Self-Supervised Image Prior Learning with GMM from a Single Noisy Image

Haosen Liu[1,2], Xuan Liu[1], Jiangbo Lu[2], Shan Tan[1]*
[1]Huazhong University of Science and Technology, [2]SmartMore Corporation
{haosen.liu0803, jiangbo.lu}@gmail.com, {liuxuan99, shantan}@hust.edu.cn

Abstract

The lack of clean images undermines the practicability of supervised image prior learning methods, of which the training schemes require a large number of clean images. To free image prior learning from the image collection burden, a novel Self-Supervised learning method for Gaussian Mixture Model (SS-GMM) is proposed in this paper. It can simultaneously achieve the noise level estimation and the image prior learning directly from only a single noisy image. This work is derived from our study on eigenvalues of the GMM's covariance matrix. Through statistical experiments and theoretical analysis, we conclude that (1) covariance eigenvalues for clean images hold the sparsity; and that (2) those for noisy images contain sufficient information for noise estimation. The first conclusion inspires us to impose a sparsity constraint on covariance eigenvalues during the learning process to suppress the influence of noise. The second conclusion leads to a self-contained noise estimation module of high accuracy in our proposed method. This module serves to estimate the noise level and automatically determine the specific level of the sparsity constraint. Our final derived method requires only minor modifications to the standard expectation-maximization algorithm. This makes it easy to implement. Very interestingly, the GMM learned via our proposed self-supervised learning method can even achieve better image denoising performance than its supervised counterpart, i.e.*, the EPLL. Also, it is on par with the state-of-the-art self-supervised deep learning method,* i.e.*, the Self2Self. Code is available at https://github.com/HUST-Tan/SS-GMM.*

1. Introduction

Image modeling and prior learning play key roles in the design of an image denoising algorithm. Traditional image modeling methods are generally hand-designed. These methods can be classified into two main categories [13], *i.e.*, the analysis-based methods and the synthesis-based ones. The analysis-based methods directly model the image itself [26, 4, 5, 17, 18], while synthesis-based methods model coefficients of an image in a transform-domain [10, 12, 23, 11]. The main drawback of traditional hand-designed methods is that images in the real world are too complex to be effectively modeled with simplified assumptions. To tackle this problem, the data-driven learning method is utilized as an alternative. Some representatives in this branch include [25, 33, 27, 7, 32]. These methods have indeed achieved great success in image denoising tasks. However, their state-of-the-art performances are based on acquiring a large number of clean images, which are generally unavailable in practice. This undermines the practicability of these methods.

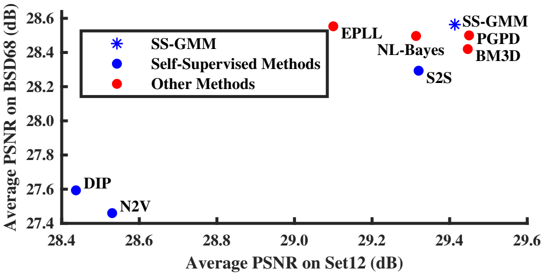


Figure 1. Average PSNR of image denoising results for three noise levels σ=15, 25, 50 on Set12 and BSD68. These results are calculated based on Table 1. Comparison methods include a) the non-learning method: BM3D [8]; b) GMM-related methods: EPLL [33], PGPD [30] and NL-Bayes [16]; c) Self-supervised deep learning methods: N2V [15], DIP [28] and S2S [22]; d) Our proposed self-supervised GMM (SS-GMM).

To reduce or even eliminate the image collection burden, the self-supervised learning method, which only requires noisy images themselves during the training process, has attracted more and more attention. One recent work in this field is the deep image prior (DIP) [28]. It performs the self-supervised learning for the generator network with an early-stopping strategy. Since the early-stopping strategy acts as an implicit constraint on network parameters [14], an idea coming to our mind is that *imposing reason-*

*Corresponding author. This work was supported in part by the National Natural Science Foundation of China (NNSFC), under Grant Nos. 61672253 and 62071197. Part of this work was done when Haosen was interning in SmartMore Co., Ltd.

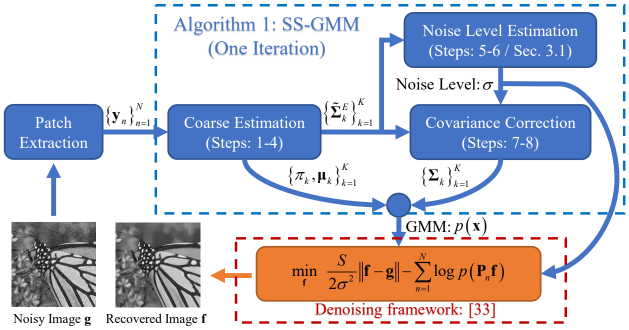

Algorithm 1: SS-GMM
(One Iteration)
Noise Level Estimation
(Steps: 5-6 / Sec. 3.1)
Noise Level: σ
Patch Extraction
$\{\mathbf{y}_n\}_{n=1}^{N}$
Coarse Estimation
(Steps: 1-4)
$\{\tilde{\mathbf{\Sigma}}_k^E\}_{k=1}^{K}$
Covariance Correction
(Steps: 7-8)
$\{\pi_k, \mathbf{\mu}_k\}_{k=1}^{K}$
$\{\mathbf{\Sigma}_k\}_{k=1}^{K}$
GMM: $p(\mathbf{x})$
$\min_{\mathbf{f}} \frac{S}{2\sigma^2}\|\mathbf{f}-\mathbf{g}\| - \sum_{n=1}^{N} \log p(\mathbf{P}_n \mathbf{f})$
Denoising framework: [33]
Noisy Image $\mathbf{g}$
Recovered Image $\mathbf{f}$

Figure 2. Overall schematic figure.

able constraints on parameters might be the key for learning generative models in a self-supervised way. Motivated by this idea, we conducted a study on how to achieve self-supervised learning for the classical generative model, *i.e.*, the Gaussian Mixture Model (GMM).

To achieve our goal, we at first did a study on the influence of noise on GMM's parameters. A common assumption that the noise follows the $i.i.d.$ Gaussian distribution is adopted in this paper. We observed that, in a statistical sense, the existence of Gaussian noise only affects eigenvalues of the GMM's covariance matrices. Therefore, we focused our further study on the property of covariance eigenvalues. Through statistical experiments and theoretical analysis, we obtained two important conclusions: 1) Covariance eigenvalues for clean images hold the sparsity, while those for noisy images do not; 2) The degree to which eigenvalues for noisy images deviate from the sparsity is determined by the noise level. The first conclusion inspires us to impose a sparsity constraint on covariance eigenvalues to suppress the influence of noise. The second one further indicates that the level of this sparsity constraint should be related to the noise level. Also, it suggests that eigenvalues for noisy images contain sufficient information for the noise estimation, *i.e.*, the degree to which these eigenvalues deviate from the sparsity.

Based on these conclusions, we newly proposed a **S**elf-**S**upervised **GMM** (SS-GMM) that can learn parameters from only a single noisy image. As Fig. 2 shows, SS-GMM at first conducts a coarse estimation of GMM's parameters. This estimation is the same as the standard Expectation-Maximization based GMM learning method (EM-GMM) [2, 3]. Covariance matrices learned by this method are vulnerable to noise. Therefore, a covariance correction module, accompanied by a self-contained noise level estimation module, is further conducted to suppress the influence of noise. To evaluate the effectiveness of SS-GMM, it is applied to the image denoising task with the framework proposed in [33]. As Fig. 1 shows, SS-GMM outperforms its supervised counterpart, *i.e.*, Expected Patch Log Likelihood (EPLL) [33]. Also, it is on par with the state-of-the-art self-supervised deep learning method, *i.e.*, Self2Self [22].

In summary, our main contributions are:

- Conducting a detailed study on the influence of Gaussian noise on GMM's parameters, especially on the covariance eigenvalues, through statistical experiments and theoretical analysis.
- Proposing a self-contained noise level estimation module for our proposed self-supervised algorithm to achieve noise level estimation and to help determine the level of the sparsity constraint.
- Developing an efficient self-supervised learning algorithm, which is easy to implement, for the GMM to achieve the image prior learning from only a single noisy image.

2. Related Work

This section provides a brief comparison of our proposed method to several highly related works.

2.1. GMM Based Prior Learning Methods

As a classical generative model, GMM has been successfully applied to patch based image prior modeling. One representative is the EPLL [33]. In [30], the Patch Group based Image Denoising (PGPD) extended the GMM to the patch group based version to model the image non-local self-similarity (NSS) prior. These methods both require a set of clean images as the training data. This undermines the practical value of these methods. Some existing solutions to this problem require either a suitable parameter initialization [31] or the search of non-local similar patches, *e.g.*, Nonlocal Bayes (NL-Bayes) [16]. Besides, all of these GMM-related methods have to be provided the noise level as a hyper-parameter in advance, since they do not contain a noise estimation module. By contrast, our proposed method does not need any extra information except such necessary parameters as the patch size, but also achieves comparable or even better performances than EPLL [33], PGPD [30] and NL-Bayes [16].

2.2. Self-Supervised Deep Learning Methods

The self-supervised deep learning method aims at achieving network training directly from the input data. The Noise2Self (N2S) [1], Noise2Void (N2V) [15], Self2Self (S2S) [22], and Deep Image Prior (DIP) [28] all belong to this kind of methods. The former three methods are based on the blind-spot strategy and the dropout strategy. The key for these two strategies is to remove part of pixels from the network receptive field to avoid learning an identity function. However, recovering the removed pixels itself is an image inpainting problem. Therefore, these strategies actually increase the problem complexity. The DIP provides another possibility of realizing self-supervised learning methods. As introduced above, it regularizes network parameters with the early stopping strategy. Inspired by DIP, we propose to impose a sparsity constraint on covariance eigenvalues. Different from the early-stopping, this sparsity constraint is supported by strict theoretical analysis. Owing to this, our proposed method outperforms the DIP with a large margin, although the iteration number of DIP has been tuned by hand to achieve the highest PSNR. See Fig. 1 for an intuitive comparison.

2.3. Noise Estimation Methods

Traditional methods such as [24] can only accurately estimate noise levels for images with sufficient flat regions. Several recently proposed methods [19, 21] overcome this limitation by taking the smallest covariance eigenvalues of selected low-rank patches as the noise level. Through the statistical analysis, Chen *et al*. [6] demonstrated that these methods systematically underestimate the noise level, and proposed to take the mean value of the smallest eigenvalue to the m-th smallest one as the noise level, where m is a hyper-parameter that can be automatically determined. In this paper, we extend the statistical analysis in [6] from a single Gaussian distribution to the GMM. Our experiments showed that the histogram of covariance eigenvalues for GMM holds obvious asymmetry. This violates the theoretical basis of [6]. To tackle this problem, we propose a new method focusing the mean value calculation on the histogram's peak region, which is locally symmetrical with the noise level as the center. More importantly, our work reveals how the noise level estimation is related to the self-supervised image prior learning process, *i.e*., the estimated noise level can be further used to determine the level of constraint imposed on parameters. This is not covered by previous works in the field of noise level estimation.

3. Method

Theoretically speaking, the GMM can fit any distribution [3]. It has been successfully used to model the image patch prior, using a set of clean images as the training set [33]. However, clean images are unavailable in many applications. To overcome this problem, a novel self-supervised learning method is proposed for the GMM in this section.

The GMM is a linear composition of K Gaussian distributions. It can be written as

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \cdot \mathcal{N}\left(\mathbf{x}; \boldsymbol{\mu}_k, \mathbf{\Sigma}_k\right), \tag{1}$$

where $\mathbf{x}$ is an S-dimensional vector denoting the image patch in this paper, π_k is the mixing coefficient that adds up to 1 and satisfies $0 \leq \pi_k \leq 1$, $\mathcal{N}(\cdot)$ is the Gaussian distribution calculated as

$$\frac{1}{(2\pi)^{S/2}} \frac{1}{|\mathbf{\Sigma}_k|^{1/2}} exp\left(-\frac{1}{2}\left(\mathbf{x}-\boldsymbol{\mu}_k\right)^T \mathbf{\Sigma}_k^{-1}\left(\mathbf{x}-\boldsymbol{\mu}_k\right)\right), \tag{2}$$

where $|\cdot|$ denotes the determinant operator, $\boldsymbol{\mu}_k$ and $\mathbf{\Sigma}_k$ are the mean vector and the covariance matrix, respectively.

According to the Gaussian distribution's property, if an image patch $\mathbf{x}$ belongs to a GMM with parameters $\{\pi_k, \boldsymbol{\mu}_k, \mathbf{\Sigma}_k\}_{k=1}^{K}$, and the noise $\mathbf{n}$ is assumed to be the $i.i.d.$ additive Gaussian noise of noise level[1] σ, then the noisy image patch $\mathbf{y}$ will belong to the GMM with parameters $\left\{\pi_k, \boldsymbol{\mu}_k, \mathbf{\Sigma}_k + \sigma^2 \cdot \mathbf{I}\right\}_{k=1}^{K}$. Here, $\mathbf{I}$ denotes the identity matrix. As a result, learning from noisy image patches without any other constraints can only estimate the matrix $\tilde{\mathbf{\Sigma}}_k = \mathbf{\Sigma}_k + \sigma^2 \cdot \mathbf{I}$, where the latent covariance matrix and noise level are mutually coupled. This is the main challenge for learning GMM with noisy data.

For each Gaussian component, its covariance matrix must be positive definite and thus can be decomposed as $\mathbf{\Sigma}_k = \mathbf{D}_k \mathbf{\Lambda}_k \mathbf{D}_k^T$, where $\mathbf{D}_k$ is an orthogonal matrix and $\mathbf{\Lambda}_k$ denotes a diagonal matrix with eigenvalues $\{\lambda_{ks}\}_{s=1}^{S}$ as the diagonal element. Using this decomposition, $\tilde{\mathbf{\Sigma}}_k$ can be written as

$$\tilde{\mathbf{\Sigma}}_k = \mathbf{\Sigma}_k + \sigma^2 \mathbf{I} = \mathbf{D}_k\left(\mathbf{\Lambda}_k + \sigma^2 \mathbf{I}\right)\mathbf{D}_k^T = \mathbf{D}_k \tilde{\mathbf{\Lambda}}_k \mathbf{D}_k^T, \tag{3}$$

which is just the eigenvalue decomposition of $\tilde{\mathbf{\Sigma}}_k$. This means that $\mathbf{\Sigma}_k$ and $\tilde{\mathbf{\Sigma}}_k$ have the same eigenvectors, and the difference between eigenvalues of these two matrices is determined by the noise level, *i.e*., $\tilde{\lambda}_{ks} = \lambda_{ks} + \sigma^2$. Now, the key problem is how to decouple λ_{ks} and the noise level σ from $\tilde{\lambda}_{ks}$.

3.1. Statistical Property of Covariance Eigenvalues

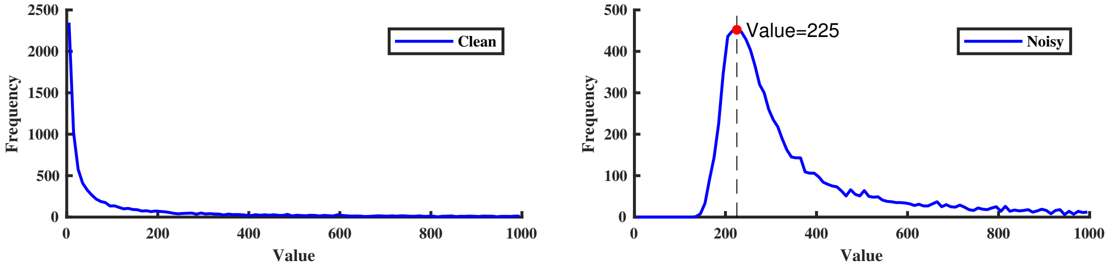
To tackle this problem, we at first did a study on the statistical property of covariance eigenvalues. In Fig. 3, histograms of covariance eigenvalues for (a) clean images and (b) noisy images are plotted. To obtain Fig. 3(a), 400×3600 patches of the size 7×7 are extracted from 400 clean images of the size 180×180 [7] to train a GMM model of 200 components with the EM-GMM algorithm [2, 3]. Then, eigenvalues are calculated and used to plot the histogram. For Fig. 3(b), images are added with Gaussian noise of σ=15.

Intuitively, most of eigenvalues λ_{ks} learned from clean images accumulate around zero, implying that eigenvalues λ_{ks} should hold the sparsity. More interestingly, eigenvalues $\tilde{\lambda}_{ks}$ learned from noisy images gather around $\sigma^2 = 225$, where the peak frequency occurs. This raises a question, *i.e*., *why does this histogram peak at* σ^2? To answer this, the theoretical analysis of Fig. 3 is provided as follows.

In practice, $\mathbf{\Sigma}_k$ and $\tilde{\mathbf{\Sigma}}_k$ are not available. We can only estimate them empirically with the training data. For the EM-GMM algorithm, $\mathbf{\Sigma}_k$ is estimated as

$$\mathbf{\Sigma}_k^E = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk}(\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T, \tag{4}$$

[1] For convenience, we use σ and σ^2 for the noise level interchangeably.



(a) Histogram of λ^E

(b) Histogram of $\tilde{\lambda}^E$

Figure 3. Histograms of covariance eigenvalues learned by EM-GMM from (a) clean images and (b) images added with Gaussian noise of the noise level $\sigma = 15$.

where N is the number of training patches, γ_{nk} denotes the probability of which the n-th image patch belongs to the k-th Gaussian component, N_k is the summation of γ_{nk} over n. If the GMM is well trained and it can classify each image patch with high accuracy, implying $\gamma_{nk} \approx 1$ or 0, there is

$$\mathbf{\Sigma}_k^E \approx \frac{1}{N_k} \sum_{n \in S_k} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T, \tag{5}$$

where S_k denotes the set containing all of the patches that belong to the k-th Gaussian component, and N_k denotes the number of patches in S_k.

Based on [6], if N_k is large enough (*e.g.*, $N_k > 1000$), the eigenvalue of $\mathbf{\Sigma}_k^E$ in Eq. (5) approximately belongs to a Gaussian distribution, *i.e.*,

$$p\left(\lambda_{ks}^E\right) \approx \mathcal{N}\left(\lambda_{ks}^E; \lambda_{ks}, \frac{2\lambda_{ks}^2}{N_k}\right), \tag{6}$$

which further means that an eigenvalue λ^E randomly picked from $\{\lambda_{ks}^E\}_{k=1,s=1}^{K,S}$ will belong to a GMM, *i.e.*,

$$p\left(\lambda^E\right) \approx \frac{1}{S \cdot N} \sum_{k=1}^{K} \sum_{s=1}^{S} \mathcal{N}\left(\lambda_{ks}^E; \lambda_{ks}, \frac{2\lambda_{ks}^2}{N_k}\right). \tag{7}$$

The histogram shown in Fig. 3(a) is essentially determined by the probability distribution $p(\lambda^E)$. To generate such a histogram, most of the eigenvalues λ_{ks} have to be close to zero, and others will distribute over a wide range. In other words, eigenvalues for clean images hold the sparsity.

Similarly, if N_k is large enough, for eigenvalues corresponding to noisy images, there is

$$p\left(\tilde{\lambda}^E\right) \approx \frac{1}{S \cdot N} \sum_{k=1}^{K} \sum_{s=1}^{S} \mathcal{N}\left(\tilde{\lambda}_{ks}^E; \tilde{\lambda}_{ks}, \frac{2\tilde{\lambda}_{ks}^2}{N_k}\right). \tag{8}$$

Since most of λ_{ks} are close to zero, most of $\tilde{\lambda}_{ks}$ will be approximate to be σ^2, *i.e.*, $\tilde{\lambda}_{ks} = \lambda_{ks} + \sigma^2 \approx \sigma^2$. That is to say, a large proportion of components in Eq. (8) are Gaussian distributions with σ^2 as their mean values. Since a Gaussian distribution peaks at its mean value point, these components collectively constitute the most significant feature of $p(\tilde{\lambda}^E)$, making its corresponding histogram shown in Fig. 3(b) peak at σ^2. This answers the question mentioned above.

Besides components whose mean values are about σ^2, Eq. (8) also includes components whose mean values are much larger than σ^2. As Fig. 3(b) shows, these components collectively make the histogram curve fall much more slowly on the right side than on the left side. This obvious asymmetry undermines the theoretical basis of applying the noise level estimation method proposed in [6] to the GMM. The work [6] is originally designed for a single Gaussian distribution. Its basic assumption can be regarded as that there exists a separate point that can divide the histogram of eigenvalues into a symmetrical part and the other part. However, it is obvious that such a point does not exist in Fig. 3(b). To tackle this problem, we proposed to focus the noise level estimation on the histogram's peak region, which is locally symmetrical with $\tilde{\lambda}_{ks}^E = \sigma^2$ as the symmetry axis. This is achieved with the following steps: a) Divide $\{\tilde{\lambda}_{ks}^E\}_{k=1,s=1}^{K,S}$ into several bins, and count the number of $\tilde{\lambda}_{ks}^E$ for each bin; b) Find the bin having the largest number. Denote this number as c_{max}; c) Find bins whose numbers are larger than $r \cdot c_{max}$ $(0 < r < 1)$, and take the average value of eigenvalues located in these bins as the estimated noise level. In this procedure, parameter r is used to control the peak region's size. It is set as 0.5 in this paper.

3.2. Self-Supervised Learning Method

The standard parameter learning method for GMM, *i.e.*, EM-GMM [2, 3], is to maximize the likelihood function as

$$\max_{\pi_k, \boldsymbol{\mu}_k, \tilde{\mathbf{\Sigma}}_k} \frac{1}{N} \sum_{n=1}^{N} \ln\left(\sum_{k=1}^{K} \pi_k \cdot \mathcal{N}\left(\mathbf{y}_n; \boldsymbol{\mu}_k, \tilde{\mathbf{\Sigma}}_k\right)\right),$$

$$s.t. \quad 0 \leq \pi_k \leq 1, \quad \sum_{k=1}^{K} \pi_k = 1, \tag{9}$$

of which the solution is

$$\pi_k = \frac{N_k}{N}, \tag{10a}$$

$$\boldsymbol{\mu}_k = \frac{1}{N_k}\sum_{n=1}^{N}\gamma_{nk}\cdot\mathbf{y}_n, \tag{10b}$$

$$\tilde{\boldsymbol{\Sigma}}_k^E = \frac{1}{N_k}\sum_{n=1}^{N}\gamma_{nk}(\mathbf{y}_n-\boldsymbol{\mu}_k)(\mathbf{y}_n-\boldsymbol{\mu}_k)^T \tag{10c}$$

where $N_k = \sum_{n=1}^{N}\gamma_{nk}$ and γ_{nk} is calculated as

$$\gamma_{nk} = \frac{\pi_k\cdot\mathcal{N}\left(\mathbf{y}_n;\boldsymbol{\mu}_k,\tilde{\boldsymbol{\Sigma}}_k\right)}{\sum\limits_{k=1}^{K}\pi_k\cdot\mathcal{N}\left(\mathbf{y}_n;\boldsymbol{\mu}_k,\tilde{\boldsymbol{\Sigma}}_k\right)}. \tag{11}$$

As shown in Fig. 3(b), covariance estimated by EM-GMM is vulnerable to noise. To suppress the influence of noise, we propose to introduce several extra constraints on covariance matrices into Eq. (9), which leads to

$$\begin{aligned}
&\max_{\boldsymbol{\theta}} \quad \frac{1}{N}\sum_{n=1}^{N}\ln\left(\sum_{k=1}^{K}\pi_k\cdot\mathcal{N}\left(\mathbf{y}_n;\boldsymbol{\mu}_k,\tilde{\boldsymbol{\Sigma}}_k\right)\right),\\
&s.t. \quad 0\le\pi_k\le 1, \quad \sum_{k=1}^{K}\pi_k = 1,\\
&\qquad \tilde{\boldsymbol{\Sigma}}_k = \mathbf{D}_k\cdot\tilde{\boldsymbol{\Lambda}}_k\cdot\mathbf{D}_k^T, \quad \mathbf{D}_k^T\cdot\mathbf{D}_k = \mathbf{I},\\
&\qquad \tilde{\boldsymbol{\Lambda}}_k = \boldsymbol{\Lambda}_k + \sigma^2\cdot\mathbf{I}, \quad \lambda_{ks}\ge 0,\\
&\qquad \sum_{k=1}^{K}\sum_{s=1}^{S}\|\lambda_{ks}\|_0 \le L,
\end{aligned} \tag{12}$$

where parameters $\boldsymbol{\theta}$ to be optimized here include the noise level σ^2 and GMM parameters $\{\pi_k,\boldsymbol{\mu}_k,\mathbf{D}_k,\boldsymbol{\Lambda}_k\}_{k=1}^{K}$, λ_{ks} is the s-th diagonal element of the diagonal matrix $\boldsymbol{\Lambda}_k$, and $\|\cdot\|_0$ denotes the l_0-norm,

In this newly proposed problem, constraints at the 3-th line and the 4-th line are obtained from Eq. (3), which clearly shows how the noise level and covariance eigenvalues are coupled. The constraint at the 5-th line is our proposed sparsity constraint on covariance eigenvalues. It is used to decouple the noise level and covariance eigenvalues. The parameter L controls the level of the sparsity constraint. As we will introduce later, it can be determined automatically.

Similar to EM-GMM, this new problem can also be effectively optimized with the EM framework [9], which alternately iterates between the so-called 'E-Step' and 'M-Step'. For this problem, its corresponding 'E-Step' is to determine γ_{nk}. This step is the same as Eq. (11). The 'M-Step' is to maximize the function

$$Q(\boldsymbol{\theta}) = \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{nk}\left(\ln\pi_k + \ln\mathcal{N}\left(\mathbf{y}_n;\boldsymbol{\mu}_k,\tilde{\boldsymbol{\Sigma}}_k\right)\right) \tag{13}$$

with constraints listed in Eq. (12). This maximization problem can be further divided into several sub-problems related to π_k, $\boldsymbol{\mu}_k$, $\mathbf{D}_k$, σ^2 and λ_{ks} (or $\boldsymbol{\Lambda}_k$), respectively. These sub-problems and their solutions are introduced as follows:

```
Algorithm 1: Self-Supervised GMM (SS-GMM)
  Input: Noisy image patches $\{\mathbf{y}_n\}_{n=1}^N$
  Output: Noise level $\sigma^2$,
          GMM parameters $\{\pi_k,\boldsymbol{\mu}_k,\mathbf{D}_k,\boldsymbol{\Lambda}_k\}_{k=1}^K$
  Initialize parameters with EM-GMM [2, 3];
  while not converge do
      1). Calculate probability $\gamma_{nk}$ as Eq. (11);
      2). Calculate mixing coefficient $\pi_k$ as Eq. (10a);
      3). Calculate mean vectors $\boldsymbol{\mu}_k$ as Eq. (10b);
      4). Calculate covariance $\tilde{\boldsymbol{\Sigma}}_k^E$ as Eq. (10c);
      5). Do eigenvalue decomposition for $\tilde{\boldsymbol{\Sigma}}_k^E$;
      6). Estimate noise level as Sec. 3.1 introduces;
      7). Determine parameter $L$ by Eq. (19);
      8). Calculate eigenvalues $\lambda_{ks}$ as Eq. (18);
  end
```

(a). **Sub-1**: Optimizing π_k and $\boldsymbol{\mu}_k$

Since the extra constraints introduced in Eq. (12) are all imposed on covariance matrices, the $(\pi_k,\boldsymbol{\mu}_k)$-related sub-problems are the same as those corresponding to Eq. (9). That is to say, the optimization of π_k and $\boldsymbol{\mu}_k$ are just the same as Eq. (10a) and Eq. (10b).

(b). **Sub-2**: Optimizing $\mathbf{D}_k$

The $\mathbf{D}_k$-related terms in 'M-Step' can be written as

$$\begin{aligned}
&\min_{\mathbf{D}_k} \quad \mathrm{Tr}\left(\tilde{\boldsymbol{\Lambda}}_k^{-1}\mathbf{D}_k^T\tilde{\boldsymbol{\Sigma}}_k^E\mathbf{D}_k\right)\\
&s.t. \quad \mathbf{D}_k^T\cdot\mathbf{D}_k = \mathbf{I},
\end{aligned} \tag{14}$$

where Tr$(\cdot)$ denotes the trace operator and $\tilde{\boldsymbol{\Sigma}}_k^E$ is calculated as Eq. (10c). This is a quadratic optimization problem with orthogonality constraints. Based on the Lemma 1 proposed in [29], we proved in the supplementary material that if diagonal elements of $\tilde{\boldsymbol{\Lambda}}_k$ are sorted in the same order as those of $\tilde{\boldsymbol{\Lambda}}_k^E$, this problem has an analytical solution as

$$\mathbf{D}_k = \tilde{\mathbf{D}}_k^E, \tag{15}$$

where $\tilde{\mathbf{D}}_k^E$ is formed from eigenvectors of $\tilde{\boldsymbol{\Sigma}}_k^E$. This is consistent with the conclusions implied by Eq (3) that $\boldsymbol{\Sigma}_k$ and $\tilde{\boldsymbol{\Sigma}}_k$ have the same eigenvectors, and that the key problem is how to decouple λ_{ks} and noise level σ.

(c). **Sub-3**: Optimizing λ_{ks} and σ^2

The (λ_{ks}, σ^2)-related terms in 'M-Step' are

$$\begin{aligned} \min_{\sigma^2, \{\lambda_{ks}\}} \quad & \ln\left|\tilde{\mathbf{\Lambda}}_k\right| + \mathrm{Tr}\left(\tilde{\mathbf{\Lambda}}_k^{-1}\mathbf{D}_k^T\tilde{\mathbf{\Sigma}}_k^E\mathbf{D}_k\right) \\ s.t. \quad & \tilde{\lambda}_{ks} = \lambda_{ks} + \sigma^2, \quad \lambda_{ks} \geq 0, \\ & \sum_{k=1}^{K}\sum_{s=1}^{S}\|\lambda_{ks}\|_0 \leq L. \end{aligned} \tag{16}$$

Since there is $\mathbf{D}_k = \tilde{\mathbf{D}}_k^E$, the objective function of this problem can be re-written as

$$\min_{\sigma^2, \{\lambda_{ks}\}} \quad \sum_{k=1}^{K}\sum_{s=1}^{S}\ln\left|\tilde{\lambda}_{ks}\right| + \frac{\tilde{\lambda}_{ks}^E}{\tilde{\lambda}_{ks}}. \tag{17}$$

If we denote the L-th largest $\tilde{\lambda}_{ks}^E$ by $\tilde{\lambda}_L^E$, the solution of this problem can be written as

$$\lambda_{ks} = \begin{cases} 0, & \tilde{\lambda}_{ks}^E < \tilde{\lambda}_L^E, \\ \tilde{\lambda}_{ks}^E - \sigma^2, & \tilde{\lambda}_{ks}^E \geq \tilde{\lambda}_L^E, \end{cases} \tag{18}$$

and

$$\sigma^2 = \frac{\sum\limits_{k=1}^{K}\sum\limits_{s=1}^{S}\left[\tilde{\lambda}_{ks}^E < \tilde{\lambda}_L^E\right]\cdot\tilde{\lambda}_L^E}{\sum\limits_{k=1}^{K}\sum\limits_{s=1}^{S}\left[\tilde{\lambda}_{ks}^E < \tilde{\lambda}_L^E\right]}, \tag{19}$$

where $[\cdot]$ is the Iverson bracket defined as

$$[\mathrm{A}] = \begin{cases} 1, & \text{if A is true;} \\ 0, & \text{otherwise.} \end{cases} \tag{20}$$

Eqs. (18)-(19) clearly demonstrate the necessity of our proposed sparsity constraint. When the sparsity constraint is invalid (*i.e.*, $L = K \cdot S$), these two equations will give undesirable results $\sigma^2 = 0$ and $\lambda_{ks} = \tilde{\lambda}_{ks}^E$. Then, *how to set a reasonable L?* Note, Eq. (19) provides an estimation of the noise level. Therefore, a reasonable L should be able to lead to an accurate noise level estimation. This inspires us to determine L based on the noise level estimated as Sec. 3.1 introduces. In our implementation, we decrease L from $KS - 1$ till the σ^2 calculated by Eq. (19) is equal or close enough to that estimated as Sec. 3.1 introduces.

In Alg. 1, the whole optimization process of problem Eq. (12) is summarized. In this algorithm, steps 1-4 are the same as the standard GMM parameter learning algorithm. Steps 5-7 are employed by the sparsity constraint. These steps serve to correct the parameters estimated by steps 1-4 to suppress the influence of noise on them. Since these steps are not complex, the whole algorithm is easy to implement.

4. Experiments

4.1. Optimization Results

Our proposed self-supervised parameter learning algorithm for GMM, dubbed as SS-GMM, is essentially an optimization algorithm designed for the problem presented in Eq. (12). To validate the convergence of this algorithm, we plotted objective function values versus iterations in Fig. 4. As one can see, our algorithm successfully converges for all example cases with just a few steps. This fast convergence rate is related to our selection of the initial point. As our theoretical analysis indicates, the only differences between GMM's parameters for clean images and those for noisy images are the eigenvalues. Therefore, parameters learned from noisy images with the EM-GMM algorithm provide a good initialization of $\{\boldsymbol{\mu}_k, \mathbf{D}_k\}_{k=1}^K$ for Alg. 1. The subsequent optimization of Alg. 1 mainly focuses on the correction of covariance eigenvalues λ_{ks}. To illustrate this, histograms of eigenvalues learned by Alg. 1 and those learned from the clean/noisy 'couple' with the EM-GMM algorithm are plotted in Fig. 5. As this figure shows, the curve 'Noisy + EM' severely deviates from the curve 'Clean + EM'. By contrast, the curve 'Noisy + Alg. 1' coincides with the curve

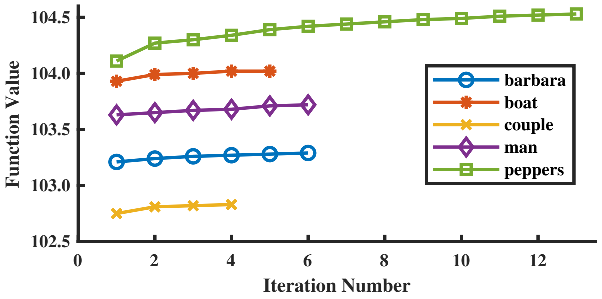


Figure 4. Objective function values versus iteration numbers.

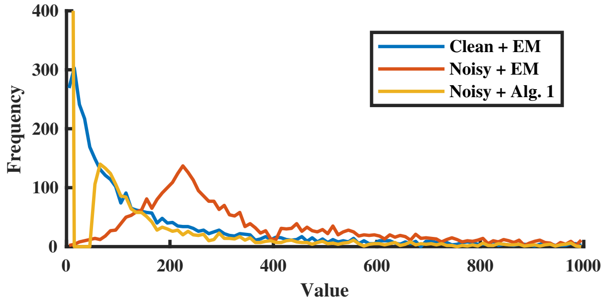


Figure 5. Histograms of eigenvalues learned by Alg. 1 and those learned from the clean/noisy 'couple' with the EM-GMM algorithm. The noise level is $\sigma = 15$.

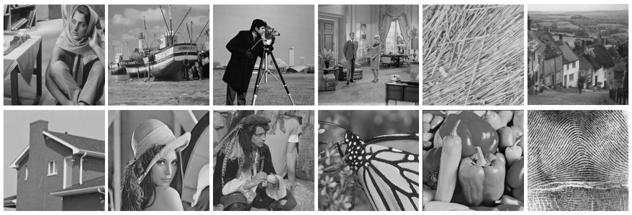


Figure 6. The 12 widely used test images. From left to right and from top to bottom: Barbara, Boat, Cameraman, Couple, Straw, Hill, House, Lena, Man, Monarch, Pepper, Fingerprint.

Table 1. Image Denoising Performances on Set12 and BSD68. The best results are highlighted in bold. The results marked with '*' are quoted from [22]. Comparison methods include BM3D [8], EPLL [33], PGPD [30], NL-Bayes [16], N2V [15], DIP [28], S2S [22] and our proposed SS-GMM.

Dataset	σ	BM3D	EPLL	PGPD	NL-Bayes	N2V	DIP	S2S	SS-GMM
Set12	15	32.12	31.83	32.13	31.98	30.73	30.90	31.83	**32.18**
	25	29.73	29.38	29.69	29.61	28.86	28.89	**29.75**	29.68
	50	26.49	26.09	**26.53**	26.35	26.00	25.52	26.38	26.38
BSD68	15	31.08	31.22	31.13	31.14	29.25	29.70	30.26	**31.26**
	25	28.56	28.72	28.62	28.69	27.69	28.00	28.70*	**28.73**
	50	25.62	25.72	25.75	25.66	25.44	25.08	**25.92***	25.70

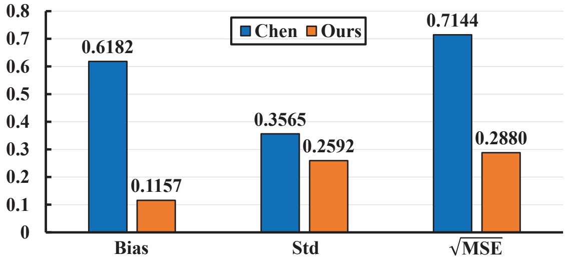


Figure 7. Noise level estimation results on Set12. The smaller these measurements are, the better an estimator is.

'Clean + EM' at most places. They are different only at small eigenvalues, of which the corresponding eigenvectors generally represent faint features [30, 33]. This clearly indicates that Alg. 1 succeeds in suppressing the influence of noise on eigenvalues.

4.2. Noise Level Estimation Results

To evaluate the performance of the self-contained noise level estimation module in Alg. 1, we apply it on 12 gray images (Set12) shown in Fig. 6. In this test, the patch size is set as 9×9 and the number of Gaussian components is set as $K = 20$ for Alg. 1. The state-of-the-art noise level estimation method proposed by Chen *et al.* [6] is adopted as the benchmark. Following [6], the qualities of noise level estimation results are quantified with three measurements, *i.e.*, Bias, Std and $\sqrt{\text{MSE}}$, which respectively evaluate the accuracy, the robustness and the overall performance of an estimator. The smaller these indexes are, the better an estimator is. The test noise levels include σ = 15, 25, 50. The average performances of Chen *et al.* [6] and our proposed method are summarized in Fig. 7. As this figure shows, our proposed estimator outperforms [6] in all measurements. This demonstrates that our proposed method succeeds in overcoming the asymmetry problem as shown in Fig. 3(b) and thus it can estimate the noise level with high accuracy.

4.3. Image Denoising Results

The effectiveness of our proposed self-supervised prior learning algorithm is further validated on the image denoising task, which is to recover a latent clean image **f** from its noisy observation **g**. As shown in Fig. 2, the framework proposed in [33] is adopted to apply the GMM based image prior learned with Alg. 1 to the image denoising task. Since [33] is originally designed for the gray image, the Set12 and 68 gray images (BSD68) from the dataset [20] are selected as test images. In our experiments, each test image is added to $i.i.d.$ Gaussian noise with standard deviations of σ = 15, 25, 50 to generate noisy images with three noise levels. The quality of each denoised result is quantified by the peak signal-to-noise ratio (PSNR). The average PSNR results of each dataset and noise level are provided in Table 1 for all methods. The results marked with '*' are quoted from [22]. To allow for visual assessments, several denoising results on cropped regions are shown in Fig. 8.

The comparison methods include a) the non-learning method: BM3D [8]; b) GMM-related methods: EPLL [33], PGPD [30] and NL-Bayes [16]; c) self-supervised deep learning methods: N2V [15], DIP [28] and S2S [22]. All of these methods are implemented with the source codes and/or trained models provided by their authors. The implementation details are introduced in the supplementary material. Their comparisons to our proposed method are analyzed below.

Comparison to the non-learning method. The BM3D is widely adopted as the benchmark due to its excellent performance. The success of BM3D relies on the search of non-local similar patches. Images from BSD68 usually have many irregular patterns. As a result, the patch search task is difficult on these images. Therefore, BM3D's performance is inferior to our proposed SS-GMM about 0.1dB-0.2dB by average on BSD68. As for Set12 which holds relatively strong non-local similarity, our proposed method still outperforms BM3D at the noise level σ=15. This indicates that the prior learned by our proposed method is still better than the non-local similarity in this case.

Comparison to GMM-related methods. Both EPLL and PGPD adopt supervised image prior learning methods. Our proposed SS-GMM is a self-supervised version of the EPLL. The EPLL requires an extra set of clean images to train GMM with EM-GMM, while SS-GMM only needs a

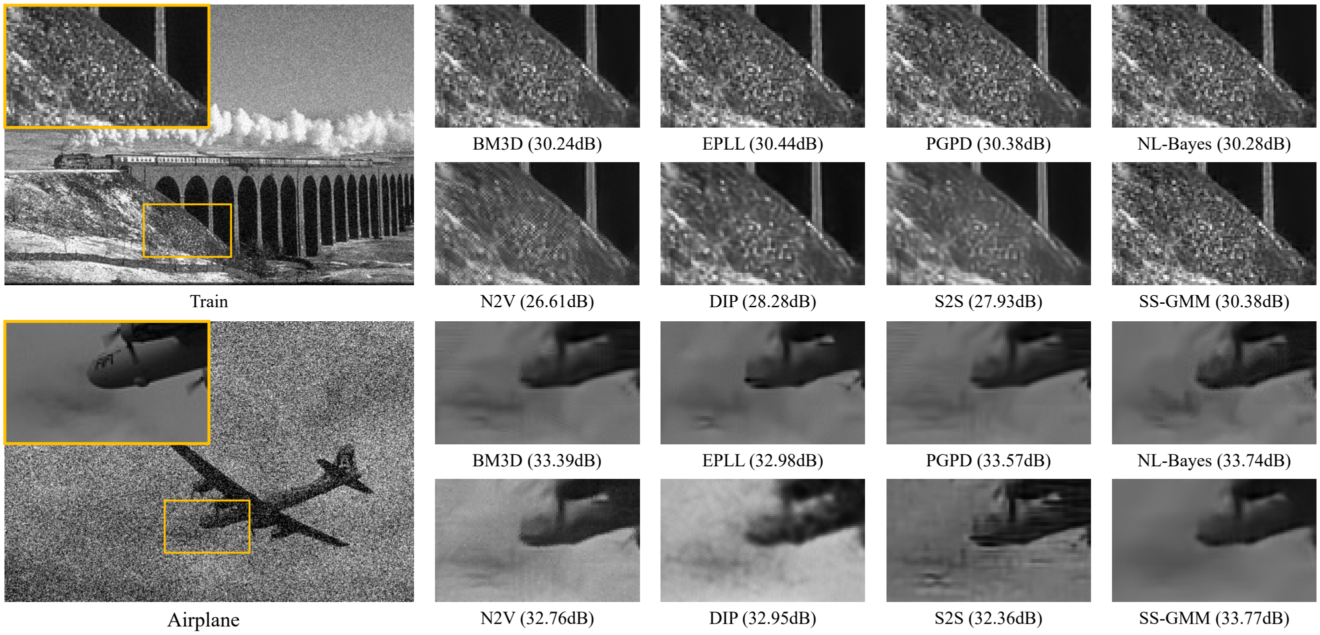


Figure 8. Visual results of comparison algorithms on regions of the image 'Train' ($\sigma = 15$) and regions of the image 'Airplane' ($\sigma = 50$). The inserts shown in the first column are clean GT regions. The whole images for these regions are provided in the supplementary materials.

single noisy image. As Table 1 shows, SS-GMM outperforms the EPLL on Set12 by a large margin (over 0.3dB on average). On the other hand, these two methods' performances are close on BSD68. Observing that the model provided by EPLL's authors [33] is trained on images from the same dataset as the BSD68, we believe that the comparison between EPLL and SS-GMM reveals the sensitivity of supervised methods to the training images. The NL-Bayes is an existing method that can determine GMM's parameters from noisy images. However, its parameter estimation process is based on the search of non-local similar patches and it has to be provided with the noise level in advance. Compared with NL-Bayes, SS-GMM requires no extra information except the noisy image itself as the input, but also achieves better performances for all cases shown in Table 1.

Comparison to self-supervised deep learning methods. The N2V, S2S, and DIP are all self-supervised deep learning methods. As Table 1 shows, the performances of N2V and S2S are very limited at σ=15, while they perform significantly better at σ=50. By contrast, the best performance of our proposed SS-GMM occurs at σ=15. This reveals the essential difference between these two kinds of methods. Strategies adopted by N2V and S2S remove part of pixels to avoid learning an identity function. Even though there is no noise, these strategies have to recover the pixels removed by themselves. When the noise level gradually increases, the proportion of this additional complexity is getting less and less. This explains why these strategies present limited performances at low noise levels and relatively good performance at high noise levels. By contrast, SS-GMM is essentially aimed at suppressing the influence of noise on model parameters, which is easier when σ is relatively smaller. Therefore, SS-GMM is superior to all of the other comparison methods at σ=15. In this sense, these two kinds of methods complement each other. As for DIP, it is significantly inferior to SS-GMM. The advantage of SS-GMM over DIP should be owing to our strict analysis on how to regularize model parameters. Imposing a sparsity constraint on covariance eigenvalues is much more reasonable than simply regularizing all of the parameters with the early-stopping strategy.

5. Conclusion

In this paper, we conducted a detailed study on the statistical property of GMM's covariance eigenvalues under the influence of Gaussian noise. This study finally leads to a self-supervised learning method incorporating a self-contained noise level estimation module. With this method, we successfully achieved image prior learning from a single noisy image. The effectiveness of the learned prior was further validated through image denoising experiments, which demonstrated that our proposed method holds obvious advantages over its supervised counterpart EPLL and is on par with state-of-the-art self-supervised deep learning methods. In the future, we will try to extend the application of our proposed method to other noise types and other image restoration tasks. Also, we will explore how to achieve self-supervised learning by imposing reasonable constraints on other generative models.

References

[1] Joshua Batson and Loic Royer. Noise2Self: Blind denoising by self-supervision. In *International Conference on Machine Learning*, pages 524–533. PMLR, 2019. 2

[2] Jeff A Bilmes et al. A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models. *International Computer Science Institute*, 4(510):126, 1998. 2, 3, 4, 5

[3] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006. 2, 3, 4, 5

[4] Kristian Bredies, Karl Kunisch, and Thomas Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010. 1

[5] Tony Chan, Antonio Marquina, and Pep Mulet. High-order total variation-based image restoration. *SIAM Journal on Scientific Computing*, 22(2):503–516, 2000. 1

[6] Guangyong Chen, Fengyuan Zhu, and Pheng Ann Heng. An efficient statistical method for image noise level estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 477–485, 2015. 3, 4, 7

[7] Yunjin Chen and Thomas Pock. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1256–1272, 2016. 1, 3

[8] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3D transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095, 2007. 1, 7

[9] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977. 5

[10] Minh N Do and Martin Vetterli. The contourlet transform: An efficient directional multiresolution image representation. *IEEE Transactions on image processing*, 14(12):2091–2106, 2005. 1

[11] Weisheng Dong, Guangming Shi, Yi Ma, and Xin Li. Image restoration via simultaneous sparse coding: Where structured sparsity meets Gaussian scale mixture. *International Journal of Computer Vision*, 114(2):217–232, 2015. 1

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

[13] Michael Elad, Peyman Milanfar, and Ron Rubinstein. Analysis versus synthesis in signal priors. *Inverse problems*, 23(3):947, 2007. 1

[14] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016. 1

[15] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2Void-Learning denoising from single noisy images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2129–2137, 2019. 1, 2, 7

[16] Marc Lebrun, Antoni Buades, and Jean-Michel Morel. A nonlocal Bayesian image denoising algorithm. *SIAM Journal on Imaging Sciences*, 6(3):1665–1688, 2013. 1, 2, 7

[17] Stamatios Lefkimmiatis, Anastasios Roussos, Petros Maragos, and Michael Unser. Structure tensor total variation. *SIAM Journal on Imaging Sciences*, 8(2):1090–1122, 2015. 1

[18] Haosen Liu and Shan Tan. Image regularizations based on the sparsity of corner points. *IEEE Transactions on Image Processing*, 28(1):72–87, 2018. 1

[19] Xinhao Liu, Masayuki Tanaka, and Masatoshi Okutomi. Single-image noise level estimation for blind denoising. *IEEE transactions on image processing*, 22(12):5226–5237, 2013. 3

[20] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423. IEEE, 2001. 7

[21] Stanislav Pyatykh, Jürgen Hesser, and Lei Zheng. Image noise level estimation by principal component analysis. *IEEE transactions on image processing*, 22(2):687–699, 2012. 3

[22] Yuhui Quan, Mingqin Chen, Tongyao Pang, and Hui Ji. Self2Self with dropout: Learning self-supervised denoising from single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1890–1898, 2020. 1, 2, 7

[23] Ignacio Ramirez and Guillermo Sapiro. Universal regularizers for robust sparse coding and modeling. *IEEE Transactions on Image Processing*, 21(9):3850–3864, 2012. 1

[24] Klaus Rank, Markus Lendl, and Rolf Unbehauen. Estimation of image noise variance. *IEE Proceedings-Vision, Image and Signal Processing*, 146(2):80–84, 1999. 3

[25] Stefan Roth and Michael J Black. Fields of experts: A framework for learning image priors. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 860–867. IEEE, 2005. 1

[26] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992. 1

[27] Uwe Schmidt and Stefan Roth. Shrinkage fields for effective image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2774–2781, 2014. 1

[28] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9446–9454, 2018. 1, 2, 7

[29] Zaiwen Wen and Wotao Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1):397–434, 2013. 5

[30] Jun Xu, Lei Zhang, Wangmeng Zuo, David Zhang, and Xiangchu Feng. Patch group based nonlocal self-similarity prior learning for image denoising. In *Proceedings of the*

IEEE international conference on computer vision, pages 244–252, 2015. 1, 2, 7

[31] Guoshen Yu, Guillermo Sapiro, and Stéphane Mallat. Solving inverse problems with piecewise linear estimators: From Gaussian mixture models to structured sparsity. *IEEE Transactions on Image Processing*, 21(5):2481–2499, 2011. 2

[32] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE transactions on image processing*, 26(7):3142–3155, 2017. 1

[33] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *2011 International Conference on Computer Vision*, pages 479–486. IEEE, 2011. 1, 2, 3, 7, 8