A Weighted Dictionary Learning Model for Denoising Images Corrupted by Mixed Noise

Jun Liu, Xue-cheng Tai, Haiyang Huang, Zhongdan Huan

Abstract

This paper proposes a general weighted $l^2 - l^0$ norms energy minimization model to remove mixed noise such as Gaussian-Gaussian mixture, impulse noise, Gaussian-impulse noise from the images. The approach is built upon maximum likelihood estimation (MLE) framework and sparse representations over a trained dictionary. Instead of optimizing the likelihood functional derived from a mixture distribution, we present a new weighting data fidelity function which has the same minimizer as the original likelihood functional but is much easier to optimize. The weighting function in the model can be determined by the algorithm itself and it plays a role of noise detection in terms of the different estimated noise parameters. By incorporating the sparse regularization of small image patches, the proposed method can efficiently remove a variety of mixed or single noise while preserving the image textures well. In addition, a modified K-SVD algorithm is designed to address weighted rank-one approximation. Experimental results demonstrate its better performance compared with some existing methods.

Index Terms

Image Denoising, Sparse Representation, Mixed Noise, K-SVD, Weighted Norms

I. Introduction

We address the classical additive noise removal problem in this paper, where the noise in the images can be modeled by

$$g = f + n, \tag{1}$$

Jun Liu, Haiyang Huang and Zhongdan Huan are with the School of Mathematical Sciences, Laboratory of Mathematics and Complex Systems, Ministry of Education, Beijing Normal University, Beijing 100875, P.R. China.

Xue-cheng Tai is with the Department of Mathematics, University of Bergen, Johannes Brunsgate 12, N-5008 Bergen, Norway.

where g, f, n are the observed image, clean image, and noise, respectively. In the overwhelming majority of literature results, the noise n is supposed to be a Gaussian distribution and the L^2-based fidelity term is adopted in many denoising methods. However, in many practical applications, the performance of imaging sensors is affected by a number of factors such as environmental conditions, sensor temperature, atmospheric disturbance, light levels and so on. Thus the distribution of noise in images may often be different from a single Gaussian. One may get some unsatisfactory reconstruction with existing denoising models. Generally speaking, images contaminated by several different types noise (e.g. with different means, variances, or even distributions), namely mixed noise, are more difficult to be restored since the noise levels of each pixel could be far different and there is no good unified standard to measure the similarity between the original and noisy pixels. In this paper, we shall address the mixed noise removal.

There are many methods for image denoising after several decades of developments in image restoration. As mentioned before, most of them are aimed to remove either Gaussian noise or impulse noise.

For Gaussian noise removal, variational method becomes one of the most popular and powerful tools for image restoration since the total variation (TV) was proposed in [1]. The TVL^2 or the so-called ROF model [1] is a classical and well-known model to remove Gaussian noise. However, the results obtained with TV could be over-smoothed and the image details such as textures could be removed together with noise. In order to better preserve the image textures, the nonlocal denoising method [2], [3] was integrated with variational method and the nonlocal TV models in [4], [5]. The nonlocal TV greatly improves the denoising results, but the nonlocal weights in these models may be difficult to determine. Another Gaussian noise removal approach is to use wavelet shrinkage. The high frequency coefficients are suppressed with some given rules such as shrinking, see [6]–[10]. Sparse representation and dictionary learning is also a highly effective image denoising technique. In [11], [12], the authors proposed a novel method to remove additive white Gaussian noise using K-SVD for learning the dictionary from the noisy image with gray scale images. Sparse representation models offer another powerful method to analyse images based on the sparsity and redundancy of their representations. These models assume that there exists a sparse linear combination of the trained dictionary for each small block of the images. This linear combination can be learned from the noisy image itself with the K-SVD algorithm [13]. Due to its good performance, methods based on sparse representation have been extended to color images in [14] and nonlocal models in [15]. Based on overlapping-patches technique and sparsity, many nonlocal image denoising methods have been proposed in very recent years, e.g. locally learned dictionaries (K-LLD) [16], learned simultaneous sparse coding (LSSC) [15], clustering-based sparse representation (CSR) [17] and so on. However, most of these methods only consider the Gaussian noise removal and they may not

work well for mixed noise.

There are two common types of impulse noise: salt-and-pepper noise and random-valued noise. For impulse noise removal, the most popular and classical method is median type filters (e.g. [18], [19]). Different from the mean filters for Gaussian noise removal, the outputs of median filters take the median value in each pixel neighborhood and the impulse noise can be efficiently identified and eliminated, especially for salt-and-pepper noise. However, the median type of filters may significantly destroy the structures of the images, such as blurring of edges and textures. In a variational setting, the data fidelity term associated with median filters is L^1 norm, see [20], [21]. This model has also been extended to deblurring problems in [22]–[25]. For images with mixed noise, these noise detectors have been combined with sparsity regularization method to deal with Gaussian plus impulse noise, see [26]. With the sparsity representation, the quality of the restored image is improved since texture parts can be represented through the dictionary. However, It is worthwhile to note that methods similar to those in [21], [24]–[26] may not work for mixed Gaussian noise.

A natural choice for mixed noise removing is to consider the combination of L^1, L^2 fidelities. For example, we can use $L^2 + L^1$+TV model to remove the Gaussian plus impulse mixed noise. However, it is not easy to precisely determine which pixel is contaminated by Gaussian noise and which one is contaminated by other noise. To overcome this difficulty, in [27], a kernel estimation method was introduced to remove Gaussian and random-valued noise. The TV regularization and EM algorithm was used in [28].

In this work, we propose a general framework to adaptively detect and remove noise of different type, including Gaussian noise, impulse noise and more importantly, their mixtures. We derive our model from the regularized maximum likelihood estimation (MLE) of the noise. Since the likelihood functional related to mixed noise is not easy to be optimized compared with the functional for a single Gaussian noise, a new functional with an additional variable is introduced. This new functional is easier to be optimized and has the same global minimizer (or maximizer) as the original likelihood functional. By minimizing the new functional, we obtain some weighted norms models, in which the weighting functions play the role of noise detectors. By integrating this with sparsity representation, our model can well restore images and textures corrupted by mixed noise. To solve the weighted rank-one approximation problem arisen from the proposed model, a new iterative scheme is given and the low rank approximation can be obtained by singular value decomposition (SVD). Our method integrates sparse coding-dictionary learning, image reconstruction, noise clustering (detection), and parameters estimation into a four-step algorithm. Each step needs to solve a minimization problem.

The rest of the paper is organized as follows. In section II, the K-SVD denoising algorithm is briefly reviewed. The proposed methods are given in section III. Details on theoretical aspects of the model, the proposed algorithms and choices for initial values are discussed. Section IV contains the experimental results. The proposed method is compared with a a number of existing models from the literature. Finally, we conclude our method in section V.

II. Brief Review of the K-SVD Denoising Algorithm

The K-SVD method for removing additive homogeneous white Gaussian noise is proposed in Aharon and Elad [11]–[13]. Since our algorithm will be built upon sparse representations, we now brief review the main mathematical ideas of the K-SVD denoising algorithm. Let $\mathbf{g}, \mathbf{f} \in \mathbb{R}^{N_1 \times N_2}$ be the $N_1 \times N_2$ size noisy and clean images, respectively. To simplify notations, we always use the lowercase letters such as $g \in \mathbb{R}^{N_1 N_2}$ to represent a column vector by stacking the columns of the matrix $\mathbf{g}$. According to the maximum a-posteriori probability (MAP) estimator and an assumption that each small image patch can be sparsely represented as a linear combination of a redundant learned dictionary, the authors of [11], [12] presented the following energy minimization problem to address the denoising problem:

$$\{\alpha^*_{\bullet,i}, \mathbf{D}^*, f^*\} = \underset{\mathbf{D},\alpha_{\bullet,i},f}{\arg\min} \left\{ \mathcal{J}(\boldsymbol{\alpha}, \mathbf{D}, f) \triangleq \frac{1}{2}||g - f||_2^2 + \frac{\lambda}{2} \sum_{i=1}^{N} ||\mathbf{D}\alpha_{\bullet,i} - \mathbf{R}_i f||_2^2 + \sum_{i=1}^{N} \mu_i ||\alpha_{\bullet,i}||_0 \right\} \quad (2)$$

In the above, each $\mathbf{R}_i \in \mathbb{R}^{n_1 n_2 \times N_1 N_2}$ is a binary extracting matrix which extracts $n_1 n_2$ components from a column vector of size $N_1 N_2$, that is to say, $\mathbf{R}_i f$ stands for extracting a $n_1 \times n_2$ patch from the image $\mathbf{f}$ at coordinates (i, j) as a $n_1 n_2$ dimensional column vector. $\mathbf{D} \in \mathbb{R}^{n_1 n_2 \times K}$ is an unknown redundant dictionary (i.e. $K > n_1 n_2$) which should be learned from the noisy image. Each column of the dictionary $\mathbf{D}$, denoting by d_k $(k = 1, 2, \cdots, K)$, is called an atom, and usually satisfies $||d_k||_2 = 1$ though this is not crucial. The vectors $\alpha_{\bullet,i} \in \mathbb{R}^K$ refer to the linear combination coefficients of these atoms, and the l^0 pseudo norm is defined by

$$||\alpha_{\bullet,i}||_0 \triangleq \#\{k : \alpha_{ki} \neq 0, 1 \leqslant k \leqslant K\}, \quad (3)$$

where $\#$ is the cardinality of a set. The l^0 pseudo norm is a sparsity measure, which counts the number of non-zero elements in a vector. $\mu_i > 0$ are some regularization parameters that control the image patch sparsity. λ is a weight parameter controls the trade-off between the data fidelity and the image prior.

The K-SVD denoising algorithm in [11], [12] is a relaxed alternating minimization method. The problem (2) can be split into three subproblems:

Sparse Coding:

$$\boldsymbol{\alpha}^{\nu+1} = \arg\min_{\boldsymbol{\alpha}} \mathcal{J}(\boldsymbol{\alpha}, \mathbf{D}^{\nu}, f^{\nu}). \tag{4}$$

This l^0 minimization problem is in general NP-hard. However, it can be approximately solved by the basis pursuit algorithms [29] such as the orthonormal matching pursuit (OMP) [30], [31]. Other recently proposed methods can also be employed such as the algorithms in [32]–[34].

Dictionary Learning:

$$\mathbf{D}^{\nu+1} = \arg\min_{\mathbf{D}, ||d_k||_2=1} \mathcal{J}(\boldsymbol{\alpha}^{\nu+1}, \mathbf{D}, f^{\nu}). \tag{5}$$

Instead of directly solving this constrained quadratic optimization, the K-SVD algorithm is to iteratively update a column of $\mathbf{D}$ by solving a rank-one approximation of (5). More details about the K-SVD algorithm, please see [13].

Reconstruction:

$$f^{\nu+1} = \arg\min_{f} \mathcal{J}(\boldsymbol{\alpha}^{\nu+1}, \mathbf{D}^{\nu+1}, f). \tag{6}$$

This subproblem has a closed-form solution

$$f^{\nu+1} = (\mathbf{I} + \lambda \sum_{i=1}^{N} \mathbf{R}_i^{\mathrm{T}} \mathbf{R}_i)^{-1} (g + \lambda \sum_{i=1}^{N} \mathbf{R}_i^{\mathrm{T}} \mathbf{D} \alpha_{\bullet,i}^{\nu+1}). \tag{7}$$

As mentioned earlier, the K-SVD denoising algorithm in [11], [12] is built on the assumption that the noise is Gaussian. It may not work well for mixed noise. Moreover, the performance of this model depends on the choice of the parameters which related to the noise variance. In [11], [12], the noise variance are supposed to be known. In the next, we shall propose a general framework to remove mixed noise with sparse representations.

III. The Proposed Method

A. The Probability Density Functions of Mixed Noise

We focus on additive mixed noise removal via energy minimization method. For real images, the probability density function (PDF) is often not a single standardized distribution such as Gaussian. Thus its MLE is often difficult to solve. Here we consider the case that the noise is sampled from several different distributions. This mixed noise in images is more difficult to remove than the standardized Gaussian noise. In this paper, we address this issue and give a general framework for restoring images corrupted by mixed noise.

Suppose the mixed noise $n \in \mathbb{R}^{N_1 N_2}$ is constituted by M different groups $n_l, l = 1, 2, \cdots, M$, each n_l is some realizations of a random variable $\mathfrak{N}_l$ with PDF $p_l(x)$, and the ratio of each n_1 is r_l. Here r_l satisfies $\sum_{l=1}^{M} r_l = 1$. Similarly, n can also be regarded as some realizations of a random variable $\mathfrak{N}$ whose PDF is $p(x)$. With these assumptions, one can get the PDF of mixed noise

$$p(x) = \sum_{l=1}^{M} r_l p_l(x). \tag{8}$$

In this paper, we suppose all values of the pixels in the original and observed images range from $[0, 255]$. A special mixed noise is the Gaussian noise plus impulse noise. For such noise model, it can be written as

$$n = \begin{cases} n_1, & \text{with probability } 1 - r, \\ n_2, & \text{with probability } r, \end{cases} \tag{9}$$

where n_1 is the Gaussian noise and n_2 is the changed pixels values by the impulsive process. Thus n_2 is a uniformly distributed random number in intensity range $[0, 255]$ for random-valued noise and has a value at either 0 or 255 for the salt-and-pepper noise. In real scenario, the noise model maybe more complicated than (9), here (9) is a theoretical formulation with some mathematical simplifications. It can be proven that

Proposition 1: the PDFs of Gaussian plus random-valued noise and Gaussian plus salt-and-pepper noise have the following expression respectively,

$$p(x) = \begin{cases} (1 - r)p_1(x) + \frac{r}{255} \int_{-x}^{255-x} p_2(y)\, \mathrm{d}y, \\ (1 - r)p_1(x) + \frac{r}{2} p_2(-x) + \frac{r}{2} p_2(255 - x), \end{cases} \tag{10}$$

where p_1 is a Gaussian function and p_2 is the PDF of the clean image f, which is a compactly supported function with support $[0, 255]$, i.e. $p_2(y) = 0$ when $y \notin [0, 255]$.

Proof: The proof is similar to the one given in appendix A in [35]. For brevity, we omit it here.

For another special mixed noise, namely impulse noise, its PDF has been analyzed in [27], [35] etc., and the mixture model $p(x)$ in (8) is a general formulation since it can represent any mixed noise such as Gaussian-Gaussian, Gaussian-impulse, Gaussian-Poisson and so on.

Once the PDF of noise is known, a natural way to construct the fidelity term for image denoising is MLE. However, a direct use of the MLE method would lead to a log-likelihood functional which is difficult to be optimized in the case of mixed noise. For instance, by the independent assumption and mixture model (8), one can get the log-likelihood functional

$$\mathcal{L}(f, \boldsymbol{\Theta}) = \ln \prod_{i=1}^{N} \sum_{l=1}^{M} r_l p_l(g_i - f_i) = \sum_{i=1}^{N} \ln \sum_{l=1}^{M} r_l p_l(g_i - f_i), \tag{11}$$

where $N = N_1 N_2$ is the total number of the pixels, and $\mathbf{\Theta}$ is a parameter vector of the distributions. Here $\mathcal{L}$ is not easy to be efficiently maximized since the the existing of ln-sum operator. For Gaussian mixture (i.e. each p_l is a Gaussian function), a classical approach to solve this optimization problem is the well-known expectation-maximization (EM) algorithm [35], [36]. Here we give another method to address this problem for the MLE of general mixture model. This method is intuitive and built upon continuous constraint optimization. We shall use it to design the denosing cost functional.

B. Optimizing the Log-likelihood Functional Indirectly

We found that the essential difficulty of optimizing $\mathcal{L}$ comes from the ln-sum function since the logarithm and the summation operations are noncommutative in general. However, the commutativity of logarithm and summation operations can be achieved under certain conditions. Based on [37]–[39], we have the following more general property on the commutativity of log-sum operations

Proposition 2 (Commutativity of log-sum operations): Given two functions $\gamma_l(x) > 0$, $p_l(x) > 0$, we have

$$-\ln \sum_{l=1}^{M} \gamma_l(x) p_l(x) = \min_{\mathbf{u}(x)\in\Delta_+} \left\{ -\sum_{l=1}^{M} \ln[\gamma_l(x) p_l(x)] u_l(x) + \sum_{l=1}^{M} u_l(x) \ln u_l(x) \right\}, \tag{12}$$

where $\mathbf{u}(x) = (u_1(x), u_2(x), \cdots, u_M(x))$ is a vector-valued function, and $\Delta_+ = \{\mathbf{u}(x) : 0 < u_l(x) < 1, \text{and} \sum_{l=1}^{M} u_l(x) = 1\}$.

Proof: The proof can be done using Lagrangian multiplier method.

This proposition is very useful in simplifying the optimization problem of $\mathcal{L}$. More precisely, after changing the order of log and sum operators, one obtains a new functional with more variables but which can be efficiently minimized (e.g. quadratic problem).

We now show the details on how to apply this proposition. Considering the following minimizing problem

$$\begin{aligned} &\min_{f,\mathbf{\Theta}} \{-\mathcal{L}(f, \mathbf{\Theta})\} = \min_{f,\mathbf{\Theta}} \left\{ -\textstyle\sum_{i=1}^{N} \ln \sum_{l=1}^{M} r_l p_l(g_i - f_i) \right\} \\ &= \min_{f,\mathbf{\Theta},\mathbf{u}\in\Delta_+} \left\{ -\textstyle\sum_{i=1}^{N} \sum_{l=1}^{M} \ln[r_l p_l(g_i - f_i)] u_{il} + \sum_{i=1}^{N} \sum_{l=1}^{M} u_{il} \ln u_{il} \right\}. \end{aligned} \tag{13}$$

Here g_i, f_i, and u_{il} are the discrete representation of $g(x), f(x)$, and $u_l(x)$, respectively. $\mathbf{u}$ is a matrix whose (i,l)-th element is u_{il} and $\mathbf{u} \in \Delta_+$ means that each row of $\mathbf{u}$ (i.e. $u_{i,\bullet}$) in Δ_+.

Let us introduce a new functional

$$\mathcal{H}(f, \mathbf{\Theta}, \mathbf{u}) \triangleq -\sum_{i=1}^{N} \sum_{l=1}^{M} \ln[r_l p_l(g_i - f_i)] u_{il} + \sum_{i=1}^{N} \sum_{l=1}^{M} u_{il} \ln u_{il}. \tag{14}$$

Compared with the original log-likelihood functional $\mathcal{L}(f, \mathbf{\Theta})$, there is an extra variable $\mathbf{u}$ in $\mathcal{H}$. However, minimizing $\mathcal{H}$ is easier than $\mathcal{L}$ in most of the cases. For example, taking each p_l as Gaussian function, then $\mathcal{H}$ becomes quadratic with respect to f, and Θ has a closed-form solution.

Instead of optimizing the original MLE problem, we can turn to minimize $\mathcal{H}$. Usually, the minimizer of multi-variables functional $\mathcal{H}$ can be obtained by the alternating algorithm:

$$\begin{cases} \mathbf{u}^{\nu+1} &= \arg\min\limits_{\mathbf{u}\in\Delta_+} \ \mathcal{H}(f^{\nu}, \mathbf{\Theta}^{\nu}, \mathbf{u}), \\ (f^{\nu}, \mathbf{\Theta}^{\nu+1}) &= \arg\min\limits_{f,\mathbf{\Theta}} \ \mathcal{H}(f, \mathbf{\Theta}, \mathbf{u}^{\nu+1}). \end{cases} \tag{15}$$

For the above iterative scheme, we have:

Proposition 3 (Energy Descent): The sequence $(f^{\nu}, \mathbf{\Theta}^{\nu})$ produced by iteration scheme (15) satisfies

$$-\mathcal{L}(f^{\nu+1}, \Theta^{\nu+1}) \leqslant -\mathcal{L}(f^{\nu}, \Theta^{\nu}). \tag{16}$$

Proof: See appendix A for details.

The equation in (13) shows that both $\mathcal{H}$ and $-\mathcal{L}$ have the same minimum. However, our interest is to know whether they have the same minimizer (not the minimum value). For this aspect, we have:

Proposition 4: Both $\mathcal{H}$ and $-\mathcal{L}$ have the same global minimizer $(f^*, \mathbf{\Theta}^*)$.

Proof: See appendix B.

The proposition 3 can ensure that the iterative scheme (15) can at least find a local minimizer of $-\mathcal{L}$. Moreover, once we obtain the global minimizer of $\mathcal{H}$ by iteration (15), we know it also gives the global minimizer of $-\mathcal{L}$ thanks to proposition 4.

Let us mention that the iteration (15) is essentially equivalent to the EM algorithm [35], [36], [40]: Updating $\mathbf{u}$ in the first step plays a role of the E-step in the EM algorithm and the second step is the M-step. Indeed, u_{il} is a probability of noise at location i belongs to the l-th distribution. However, here the theoretical foundation is totally different from the probabilistic EM algorithm. In this paper, we show that EM algorithm is just a special alternating algorithm with some constraint conditions.

Next, we shall construct a model for mixed noise removing based on $\mathcal{H}$ and sparse representation.

C. Weighted Norms Model

In section III-B, we have shown that the MLE problem (i.e. to maximize $\mathcal{L}$) of mixed noise can be realized by minimizing a new functional $\mathcal{H}$. By incorporating the patch-based sparsity, we propose the

following denoising cost functional

$$\begin{aligned}
&\mathcal{J}(\boldsymbol{\alpha}, \mathbf{D}, \mathbf{u}, \boldsymbol{\Theta}, f) = \\
&-\sum_{i=1}^{N}\sum_{l=1}^{M} u_{il} \ln\left(r_l p_l(g_i - f_i)\right) + \sum_{i=1}^{N}\sum_{l=1}^{M} u_{il} \ln u_{il} \\
&-\lambda \sum_{i=1}^{N}\sum_{j=1}^{n_1 n_2}\sum_{l=1}^{M} [\mathbf{R}_i u_{\bullet,l}]_j \ln\left(r_l p_l([D\alpha_{\bullet,i}]_j - [\mathbf{R}_i f]_j)\right) \\
&+\lambda \sum_{i=1}^{N}\sum_{j=1}^{n_1 n_2}\sum_{l=1}^{M} [\mathbf{R}_i u_{\bullet,l}]_j \ln[\mathbf{R}_i u_{\bullet,l}]_j + \sum_{i=1}^{N} \mu_i ||\alpha_{\bullet,i}||_0.
\end{aligned} \tag{17}$$

In this formulation, the first and second terms are $\mathcal{H}$ (14), which is a global data-fitting term related to the MLE of the mixed noise; the third and fourth terms measure the difference between each $n_1 \times n_2$ image patch and the approximation with an over-complete dictionary $\mathbf{D}$; here the measurement is also constructed in term of $\mathcal{H}$; the last term demands the representation is sparse; $\lambda > 0$ and $\mu_i > 0$ are parameters that control the trade-off between the different terms.

Equation (17) is a general functional for denoising mixed noise. For particular mixture such as Gaussian-Possion, we only need to replace the PDF p_l with the relevant expression. In this paper, we only consider the case that each p_l is a Gaussian function parameterized by variance σ_l^2, that is to say

$$p_l(x) = \frac{1}{\sqrt{2\pi\sigma_l^2}} \exp(-\frac{x^2}{2\sigma_l^2}). \tag{18}$$

The Gaussian-Gaussian mixture model would lead to a weighted l^2 norm which can be easily optimized. Taking (18) into (17), and ignoring any constant term, one can get

$$\begin{aligned}
&\mathcal{J}(\boldsymbol{\alpha}, \mathbf{D}, f, \mathbf{u}, \boldsymbol{\Theta}) = \\
&\tfrac{1}{2}||w \circ (g - f)||_2^2 + \tfrac{\lambda}{2} \textstyle\sum_{i=1}^{N} ||\mathbf{R}_i w \circ (\mathbf{D}\alpha_{\bullet,i} - \mathbf{R}_i f)||_2^2 \\
&+\tfrac{1}{2} < \mathbf{u}, \mathbf{1} > (\ln \sigma_l^2 - 2\ln r_l) + \tfrac{\lambda}{2} \textstyle\sum_{i=1}^{N} < \mathbf{R}_i \mathbf{u}, \mathbf{1} > (\ln \sigma_l^2 - 2\ln r_l) \\
&+ < \mathbf{u}, \ln \mathbf{u} > + \lambda \textstyle\sum_{i=1}^{N} < \mathbf{R}_i \mathbf{u}, \ln(\mathbf{R}_i \mathbf{u}) > + \textstyle\sum_{i=1}^{N} \mu_i ||\alpha_{\bullet,i}||_0,
\end{aligned} \tag{19}$$

where $w \in \mathbb{R}^N$ and its elements $w_i = \sqrt{\sum_{l=1}^{M} \frac{u_{il}}{\sigma_l^2}}$, the symbol $\circ$ stands for element-wise multiplication between two vectors, while $\mathbf{u} \in \mathbb{R}^{N \times M}$ and $<,>$ is the Frobenius inner product, and $\boldsymbol{\Theta} = (\sigma_1^2, \cdots, \sigma_M^2, r_1, \cdots, r_M)$ represents some statistic parameters of the noise.

D. Algorithms

We apply the relaxed alternating algorithm to iteratively minimize (19). In each iteration, one or two variables are updated by fixing the others. More precisely, we need to solve the following four sub-minimization problems.

i. Sparse Coding and Dictionary Learning:

The first minimization problem is

$$(\boldsymbol{\alpha}^{\nu+1}, \mathbf{D}^{\nu+1}) = \arg\min_{\boldsymbol{\alpha},\mathbf{D}} \mathcal{J}(\boldsymbol{\alpha}, \mathbf{D}, f^{\nu}, \mathbf{u}^{\nu}, \boldsymbol{\Theta}^{\nu}). \tag{20}$$

Applying the alternating algorithm again to this subproblem, this problem can be split into two convex subproblems corresponding to the so-called sparse coding step and the dictionary learning step, respectively. Let ν_1 be an inner iteration number, then $\boldsymbol{\alpha}^{\nu+1}$ and $\mathbf{D}^{\nu+1}$ can be obtained by solving the following two minimization problems iteratively:

Sparse Coding: (**Conjugated OMP**)

$$\begin{aligned} &\boldsymbol{\alpha}^{\nu_1+1} = \arg\min_{\boldsymbol{\alpha}} \mathcal{J}(\boldsymbol{\alpha}, \mathbf{D}^{\nu_1}, f^{\nu}, \mathbf{u}^{\nu}, \boldsymbol{\Theta}^{\nu}) = \\ &\arg\min_{\boldsymbol{\alpha}} \left\{ \tfrac{\lambda}{2} \textstyle\sum_{i=1}^{N} ||W_i \mathbf{D}^{\nu_1} \alpha_{\bullet,i} - W_i \mathbf{R}_i f^{\nu}||_2^2 + \sum_{i=1}^{N} \mu_i ||\alpha_{\bullet,i}||_0 \right\}. \end{aligned} \tag{21}$$

In the above, W_i is a diagonal matrix whose diagonal elements are $\mathbf{R}_i w$, i.e. $W_i = \mathrm{diag}(\mathbf{R}_i w)$. This l_0-minimization problem can be approximately solved with the OMP algorithm [30], [31] by redefining $\overline{\mathbf{D}} \triangleq W_i \mathbf{D}$ and $\overline{\mathbf{R}_i f^{\nu}} \triangleq W_i \mathbf{R}_i f^{\nu}$. This process is related to a conjugated orthonormal matching pursuit, and its convergence can be proven similarly as in [30], [31].

Dictionary Learning: (**Modified K-SVD**)

$$\begin{aligned} \mathbf{D}^{\nu_1+1} = &\quad \arg\min_{\mathbf{D},||d_k||_2=1} \mathcal{J}(\boldsymbol{\alpha}^{\nu_1+1}, \mathbf{D}, f^{\nu}, \mathbf{u}^{\nu}, \boldsymbol{\Theta}^{\nu}) \\ = &\quad \arg\min_{\mathbf{D},||d_k||_2=1} \left\{ \textstyle\sum_{i=1}^{N} ||\mathbf{R}_i w \circ (\mathbf{D}\alpha_{\bullet,i}^{\nu_1+1} - \mathbf{R}_i f^{\nu})||_2^2 \right\}. \end{aligned} \tag{22}$$

Although (22) is very similar to (5) except for a weight $\mathbf{R}_i w$, we should note that the above problem can not be directly solved by the K-SVD algorithm since the linear structure is significantly changed by the non-uniform weights. We denote

$$\mathbf{W} = \begin{pmatrix} \mathbf{R}_1 w & \cdots & \mathbf{R}_N w \end{pmatrix}, \mathbf{X} = \begin{pmatrix} \mathbf{R}_1 f & \cdots & \mathbf{R}_N f \end{pmatrix}, \tag{23}$$

then (22) becomes

$$\mathbf{D}^{\nu_1+1} = \arg\min_{\mathbf{D},||d_k||_2=1} \left\{ ||\mathbf{W} \circ (\mathbf{D}\boldsymbol{\alpha}^{\nu_1+1} - \mathbf{X}^{\nu})||_F^2 \right\}. \tag{24}$$

Similar to the K-SVD learning algorithm of [13], a natural approach is to minimize each atom d_k from following energy:

$$d_k^{\nu_1+1} = \arg\min_{||d_k||_2=1} ||\mathbf{W} \circ (\mathbf{E}^k - d_k \alpha_{k,\bullet}^{\nu_1+1})||_F^2. \tag{25}$$

In the above, the error $\mathbf{E}^k \triangleq \mathbf{X}^\nu - \sum_{l=1,l\neq k}^{K} d_l^{\nu_1}\alpha_{l,\blacksquare}^{\nu_1+1}$. This problem is known as a weighted rank-one approximation. It is not simple and has no closed-form solution [41]. Srebro and Jaakkola [41] proposed an iterative algorithm to address this difficulty. Their method is to solve

$$d_k^{\nu_1+1} = \underset{||d_k||_2=1}{\arg\min}||\mathbf{W}\circ(\mathbf{E}^k - d_k^{\nu_1}\alpha_{k,\blacksquare}^{\nu_1+1}) + d_k^{\nu_1}\alpha_{k,\blacksquare}^{\nu_1+1} - d_k\alpha_{k,\blacksquare}||_F^2, \tag{26}$$

via SVD. We note that this algorithm can not be used for the unweighted case when $\mathbf{W} = \tau\mathbf{I}$ is a scalar matrix. Here we shall use another new iteration. Recall that the key idea of K-SVD algorithm is the Gauss-Seidel iteration for matrix equations and low-rank approximations, thus the key step is to separate the diagonal element d_k from the expression. Note that there are N terms $W_i d_k\alpha_{k,\blacksquare}^{\nu_1+1}$ in (22), and we can not get a nice linear representation for d_k since each weight W_i may be different. However, we can get an approximated one via the following minimization problem

$$\tau^* = \arg\min_{\tau}\sum_{i=1}^{N}||W_i d_k\alpha_{k,\blacksquare}^{\nu_1+1} - \tau d_k\alpha_{k,\blacksquare}^{\nu_1+1}||_F^2, \tag{27}$$

where τ is a scalar variable. It is easy to see that $\tau^* = d_k^{\mathrm{T}}\left(\frac{\sum_{i=1}^{N}W_i}{N}\right)d_k$. Hence, we solve the minimization problem

$$d_k^{\nu_1+1} = \underset{||d_k||_2=1}{\arg\min}||\mathbf{W}\circ(\mathbf{E}^k - d_k^{\nu_1}\alpha_{k,\blacksquare}^{\nu_1+1}) + \tau_k d_k^{\nu_1}\alpha_{k,\blacksquare}^{\nu_1+1} - \tau_k d_k\alpha_{k,\blacksquare}||_F^2 \tag{28}$$

to update the atoms, where $\tau_k = (d_k^{\nu_1})^{\mathrm{T}}\left(\frac{\sum_{i=1}^{N}W_i}{N}\right)d_k^{\nu_1}$. Let us mention that the modified scheme would reduce to the original K-SVD algorithm when all weights $W_i = \tau\mathbf{I}$ are the same.

Incorporating the sparse constraint, we get our modified K-SVD algorithm for weighted norm model as follows:

- Select the index set of patches $\mathcal{S}_k$ that use atom d_k, i.e. $\mathcal{S}_k = \{i : \alpha_{k,i}^{\nu_1+1} \neq 0, 1\leqslant i\leqslant N\}$.
- Let $\tau_k = (d_k^{\nu_1})^{\mathrm{T}}\left(\frac{\sum_{i=1}^{N}W_i}{N}\right)d_k^{\nu_1}$, for each image patch with index $i\in\mathcal{S}_k$, calculate the residual $\tilde{e}_i^k = W_i(\mathbf{R}_i f^\nu - \mathbf{D}^{\nu_1}\alpha_{\blacksquare,i}^{\nu_1+1}) + \tau_k d_k^{\nu_1}\alpha_{k,i}^{\nu_1+1}$.
- Set $\tilde{\mathbf{E}}^k \in \mathbb{R}^{n_1 n_2\times|\mathcal{S}_k|}$ with its columns being the $\tilde{e}_i^k$ and update $d_k^{\nu_1+1}$ by minimizing

$$(d_k^{\nu_1+1}, \beta^*) = \underset{||d_k||_2=1,\beta}{\arg\min}||\tilde{\mathbf{E}}^k - \tau_k d_k\beta^{\mathrm{T}}||_F^2, \tag{29}$$

  where $\beta\in\mathbb{R}^{|\mathcal{S}_k|}$. This rank-one approximation can be solved using SVD decomposition of $\tilde{\mathbf{E}}^k$.
- Replace $\alpha_{k,i}^{\nu_1+1}, i\in\mathcal{S}_k$ by the relevant element of β^*.

In our experiments, we choose the inner iteration number $\nu_1 = 10$.

ii. Reconstruction:

The minimization problem we need to solve is:

$$\begin{aligned} f^{\nu+1} &= \arg\min_{f} \mathcal{J}(\boldsymbol{\alpha}^{\nu+1}, \mathbf{D}^{\nu+1}, f, \mathbf{u}^{\nu}, \boldsymbol{\Theta}^{\nu}) \\ &= \arg\min_{f} \left\{ \tfrac{1}{2}||w \circ (g-f)||_2^2 + \tfrac{\lambda}{2}\sum_{i=1}^{N} ||\mathbf{R}_i w \circ (\mathbf{D}^{\nu+1}\alpha_{\bullet,i}^{\nu+1} - \mathbf{R}_i f)||_2^2 \right\}. \end{aligned} \tag{30}$$

Since $\mathcal{J}$ is quadratic with respect to f, thus

$$\begin{aligned} f^{\nu+1} = \quad & \left(\mathrm{diag}(w \circ w) + \lambda \sum_{i=1}^{N} \mathbf{R}_i^{\mathrm{T}} \mathrm{diag}((\mathbf{R}_i w) \circ (\mathbf{R}_i w)) \mathbf{R}_i \right)^{-1} \\ & \left(\mathrm{diag}(w \circ w) g + \lambda \left(\sum_{i=1}^{N} \mathbf{R}_i^{\mathrm{T}} \mathrm{diag}((\mathbf{R}_i w) \circ (\mathbf{R}_i w)) \mathbf{R}_i \right) \mathbf{D}^{\nu+1} \alpha_{\bullet,i}^{\nu+1} \right). \end{aligned} \tag{31}$$

Note that every $\mathbf{R}_i$ is a diagonal matrix. Thus the inverse matrix in the above equation can be directly obtained.

iii. Noise Clustering: (Expectation step)

The minimization problem we need to solve is

$$\begin{aligned} \mathbf{u}^{\nu+1} &= \arg\min_{\mathbf{u}\in\Delta_+} \mathcal{J}(\boldsymbol{\alpha}^{\nu+1}, \mathbf{D}^{\nu+1}, f^{\nu+1}, \mathbf{u}, \boldsymbol{\Theta}^{\nu}) \\ &= \arg\min_{\mathbf{u}\in\Delta_+} \left\{ \begin{array}{l} \frac{1}{2}||w \circ (g - f^{\nu+1})||_2^2 + \frac{\lambda}{2}\sum_{i=1}^{N} ||\mathbf{R}_i w \circ (\mathbf{D}^{\nu+1}\alpha_{\bullet,i}^{\nu+1} - \mathbf{R}_i f^{\nu+1})||_2^2 \\ + \frac{1}{2} < \mathbf{u}, \mathbf{1} > (\ln(\sigma_l^2)^{\nu} - 2\ln r_l^{\nu}) + \lambda \sum_{i=1}^{N} < \mathbf{R}_i \mathbf{u}, \ln(\mathbf{R}_i \mathbf{u}) > \\ + < \mathbf{u}, \ln \mathbf{u} > + \frac{\lambda}{2} \sum_{i=1}^{N} < \mathbf{R}_i \mathbf{u}, \mathbf{1} > (\ln(\sigma_l^2)^{\nu} - 2\ln r_l^{\nu}) \end{array} \right\}. \end{aligned} \tag{32}$$

This problem has a closed-form solution. For simplicity of notations, let us denote

$$\begin{aligned} \mathbf{M} &\triangleq \sum_{i=1}^{N} \mathbf{R}_i^{\mathrm{T}} \mathbf{R}_i, \\ T_l &\triangleq \frac{(g-f^{\nu+1}) \circ (g-f^{\nu+1}) + \lambda \sum_{i=1}^{N} (\mathbf{R}_i^{\mathbf{T}} (\mathbf{D}^{\nu+1}\alpha_{\bullet,i}^{\nu+1} - \mathbf{R}_i f^{\nu+1})) \circ (\mathbf{R}_i^{\mathbf{T}} (\mathbf{D}^{\nu+1}\alpha_{\bullet,i}^{\nu+1} - \mathbf{R}_i f^{\nu+1}))}{2\sigma_l^2 (\mathbf{1} + \lambda \mathbf{M}\mathbf{1})}, \end{aligned} \tag{33}$$

where the division symbol means element-wise division between two vectors. Then $\mathbf{u}^{\nu+1}$ can be computed by

$$u_{\bullet,l}^{\nu+1} = \frac{\frac{r_l^{\nu}}{\sigma_l^{\nu}} \exp(-T_l)}{\sum_{s=1}^{M} \frac{r_s^{\nu}}{\sigma_s^{\nu}} \exp(-T_s)}. \tag{34}$$

iv. Parameters Estimation:

The minimization for this step is

$$\begin{aligned} \boldsymbol{\Theta}^{\nu+1} &= \arg\min_{\boldsymbol{\Theta}, \sum r_l = 1} \mathcal{J}(\boldsymbol{\alpha}^{\nu+1}, \mathbf{D}^{\nu+1}, f^{\nu+1}, \mathbf{u}^{\nu+1}, \boldsymbol{\Theta}) \\ &= \arg\min_{\boldsymbol{\Theta}, \sum r_l = 1} \left\{ \begin{array}{l} \frac{1}{2}||w \circ (g - f^{\nu+1})||_2^2 + \frac{\lambda}{2}\sum_{i=1}^{N} ||\mathbf{R}_i w \circ (\mathbf{D}^{\nu+1}\alpha_{\bullet,i}^{\nu+1} - \mathbf{R}_i f^{\nu+1})||_2^2 \\ + \frac{1}{2} < \mathbf{u}^{\nu+1}, \mathbf{1} > (\ln(\sigma_l^2) - 2\ln r_l) \\ + \frac{\lambda}{2} \sum_{i=1}^{N} < \mathbf{R}_i \mathbf{u}^{\nu+1}, \mathbf{1} > (\ln(\sigma_l^2) - 2\ln r_l) \end{array} \right\}. \end{aligned} \tag{35}$$

From equation $\frac{\partial \mathcal{J}}{\partial \boldsymbol{\Theta}} = 0$, we get the closed-form solution of $\boldsymbol{\Theta}^{\nu+1}$:

$$\begin{aligned} r_l^{\nu+1} &= \frac{< u_{\bullet,l}^{\nu+1}, \mathbf{1} > + \lambda < \mathbf{M} u_{\bullet,l}^{\nu+1}, \mathbf{1} >}{< \mathbf{1}, \mathbf{1} > + \lambda < \mathbf{M}\mathbf{1}, \mathbf{1} >}, \\ (\sigma_l^2)^{\nu+1} &= \frac{< u_{\bullet,l}^{\nu+1}, (g-f^{\nu+1}) \circ (g-f^{\nu+1}) > + \lambda \sum_{i=1}^{N} < \mathbf{R}_i u_{\bullet,l}^{\nu+1}, (\mathbf{D}^{\nu+1}\alpha_{\bullet,i}^{\nu+1} - \mathbf{R}_i f^{\nu+1}) \circ (\mathbf{D}^{\nu+1}\alpha_{\bullet,i}^{\nu+1} - \mathbf{R}_i f^{\nu+1}) >}{< u_{\bullet,l}^{\nu+1}, \mathbf{1} > + \lambda \sum_{i=1}^{N} < \mathbf{R}_i u_{\bullet,l}^{\nu+1}, \mathbf{R}_i \mathbf{1} >}. \end{aligned} \tag{36}$$

E. Tuning of Initial Values and Parameters

It is well-known that the alternating algorithm may immerse at a local minimum. In addition, the quality of the restorations is partly influenced by the choice of control parameters such as λ in the model. Thus good initial values and parameters can help us to improve the performance of the algorithm. In this section, we first discuss the choice of initial variance of noise $(\sigma_l^2)^0$. Tuning of the other optimization initial values and parameters would be discussed later.

Let us begin with the following variance properties of Gaussian mixture distribution:

Proposition 5: Suppose $\mathfrak{N}$ is a random variable with Gaussian mixture PDF $p(x)$, i.e.

$$p(x) = \sum_{l=1}^{M} \frac{r_l}{\sqrt{2\pi\sigma_l^2}} \exp(-\frac{x^2}{2\sigma_l^2}), \tag{37}$$

then its variance $\mathrm{E}(\mathfrak{N} - \mathrm{E}(\mathfrak{N}))^2 = \sum_{l=1}^{M} r_l\sigma_l^2$.

Proof: The proof can be done through direct calculations using the definition of variance.

Proposition 6: Suppose $\mathfrak{N}_1$ and $\mathfrak{N}_2$ are two independent identically distributed random variables with the same 2-components Gaussian mixture PDF $p(x)$, i.e.

$$p(x) = \sum_{l=1}^{2} \frac{r_l}{\sqrt{2\pi\sigma_l^2}} \exp(-\frac{x^2}{2\sigma_l^2}). \tag{38}$$

Let $\mathfrak{N} = \rho\mathfrak{N}_1 + \rho\mathfrak{N}_2$, where ρ is a constant, then we have the variance of $\mathfrak{N}$

$$\mathrm{E}(\mathfrak{N} - \mathrm{E}(\mathfrak{N}))^2 = \rho^2 \sum_{i=0}^{2} C_2^i r_1^{2-i} r_2^i \left((2-i)\sigma_1^2 + i\sigma_2^2\right). \tag{39}$$

In general, if $\mathfrak{N} = \rho\sum_{\kappa=1}^{K} \mathfrak{N}_\kappa$ and each $\mathfrak{N}_\kappa$ satisfies the independent conditions, then we have

$$\mathrm{E}(\mathfrak{N} - \mathrm{E}(\mathfrak{N}))^2 = \rho^2 \sum_{i=0}^{K} C_K^i r_1^{K-i} r_2^i \left((K-i)\sigma_1^2 + i\sigma_2^2\right). \tag{40}$$

Proof: For the length limit of the paper, the proof is omitted here and we leave it to the readers.

Corollary 1: Let $\mathfrak{N} = \mathfrak{N}_1 + \mathfrak{N}_2 + \mathfrak{N}_3 + \mathfrak{N}_4 - 4\mathfrak{N}_5$, where each $\mathfrak{N}_k, k = 1, \cdots, 5$ are independent and identically distributed with 2-components Gaussian mixture PDF, then

$$\begin{aligned} \mathrm{E}(\mathfrak{N} - \mathrm{E}(\mathfrak{N}))^2 = \; & \textstyle\sum_{i=0}^{4} C_4^i r_1^{5-i} r_2^i \left((20-i)\sigma_1^2 + i\sigma_2^2\right) \\ & + \textstyle\sum_{i=0}^{4} C_4^i r_1^{4-i} r_2^{i+1} \left((4-i)\sigma_1^2 + (i+16)\sigma_2^2\right). \end{aligned} \tag{41}$$

In particular, if $r_1 = r_2 = 0.5$, then

$$\mathrm{E}(\mathfrak{N} - \mathrm{E}(\mathfrak{N}))^2 = 10(\sigma_1^2 + \sigma_2^2). \tag{42}$$

Now, we present a rough variance estimation for 2-components mixed Gaussian noise using corollary 1. Denote the Laplace operator of the observed image by

$$\triangle \mathbf{g}_{ij} = \mathbf{g}_{i+1,j} + \mathbf{g}_{i-1,j} + \mathbf{g}_{i,j+1} + \mathbf{g}_{i,j-1} - 4\mathbf{g}_{i,j}. \tag{43}$$

According to the noise model, one gets $\triangle \mathbf{g}_{ij} - \triangle \mathbf{f}_{ij} = \triangle \mathbf{n}_{ij}$, which results in

$$\sigma_1^2 + \sigma_2^2 = \frac{\mathrm{Var}(\triangle \mathbf{g} - \triangle \mathbf{f})}{10} \tag{44}$$

by the corollary 1. Here 'Var' represents the sample variance. We simply replace the left side of the above equation by $\mathrm{Var}(\triangle \mathbf{g})$ since $\mathbf{f}$ is unknown. That is to say, we take

$$\sigma^2 \approx \frac{\sum_{i=1}^{N_1}\sum_{j=1}^{N_2} \triangle \mathbf{g}_{ij}}{10N_1N_2} \tag{45}$$

to roughly estimate the sum of variances of the mixed noise. Generally speaking, $\mathrm{Var}(\triangle \mathbf{f})$ is greater than 0 due to existence of edges and textures in the true image $\mathbf{f}$, this implies that the estimated variance σ^2 is always larger than the true one $\sigma_1^2 + \sigma_2^2$. This rough estimation provides a good enough initial parameter for our model to get some satisfactory reconstructions.

In order to get the initial $(\sigma_2^2)^0$ and $(\sigma_1^2)^0$, we empirically set $(\sigma_2^2)^0 = 2(\sigma_1^2)^0$. Thus we get the initial parameter as

$$\begin{cases} r_1^0 &= 0.5, \\ r_2^0 &= 0.5, \end{cases}, \qquad \begin{cases} (\sigma_1^2)^0 &= \frac{\sigma^2}{3}, \\ (\sigma_2^2)^0 &= \frac{2\sigma^2}{3}. \end{cases} \tag{46}$$

Other initial values are selected as: $\mathbf{D}^0$ is set to the overcomplete DCT dictionary; $f^0 = g$, and $u_{\bullet,1}^0 = 0, u_{\bullet,2}^0 = 1$.

According to the proposition 5 and [11], [12], the parameter λ in our model is empirically set as $\lambda = \frac{1}{30}\sqrt{\sum_{l=1}^{2} r_l^\nu (\sigma_l^2)^\nu}$ in the ν-th outer iteration, and for each example $\mathbf{R}_i f$, the conjugated OMP algorithm in (21) is terminated when $\frac{||\mathbf{R}_i w \circ (\mathbf{R}_i f - \mathbf{D}\boldsymbol{\alpha}_{\bullet,i})||_2^2}{n_1 n_2} < 1.15^2$. Indeed, this is an implicit method to select the sparse regularization parameter μ_i. Here $n_1 \times n_2$ is the size for the extracted image patch $\mathbf{R}_i f$, and in our experiments, $n_1 \times n_2$ is set as a 8×8 block.

IV. Experimental Results

Let us first give some details of the implementation of the proposed algorithm. Firstly, we do not update $\mathbf{R}_i f^\nu$ during the iterations, that is to say, we keep $\mathbf{R}_i f^\nu$ or $\mathbf{R}_i f^{\nu+1}$ as $\mathbf{R}_i f^0 = \mathbf{R}_i g$ in our algorithm. It implies that we use the noisy image g to train the dictionary $\mathbf{D}$. The reason for doing so is that the error limit in the OMP should be smaller since the residual error $||\mathbf{R}_i w \circ (\mathbf{D}\boldsymbol{\alpha}_{\bullet,i} - \mathbf{R}_i f^\nu)||_2^2$ would contain less noise as the iterative process progresses if we update f^ν. However, this error limit is

(a) Lena (b) Barbara (c) Boat (d) House (e) Peppers

Fig. 1. The original test images.

related to the true image f and it is an unknown. This leads to the fact that the stopping criterion of the OMP algorithm is difficult to be chosen. Thus we keep $\mathbf{R}_i f^\nu$ as $\mathbf{R}_i g$ and the main residual error to be the noise variance in this case. Secondly, in this paper, the total number of the outer iteration number ν is set to 20 if not stated otherwise.

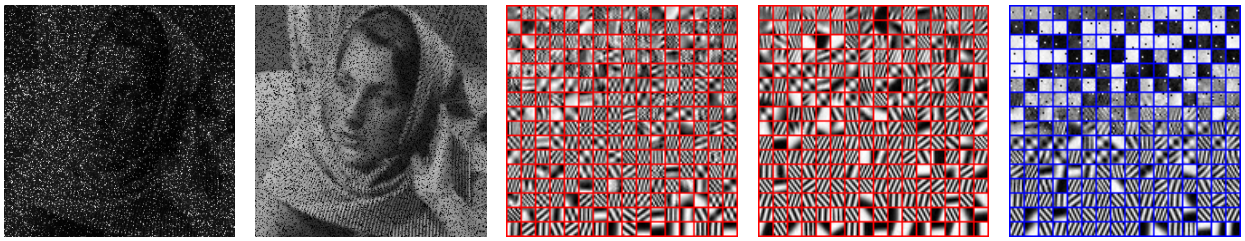
Fig. 1 shows five test images for our experiments: Lena(512×512), Barbara(512×512), Boat (512×512), House (256×256), and Peppers (256×256). In order to compare with other methods, the $\mathrm{PSNR} = 10 \log_{10} \frac{255^2}{\mathrm{Var}(f^* - f)}$ is used to measure the improvement of image quality. Here f^* and f are the restored image and clean image, respectively.

A. Gaussian Mixture Noise

In this experiment, the original image 'Barbara' is contaminated by two Gaussian noise with different standard deviations $\sigma_1 = 10$ and $\sigma_2 = 50$, respectively, while the mixture ratio is $r_1 : r_2 = 0.7 : 0.3$. We implement the proposed algorithm two times with two different noise priors. In case 1, all the noise information, including the true standard deviations (σ_1, σ_2), mixture ratio (r_1, r_2) and spatial distribution (u_1, u_2), are supposed to be known. For such a case, we set the outer iteration number $\nu = 1$ in our algorithm and use the true noise parameters for the initialization. In case 2, none priors of noise is given and all the parameters need to be estimated from the noisy image. The denoising results for both cases are shown in Fig.2. As can be seen from the figure, it is understandable that the reconstructed image in case 1 has better visual effect (preserving textures better) and higher PSNR value than the denoising result in case 2. This is because the estimated noise parameters are often less accurate than the given true ones. To illustrate the superiority of our model, we take the original K-SVD [12] for comparison though it is designed for a single Gaussian distribution. According to proposition 5, for 2-components Gaussian mixture noise, the denoising image would have the highest PSNR value when the noise variances are set as $r_1\sigma_1^2 + r_2\sigma_2^2$ in K-SVD algorithm. In all the experiments, the parameter of noise variances in K-SVD

(a) Noisy (PSNR=19.02) (b) Ours (case 1, with known parameters, PSNR=32.39) (c) Ours (case 2, with unknown parameters, PSNR=30.07) (d) K-SVD [12] (with known parameters, PSNR=26.95)

Fig. 2. A comparison of denoising results under Gaussian mixture noise. For better visual effects, only part of "barbara" image and its corresponding reconstructions are displayed.



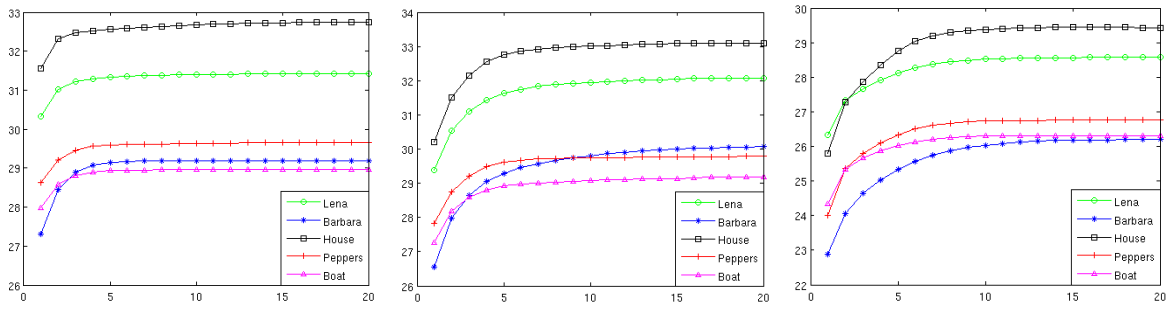
(a) $u \circ g$ (b) $(1-u) \circ g$ (c) Learned $\mathbf{D}$ (Case 1) (d) Learned $\mathbf{D}$ (Case 2) (e) Learned $\mathbf{D}$ (K-SVD)

Fig. 3. Some estimated variables with the proposed algorithm. From left to right: the estimated images $u \circ g$ and $(1-u) \circ g$ of our method; the learned dictionaries $\mathbf{D}$ by the proposed algorithm and K-SVD [12], respectively.

algorithm [12] are supposed to be known as $r_1\sigma_1^2 + r_2\sigma_2^2$. The denoised image with K-SVD are displayed in Fig.2(d). One can see that there are some speckles in the restored image since this model can not discriminate the pixels with different noise levels. In this experiment, our method improves the PSNR values of the reconstructions more than 3 db. For case 2, some estimated functions and parameters by our algorithm are shown in Fig.3 and TABLE I. In Fig.3, one can see that the learned dictionaries by the proposed method are less noisy and can better describe the character of "Barbara" than the K-SVD's. This is one of the reasons why our model can produce better denoising results than K-SVD under mixed noise.

Let us mention that the first iteration in the proposed algorithm is exactly the K-SVD denoising method in [12] with some initially estimated parameters. Our algorithm can improve the quality of the reconstructions with the help of the estimated noise parameters. Figure Fig.4 illustrates parts of the denoising images during the iterations. As can be seen from this figure, the result f^1 is a little oversmoothed mainly because the initially estimated sum of the variances of noise is always larger than

(a) initial f^0 (b) intermediate f^1 (c) intermediate f^5 (d) final f^{20}



(e) $\sigma_1 = 5, \sigma_2 = 30, r_1 : r_2 = 0.3 : 0.7$ (f) $\sigma_1 = 10, \sigma_2 = 50, r_1 : r_2 = 0.7 : 0.3$ (g) $\sigma_1 = 15, \sigma_2 = 75, r_1 : r_2 = 0.5 : 0.5$

Fig. 4. First row: the denoising results f^ν of our algorithm. Second row: the PSNR values for the five test images versus the iteration number. X-axis: the iteration number, Y-axis: the PSNR values.

the true one. However, our algorithm can iteratively improve the parameters of noise and it can provide nicer results, just as shown in Fig.4(c)-4(d). To detail this development, figures in the second row of Fig.4 illustrate the improved PSNR values versus the iteration for the five test images under different levels noise.

More denoising results obtained by the K-SVD and ours under different levels of Gaussian mixture noise are given in TABLE I. For the PSNR values obtained by the proposed algorithm, all the noise parameters are unknown and need to be estimated from the given noisy images. As for the K-SVD algorithm in [12], the noise variances are all supposed to be known as $r_1\sigma_1^2 + r_2\sigma_2^2$. It can be seen from this table that the proposed model still has better performance though our method is blind (all the noise parameters need to be estimated).

B. Impulse Noise

Salt-and-pepper noise and random-valued noise are two common types of the impulse noise. Strictly speaking, they are not additive noise. Existing methods to remove the impulse noise include median type filters such as [18], [19], l^1-norm based variational method [20]–[22], [24], [25], in which two-phase

Images		$\sigma_1 = 5$	$\sigma_2 = 30$		$\sigma_1 = 10$	$\sigma_2 = 50$		$\sigma_1 = 15$	$\sigma_2 = 75$	
$\downarrow$	$r_1 : r_2 \rightarrow$	0.3:0.7	0.5:0.5	0.7:0.3	0.3:0.7	0.5:0.5	0.7:0.3	0.3:0.7	0.5:0.5	0.7:0.3
Lena	K-SVD	31.11	31.60	31.30	28.49	28.92	28.43	26.41	26.79	26.05
	Proposed	**31.43**	**32.69**	**34.24**	**29.00**	**30.43**	**32.07**	**27.04**	**28.57**	**30.36**
Barbara	K-SVD	**29.37**	30.08	30.65	26.40	27.08	26.95	23.70	24.38	24.21
	Proposed	29.19	**30.50**	**32.69**	**26.46**	**28.15**	**30.07**	**23.75**	**26.20**	**28.22**
Boat	K-SVD	**29.08**	29.63	29.78	26.60	27.07	26.81	24.61	25.02	24.60
	Proposed	28.96	**29.85**	**31.19**	**26.79**	**27.82**	**29.19**	**25.02**	**26.31**	**27.77**
House	K-SVD	31.81	32.25	31.74	28.83	29.42	28.88	26.13	26.74	26.13
	Proposed	**32.73**	**33.66**	**34.83**	**30.16**	**31.56**	**33.10**	**27.24**	**29.44**	**31.42**
Peppers	K-SVD	29.47	30.10	30.32	26.82	27.40	27.10	24.53	25.16	24.69
	Proposed	**29.66**	**30.55**	**31.69**	**27.37**	**28.47**	**29.79**	**25.21**	**26.76**	**28.37**

TABLE I

THE PSNR VALUES OF THE DENOISED IMAGES WITH THE PROPOSED METHOD AND K-SVD [12] IN THE PRESENCE OF GAUSSIAN MIXTURE NOISE.

methods [21], [24], [25] can provide reasonably good results. Here we shall illustrate that the proposed model can work well for impulse noise and produce better denoising results than these mentioned methods. The true PDFs of the impulse noise have been analyzed in our previous work [35] and it has been shown that the PDFs can be well approximated with mixture models.

As for impulse noise, the initial u^0 in our algorithm can be better chosen as the output from the first phase in the two-phase methods [24], [25], that is, u^0 can be set as

$$u_i^0 = \begin{cases} 1, & \text{if} \quad g_i == f_i^{\text{med}}, \\ 0, & \text{else.} \end{cases} \tag{47}$$

In the above expression, f^{med} is a filtered image by median-type filter. Similar to [24], [25] we can use adaptive median filter (AMF) [18] and adaptive center-weighted median filter (ACWMF) [42] for salt-and-pepper noise and random-valued impulse noise detection, respectively. Besides, in order to get to the converged solution quickly, here for impulse noise, we set the initial variances as

$$\begin{cases} (\sigma_1^2)^0 & = \frac{\sigma^2}{10}, \\ (\sigma_2^2)^0 & = \frac{9\sigma^2}{10}, \end{cases} \tag{48}$$

where σ^2 is the estimated variance from the noisy images according to equation (45). This choice of initial variances is empirical and maybe not be optimal. Our goal is to increase the difference of the initial variances since the true one is $\sigma_1^2 = 0$.

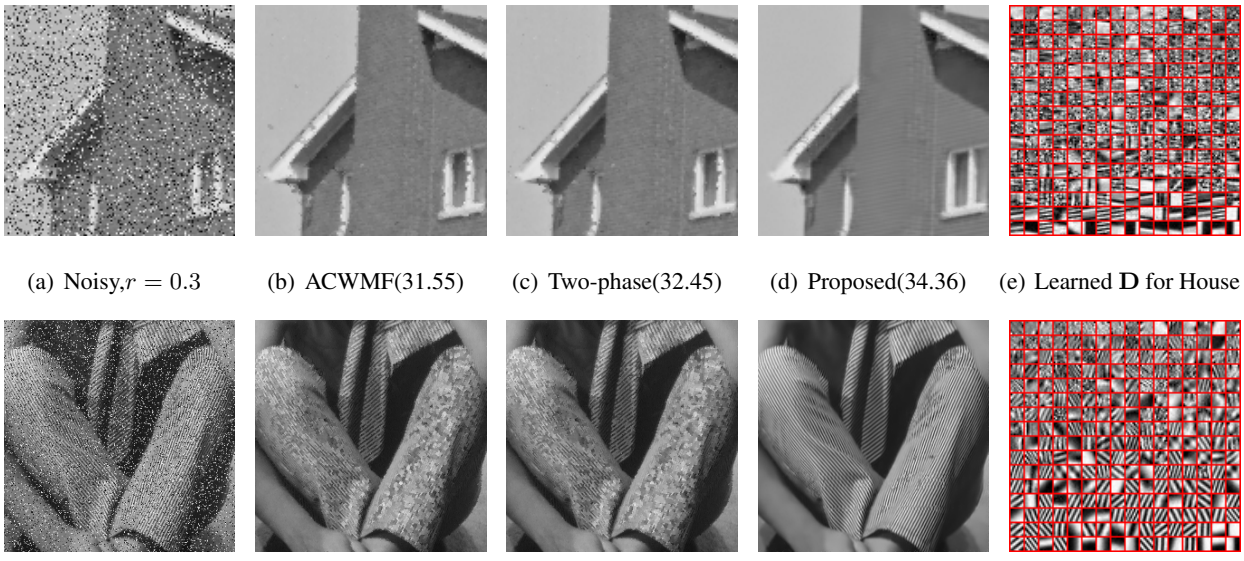
Images	Noise density r	Noisy	[42]	[25]	Proposed
Lena	0.1	19.27	37.91	38.22	**38.38**
	0.2	15.42	34.95	35.45	**36.51**
	0.3	13.32	32.73	33.49	**34.42**
	0.4	11.99	30.24	31.37	**31.75**
	0.5	11.62	27.49	28.79	**29.47**
	0.6	10.82	23.86	25.33	**25.55**
Barbara	0.1	18.83	26.25	25.98	**35.41**
	0.2	15.83	25.36	25.20	**33.58**
	0.3	14.07	24.50	24.47	**31.14**
	0.4	12.81	23.54	23.76	**26.60**
	0.5	11.85	22.37	22.86	**23.40**
	0.6	11.06	20.49	21.31	**21.68**
House	0.1	19.35	37.06	37.46	**40.88**
	0.2	16.28	34.22	35.06	**37.97**
	0.3	14.52	31.55	32.45	**34.36**
	0.4	13.27	29.27	30.28	**30.70**
	0.5	12.28	26.85	27.96	**28.23**
	0.6	11.51	23.61	24.84	**25.67**

TABLE II

PSNR VALUES FOR DIFFERENT METHODS AND TEST IMAGES WITH RANDOM VALUED NOISE.

Since salt-and-pepper noise can be well detected by median filters and thus two-phase method can give some good denoising results even though the salt-and-pepper noise density is as high as 90% [24], [25]. However, random-valued noise is hard to identify and there is no good detector for it. Compared to two-phase method, our method has a superiority that the noise estimation and denoising process are done alternately, which means one of the two procedures can guide the other using the estimated information. This explains why the proposed method has better performance for random-valued noise, especially when the noise levels are high.

TABLE II lists some denoising results in the presence of random-valued noise. In this table, the density of random-valued noise varies from 0.1 to 0.6. As can be seen from this table, our method gives the best restoration with the highest PSNR values with all different noise density. Because of the sparsity regularization of small blocks, our method significantly improves the texture preservation, see Barbara test image in Fig.5 for more details.



(a) Noisy,$r = 0.3$ (b) ACWMF(31.55) (c) Two-phase(32.45) (d) Proposed(34.36) (e) Learned $\mathbf{D}$ for House

(f) Noisy,$r = 0.2$ (g) ACWMF(25.36) (h) Two-phase(25.20) (i) Proposed(33.58) (j) Learned $\mathbf{D}$ for Barbara

Fig. 5. The denoising results of different methods under random-valued noise.

C. Gaussian Noise Plus Impulse Noise

We consider Gaussian plus impulse mixed noise removal in this subsection. Recall that the PDF of Gaussian plus impulse noise has the expression (9). Specially, if we suppose the normalized histogram of the clean image, namely $p_2(y)$, is an uniformly distributed PDF in $[0, 255]$, then for random-valued noise, (9) becomes

$$p(x) = (1-r)p_1(x) + \begin{cases} \frac{r(255+x)}{255^2}, & -255 \leqslant x \leqslant 0, \\ \frac{r(255-x)}{255^2}, & 0 \leqslant x \leqslant 255, \\ 0, & \text{else.} \end{cases} \tag{49}$$

The above triangular-type PDF is the key point of the kernel regression method in [27] for addressing mixture of Gaussian and random-valued noise. One can use this PDF and the framework proposed in this paper to get a new fidelity term. However, for real images, the supposition of uniformly distributed histogram may fail. Moreover, the indifferentiability of the triangular-type piecewise PDF would lead to a non-smoothed minimization problem. On the other hand, our experiments show that good denoising results also can be achieved by replacing the second part of the PDF by an easily optimized Gaussian function. Thus the proposed method can also efficiently remove mixture of Gaussian and impulse noise.

Let us mention that if we use the two-sided exponential distribution to approximate the second part of the PDF of the noise and when p_1 is the delta function, together with the total variational regularization,

	$\sigma = 5$			$\sigma = 10$			$\sigma = 15$		
$r \rightarrow$	0.1	0.2	0.3	0.1	0.2	0.3	0.1	0.2	0.3
Noisy	18.76	15.76	14.04	18.43	15.61	13.95	17.94	15.38	13.81
Two-phase	25.40	24.77	24.13	24.34	23.94	23.45	23.32	23.02	22.67
ACWMD+K-SVD	26.07	25.27	24.51	25.50	24.91	24.31	24.64	24.19	23.77
[26]	30.45	27.75	25.95	28.45	26.59	25.34	27.33	25.69	**24.55**
Proposed	**32.97**	**31.52**	**28.95**	**30.42**	**28.32**	**26.30**	**28.37**	**25.98**	24.01

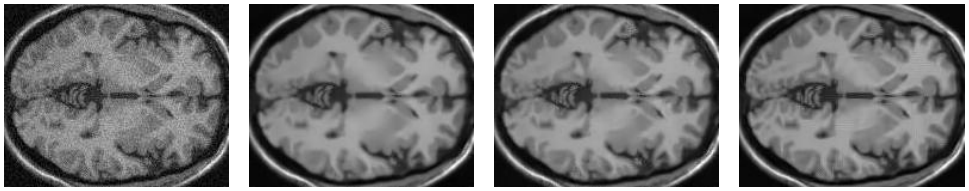
TABLE III

PSNR VALUES FOR DIFFERENT METHODS FOR THE BARBARA IMAGE WITH GAUSSIAN NOISE PLUS RANDOM VALUED NOISE.

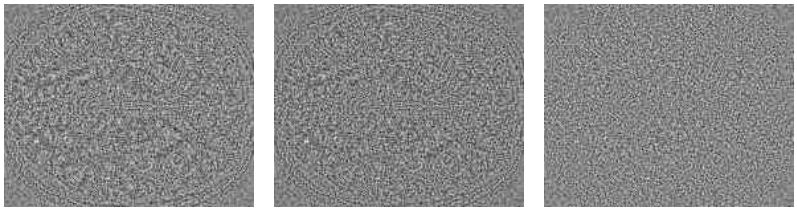
then our method would reduce to the two-phase method [24], [25] under some proper conditions.

TABLE III gives the denoising results of Barbara test images corrupted by Gaussian noise and random-valued noise. In our experiment, the Gaussian noise level varies with $\sigma = 5, 10, 15$ and the density of random-valued noise values with $r = 0.1, 0.2, 0.3$, respectively. For comparison, we test on the performance of the median filter ACWMF [42] plus K-SVD [12], that is, we first filter the noisy images by ACWMF and then denoise the output of ACWMF with K-SVD. The two-phase based methods [25] and [26] will be compared. For the tuning of parameters for different methods, in "ACWMF+K-SVD" algorithm, the variances of Gaussian noise are all supposed to be known. We test several parameters (including the suggested ones in [24], [25]) for the two-phase method and pick the results with the highest PSNR in our comparisons. The choice of initial values for the proposed method is the same as section IV-B. It can be seen that the parameters of our method do not need to be tuned manually for different noise levels. The control parameters in the proposed method can be adaptively adjusted. This is another advantage of our method in removing mixed noise.

To show the efficiency of the proposed method and make a comparison with other methods for some real noisy images, the denoising results of KSVD [12], the proposed method and a wavelets-based method BM3D [7] are shown in Fig.6. The methods in KSVD and BM3D require the true noise variance, thus in this experiment, we use the right side of (44) to estimate the noise variance for KSVD and BM3D. For real images, it is difficult to give an objective index such as PSNR values to evaluate the quality of the restorations since there is no true image. However, we can give the noise removed by different methods to make a comparison. From the noise images shown in the second row, it can be seen that the proposed method gives better result than KSVD's since there is less information in the noise image removed by



(a) Real noisy image (b) KSVD [12] (c) Proposed (d) BM3D [7]



(e) Noise removed by KSVD (f) Noise removed by proposed (g) Noise removed by BM3D

Fig. 6. Denoising the real Brain MR image.

the proposed algorithm. Let us mention that the proposed model would have a better performance if the noise difference is larger. In this experiment, it seems that the BM3D has a slightly better performance than the proposed model. This is because the noise difference in this real image is relatively small and the BM3D method has a better performance than the dictionary learning methods. However, one can integrate our method with BM3D to further improve the denoising result.

V. Conclusion and Discussion

We provide a general framework to remove mixed noise using the PDF. A method is proposed to solve the MLE problem of mixture distribution. Though it is essentially equivalent to the well-known EM algorithm, here we give the EM algorithm another interpretation through continuous constraint optimization. We also further explain the connections between the classical EM algorithm and alternating algorithm. By combining the sparsity regularization and dictionary learning techniques, a novel and efficient model is designed to removing mixed noise such as Gaussian-Gaussian mixture noise, impulse noise, Gaussian plus impulse noise and others. Besides, we study the variance estimation from the given noisy images and offer a method to tune the initial parameters. This makes the proposed method to be very practical. Our model is a general one since it can work well under Gaussian, non-Gaussian noise and even their mixtures. Experimental results have demonstrated its better performance compared with other related state-of-the-art algorithms.

The computation complexity for each outer iteration in the proposed method is slightly more than K-SVD algorithm. Thus our method is slower than some methods such as two-phase methods since the sparse coding and dictionary learning are often time consuming. Generally speaking, for 256×256 images, the CPU time for one outer iteration for our method is about 40 seconds with our unoptimized Matlab codes on a PC equipped with 3.2 GHz CPU. The sparse coding and dictionary learning steps cost about 12 and 26 seconds, respectively. It costs less than 1 second for the reconstruction step, and about 1 second for the noise clustering step include the parameters estimation step. It was observed that the sparse coding and dictionary learning part are consuming the most of the computation time.

Note that our method is easy to be paralleled. Besides, fast split algorithms to solve l^0 minimization algorithm can also be used to improve the computation. This will be for our future research. For sparse coding and dictionary learning, some recently algorithms such as [33], [34] can be adopted with some modifications for our model.

For different types of noise, the computational complexity of our algorithm for one outer iteration is almost the same since the mathematical models are the same for different kinds of the noise. One important factor that influences the computational cost is the noise level. Generally speaking, images with high noise levels need more outer iterations.

Although we only discuss the $L^2 + L^2$ type model in this paper, it can be seen that other $L^p + L^q$ models can also be used.

Our method has the potential to be used for images segmentation and registration. For example, a possible extension is to segment images with non-parameterized mixture distribution with the proposed optimization framework.

VI. Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (Nos. 11071023 and 11201032), and in part by the MOE (Ministry of Education) Tier II project T207N2202.

Appendix A
Proof of proposition 3

By proposition 2, equation (13) and the first formulation of (15), we get

$$\begin{cases} -\mathcal{L}(f^{\nu+1}, \mathbf{\Theta}^{\nu+1}) = & \mathcal{H}(f^{\nu+1}, \mathbf{\Theta}^{\nu+1}, \mathbf{u}^{\nu+2}), \\ -\mathcal{L}(f^{\nu}, \mathbf{\Theta}^{\nu}) = & \mathcal{H}(f^{\nu}, \mathbf{\Theta}^{\nu}, \mathbf{u}^{\nu+1}). \end{cases} \tag{50}$$

On the other hand, the second equation in (15) provides

$$\mathcal{H}(f^{\nu+1}, \mathbf{\Theta}^{\nu+1}, \mathbf{u}^{\nu+2}) \leqslant \mathcal{H}(f^{\nu+1}, \mathbf{\Theta}^{\nu+1}, \mathbf{u}^{\nu+1}) \leqslant \mathcal{H}(f^{\nu}, \mathbf{\Theta}^{\nu}, \mathbf{u}^{\nu+1}), \tag{51}$$

thus $-\mathcal{L}(f^{\nu+1}, \Theta^{\nu+1}) \leqslant -\mathcal{L}(f^{\nu}, \Theta^{\nu})$, and the conclusion holds.

APPENDIX B
PROOF OF PROPOSITION 4

Suppose $(f^*, \mathbf{\Theta}^*)$ is a global minimizer of $-\mathcal{L}$, and let $\mathbf{u}^* = (u_1^*, u_2^*, \cdots, u_M^*)$ with its component function $u_{il}^* = \frac{r_l^* p_l(g_i - f_i^*)}{\sum_{s=1}^M r_s^* p_s(g_i - f_i^*)}$, then $\mathbf{u}^* \in \Delta_+$ and

$$\mathcal{H}(f^*, \mathbf{\Theta}^*, \mathbf{u}^*) = \sum_{i=1}^N \ln \sum_{l=1}^M r_l^* p_l(g_i - f_i^*) = -\mathcal{L}(f^*, \mathbf{\Theta}^*). \tag{52}$$

Please recall that $\min\limits_{f,\mathbf{\Theta}} -\mathcal{L}(f, \mathbf{\Theta}) = \min\limits_{f,\mathbf{\Theta},\mathbf{u}\in\Delta_+} \mathcal{H}(f, \mathbf{\Theta}, \mathbf{u})$ in equation (13), thus we have $\min\limits_{f,\mathbf{\Theta},\mathbf{u}\in\Delta_+} \mathcal{H}(f, \mathbf{\Theta}, \mathbf{u}) = -\mathcal{L}(f^*, \mathbf{\Theta}^*) = \mathcal{H}(f^*, \mathbf{\Theta}^*, \mathbf{u}^*)$, which means $(f^*, \mathbf{\Theta}^*)$ is a global minimizer of $\mathcal{H}$.

Conversely, we assume $(f^*, \mathbf{\Theta}^*, \mathbf{u}^*)$ is a global minimizer of $\mathcal{H}$ but $(f^*, \mathbf{\Theta}^*)$ is not the global minimizer of $-\mathcal{L}$, then there must be a point $(\hat{f}, \hat{\mathbf{\Theta}})$ such that $-\mathcal{L}(\hat{f}, \hat{\mathbf{\Theta}}) < -\mathcal{L}(f^*, \mathbf{\Theta}^*)$. Similarly, we let $\hat{\mathbf{u}} = (\hat{u}_1, \hat{u}_2, \cdots, \hat{u}_M)$ with $\hat{u}_{il} = \frac{\hat{r}_l p_l(g_i - \hat{f}_i)}{\sum_{s=1}^M \hat{r}_s p_s(g_i - \hat{f}_i)}$, one can calculate $\mathcal{H}(\hat{f}, \hat{\mathbf{\Theta}}, \hat{\mathbf{u}}) = -\mathcal{L}(\hat{f}, \hat{\mathbf{\Theta}}) < -\mathcal{L}(f^*, \mathbf{\Theta}^*) = \mathcal{H}(f^*, \mathbf{\Theta}^*, \bar{\mathbf{u}}) \leqslant \mathcal{H}(f^*, \mathbf{\Theta}^*, \mathbf{u}^*)$, which is contradicted with the assumption. Here $\bar{\mathbf{u}} = (\bar{u}_1, \bar{u}_2, \cdots, \bar{u}_M)$ and $\bar{\mathbf{u}}_{il} = \frac{r_l^* p_l(g_i - f_i^*)}{\sum_{s=1}^M r_s^* p_s(g_i - f_i^*)}$, the last inequality is obtained in terms of $\bar{\mathbf{u}}$ is a minimizer of $\mathcal{H}(\cdot, \cdot, \mathbf{u})$ for fixed $f^*, \mathbf{\Theta}^*$.

REFERENCES

[1] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, pp. 259–268, 1992.

[2] A. Buades, B. Coll, and J. Morel, "A non-local algorithm for image denoising," *Computer Vision and Pattern Recognition*, 2005.

[3] ——, "A review of image denoising algorithms, with a new one,," *Multiscale Modeling and Simulation*, vol. 4, no. 2, pp. 490–530, 2005.

[4] G. Gilboa and S. Osher, "Nonlocal linear image regularization and supervised segmentation," *Multiscale Modeling and Simulation*, vol. 6, no. 2, pp. 595–630, Jan. 2007.

[5] ——, "Nonlocal operators with applications to image processing," *SIAM: Multiscale Modeling and Simulation*, vol. 7, no. 3, pp. 1005–1028, 2008.

[6] J. Portilla, V. Strela, M. Wainwright, and E. Simoncelli, "Image denoising using scale mixture of gaussians in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1338–1351, 2001.

[7] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[8] ——, "A nonlocal and shape-adaptive transform-domain collaborative filtering," in *2008 int. workshop on local and non-local approximation in image processing*, Lausanne, Switzerland, 2008.

[9] ——, "Bm3d image denoising with shape-adaptive principal component analysis." in *Proc. workshop on signal processing with adaptive sparse structured representations (SPARS09)*, Saint-Malo, France, 2009.

[10] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola, "Form local kernel to nonlocal multiple-model image denosing," *International Journal of Computer Vision*, vol. 86, pp. 1–32, 2010.

[11] M. Elad and M. Aharon, "Image denoising via learned dictionaries and sparse representation," in *IEEE Computer Vision and Pattern Recognition*, New York, June 2006.

[12] ——, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[13] M. Aharon, M. Elad, and A. Bruckstein, "The k-svd: An algorithm for designing of overcomplete dictionaries for sparse representations," *IEEE Transactions on Image Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[14] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53–69, 2008.

[15] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *IEEE 12th International Conference on Computer Vision*, 2009, pp. 2272–2279.

[16] P. Chatterjee and P. Milanfar, "Clustering-based denoising with locally learned dictionaries," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1438–1451, 2009.

[17] W. Dong, X. Li, L. Zhang, and G. Shi, "Sparsity-based image denoising via dictionary learning and structural clustering," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[18] H. Wang and R. Haddad, "Adaptive median filters: New algorithms and results," *IEEE Transactions on Image Processing*, vol. 4, pp. 499–502, 1995.

[19] H. Eng and K. Ma, "Noise adaptive soft-switching median filter," *IEEE Transactions on Image Processing*, vol. 10, pp. 242–251, 2001.

[20] M. Nikolova, "A variational approach to remove outliers and impulse noise," *Journal of Mathematical Imaging and Vision*, vol. 20, no. 1, pp. 99–120, Jan. 2004.

[21] R. Chan, C. Ho, and M. Nikolova, "Salt-and-pepper noise removal by median-type noise detectors and edge-preserving regularization," *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1479–1485, 2005.

[22] L. Bar, N. Kiryati, and N. Sochen, "Image deblurring in the presence of impulsive noise," *International Journal of Computer Vision*, vol. 70, no. 3, pp. 279–298, Dec. 2006.

[23] L. Bar, A. Brook, N. Sochen, and N. Kiryati, "Deblurring of color images corrupted by impulsive noise," *IEEE Transactions on Image Processing*, vol. 16, no. 4, pp. 1101–1111, 2007.

[24] J.-F. Cai, R. Chan, and M. Nikolova, "Two-phase methods for deblurring images corrupted by impulse plus gaussian noise," *Inverse Problems and Imaging*, vol. 2, pp. 187–204, 2008.

[25] ——, "Fast two-phase image deblurring under impulse noise," *Journal of Mathematical Imaging and Vision*, vol. 36, no. 1, pp. 46–53, 2009.

[26] Y. Xiao, T. Zeng, j. Yu, and M. K. Ng, "Restoration of images corrupted by mixed gaussian-impulse noise via l1-l0 minimization," *Pattern Recognition*, vol. 44, pp. 1708–1720, 2011.

[27] E. Lopez-Rubio, "Restoration of images corrupted by gaussian and uniform impulsive noise," *Pattern Recognition*, vol. 43, no. 5, pp. 1835–1846, 2010.

[28] J. Liu, Z.-D. Huan, H.-Y. Huang, and H.-L. Zhang, "An adaptive method for recovering image from mixed noisy data," *International Journal of Computer Vision*, vol. 85, no. 2, pp. 182–191, 2009.

[29] S. Mallat and Z. Zhang, "Matching pursuit with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[30] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," *Proceeding of the 27 th Annual Asilomar Conference on Signals, Systems, and Computers*, vol. 1, pp. 40–44, 1993.

[31] G. Davis, S. Mallat, and Z. Zhang, "Adaptive time-frequency decompositions," *Optical Engineering*, vol. 33, no. 7, pp. 2183–2191, 1994.

[32] H. Lee, A. Battle, R. Raina, and A. Ng, "Efficient sparse coding algorithms," in *In Advances in Neural Information Processing Systems*. MIT Press, 2007.

[33] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding." *Journal of Machine Learning Research,*, vol. 11, pp. 19–60, 2010.

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

[35] J. Liu, H. Huang, Z. Huan, and H. Zhang, "Adaptive variational method for restoring color images with high density impulse noise," *International Journal of Computer Vision*, vol. 90, no. 2, pp. 131–149, 2010.

[36] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm (with discussion)," *Journal of the Royal Statistical Society*, vol. 39, pp. 1–38, 1977.

[37] R. Rockafellar, "A dual approach to solving nonlinear programming problems by unconstrained optimization,," *Mathematical Programming*, vol. 5, pp. 354–373, 1973.

[38] M. Teboulle, "A unified continuous optimization framework for center-based clustering methods," *J. Mach. Learn. Res.*, vol. 8, pp. 65–102, 2007.

[39] E. Bae, J. Yuan, and X. Tai, "Global minimization for continuous multiphase partitioning problems using a dual approach," *International Journal of Computer Vision*, vol. 92, no. 1, pp. 112–129, 2011.

[40] J. Bilmes, "A gentle tutorial on the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," 1997. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.28.613

[41] N. Srebro and T. Jaakkola, "Weighted low-rank approximations," in *Proceedings of the Twentieth International Conference on Machine Learing (ICML,2003)*, 2003, pp. 720–727.

[42] S. Ko and Y. Lee, "Center weighted median filters and their applications to image enhancement," *IEEE Transactions on Circuits and Systems*, vol. 38, pp. 984–993, 1991.