proposed to handle with realistic noise.

Image denoising aims to obtain a clean image $\boldsymbol{x}$ from its noisy observation $\boldsymbol{y}$. Even though noise has different characteristics, we commonly assume the forward imaging process can be modeled as a linear system, i.e. $\boldsymbol{y} = \boldsymbol{x}+\boldsymbol{\sigma}$, where $\boldsymbol{\sigma}$ denotes the noise term. In fact, this linear system is underdetermined, and thus image-denoising problems have an intrinsically ill-posed property. There are infinite possible solutions for the image-denoising problems. From the Bayesian perspective, the clean image $\boldsymbol{x}$ can be obtained by maximizing a posterior distribution, which is to find the maximum mode of the posterior distribution. Based on the Bayesian theorem, we have

$$p(\boldsymbol{x}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x}). \tag{1}$$

By applying the logarithm to both sides of Eq. (1), it can be rewritten as

$$\log p(\boldsymbol{x}|\boldsymbol{y}) \propto \log p(\boldsymbol{y}|\boldsymbol{x}) + \log p(\boldsymbol{x}), \tag{2}$$

where $p(\boldsymbol{x}|\boldsymbol{y})$ is the posterior distribution of the clean image $\boldsymbol{x}$, $p(\boldsymbol{y}|\boldsymbol{x})$ represents the forward imaging process, which is usually fixed in the image-denoising problem, and $p(\boldsymbol{x})$ denotes the prior knowledge of the clean image $\boldsymbol{x}$. Therefore, we can see that the image prior is crucial for designing a noise-removal algorithm, and many algorithms have been proposed to better exploit image priors for the image-denoising problem. For example, the well-known bilateral filter [20] utilizes spatial information and intensity information simultaneously, yielding a promising result for image denoising. In general, image-denoising approaches can be divided into two categories, i.e. learning-based methods and model-based methods.

Generative learning-based methods and discriminative learning-based methods are the two major learning-based approaches. Generative learning refers to learning image priors from either an external, clean image dataset [21] or given noisy images [4], and the learned prior is utilized to perform image denoising. Recently, researchers have turned their attention to the deep convolutional neural network (CNN), because CNN-based methods have witnessed many unprecedented successes in the computer-vision fields, due to the significant learning capacity of CNN. CNN-based methods learn the image priors from given pairs of noisy images and the corresponding clean images, and perform image denoising simultaneously [22–24]. Given pairs of noisy and clean images, CNN-based methods learn a highly nonlinear function to map degraded images to clean images. This learning strategy is viewed as discriminative learning. Specifically, DnCNN [16] is a competitive approach, based on CNN, which demonstrates the effectiveness of residual learning and batch normalization. To address the speed issue and obtain further improved performance, FFDNet [17] was proposed to concatenate the noise-level map with the noisy image to form the input of a CNN. This approach achieved state-of-the-art performance for image denoising, in the presence of both AWGN and realistic noise. CBDNet [18] is a two-stage denoising network, which consists of a noise estimator and a denoiser. CBDNet first simulates the noise by training a network to function as the noise estimator, whose output is a noise-level map. This noise-level map represents the noise information, which is concatenated with the corresponding noisy image to feed into the denoiser. The performance is further improved by using the FFDNet. However, the performance of the CNN-based models highly relies on the training data, and the performance becomes poor on other noise types that have not been considered in the training process. This leads to limited capacities for generalization and flexibility.

Instead of learning the prior from training data, model-based methods investigate the intrinsic characteristics of natural images, such as the nonlocal self-similarity and the sparsity, which can be utilized as useful image priors. Compared with CNN-based methods, model-based approaches have several advantages: 1). They do not require a large number of pairs of noisy images and clean images for training. 2). Their performances are robust and have good generalization ability across different domains, without requiring any fine-tuning techniques. 3). They are efficient and can achieve real-time performance if fast algorithms are adopted, or the analytical solution exists. The sparse model or low-rank model has been widely used in image-restoration problems [25–35]. The sparse model assumes that the main energy of an image is distributed sparsely in some transformed domains, such as wavelet domain [36, 37], curvelet domain [2], or a generalized over-complete dictionary [5]. Most of the existing sparse representation models are essentially based on the $l_0$ norm regularization. However, the $l_0$ norm penalty leads to a highly non-convex property, and solving the $l_0$ norm regularization problem is NP-hard. Therefore, the $l_1$ norm penalty is commonly adopted for convex relaxation in the sparse model, while the sparsity of the solution is preserved. Combined with the nonlocal self-similarity property, Zhang *et al.* [3] proposed to encode similar image patches as atoms in the dictionary, and this gives rise to a better result. Besides that, Gu *et al.* [38] characterized the nonlocal self-similarity property of images by minimizing a weighted nuclear norm model (WNNM), and achieved state-of-the-art performance for AWGN denoising. Furthermore, to focus on the statistical properties of the noise in different channels, Xu *et al.* [39] adopted different weights in the respective RGB channels in WNNM, called multi-channel WNNM (MCWNNM), and improved the performance of WNNM. Xu *et al.* [40] considered noise, with different characteristics, in the RGB channels, and proposed a trilateral weighted scheme for the sparse coding model (TWSC), leading to state-of-the-art performance for denoising AWGN and realistic noise removal. All the methods above mentioned are based on using a soft-threshold operator. However, many previous works [41–43] have shown that using a soft-threshold operator may result in forming a biased estimator, and this biased estimator cannot achieve satisfactory results for high-dimensional data.

In this paper, we follow the Bayesian analysis and adopt the improper prior distribution for image denoising, leading to a Bayesian hierarchical sparse model. The major contributions of this paper are summarized as follows:

1. Instead of setting the threshold values as a hyper-parameter, our proposed prior distribution treats it as a random variable and is affected by another underlying distribution.
2. Our proposed method imposes different threshold values and feature-selection ranges onto denoised images. Moreover, we propose an adaptive weight-updating scheme for image denoising.
3. Compared with the other state-of-the-art models, our proposed Bayesian hierarchical sparse model with the adaptive weighting strategy, can achieve state-of-the-art performance for image denoising.

The rest of this paper is organized as follows. In Section 2, related works will be reviewed. In Section 3, our proposed method will be introduced in detail, and the experimental results are shown in Section 4. The conclusion is given in Section 5.

## 2. Related Works

Over the past decades, sparse models and their variants have shown their effectiveness for image denoising. The sparse model assumes that given the observed corrupted image $\boldsymbol{y}$, the goal is to recover a latent clean image $\boldsymbol{x}$, which can be encoded by using a dictionary $\boldsymbol{D}$ with the corresponding sparse code $\boldsymbol{\theta}$. Hence, the problem of image denoising based on the sparse model can be formulated as a $l_0$ regularized problem, as follows:

$$\min_{\boldsymbol{\theta},\boldsymbol{D}} \|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\theta}\|_2^2 + \|\boldsymbol{\theta}\|_0. \tag{3}$$

From Eq. (3), we can see that the image-denoising problem is composed of two subproblems: 1). dictionary learning, and 2). sparse-code estimation.

Dictionary learning aims to learn an effective dictionary, a transformed domain of the training data, such that a latent, clean image can be effectively represented by a linear combination of a few atoms of the dictionary. To solve this problem, the K-SVD algorithm was proposed in [5], which generalizes the K-mean clustering method to update the sparse code and the dictionary alternatively. However, the method is very time-consuming and cannot be generalized to a large-scale dataset. In [44], an online model was proposed, which can solve these two subproblems efficiently. In [45], a tree structure is adopted for sparse representation, so the computational complexity is reduced by a great margin. Instead of learning a dictionary to adapt the image domain each time, [3, 44] utilized the nonlocal self-similarity property of natural images for encoding similar image patches as atoms of the dictionary, and achieved promising results for image denoising. Xu *et al.* [4] proposed to use the Gaussian mixture model (GMM) to describe the statistical properties of image patches, and the dictionary is learned from the covariate matrix of the GMM model by performing singular value decomposition (SVD). Xu *et al.* [21] proposed a hybrid dictionary learning method for image denoising. A dictionary, based on the external image prior, is learned from external datasets of clean images, while the other dictionary, based on the internal image prior, is learned from the noisy input image, under the guidance of the external image prior.

Apart from the dictionary learning problems, another issue of the sparse model is how to accurately estimate the sparse code given a dictionary. The $l_0$ regularization gives rise to the solution of the sparsity, but solving the $l_0$ regularized problem is equivalent to solving the best subset-selection problem. To the best of our knowledge, the best-subset selection problem is NP-hard. Thus, it is impractical to employ $\ell_0$ regularization in the sparse model for real-world applications. Instead of $\ell_0$ norm penalty, some tractable penalty functions have been proposed as alternatives of the $l_0$ norm penalty, while the sparsity of the solution can still be preserved. The $l_1$ norm penalty is one well-known regularization, and commonly adopted as convex relaxation of the $l_0$ norm penalty in the sparse model [46]. The $\ell_1$ is convex, and it makes the solution of the regularized problem more tractable. Given a fixed dictionary, solving the $l_1$ regularized problem is equivalent to solving the problem of least absolute shrinkage and selection operator (LASSO) problem. Besides that, many algorithms have been developed to solve this problem efficiently, such as LARS [47], coordinate-descent algorithm [48], etc. One advantage of considering the $\ell_1$ regularized problem is that an analytical solution can be obtained by performing the soft threshold operator, when the dictionary is orthogonal. Unfortunately, the LASSO estimator suffers from several drawbacks. In particular, a LASSO estimator over-penalizes the estimated value in the far region, and causes a biased estimation of the sparse code [41–43], leading to degraded performance. To solve such an issue in the $\ell_1$ regularized image-denoising problem, a weighted sparse model [4] is adopted, and the sparse code is obtained by weighted shrinkage. This weighted scheme is based on the significance of similar image patches. By considering the realistic noise with characteristics of the spatial variance and the channel variance, a trilateral weighted scheme for the sparse model was proposed in [40], and it gives rise to a state-of-the-art performance for image denoising. In order to achieve a better approximation for the $l_0$ norm penalty, some non-convex penalty functions have been proposed, such as the smoothly clipped absolute deviation (SCAD) penalty [49], the log penalty [50], and **the minimax concave penalty (MCP) [51]. In particular, MCP has shown its effectiveness when it approximates th $L_0$ norm, which is defined as follows:**

$$p(t;\lambda) = \lambda \int_0^t (1 - \frac{x}{\lambda\gamma})_+ \mathrm{d}x \tag{4}$$

**where the regularization parameter $\gamma > 0$ and $(x)_+ = max(0, x)$. MCP is a highly non-convex function, but enjoys a better sparsity property,**

**compared with the $\ell_1$ norm.** The sparse model with MCP has achieved a promising performance, compared to other non-convex penalties and the $l_1$ penalty. However, this method cannot be generalized to a large-scale image dataset, because it is not computationally efficient. Moreover, it does not consider the noise with spatial variances, and adopts the same threshold for all the patches of an image in the denoising process, which is not suitable for realistic noise.

In this paper, we follow the Bayesian analysis like previous works, but we adopt a hierarchical improper prior in the sparse model, leading to a Bayesian sparse hierarchical model for image denoising. This hierarchical structure of the sparse model allows us to impose an adaptive weight strategy on different images based on the noise characteristics within the image. Additionally, we show that such a hierarchical sparse model with adaptive weight is equivalent to a generalized MCP regularized model. Our proposed method is different from [51] in two aspects: 1) The threshold value and the selection range of the penalty function in [51] remain constant in the denoising process, but they are varying in our proposed method, and adaptive in the denoising process according to the noise characteristics within an image. 2) We show that our proposed prior term is more general, and the $l_1$ norm penalty function and MCP are the special cases of our proposed method. The details of our proposed method are given in Section 3.

## 3. The Proposed Method

In this section, we introduce our proposed Bayesian sparse hierarchical model in detail. First, we consider the image-denoising problem as the MAP problem, and the assumptions behind our proposed method will be described. Then, dictionary learning, combined with the nonlocal self-similarity property, will be introduced. After that, the proposed hierarchical improper prior, adopted in the sparse model, will be described. The image-denoising problem is formulated as the MAP problem of a scale mixture of normal distributions under the Bayesian analytical framework. Finally, the overall denoising algorithm will be described.

*3.1. Problem Formulation*

Our proposed sparse model is learned on image patches, and the local noise contained in an image local patch is assumed to be approximated by AWGN, i.e. a local noisy image patch $\boldsymbol{y}_n \sim \mathcal{N}(\boldsymbol{x}_n, \sigma_n^2)$, where $\boldsymbol{x}_n$ is its corresponding clean image patch, and $\sigma_n^2$ denotes the noise level of the $n$-th image local patch. In addition, the clean image patch $\boldsymbol{x}_n$ is assumed to be encoded by a sparse code $\boldsymbol{\theta}$ based on a dictionary $\boldsymbol{D}$, i.e. $\boldsymbol{x}_n = \boldsymbol{D}\boldsymbol{\theta}_n$. Therefore, we have the statistical model of the local noisy image patch, i.e. $\boldsymbol{y}_n \sim \mathcal{N}(\boldsymbol{D}\boldsymbol{\theta}_n, \sigma_n^2)$. We aim to estimate the clean image patch $\boldsymbol{x}_n$ based on the Bayesian perspective. That is

$$\begin{aligned} p(\boldsymbol{x}_n|\boldsymbol{y}_n) &= \frac{p(\boldsymbol{y}_n|\boldsymbol{x}_n)p(\boldsymbol{x}_n)}{p(\boldsymbol{y}_n)} \\ &\propto p(\boldsymbol{y}_n|\boldsymbol{x}_n)p(\boldsymbol{x}_n). \end{aligned} \tag{5}$$

We take the logarithm of both sides of Eq. (5), leading to

$$\log p(\boldsymbol{x}_n|\boldsymbol{y}_n) \propto \log p(\boldsymbol{y}_n|\boldsymbol{x}_n) + \log p(\boldsymbol{x}_n). \tag{6}$$

Based on the assumption, the above objective function for determining $\boldsymbol{x}_n = D\boldsymbol{\theta}_n$ can be rewritten as follows:

$$\min_{\boldsymbol{D},\boldsymbol{\theta}_n} \|\boldsymbol{y}_n - \boldsymbol{D}\boldsymbol{\theta}_n\|_2^2 + \log p(\boldsymbol{\theta}_n). \tag{7}$$

We can see that the dictionary $\boldsymbol{D}$ and the prior knowledge $p(\boldsymbol{\theta}_n)$ are the key elements in Eq. (7). **The Laplacian distribution is widely used prior knowledge in most of the existing sparse models, which is defined as:**

$$p(\boldsymbol{\theta}_n|\lambda) = \left(\frac{\lambda}{2}\right)^d \exp\left(-\lambda \sum_{i=1}^{d} |\theta_i|\right), \tag{8}$$

**where $d$ denotes the dimension of the sparse code, and $\lambda$ is regarded as a known hyperparameter. However, this is not suitable for realistic noise, because it is spatially variant. Rather than treating $\lambda$ to be**

**fixed in the denoising process, which causes biased estimation, we maximize the conditional joint prior distribution of $\boldsymbol{\theta}_n$ and $\lambda$. In the following the section, we will explicitly describe how to learn the dictionary and the prior term adopted in our proposed method.**

### *3.2. Dictionary Learning*

Given a noisy color image $\boldsymbol{I}_y \in \mathbb{R}^{H\times W\times 3}$, where $H$ and $W$ represent the height and width of the image, respectively, an image local patch is extracted from the noisy image $\boldsymbol{I}_y$ with the size of $p \times p \times 3$, with a patch size of $p \times p$. Image local patches are densely extracted with the step size of $s$, and then each image local patch is stretched into a vector. Let $\boldsymbol{y}_n \in \mathbb{R}^{3p^2}$ denote the $n$-th image local patch vector. For each local patch, the $N$ most similar image patches are searched around the current local patch based on a window size of $W' \times W'$. For each current image local patch, $N$ similar patches form a patch group (PG), denoted as $\boldsymbol{Y} = \{\boldsymbol{y}_n\}_{n=1}^{N}$. For each PG, each patch is subtracted by its mean vector $\boldsymbol{\mu}_n = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{y}_n$, to form a mean subtracted PG $\bar{\boldsymbol{Y}}$, which is defined as $\bar{\boldsymbol{Y}} = \{\bar{\boldsymbol{y}}_n = \boldsymbol{y}_n - \boldsymbol{\mu}_n\}_{n=1}^{N}$.

**Assume that we have obtained $M$ mean-subtracted PGs from the given noisy image, and the $m$-th mean-subtracted PG is denoted as $\bar{\boldsymbol{Y}}_m = \{\bar{\boldsymbol{y}}_{n,m}\}_{n=1}^{N}, m = 1, 2, \cdots, M$. In order to better describe the statistical property of each PG, singular value decomposition (SVD) is applied to each mean-subtracted PG $\bar{\boldsymbol{Y}}_m$, as follows:**

$$\bar{\boldsymbol{Y}}_m = \boldsymbol{U}_m \boldsymbol{S}_m \boldsymbol{V}_m^T, \tag{9}$$

**where $\boldsymbol{U}_m \in \mathbb{R}^{3p^2\times 3p^2}$ is the left eigenvector matrix, $\boldsymbol{V}_m \in \mathbb{R}^{N\times N}$ is the right eigenvector matrix, and $\boldsymbol{S}_m \in \mathbb{R}^{3p^2\times N}$ is the diagonal matrix of singular values. The columns of the matrix $\boldsymbol{U}_m$ are the principal components of the local patches of the $m$-th PG, and sorted in descending order according to their corresponding eigenvalues. The matrix $\boldsymbol{U}_m$ projects a mean-subtracted image local patch into a low-dimensional space to form a compact representation, and thus is adopted as the**

**dictionary in our proposed method. This is then used to estimate the sparse code of an image local patch in the patch group. The singular values represent the significance of the corresponding eigenvectors in $\boldsymbol{U}_m$, and are used as the prior in our proposed sparse model.**

### *3.3. Hierarchical Prior*

**For estimating the high-dimensional sparse code $\boldsymbol{\theta} \in \mathbb{R}^{3p^2}$, we use the conditional joint probability in $\boldsymbol{\theta}$ and $\boldsymbol{\gamma}$, where $\boldsymbol{\gamma}$ is a parameter of the threshold value. In contrast to being a fixed value in the Laplacian distribution, it is considered as a random variable in the proposed prior. Furthermore, we expand the joint distribution by using the improper prior distributions, leading to a hierarchical structure. The hierarchical prior is defined as follows:**

$$p(\boldsymbol{\theta}, \boldsymbol{\gamma} | \boldsymbol{w}, \alpha, \lambda) \propto p(\boldsymbol{\theta}|\boldsymbol{\gamma}) p(\boldsymbol{\gamma}|\boldsymbol{w}, \alpha, \lambda), \tag{10}$$

where the prior term $p(\boldsymbol{\theta}|\boldsymbol{\gamma})$ is defined as follows:

$$p(\boldsymbol{\theta}|\boldsymbol{\gamma}) = \prod_{i=1}^{3p^2} p(\theta_i|\gamma_i) \propto \prod_{i=1}^{3p^2} \gamma_i \exp\left(-\gamma_i|\theta_i|\right), \tag{11}$$

and the improper prior $p(\boldsymbol{\gamma}|\boldsymbol{w}, \alpha, \lambda)$ is formulated as follows:

$$\begin{aligned} p(\boldsymbol{\gamma}|\boldsymbol{w}, \alpha, \lambda) &= \prod_{i=1}^{3p^2} p(\gamma_i|w_i, \alpha, \lambda) \\ &\propto \prod_{i=1}^{3p^2} (w_i \alpha)^{1/2} \gamma_i^{-1} \exp\left(-w_i \alpha (\gamma_i - \lambda)^2\right), \end{aligned} \tag{12}$$

**for $\gamma_i \geq 0, i = 1, 2, \cdots, 3p^2$. $\alpha$ and $\lambda$ are the scaling and location parameters, respectively, which determine the threshold value of the corresponding dimension. The multiplicative term $\gamma_i^{-1}$ in Eq. (12) is introduced in order to offset the contribution $\gamma_i$ in Eq. (11). The weight $w_i$ is the decay factor for $\gamma_i$, where $w_i > 0$, for $i = 1, 2, \cdots, 3p^2$. This improper prior imposes different threshold values in different dimensions, and hence, it can be viewed as a kind of scale mixture**

**prior. In this formulation, we assume that the sparse code $\boldsymbol{\theta}_n$ is independent of $\boldsymbol{w}, \alpha$ and $\lambda$. The threshold value $\boldsymbol{\gamma}$ in each dimension is considered as a variable, which is implicitly captured through the improper prior $p(\boldsymbol{\gamma}|\boldsymbol{w}, \alpha, \lambda)$. In such a way, we introduce more flexibility into the denoising algorithm.**

Let $\boldsymbol{\alpha} = \alpha \boldsymbol{I}, \hat{\alpha}_i = w_i \alpha_i$, and $\boldsymbol{\lambda} = \lambda \boldsymbol{I} \in \mathbb{R}^{3p^2}$, then Eq. (12) can be rewritten as follows

$$p(\boldsymbol{\gamma}|\hat{\boldsymbol{\alpha}}, \boldsymbol{\lambda}) \propto \prod_{i=1}^{3p^2} \hat{\alpha}^{1/2} \gamma_i^{-1} \exp\left(-\hat{\alpha}_i (\gamma_i - \lambda_i)^2\right). \tag{13}$$

When $\gamma_i = \lambda_i$, for $i = 1, 2, \cdots, 3p^2$, the hierarchical prior distribution with the improper prior is degenerated to the Laplacian prior, which is a proper prior.

In order to adopt this improper prior under the Bayesian framework, we should first verify that the posterior distribution is well-defined under the improper prior. Theorem 1 provides a sufficient and necessary condition to guarantee the existence of posterior distribution under the improper prior.

**Theorem 1.** *Let the prior term be as follows:*

$$p(\boldsymbol{\theta}, \boldsymbol{\gamma}) \propto \prod_{i=1}^{N} \alpha^{1/2} \exp\left(-\gamma_i |\theta_i|\right) \exp\left(-\alpha_i (\gamma_i - \lambda_i)^2\right). \tag{14}$$

*The posterior distribution based on the prior $p(\boldsymbol{\theta}, \boldsymbol{\gamma})$ is defined as follows:*

$$p(\boldsymbol{\theta}, \boldsymbol{\gamma}|\boldsymbol{z}) = \frac{\prod_{i=1}^{N} \phi(z_i - \theta_i) p(\theta_i, \gamma_i)}{p(\boldsymbol{z})}. \tag{15}$$

*where $\phi(\cdot)$ is the standard Gaussian distribution. The evident term $p(\boldsymbol{z})$ is defined as follows:*

$$p(\boldsymbol{z}) = \prod_{i=1}^{N} \int_{\mathbb{R}} \int_0^{\infty} \phi(z_i - \theta_i) p(\theta_i, \gamma_i) \mathrm{d}\gamma_i \mathrm{d}\theta_i, \tag{16}$$

*If $p(z_i) < \infty, \forall z_i \in \mathbb{R}$, then it implies that the posterior distribution $p(\boldsymbol{\theta}, \boldsymbol{\gamma}|\boldsymbol{z})$ exists.*

The proof of Theorem 1 can be found in Appendix A.

### *3.4. Image-Denoising Model*

**For each mean-subtracted PG $\bar{\boldsymbol{Y}}_m = \{\bar{\boldsymbol{y}}_{n,m}\}_{n=1}^{N}$, $m = 1, 2, \cdots, M$, we obtain the dictionary $\boldsymbol{D}_m = \boldsymbol{U}_m$, and the improper prior is adopted in the sparse model to form a Bayesian sparse model with hierarchical priors, which is**

$$p(\boldsymbol{\theta}_{n,m}|\boldsymbol{y}_{n,m}, \boldsymbol{D}_m) \propto p(\boldsymbol{y}_{n,m}|\boldsymbol{\theta}_{n,m}, \boldsymbol{D}_m) \times p(\boldsymbol{\theta}_{n,m}|\gamma_{n,m}) \times p(\boldsymbol{\gamma}_{n,m}|\boldsymbol{w}, \alpha, \lambda). \tag{17}$$

**Eq. (17) can be represented by a probabilistic graphical model, as shown in Figure 1, which demonstrates the generation procedure of each observed image local patch in the PGs. In particular, the $n$-th clean image local patch in the $m$-th group, i.e., $\boldsymbol{x}_{n,m}$, is corrupted by the noise $\boldsymbol{\sigma}_{n,m}$, leading to the corresponding observed image local patch $\boldsymbol{y}_{n,m}$. In addition, the underlying image local patch $\boldsymbol{x}_{n,m}$ can be represented as a linear combination of the dictionary $\boldsymbol{D}_m$ and the sparse code $\boldsymbol{\theta}_{n,m}$. $\boldsymbol{\theta}_{n,m}$ follows the distribution of $\boldsymbol{\gamma}_{\boldsymbol{n,m}}$, and the $\gamma_{n,m}$ is controlled by the distribution of $\boldsymbol{\alpha}_{n,m}, \boldsymbol{\lambda}_{n,m}$ and $\boldsymbol{w}_{n,m}$, forming a hierarchical structure.**

In this case, the forward image process is assumed to be a normal distribution with a scale mixture of improper priors. The image-denoising problem is formulated as a MAP estimation problem of a scale mixture of normal distributions. The MAP estimator can be obtained by jointly minimizing an optimization problem, as follows:

$$\min_{\boldsymbol{\theta},\boldsymbol{\gamma}} \|\boldsymbol{y}_{n,m} - \boldsymbol{D}_m\boldsymbol{\theta}_{n,m}\|^2 + \sum_{i=1}^{3p^2} \gamma_i|\theta_i| + \sum_{i=1}^{3p^2} \hat{\alpha}_i(\gamma_i - \lambda_i)^2. \tag{18}$$

Eq. (18) is an optimization problem in terms of $\boldsymbol{\theta}$ and $\boldsymbol{\gamma}$, which is non-convex and difficult to be solved. Rather than solving the primal problem, Theorem 2 shows that the problem of Eq. (18), with the improper prior, can be converted to the problem regularized by the generalized weighted MCP penalty.

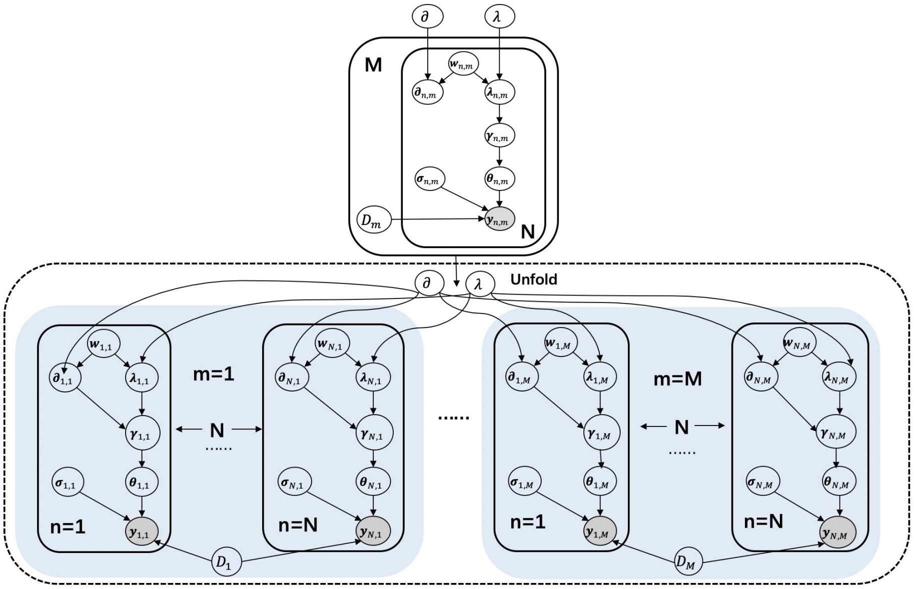


Figure 1: The probabilistic graphical model of Eq. (17). The shaded node denotes the observable data.

**Theorem 2.** *Let $\lambda > 0$, and $\alpha > 0$, then*

$$\begin{aligned}\int_0^{|x_i|}(\lambda_i - \frac{t}{\hat{\alpha}_i})\mathrm{d}t &= \min_{\gamma_i \geq 0}(\gamma_i|x| + \frac{\hat{\alpha}_i}{2}(\gamma_i - \lambda_i)^2), \\ &= \min_{\gamma_i \geq 0}(\gamma_i|x| + \frac{w_i\alpha_i}{2}(\gamma_i - \lambda_i)^2),\end{aligned} \tag{19}$$

*and the optimal $\hat{\gamma}_i = \nu(\lambda_i - \frac{|x|}{w_i\alpha})_+$ is the unique solution to the right-hand side of Eq. (19), where $\nu(x)_+ = max(x, 0)$.*

By Theorem 2, Eq. (18) can be rewritten as follows:

$$\min_{\boldsymbol{\theta}} \|\boldsymbol{y}_{n,m} - \boldsymbol{D}_m\boldsymbol{\theta}_{n,m}\|^2 + \sum_{i=1}^{3p^2} p_{\lambda_i,\alpha_i}(\theta_i), \tag{20}$$

where

$$\begin{aligned} p_{\lambda_i,\alpha_i}(\theta_i) &= \int_0^{|\theta_i|} \nu(\lambda_i - \frac{t}{w_i\alpha_i})_+ \mathrm{d}t \\ &= \begin{cases} \lambda_i|\theta_i| - \frac{\theta_i^2}{2w_i\alpha_i}, & \text{if } |\theta_i| \leq w_i\alpha_i\lambda_i, \\ \frac{1}{2}w_i\alpha_i\lambda_i^2, & \text{if } |\theta_i| > w_i\alpha_i\lambda_i, \end{cases} \end{aligned} \tag{21}$$

for $\alpha_i = \alpha > 1, \lambda_i = \lambda > 0$, for $i = 1, 2, \cdots, 3p^2$. When $w_i = 1$, for all $i = 1, 2, \cdots, 3p^2$, Eq. (21) is reduced to the MCP function. Compared to the MCP function, our derived penalty function considers that the different dimensions of an image local patch have different contributions in image denoising, and thus, we impose different threshold values and selection ranges for different dimensions. Due to the orthogonality of the dictionary $\boldsymbol{D}$, and with the i.i.d assumption, we can obtain an analytical solution, as follows :

$$\theta_i = \begin{cases} \frac{w_i\alpha_i}{w_i\alpha_i - 1} T_{\lambda_i}(\boldsymbol{d}_i^T \boldsymbol{y}_{n,m}), & \text{if } |\boldsymbol{d}_i^T \boldsymbol{y}_{n,m}| \leq w_i\lambda_i\alpha_i, \\ \boldsymbol{d}_i^T \boldsymbol{y}_{n,m}, & \text{if } |\boldsymbol{d}_i^T \boldsymbol{y}_{n,m}| > w_i\lambda_i\alpha_i, \end{cases} \tag{22}$$

where $T_\lambda(\cdot)$ is the soft-threshold operator, which is defined as follows:

$$T_\lambda(x) = \mathrm{sgn}(x) \cdot \max(x - \lambda, 0). \tag{23}$$

More details of the derivation is shown in Appendix B.

We can see that the solution consists of two parts. Within the neighborhood of the original point, the soft threshold function, with the scale factor of $\frac{w_i a_i}{w_i a_i - 1}$, is performed to shrink the variables, while for a region far from the origin, the ordinary regression is performed, which is an unbiased estimation under the Gauss-Markov assumption. Therefore, the obtained solution can achieve a better approximation, compared with the LASSO estimator, derived from the $l_1$ norm penalty. A different weight $w_i$ corresponds to a different selection scale. When $\alpha_i \to \infty$, Eq. (22) will become a soft threshold. When $\alpha_i \to 1$, Eq. (22) will become the hard threshold. In particular, when $w_i = 1$, for $i = 1, 2, \cdots, 3p^2$, Eq. (22) will become the firm threshold derived from the MCP function. Another hyperparameter $\lambda_i$ decides the shrinking value of the threshold function.

The weight $w_i$ is set based on the matrix of singular values obtained from the SVD decomposition of the PG, and the matrix of singular values contains the information of the nonlocal self-similarity of the PG. Diagonal values in the singular matrix $\boldsymbol{S}$ are extracted to form a vector $\boldsymbol{s} = [s_1, s_2, \cdots, s_{3p^2}]^T$, where $s_1, s_2, \cdots, s_{3p^2}$ are the diagonal values in the singular matrix $\boldsymbol{S}$, and the weight $w_i$ is computed as follows

$$w_i = 1 - \frac{s_i}{\sum_{i=1}^{3p^2} s_i}. \tag{24}$$

This weight setting implies the nonlocal self-similarity property. A small selection scale is imposed on more significant atoms, and the unbiased estimation is performed on these atoms as much as possible.

### *3.5. The denoising Algorithm*

When we obtain the solution of the sparse coding vectors $\{\hat{\boldsymbol{\theta}}_{n,m}\}$ in Eq. (22), the clean image patch $\hat{\boldsymbol{y}}_{n,m}$ of the $n$-th noisy patch in the PG $\boldsymbol{Y}_m$ is reconstructed as follows:

$$\hat{\boldsymbol{y}}_{n,m} = \boldsymbol{D}\hat{\boldsymbol{\theta}}_{n,m} + \boldsymbol{\mu}_m, \tag{25}$$

where $\boldsymbol{\mu}_m$ is the group mean of $\boldsymbol{Y}_m$. The clean image is then reconstructed by aggregating all the reconstructed image local patches in all PGs, with Gaussian weights. Given a noisy image, we perform the denoising algorithm iteratively for achieving better denoising outputs. In each iteration, we propose a decay scheme for $\alpha_i$, and $\lambda_i$, based on the noise variation. Therefore, $\alpha_i$, and $\lambda_i$ can be adjusted adaptively, based on the noise level for each image local patch.

We adopt the noise-estimation method used in [40]. In the $t$-th iteration, we assume that the noise level of an image local patch is computed iteratively as follows:

$$\sigma_{t,n}^2 = max(\sigma^2 - \|\boldsymbol{y}_{n,m} - \boldsymbol{x}_{n,m}^{t-1}\|_2^2, 0), \tag{26}$$

where $\boldsymbol{y}_{n,m}$ is the $n$-th image patch in $\boldsymbol{Y}_m$, $\boldsymbol{x}_{n,m}^{t-1}$ is the $n$-th patch in $\boldsymbol{Y}_m$ recovered in the $(t-1)$-st iteration, and $\sigma^2$ is the initial noise level of the $n$-th patch of $\boldsymbol{Y}_m$, which is computed as follows:

$$\sigma^2 = (\sigma_r^2 + \sigma_g^2 + \sigma_b^2)/3. \tag{27}$$

The variation of noise level is defined as follows:

$$\delta = \frac{|\sigma_t^2 - \sigma_{t-1}^2|}{\sigma_{t-1}^2}. \tag{28}$$

To update $\boldsymbol{w}$ of that local PG, where

$$\boldsymbol{w}^{t+1} = \boldsymbol{w}^t f(\delta, \delta_{low}, \delta_{high}), \tag{29}$$

where $f(\cdot)$ is a weight decay function, with its lower bound value $\delta_{low}$, and upper bound value $\delta_{high}$. In this paper, we use the clipped function as the weight decay function in our model, which is defined as follows:

$$f(x) = \begin{cases} x_{low}, & \text{if } x \leq x_{low}, \\ x_{high}, & \text{if } x \geq x_{high}, \\ x, & \text{otherwise}. \end{cases} \tag{30}$$

The update of $\lambda_i$ and $\alpha_i$ in the $(t+1)$-st iteration is as follows:

$$\lambda_i^{t+1} = w_i^{t+1} \lambda_i^t, \tag{31}$$

$$\alpha_i^{t+1} = \frac{\alpha_i^t}{w_i^{t+1}}. \tag{32}$$

We can see that $\lambda_i^{t+1} a_i^{t+1} = \lambda_i^t a_i^t$, and hence the selection range is consistent in all iterations. Specifically, the shrinking threshold parameter $\lambda$ is adaptive to the noise level. The proposed denoising algorithm is summarized in Algorithm 1.

## 4. Experiments

In this section, we evaluate our proposed Bayesian sparse hierarchical model on several denoising datasets, including one synthesized AWGN dataset and three publicly available real-world noisy image datasets. Different types of sensors can capture images under different light conditions and camera settings, which may cause varying characteristics of the noise. Therefore, it is challenging to adopt a single set of parameters in our proposed sparse model for all images

**Algorithm 1** Bayesian sparse hierarchical model for image denoising.

```
 Input: Noisy image $\boldsymbol{y}$
 Initialization: $\hat{\boldsymbol{x}}^{(0)} = \boldsymbol{y}$, $\boldsymbol{\alpha}^0$, $\boldsymbol{\lambda}^0$
for i=1:Num_of_iterations
1.  Extract PGs $\{\boldsymbol{Y}_m\}_{m=1}^{M}$;
  for each PG $Y_m$
2.  Compute the mean $\boldsymbol{\mu}_m$ and the mean-subtracted PG $\bar{\boldsymbol{Y}}_n$;
3.  Apply SVD to each mean subtracted PG via Eq. (11);
4.  Compute the noise increment, via Eq. (28) and Eq. (29), and the corresponding $\alpha^{Ite}$ and $\lambda^{Ite}$;
5.  Recover each patch in all PGs, via Eq. (25).
  end for
6. Aggregate the recovered PGs of all the subspaces to form the recovered image $\hat{\boldsymbol{x}}^{(Ite)}$.
end for
 Output: The denoised image $\hat{\boldsymbol{x}}$
```

across different domains. However, it is impractical to tune all parameters of the model to obtain the best performance for each image. In this experiment, the patch size is set to 6 × 6 for all datasets. The size of the searching window is set to 20 × 20, and 100 similar image patches are searched to form a PG. Furthermore, image patches are densely extracted with a step size of 3. The upper bound value and the lower bound value of the weight decay factor are fixed at 0.75 and 0, respectively. We implement the denoising algorithm in fixed iterations under the Matlab2017b environment on a computer with Intel(R) Core(TM) i7-8700K CPU of 3.70GHz and 32 GB RAM. The code will be released with the publication of this paper.

### *4.1. Ablation Study*

Our proposed Bayesian sparse hierarchical model can be considered as a sparse model with a generalized weighted MCP function. In order to evaluate the effectiveness of our proposed sparse hierarchical model and weight decay scheme based on noise variation, we adopt the sparse model with MCP prior proposed by [51] as the baseline, denoted as MCP. In this model, the threshold value and the feature-selection range are fixed for all image patches, independent of the noise level. In addition, we adopt two settings of the weight decay scheme in our proposed method for self-comparison. For the proposed model without the weight decay scheme, we denote it as WMCP. For the proposed model with the weight decay scheme, we denote it as Ada-WMCP.

All the parameters of these three models are set the same, and we apply them to the CC15 dataset [52] for denoising, with ten iterations. The CC15 dataset [52] contains 15 cropped real-world noisy images from the CC dataset. Since the image size of the CC dataset is about 7000 × 5000, smaller images with a size of $512 \times 512$ are cropped for conducting our experiments. The noisy images of the CC15 dataset include 11 static scenes, which are controlled under the indoor environments. Each scene was shot 500 times, under the same camera settings. The mean image of these 500 shots is taken as the ground truth.

**The average result, in terms of peak signal-to-noise ratio (PSNR),**

**up to 10 iterations, are shown in Figure 2. We can see that our proposed WMCP model and Ada-WMCP model can achieve a better performance than the MCP model, because the use of adaptive weights leads to different selection scales, so the denoising performance can be effectively improved. Furthermore, the performance of the Ada-WMCP model converges eventually, while the performance of WMCP and MCP increases in the first several iterations, and then, drops gradually after the 3rd iteration. Images in the CC15 dataset contain realistic noise, and the noise is not uniformly distributed in the whole image. This means that those image patches, with strong high-level noise, will require more iterations for the fixed-threshold scheme, while those image patches, with weak low-level noise, requires less iterations. In this case, both WMCP and MCP easily cause oversmoothing, and lead to degraded performance. Compared with MCP, WMCP imposes different threshold values, and feature-selection ranges based on the estimated noise level. Thus, this gives rise to better results. By considering the weight decay scheme in the denoising processing, the threshold values and feature-selection ranges are adaptive to the noise within each image patch, so Ada-MCP can effectively prevent the images from being oversmoothed or undersmoothed, leading to the best performance. In real-world applications, the adaptive weight decay scheme is indispensable, because no ground-truth images are referred to measuring the quality of denoising images, so the optimal number of iterations for image denoising is unknown. In the rest of the paper, we employ Ada-MCP for comparison without explicitly mentioning it.**

### *4.2. Evaluation on Synthesized AWGN Corrupted Images*

In this work, the Set12 dataset is adopted to verify our proposed method for AWGN corrupted images. The dataset consists of 12 grayscale images and is widely used in image denoising. The noisy image $\boldsymbol{y}$ is synthesized by adding

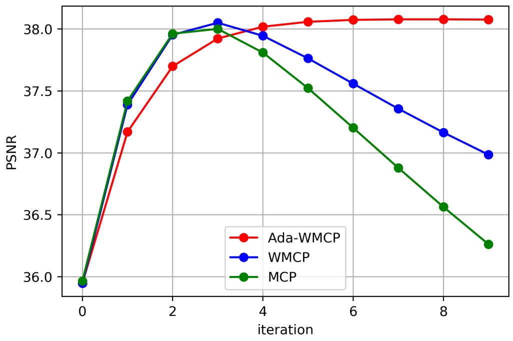


Figure 2: The PSNR of MCP, WMCP, and Ada-MCP on the CC15 dataset in 10 iterations.

AWGN with the noise level $\sigma$ to the corresponding clean image $\boldsymbol{x}$, following the degradation process $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{\sigma}$, where $\boldsymbol{\sigma}$ is the noise component. The noise level $\sigma$ is set to $\{15, 25, 50\}$ for evaluating our proposed algorithm.

We compare our proposed method with BM3D [53], EPLL [54], PGPD [4], and ACPT [55]. The BM3D is a classic two-stage denoising method, and its effectiveness has been proven. EPLL, PGPD, and ACPT are variants of the sparse model based on the soft-threshold estimator. Specifically, EPLL and PGPD are patch-based processing methods, which apply clustering methods to construct the dictionary for noise removal. EPLL used extra datasets, which can be considered as a kind of extra image prior, while our proposed method only adopts the property of nonlocal self-similarity of an image to construct the dictionary for image-denoising. This is viewed as a kind of internal image prior, and the condition of the internal image in our proposed method is weaker than PGPD and EPLL. ACPT is a recently proposed method, which is effective in preserving the texture and detail information in image denoising, and thus it can prevent images suffered from oversmoothing in the denoising process. All of these methods have their models publicly available as released by the authors, and we use their default settings in our experiments. We adopt the PSNR as our criterion in the evaluation. The average PSNR results of different denoising methods for AWGN removal are tabulated in Table 1, where the best average

Table 1: Average PSNR(dB) results of different methods on the Set12 dataset. The best average results are highlighted in bold.

| No. | BM3D [53] | | | PGPD [4] | | | EPLL [54] | | | ACPT [55] | | | Ours | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 25 | 50 | 15 | 25 | 50 | 15 | 25 | 50 | 15 | 25 | 50 | 15 | 25 | 50 |
| 01 | 31.69 | 29.07 | 26.06 | 30.46 | 27.27 | 23.05 | 31.74 | 29.15 | **26.19** | 31.84 | **29.35** | 25.93 | **32.01** | 29.04 | 26.12 |
| 02 | 34.60 | 32.28 | 29.08 | 30.46 | 27.27 | 23.05 | 34.19 | 32.24 | 28.87 | **35.11** | **32.90** | 28.86 | 34.73 | 32.77 | **29.34** |
| 03 | 32.50 | 29.92 | 26.39 | 31.12 | 27.66 | 23.01 | 32.58 | 30.13 | **26.70** | 32.58 | 29.86 | 26.16 | **32.84** | **30.31** | 26.65 |
| 04 | 30.84 | 28.05 | 24.59 | 30.10 | 26.42 | 21.82 | 31.16 | 28.39 | 24.98 | 31.44 | **28.80** | **25.07** | **31.53** | 28.77 | 24.97 |
| 05 | 31.56 | 28.93 | 25.39 | 30.62 | 26.94 | 22.27 | 31.98 | 29.45 | **25.86** | 32.19 | 29.41 | 25.60 | **32.36** | 29.50 | 25.77 |
| 06 | 30.79 | 28.16 | 25.17 | 29.97 | 26.58 | 22.22 | 31.21 | 28.50 | **25.22** | 31.08 | 28.40 | 24.76 | **31.29** | **28.64** | 25.18 |
| 07 | 31.28 | 28.75 | 25.71 | 30.25 | 26.94 | 22.66 | 31.34 | 28.76 | 25.79 | 31.41 | 28.80 | 25.51 | **31.54** | **29.02** | **25.83** |
| 08 | 34.09 | 31.88 | 28.44 | 31.88 | 28.54 | 23.92 | 33.85 | 31.56 | 28.36 | 33.94 | 31.45 | 27.78 | **34.26** | **31.92** | **28.48** |
| 09 | 32.80 | 30.32 | **26.83** | 31.14 | 27.76 | 23.19 | 31.30 | 28.53 | 24.84 | 33.00 | 30.14 | 25.82 | **33.39** | **30.78** | 26.69 |
| 10 | 31.95 | 29.61 | 26.53 | 30.79 | 27.55 | 23.15 | 31.89 | 29.63 | **26.62** | 31.94 | 29.51 | 26.13 | **32.20** | **29.82** | 26.46 |
| 11 | 31.79 | 29.46 | 26.58 | 30.79 | 27.58 | 23.27 | 31.92 | **29.55** | **26.69** | 31.74 | 29.26 | 26.11 | **32.05** | 29.54 | 26.47 |
| 12 | 31.96 | 29.48 | 26.30 | 30.86 | 27.54 | 23.06 | 31.85 | 29.41 | 26.16 | 31.78 | 29.15 | 25.58 | **32.09** | **29.55** | **26.49** |
| Avg. | 32.15 | 29.66 | 26.42 | 30.85 | 27.47 | 22.98 | 32.09 | 29.61 | 26.36 | 32.34 | 29.75 | 26.11 | **32.52** | **30.00** | **26.49** |

performance on the 12 grayscale images is highlighted in bold. It is worth noting that BM3D is a two-step estimator, and we show the final estimated results in Table 1. All the sparse models are a one-step estimator by performing the threshold function. In particular, compared with EPLL and PGPD, our proposed method only uses the internal image prior, i.e., nonlocal self-similarity, and gives rise to the best performances, with the dictionary constructed under weak conditions. The reason is that our proposed method leads to a more accurate estimation of the sparse code than the soft-threshold estimator. Overall, we can see that all the denoising performances are degraded with increasing noise levels. Our proposed method does not achieve the best denoising result for some images in the Set12 dataset, but the average denoising performance, under three different noise levels, is the best compared with BM3D, PGPD, EPLL, and ACPT.

### *4.3. Evaluation on Realistic Noise Removal*

Due to the complexity of real-world noise, we adopt three real-world noisy-image datasets to validate our proposed method on the different characteristics of the realistic noise. We evaluate our proposed model on CC15 dataset and CC60 dataset [60]. CC60 dataset is a large-scale dataset and contains more

Table 2: PSNR(dB) results of different methods on 15 cropped real-world noisy images. The best results are highlighted in bold.

| Setting | **CBM3D** [56] | **WNNM** [38] | **CSF** [57] | **TNRD** [58] | **DnCNN** [16] | **FFDNet** [17] | **NI** | **NC** [59] | **EI** [21] | **TWSC** [40] | **Ours** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Cannon 5D ISO=3200 | 39.76 | 37.51 | 35.68 | 39.51 | 37.26 | 40.27 | 37.68 | 38.76 | 40.50 | 40.55 | **41.32** |
|  | 36.40 | 33.86 | 34.03 | 36.47 | 34.13 | 37.24 | 34.87 | 35.69 | 37.05 | 35.93 | **37.40** |
|  | 36.37 | 31.43 | 32.63 | 36.45 | 34.09 | 37.03 | 34.77 | 35.54 | 36.11 | 35.15 | **37.24** |
| Nikon D600 ISO=3200 | 34.18 | 33.46 | 34.84 | 34.79 | 33.62 | 35.26 | 34.12 | **35.57** | 34.88 | 35.36 | **35.57** |
|  | 35.07 | 36.09 | 35.16 | 36.37 | 34.48 | 36.82 | 35.36 | 36.70 | 36.31 | 37.09 | **37.27** |
|  | 37.13 | 39.86 | 39.98 | 39.49 | 35.41 | 40.94 | 38.68 | 39.28 | 39.23 | 41.13 | **41.34** |
| Nikon D800 ISO=1600 | 36.81 | 36.35 | 34.84 | 38.11 | 35.79 | 39.34 | 37.34 | 38.01 | 38.40 | 39.36 | **39.47** |
|  | 37.76 | 39.99 | 38.42 | 40.52 | 36.08 | 41.78 | 38.57 | 39.05 | 40.92 | 41.91 | **42.20** |
|  | 37.51 | 37.15 | 35.79 | 38.17 | 35.48 | 39.44 | 37.87 | 38.20 | 38.97 | 38.81 | **39.89** |
| Nikon D800 ISO=3200 | 35.05 | 38.06 | 38.36 | 37.69 | 34.08 | 40.11 | 36.95 | 38.07 | 38.66 | **40.27** | 40.20 |
|  | 34.07 | 36.04 | 35.53 | 35.90 | 33.70 | 37.58 | 35.09 | 35.72 | 37.07 | 37.22 | **37.86** |
|  | 34.42 | 39.73 | 40.05 | 38.21 | 33.31 | 41.83 | 36.91 | 36.76 | 38.52 | **42.09** | 41.52 |
| Nikon D800 ISO=6400 | 31.13 | 33.29 | 34.08 | 32.81 | 29.83 | 35.34 | 31.28 | 33.49 | 33.76 | **35.53** | 34.85 |
|  | 31.22 | 31.16 | 32.13 | 32.33 | 30.55 | 34.01 | 31.38 | 32.79 | 33.43 | **34.15** | 34.02 |
|  | 30.97 | 31.98 | 31.52 | 32.29 | 30.09 | 34.04 | 31.40 | 32.86 | 33.58 | 33.93 | **34.19** |
| Average | 35.19 | 35.77 | 35.33 | 36.61 | 33.86 | 38.07 | 35.49 | 36.43 | 37.15 | 37.89 | **38.29** |

characteristics of natural images, compared with the CC15 dataset. The third dataset is the PolyU-Xu dataset [61], which contains 100 different pairs of images of 40 scenes. The size of each image is 512 × 512, and these images were captured by different camera settings, including Canon Mark 5D, Canon Mark 80D, Canon Mark 600D, Nikon D800, and Sony A7 II. Each scene was captured with three lighting conditions, including the indoor normal lighting condition, dark lighting condition, and outdoor normal light condition. Six different ISO settings were used, which are 800, 1600, 3200, 6,400, 12,800, and 25,600. Each static scene was shot at about 500 to 1000 times under the same camera setting, and the captured images were averaged to obtain the "ground-truth" image. We compare our proposed method with other state-of-the-art denoising methods, including CBM3D [56], WNNM [38], MCWNNM [39], CSF [57], TNRD [58], External and internal prior hybrid method (EI) [21], TWSC [40], Noise clinic (NC) [59], Neat Clinic (NI), DnCNN [16] and FFDNet [17]. Among these methods, NC is a blind image-denoising method, and NI is commercial software for image denoising. Both methods have been embedded in Photoshop and Core Paintshop. WNNM, CSF, and TNRD are effective for grayscale image denois-

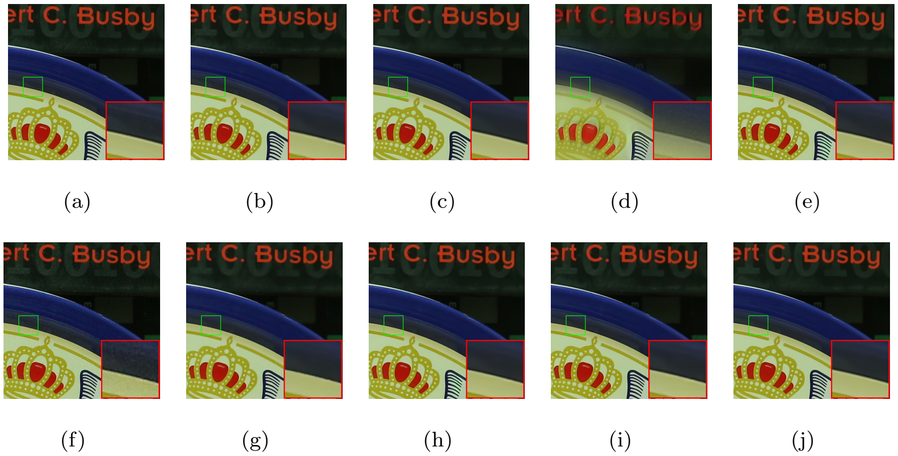


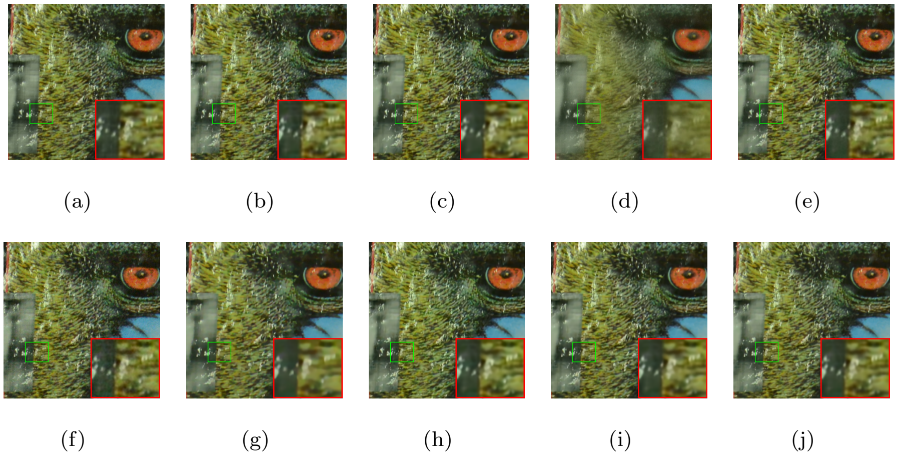
Figure 3: Visualization of the denoised image “Cannon5D ISO=3200” and its cropped region by different methods. (a) Ground-truth image, (b) CBM3D, (c) TWSC, (d) NI, (e) NC, (f) Noisy, (g) CSF, (h) DnCNN, (i) FFDNet, and (j) ours.

ing, and hence, these three methods are applied to each channel of the color images for denoising. CBM3D, MCWNNM, TWSC, DnCNN, and FFDNet can be directly used for color image denoising, and all of them have publicly available models, released by the authors. We adopt the default settings in this work.

Table 3: PSNR(dB) results of different methods on the CC60 dataset. The best result is highlighted in bold.

| | CBM3D [56] | MCWNNM [39] | NI | NC [59] | CSF [57] | TNRD [58] | TWSC [40] | DnCNN [16] | FFDNet [17] | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| PSNR | 39.40 | 39.03 | 36.53 | 37.57 | 37.40 | 38.32 | 39.66 | 34.99 | 39.73 | **40.27** |

*a). Results of the CC15 dataset.* Detailed results, in terms of PSNR, of different methods, are listed in Table 2. (The results of CBM3D, WNNM, CSF, TRND, DnCNN, NI, NC, and EI are copied from [21], with the same parameter settings). The best performance of each image is highlighted in bold. We can see that our proposed method can achieve the best PSNR results on 11 out of 15 images, and TWSC achieves the best PSNR results on 4 out of 15 images. The performance of deep-learning-based methods, i.e., DnCNN and



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Figure 4: Visualization of the denoised image “NikonD600 ISO3200 C37” and its cropped region, by different methods. (a) Ground-truth image, (b) CBM3D, (c) TWSC, (d) NI, (e) NC, (f) Noisy, (g) CSF, (h) DnCNN, (i) FFDNet, and (j) ours.

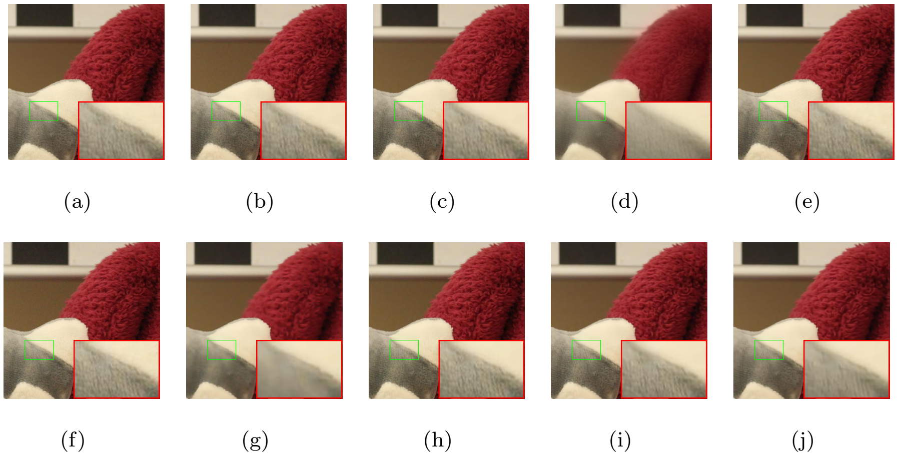
FFDNet, degrades on the CC15 datasets because the performance of deep-learning-based methods greatly depends on the training data, i.e., the types of noise in the training images. These domain-specific characteristics limit the generalization of deep-learning-based methods for those images across different domains. On average, our proposed method gains an improvement of 0.22 dB in terms of PSNR over the second-best method, i.e. FFDNet, and outperforms other competitive methods by a large margin. The visualized results of the denoised image “Cannon5D ISO=3200”, by different methods, are shown in Figure 3. We can see that NI and NC tend to oversmooth the image, while CBM3D and CSF cannot effectively preserve the edges, and some visible ripples are generated along the edges. In fact, our proposed method can preserve the edges and texture better than other competitive methods, i.e., TWSC, DnCNN, and FFDNet. More visually pleasant outputs are generated by our proposed method.

*b). Results on the CC60 dataset.* Average PSNR on the CC60 dataset, by the different methods, are shown in Table 3. On average, we can see that our proposed method can achieve state-of-the-art performance compared with other competitive methods. Compared with the second-best method (FFDNet), our proposed method gains an improvement of 0.54dB. The performance of DnCNN is worse than TWSC, but FFDNet outperforms TWSC. This reflects that deep learning methods are domain-specific. Figure 4 shows the denoised results of the image "NikonD600 ISO3200 C37", generated by different methods. NI and CSF generate oversmoothed images and lose texture information. NC can reduce the noise effectively, but produce artifacts in the denoising process. Compared with CBM3D, TWSC, DnCNN, and FFDNet, our proposed method can reduce noise effectively, and maintain the texture information as much as possible, leading to better visual quality.

*c). Results on the PolyU-Xu dataset.* The performance, in terms of PSNR on the PolyU-Xu dataset, is shown in Table 3. On average, our proposed method achieves much better PSNR results than other competitive methods. Compared with the second-best method (TWSC), the improvement of our proposed method is 0.33dB. Specifically, we can see that the deep learning methods, i.e., DnCNN and FFDNet, obtain worse results than the sparse models (TWSC and ours). This shows that the generalization capacity of the sparse models for images across different domains is much better than deep-learning-based methods. Figure 5 shows the visual results of the image "Canon5D2 5200 3200 toy1". NI and CSF generate oversmoothed images, and lose the texture in the images. DnCNN and FFDNet can effectively remove the noise, but generate artifacts. Compared with TWSC, our proposed method suppresses the noise and preserves the texture simultaneously. A visually better result is produced by our proposed method.

Table 4: PSNR(dB) results of different methods on the PolyU-Xu dataset. The best result is highlighted in bold.

| | CBM3D [56] | MCWNNM [39] | NI | NC [59] | CSF [57] | TNRD [58] | TWSC [40] | DnCNN [16] | FFDNet [17] | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| PSNR | 38.69 | 38.51 | 35.70 | 36.76 | 37.07 | 37.48 | 38.62 | 34.74 | 38.56 | **38.95** |



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Figure 5: Visualization of the denoised image "Canon5D2 5200 3200 toy1" and its cropped region by different methods: (a) Ground-truth image, (b) CBM3D, (c) TWSC, (d) NI, (e) NC, (f) Noisy, (g) CSF, (h) DnCNN, (i) FFDNet, and (j) ours.

## 5. Conclusion

In this paper, the image-denoising problem is formulated as the maximum-a-posteriori estimator of a scaled mixture of normal distributions. Instead of employing the commonly used $\ell_1$ norm penalty in the sparse model, we propose a hierarchical model with an improper prior distribution for image denoising. We have shown that our proposed hierarchical prior distribution results in a more general model for noise removal. The soft-threshold operator, derived from the $\ell_1$ norm, and the firm-threshold operator, derived from the $\ell_0$ norm can be obtained from our proposed prior distribution under some specific conditions. In the denoising process, we have proposed a strategy for adaptively updating the weights in the sparse model based on the noise characteristics. Shrinking threshold values and the feature-selection range can be adaptive to the noise level within image patches, so the issues of oversmoothing and undersmoothing can be avoided effectively. Compared with other competitive methods, experiment results on several benchmark datasets demonstrate that our proposed method achieves a promising result for additive white Gaussian noise (AWGN) removal, and obtains state-of-the-art performance for real-world image denoising.

## Appendix A. Proof of Theorem 1

Based on the Bayes theorem, the posterior distribution $p(\boldsymbol{\theta}, \boldsymbol{\gamma}|\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{\lambda})$ is defined as follows:

$$p(\boldsymbol{\theta}, \boldsymbol{\gamma}|\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{\lambda}) = \frac{p(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{\gamma}, \boldsymbol{\alpha}, \boldsymbol{\lambda}) p(\boldsymbol{\theta}|\boldsymbol{\alpha}, \boldsymbol{\lambda})}{p(\boldsymbol{x}|\boldsymbol{\alpha}, \boldsymbol{\lambda})}, \tag{A.1}$$

where $\boldsymbol{\theta}, \boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{\lambda} \in \mathbb{R}^n$, and $\boldsymbol{\gamma} \in \mathbb{R}^n_+$. In order to show the existence of the posterior distribution $p(\boldsymbol{\theta}, \boldsymbol{\gamma}|\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{\lambda})$, we need to prove that the evidence term $p(\boldsymbol{x}|\boldsymbol{\alpha}, \boldsymbol{\lambda}) < \infty$, which is defined as follows:

$$p(\boldsymbol{x}|\boldsymbol{\alpha}, \boldsymbol{\lambda}) = \int_{\mathbb{R}^n} \int_{\mathbb{R}^n_+} p(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{\gamma}|\boldsymbol{\alpha}, \boldsymbol{\lambda}) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\boldsymbol{\gamma}. \tag{A.2}$$

By the Fubini theorem, Eq. (A.2) can be rewritten as

$$p(\boldsymbol{x}|\boldsymbol{\alpha}, \boldsymbol{\lambda}) = \prod_{i=1}^{n} \int_R \int_0^\infty p(x_i, \theta_i, \gamma_i|\alpha_i, \lambda_i) \mathrm{d}\theta_i \mathrm{d}\gamma_i. \tag{A.3}$$

We utilize the conclusion claimed in [62]. For $i = 1, \cdots, N$, there exists $M_i \in \mathbb{R}_+$ such that

$$\int_R \int_0^\infty p(x_i, \theta_i, \gamma_i|\alpha_i, \lambda_i) \mathrm{d}\theta_i \mathrm{d}\gamma_i \leq M_i. \tag{A.4}$$

Therefore, we have

$$\begin{aligned} p(\boldsymbol{z}|\boldsymbol{\alpha}, \boldsymbol{\lambda}) &= \prod_{i=1}^{n} \int_R \int_0^\infty p(x_i, \theta_i, \gamma_i|\alpha_i, \lambda_i) \mathrm{d}\theta_i \mathrm{d}\gamma_i \\ &\leq \prod_{i=1}^{N} M_i \\ &< \infty. \end{aligned} \tag{A.5}$$

Then, the posterior distribution $p(\boldsymbol{\theta}, \boldsymbol{\gamma}|\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{\lambda})$ exists.

## Appendix B. Derivation of Analytical Solution

Given $\boldsymbol{y} \in \mathbb{R}^p, \boldsymbol{D} \in \mathbb{R}^{p\times N}, \boldsymbol{x} \in \mathbb{R}^N$, where $\boldsymbol{D}^T\boldsymbol{D} = \boldsymbol{I}$, the optimization problem is defined as follows:

$$\min_{x_i} \sum_{i=1}^{p}(y_i - \sum_{j=1}^{N} d_{ij}x_j)^2 + \sum_{j=1}^{N} p_{\gamma_j}(x_j;\lambda_j), \tag{B.1}$$

where

$$p_{\gamma_i}(x_i;\lambda_i) = \begin{cases} \lambda_i|x_i| - \frac{x_i^2}{2\gamma_i}, & \text{if } |x_i| \leq \gamma_i\lambda_i, \\ \frac{1}{2}\gamma_i\lambda_i^2, & \text{if } |x_i| > \gamma_i\lambda_i. \end{cases} \tag{B.2}$$

Let $L(\boldsymbol{x}) = \sum_{i=1}^{p}(y_i - \sum_{j=1}^{N} d_{ij}x_j)^2 + \sum_{j=1}^{N} p_{\gamma_j}(x_j;\lambda_j)$. We consider

1. For the case of $|x_j| \leq \gamma_i\lambda_i$, $\forall j$, we have

$$L(\boldsymbol{x}) = \sum_{i=1}^{p}(y_i - \sum_{j=1}^{N} d_{ij}x_j)^2 + \sum_{j=1}^{N}(\lambda_j|x_j| - \frac{x_j^2}{2\gamma_j}). \tag{B.3}$$

We take the derivative of both sides of Eq. (B.3), that is

$$\frac{\partial L(\boldsymbol{x})}{\partial x_j} = \sum_{i=1}^{p}(-d_{ij})(y_i - \sum_{j=1}^{N} d_{ij}x_j) + \lambda_j sgn(x_j) - \frac{x_j}{\gamma_j}. \tag{B.4}$$

Let $r_i^{(j)} = y_i - \sum_{k\neq j}^{N} d_{ik}x_k$, then

$$\begin{aligned} \frac{\partial L(\boldsymbol{x})}{\partial x_j} &= \sum_{i=1}^{p}(-d_{ij})(r_i^{(j)} - d_{ij}x_j) + \lambda_j sgn(x_j) - \frac{x_j}{\gamma_j} \\ &= \sum_{i=1}^{p}(d_{ij}d_{ij}x_j - d_{ij}r_i^{(j)}) + \lambda_j sgn(x_j) - \frac{x_j}{\gamma_j} \\ &= x_j - \boldsymbol{d}_j^T\boldsymbol{r}^{(j)} + \lambda_j sgn(x_j) - \frac{x_j}{\gamma_j}, \end{aligned} \tag{B.5}$$

where $sgn(\cdot)$ is the sign function. We expand the term $\boldsymbol{d}_j^T\boldsymbol{r}^{(j)}$ as follows:

$$\begin{aligned} \sum_{j=1}^{p} d_{ij}r_i^{(j)} &= \sum_{i=1}^{p} d_{ij}(y_i - \sum_{k\neq j}^{N} d_{ik}x_k) \\ &= \sum_{i=1}^{p} d_{ij}y_i - \sum_{i=1}^{p}\sum_{k\neq j}^{N} d_{ij}d_{ik}x_k \\ &= \sum_{i=1}^{p} d_{ij}y_i \\ &= \boldsymbol{d}_j^T\boldsymbol{y}. \end{aligned} \tag{B.6}$$

Therefore, Eq. (B.4) can be rewritten as follows:

$$\frac{\partial L(\boldsymbol{x})}{\partial x_j} = x_j - \boldsymbol{d}_j^T \boldsymbol{y} - \lambda_j sgn(x_j) - \frac{x_j}{\gamma_j}. \tag{B.7}$$

Soft thresholding is performed to solve Eq. (B.7),

$$\hat{x_j} = \frac{\gamma_j - 1}{\gamma_j} \cdot sgn(\boldsymbol{d}_j^T \boldsymbol{y}) \cdot max(1 - \frac{\lambda_j}{\boldsymbol{d}_j^T \boldsymbol{y}}, 0), \tag{B.8}$$

for $|\boldsymbol{d}_j^T \boldsymbol{y}| \leq \gamma_j \lambda_j$.

2. For the case of $|x_j| > \gamma_j \lambda_j$, $\forall j$, we have

$$L(\boldsymbol{x}) = \sum_{i=1}^{p} (y_i - \sum_{j=1}^{N} d_{ij} x_j)^2 + \sum_{j=1}^{N} \frac{1}{2} \gamma_j \lambda_j. \tag{B.9}$$

Solving Eq. (B.9) is equivalent to solving the ordinary regression, and hence, the solution can be obtained as follows:

$$x_j = \boldsymbol{d}_j^T \boldsymbol{y}, \tag{B.10}$$

for $|\boldsymbol{d}_j^T \boldsymbol{y}| > \gamma_j \lambda_j$.

Based on the above analysis, the solution of the sparse model with the generalized MCP has an analytical form, as follows:

$$x_j = \begin{cases} \frac{\gamma_j - 1}{\gamma_j} \cdot sgn(\boldsymbol{d}_j^T \boldsymbol{y}) max(1 - \frac{\lambda_j}{\boldsymbol{d}_j^T \boldsymbol{y}}, 0), & \text{if } |\boldsymbol{d}_j^T \boldsymbol{y}| \leq \gamma_j \lambda_j, \\ \boldsymbol{d}_j^T \boldsymbol{y}, & \text{if } |\boldsymbol{d}_j^T \boldsymbol{y}| > \gamma_j \lambda_j. \end{cases} \tag{B.11}$$

## Reference

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

[2] J.-L. Starck, E. J. Candès, D. L. Donoho, The curvelet transform for image denoising, IEEE Trans. on Image Processing 11 (6) (2002) 670–684.

[3] J. Zhang, D. Zhao, W. Gao, Group-based sparse representation for image restoration, IEEE Trans. on Image Processing 23 (8) (2014) 3336–3351.

[4] J. Xu, L. Zhang, W. Zuo, D. Zhang, X. Feng, Patch group based nonlocal self-similarity prior learning for image denoising, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 244–252.

[5] M. Aharon, M. Elad, A. Bruckstein, K-svd: An algorithm for designing overcomplete dictionaries for sparse representation, IEEE Trans. on Signal Processing 54 (11) (2006) 4311–4322.

[6] J. Mairal, F. R. Bach, J. Ponce, G. Sapiro, A. Zisserman, Non-local sparse models for image restoration., in: Proceedings of the IEEE International Conference on Computer Vision, 2009, pp. 54–62.

[7] W. Dong, L. Zhang, G. Shi, X. Li, Nonlocally centralized sparse representation for image restoration, IEEE Trans. on Image Processing 22 (4) (2012) 1620–1630.

[8] J. Salmon, Z. Harmany, C.-A. Deledalle, R. Willett, Poisson noise reduction with non-local pca, Journal of mathematical imaging and vision 48 (2) (2014) 279–294.

[9] L. Azzari, A. Foi, Variance stabilization for noisy+ estimate combination in iterative poisson denoising, IEEE Signal Processing Letters 23 (8) (2016) 1086–1090.

[10] F. Luisier, C. Vonesch, T. Blu, M. Unser, Fast interscale wavelet denoising of poisson-corrupted images, Signal Processing 90 (2) (2010) 415–427.

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

[12] F. Luisier, T. Blu, M. Unser, Image denoising in mixed poisson–gaussian noise, IEEE Trans. on image processing 20 (3) (2010) 696–708.

[13] M. Makitalo, A. Foi, Optimal inversion of the generalized anscombe transformation for poisson-gaussian noise, IEEE Trans. on Image Processing 22 (1) (2012) 91–103.

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

[15] C. Zou, Y. Xia, Bayesian dictionary learning for hyperspectral image super resolution in mixed poisson–gaussian noise, Signal Processing: Image Communication 60 (2018) 29–41.

[16] K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang, Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising, IEEE Trans. on Image Processing 26 (7) (2017) 3142–3155.

[17] K. Zhang, W. Zuo, L. Zhang, Ffdnet: Toward a fast and flexible solution for cnn-based image denoising, IEEE Trans. on Image Processing 27 (9) (2018) 4608–4622.

[18] S. Guo, Z. Yan, K. Zhang, W. Zuo, L. Zhang, Toward convolutional blind denoising of real photographs, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 1712–1722.

[19] K. Yu, X. Wang, C. Dong, X. Tang, C. C. Loy, Path-restore: Learning network path selection for image restoration, arXiv preprint arXiv:1904.10343.

[20] C. Tomasi, R. Manduchi, Bilateral filtering for gray and color images., in: Proceedings of the IEEE International Conference on Computer Vision, Vol. 98, 1998, p. 2.

[21] J. Xu, L. Zhang, D. Zhang, External prior guided internal prior learning for real-world noisy image denoising, IEEE Trans. on Image Processing 27 (6) (2018) 2996–3010.

[22] Z. Lyu, C. Zhang, M. Han, A nonsubsampled countourlet transform based cnn for real image denoising, Signal Processing: Image Communication 82 (2020) 115727.

[23] W. Shi, F. Jiang, S. Zhang, R. Wang, D. Zhao, H. Zhou, Hierarchical residual learning for image denoising, Signal Processing: Image Communication 76 (2019) 243–251.

[24] Y. Guo, X. Jia, B. Zhao, H. Chai, Y. Huang, Multifeature extracting cnn with concatenation for image denoising, Signal Processing: Image Communication 81 (2020) 115690.

[25] Y. Liu, S. Canu, P. Honeine, S. Ruan, Mixed integer programming for sparse coding: Application to image denoising, IEEE Trans. on Computational Imaging.

[26] L. Liu, L. Chen, C. P. Chen, Y. Y. Tang, et al., Weighted joint sparse representation for removing mixed noise in image, IEEE Trans. on Cybernetics 47 (3) (2016) 600–611.

[27] F. Wen, L. Adhikari, L. Pei, R. F. Marcia, P. Liu, R. C. Qiu, Nonconvex regularization-based sparse recovery and demixing with application to color image inpainting, IEEE Access 5 (2017) 11513–11527.

[28] D. Liu, Z. Wang, B. Wen, J. Yang, W. Han, T. S. Huang, Robust single image super-resolution via deep networks with sparse prior, IEEE Trans. on Image Processing 25 (7) (2016) 3194–3207.

[29] H. Zhuo, K.-M. Lam, Wavelet-based eigentransformation for face super-resolution, in: Pacific-Rim Conference on Multimedia, Springer, 2010, pp. 226–234.

[30] Y. Zhang, J. Liu, W. Yang, Z. Guo, Image super-resolution based on structure-modulated sparse representation, IEEE Trans. on Image Processing 24 (9) (2015) 2797–2810.

[31] Z. Hui, K.-M. Lam, Eigentransformation-based face super-resolution in the wavelet domain, Pattern Recognition Letters 33 (6) (2012) 718–727.

[32] D. Li, X. Xie, K.-M. Lam, Color correction with blind image restoration based on multiple images using a low-rank model, Journal of Electronic Imaging 23 (2) (2014) 023010.

[33] Y. Li, C. Cai, G. Qiu, K.-M. Lam, Face hallucination based on sparse local-pixel structure, Pattern Recognition 47 (3) (2014) 1261–1270.

[34] M. Jian, K.-M. Lam, J. Dong, A novel face-hallucination scheme based on singular value decomposition, Pattern Recognition 46 (11) (2013) 3091–3102.

[35] Z. Hui, K.-M. Lam, Multi-view face hallucination based on sparse representation, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, 2013, pp. 2202–2206.

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

[37] D. Cho, T. D. Bui, Multivariate statistical modeling for image denoising using wavelet transforms, Signal Processing: Image Communication 20 (1) (2005) 77–89.

[38] S. Gu, L. Zhang, W. Zuo, X. Feng, Weighted nuclear norm minimization with application to image denoising, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2862–2869.

[39] J. Xu, L. Zhang, D. Zhang, X. Feng, Multi-channel weighted nuclear norm minimization for real color image denoising, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1096–1104.

[40] J. Xu, L. Zhang, D. Zhang, A trilateral weighted sparse coding scheme for real-world image denoising, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 20–36.

[41] C.-H. Zhang, et al., Nearly unbiased variable selection under minimax concave penalty, The Annals of Statistics 38 (2) (2010) 894–942.

[42] R. Mazumder, J. H. Friedman, T. Hastie, Sparsenet: Coordinate descent with nonconvex penalties, Journal of the American Statistical Association 106 (495) (2011) 1125–1138.

[43] T. Zhang, Analysis of multi-stage convex relaxation for sparse regularization, Journal of Machine Learning Research 11 (Mar) (2010) 1081–1107.

[44] J. Mairal, F. Bach, J. Ponce, G. Sapiro, Online learning for matrix factorization and sparse coding, Journal of Machine Learning Research 11 (Jan) (2010) 19–60.

[45] R. Jenatton, J. Mairal, G. Obozinski, F. R. Bach, Proximal methods for sparse hierarchical dictionary learning., in: Proceedings of International Conference on Machine Learning, Vol. 1, Citeseer, 2010, p. 2.

[46] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society: Series B (Methodological) 58 (1) (1996) 267–288.

[47] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, et al., Least angle regression, The Annals of Statistics 32 (2) (2004) 407–499.

[48] J. Friedman, T. Hastie, H. Höfling, R. Tibshirani, et al., Pathwise coordinate optimization, The Annals of Applied Statistics 1 (2) (2007) 302–332.

[49] J. Fan, R. Li, Variable selection via nonconcave penalized likelihood and its oracle properties, Journal of the American statistical Association 96 (456) (2001) 1348–1360.

[50] J. H. Friedman, Fast sparse regression and classification, International Journal of Forecasting 28 (3) (2012) 722–738.

[51] J. Shi, X. Ren, G. Dai, J. Wang, Z. Zhang, A non-convex relaxation approach to sparse dictionary learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 1809–1816.

[52] S. Nam, Y. Hwang, Y. Matsushita, S. J. Kim, A holistic approach to cross-channel image noise modeling and its application to image denoising, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[53] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3-d transform-domain collaborative filtering, IEEE Trans. on Image Processing 16 (8) (2007) 2080–2095.

[54] D. Zoran, Y. Weiss, From learning models of natural image patches to whole image restoration, in: Proceedings of the IEEE International Conference on Computer Vision, 2011, pp. 479–486.

[55] W. Zhao, Y. Lv, Q. Liu, B. Qin, Detail-preserving image denoising via adaptive clustering and progressive pca thresholding, IEEE Access 6 (2017) 6303–6315.

[56] K. Dabov, A. Foi, V. Katkovnik, K. O. Egiazarian, Color image denoising via sparse 3d collaborative filtering with grouping constraint in luminance-chrominance space., in: Proceedings of the IEEE Conference on Image Processing, 2007, pp. 313–316.

[57] U. Schmidt, S. Roth, Shrinkage fields for effective image restoration, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2774–2781.

[58] Y. Chen, T. Pock, Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration, IEEE Trans. on Pattern Analysis and Machine Intelligence 39 (6) (2016) 1256–1272.

[59] M. Lebrun, M. Colom, J.-M. Morel, Multiscale image blind denoising, IEEE Trans. on Image Processing 24 (10) (2015) 3149–3161.

[60] Z. Kong, X. Yang, Color image and multispectral image denoising using block diagonal representation, IEEE Trans. on Image Processing.

[61] J. Xu, H. Li, Z. Liang, D. Zhang, L. Zhang, Real-world noisy image denoising: A new benchmark, arXiv preprint arXiv:1804.02603.

[62] R. L. Strawderman, M. T. Wells, E. D. Schifano, et al., Hierarchical bayes, maximum a posteriori estimators, and minimax concave penalized likelihood estimation, Electronic Journal of Statistics 7 (2013) 973–990.