Pattern Recognition 44 (2011) 1708–1720

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

# Restoration of images corrupted by mixed Gaussian-impulse noise via $l_1$–$l_0$ minimization

Yu Xiao [a], Tieyong Zeng [b,*], Jian Yu [a], Michael K. Ng [b]

[a] School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China
[b] Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong

ARTICLE INFO

*Article history:*
Received 13 May 2010
Received in revised form
19 November 2010
Accepted 2 February 2011
Available online 16 February 2011

*Keywords:*
Image restoration
Gaussian noise
Impulse noise
Dictionary learning

ABSTRACT

In this paper, we study the restoration of images corrupted by Gaussian plus impulse noise, and propose a $l_1$–$l_0$ minimization approach where the $l_1$ term is used for impulse denoising and the $l_0$ term is used for a sparse representation over certain unknown dictionary of images patches. The main algorithm contains three phases. The first phase is to identify the outlier candidates which are likely to be corrupted by impulse noise. The second phase is to recover the image via dictionary learning on the free-outlier pixels. Finally, an alternating minimization algorithm is employed to solve the proposed minimization energy function, leading to an enhanced restoration based on the recovered image in the second phase. Experimental results are reported to compare the existing methods and demonstrate that the proposed method is better than the other methods.



## 1. Introduction

Image restoration is an important task in image processing. The general idea is to estimate an ideal image $u$ from the observed noisy image $f$. In this paper, we restrict our attention to a mixed noise removal task. There are some different types of mixed noise models commonly considered in the literature, such as, blur and Gaussian (or impulse) noise [1–4]; Poisson plus Gaussian noise [5] and Gaussian plus impulse noise [6–12]. Here, we assume that the observed image $f$ is obtained from the mixed noise model:

$$f = \mathbb{N}_{imp}(u+b),$$

where $b$ is an additive zero-mean white Gaussian noise with standard variance $\sigma^2$ and $\mathbb{N}_{imp}$ denotes the image degradation by impulse noise.

Note that there are mainly two common types of impulse noise: salt-and-pepper noise and random-valued noise [13]. Denote $u_{ij}$ the gray value of an image $u$ at location $(i,j)$ and $\tilde{u}$ the noisy image. Suppose that the dynamic range of an image is $[d_{\min},d_{\max}]$, the salt-and-pepper noise model is given as follows:

$$\tilde{u}_{ij} = \begin{cases} d_{\min} & \text{with probability } s/2 \\ d_{\max} & \text{with probability } s/2 \\ u_{ij} & \text{with probability } 1-s \end{cases}, \tag{1}$$

where $0 \le s \le 1$ is the noise density level of the salt-and-pepper noise. The gray values of corrupted pixels change to either the maximum value $d_{\max}$ or the minimum value $d_{\min}$.

The random-valued impulse noise model is defined as follows:

$$\tilde{u}_{ij} = \begin{cases} d_{ij} & \text{with probability } r \\ u_{ij} & \text{with probability } 1-r \end{cases}, \tag{2}$$

where $0 \le r \le 1$ is the noise density level of the random-valued impulse noise. The gray values of $d_{ij}$ are identically and uniformly distributed random numbers between the maximum value $d_{\max}$ and the minimum value $d_{\min}$.

In the literature, most of denoising methods are aimed at removing either Gaussian or impulse noise, which are much easier than the mixed noise removal. These two types of noise affect the image in totally different ways, leading to different denoising methods. For Gaussian noise removal, the commonly used methods include total-variation methods [14–16] and wavelet shrinkage approaches [17,18]. The main drawback of the total-variation methods is that the texture information in images could be always oversmoothed [19]. Although the wavelet shrinkage methods perform much better for texture preserving, they may exhibit pseudo-Gibbs phenomena and bring artifacts in the recovered image [20,21]. Recently, Elad and Aharon introduced an effective denoising method via sparse and redundant representation over learned dictionary, called K-SVD algorithm [22], which can preserve the details and texture efficiently. In [23], Dabov et al. proposed another sparse representation-based denoising method in the transform domain. We refer to [24] for

* Corresponding author.
*E-mail address:* zeng@hkbu.edu.hk (T. Zeng).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.02.002

a comprehensive review on the developments of additive Gaussian noise removal methods.

For images corrupted by impulse noise, one of the most popular methods is median filter for its denoising ability and computational efficiency, see [25] and references therein for some previous works about nonlinear digital filtering. Recently, various modified median filters are also proposed, e.g. the adaptive median filter [26], the multi-state median filter [27], the median filter based on homogeneity information [28,29], and convolution kernels based on filter [30]. The general idea of these filters is that the location of the candidate noisy pixels are detected and replaced by some values of the pixels in the corresponding neighbor windows, while all the other pixels are unchanged. Although these filters gain fairly satisfactory results at the noise pixels detection stage, they cannot preserve the original local features well since the noisy pixels are just simply replaced by some values computed according to their neighbor pixels. Hence, much edges information will be removed if the impulse noise level is rather high.

In order to better preserve the edge structures of images, various variational approaches have been used for impulse noise removal [31–35]. In [35], a $l_1$ data-fidelity term was introduced, which is much suitable for impulse noise removal than $l_2$ norm used in [31–34]. This work makes a significant improvement for impulse noise removal. The main drawback of this method is that it changes the values of all the pixels in image, including the uncorrupted pixels, which should be unchanged during the restoration process. In order to resolve this problem, many effective two-phase methods are proposed [4,36–38], combining various variational models with different median filters. The first phase of their methods is to detect the location of noisy pixels corrupted by impulse noise using median filters, and then employ some variational methods to estimate the gray values for the noisy pixels in the second phase. Recently, this kind of two-phase methods has been further extended to handle mixed noise. For instance, Cai et al. proposed a modified two-phase method to deblur images corrupted by impulse noise plus Gaussian noise [3]. In [10], a total variation-based model was proposed for impulse and Gaussian noise removal. Lately, López–Rubio proposed a probabilistic theoretical framework to process images corrupted by Gaussian and uniform impulse noise [8]. We refer to [1,6,7] for more works on image restoration under mixed noise.

All these methods referred above for impulse noise removal are pixel-based methods. As they only consider image pixels independently, the local group features such as texture structures, repeated patterns cannot be preserved well. It is surprising that we cannot find any work in the literature handling impulse noise by sparse representation techniques of utilizing prior in images. However, it is interesting to note that sparse representation techniques have been proved to be extremely successful for Gaussian noise removal. In this paper, inspired by previous works in impulse noise removal and sparse representation, we propose a powerful patch-based three-phase denoising method. Our idea is to combine a median-type filter [26,39] with an effective dictionary learning method [40,22], and to recover images corrupted by Gaussian plus impulse noise. In the first phase, the impulse noise candidates are detected by the median-type filter, and then a modified K-SVD algorithm is used to recover the image, which shares some similarity with the truncated K-SVD method in [41,42]. Finally, we use an alternating minimization algorithm to solve a variational denoising model to enhance the processed image. The main contribution of this paper is that we propose and develop a double-sparsity approach for Gaussian plus impulse noise removal by combining both $l_1$ and $l_0$ regularization terms in the algorithm, and show that the proposed method outperforms the modified K-SVD method and the existing impulse noise removal algorithms.

The outline of the paper is as follows. In Section 2, we review some related works. In Section 3, we propose the new image restoration method for Gaussian and impulse noise removal. In Section 4, numerical experimental results are presented to illustrate the superior performance of the proposed method. The conclusion is presented in Section 5.

## 2. Review of related work

In this section, we review the well-known K-SVD method. The K-SVD method is proposed by Aharon and Elad [40], which is used for additive Gaussian noise removal in [22]. Suppose that a noisy image of $\mathbb{R}^{\sqrt{N}\times\sqrt{N}}$ indexed by $\mathcal{A}=\{1,2,\ldots,\sqrt{N}\}^2$, written as a column vector $f\in\mathbb{R}^N$, results from a zero-mean Gaussian noise $b\in\mathbb{R}^N$ with a known standard deviation $\sigma$ superimposed on an original image $u_0\in\mathbb{R}^N$. The basic assumption of the K-SVD method is that each image patch (of fixed size $\sqrt{n}\times\sqrt{n}$, and reformed as column vector $\mathbb{R}^n$) can be represented sparsely as a linear combination of atoms taken from a fixed *dictionary* $D\in\mathbb{R}^{n\times K}$. Here dictionary means a set of column vectors in $\mathbb{R}^n$. Each column vector is called an *atom* and is usually of unit norm. Note that typically, this dictionary can be overcomplete DCT dictionary (sampling the cosine wave in different frequencies to result a fixed number atoms, see [22,43] for details) or some other one learned from a noisy image (see below), and the pursuit algorithm used in this paper is the *Orthogonal Matching Pursuit* (OMP) algorithm [44]. Let us denote

$$\mathcal{P}=\{1,2,\ldots,\sqrt{N}-\sqrt{n}+1\}^2, \tag{3}$$

as the set of indices where small image patches exist.

For vector $x=(x_1,x_2,\ldots,x_m)\in\mathbb{R}^m$, let us denote the $l_0$ quantity:

$$\|x\|_0 := \#\{i|1\le i\le m,\ x_i\neq 0\},$$

as the number of the nonzero entries in the vector and denote

$$\|x\|_p=\left(\sum_{i=1}^{m}|x_i|^p\right)^{1/p},$$

as the classical $l_p$ norm in Euclidean space for $p\in[1,\infty)$.

Under the sparsity assumption, the Gaussian noise removal task can be described as an energy minimization task:

$$\{\hat{\alpha}_{ij},\hat{D},\hat{u}\}=\arg\min_{D,\alpha_{ij},u}\lambda\|f-u\|_2^2+\sum_{(i,j)\in\mathcal{P}}\|D\alpha_{ij}-R_{ij}u\|_2^2+\sum_{(i,j)\in\mathcal{P}}\mu_{ij}\|\alpha_{ij}\|_0. \tag{4}$$

In this model, the index $(i,j)$ with $1\le i,j\le\sqrt{N}-\sqrt{n}+1$ marks the location of the patch in the image, and $R_{ij}\in\mathbb{R}^{n\times N}$ is a binary matrix which extracts a $\sqrt{n}\times\sqrt{n}$ patch from the image in location $(i,j)$ and thus $R_{ij}u\in\mathbb{R}^n$. The first term demands a proximity between the measured image, $f$, and its denoised unknown version $u$. The second term demands that each patch from the reconstructed image (denoted by $R_{ij}u$) can be represented up to a bounded error by the dictionary $D\in\mathbb{R}^{n\times K}$, with coefficients vector $\alpha_{ij}\in\mathbb{R}^K$. Hence, $D\alpha_{ij}\in\mathbb{R}^n$. The third part demands that the number of coefficients required to represent any patch is small/sparse where the values $\mu_{ij}$ are patch-specific weights and are determined hiddenly by the optimization procedure. The minimization of this functional with respect to its unknowns yields the denoising algorithm.

The choice of $D$ is of high importance to the performance of the algorithm. In [40,22] it is shown that training can be done by minimization (4). The proposed algorithm in [40,22] is an iterative block-coordinate relaxation method which fixes all the unknowns except from the one to be updated. Table 1 gives a description of K-SVD algorithm, which is a generalization of the K-means clustering algorithm. The setting of the parameters will be discussed in the experimental section.

**Table 1**
K-SVD algorithm.

**Input:** Noisy image $f$ with additive Gaussian noise.
**Output:** The reconstruction image $\hat{u}$.
Parameters: $\lambda$ (Lagrange multiplier); $C$ (noise gain); $J$ (number of iterations); $K$ (number of the dictionary); $n$ (size of image patch); $\sigma$ (standard deviation of Gaussian noise).

1. Initialization: set $u=f$, Let $D=(d_l\in\mathbb{R}^{n\times 1})_{l\in 1\ldots K}$ be some initial dictionary.
2. Repeat $J$ times
   - Sparse Coding Stage: Use any pursuit algorithm to compute the representation vectors $\alpha_i$ for each example $x_i$ to minimize the function: $\forall_{ij}\min_{\alpha_{ij}}\|\alpha_{ij}\|_0$ s.t $\|R_{ij}u-D\alpha_{ij}\|_2^2\le(C\sigma)^2$
   - Dictionary Update Stage: For each column $l=1,2,\ldots,K$ in $D$.
     - Select the patches $w_l$ that use this atom $d_l$, $w_l=\{(i,j)|\alpha_{ij}(l)\neq 0\}$
     - For each patch $(i,j)\in w_l$, compute its residual without the contribution of the atom $d_l$

$e_{ij}^l=R_{ij}u-D\alpha_{ij}+d_l\alpha_{ij}$

     - Set $E_l=(e_{ij}^l)_{(i,j)\in w_l}$. Update $d_l$ and the $\alpha_{ij}(l)$ using SVD decomposition of $E_l$.
3. Reconstruction:

$\hat{u}=(\lambda I+\sum_{ij}R_{ij}^TR_{ij})^{-1}(\lambda f+\sum_{ij}R_{ij}^TD\alpha_{ij})$

## 3. Three-phase denoising approach

Inspired by the two-phase method for impulse noise removal [4] and the K-SVD [40,22] for dictionary learning, here we propose a $l_1$–$l_0$ minimization approach where the $l_1$ term is used for impulse noise denoising and the $l_0$ term is used for a sparse representation over certain unknown dictionary of images patches.

We denote $\mathcal{N}$ the set of the outlier candidate pixels corrupted by impulse noise and denote $\mathcal{U}=\mathcal{A}\backslash\mathcal{N}$ the left pixels without impulse noise, we consider the following model:

$$\min_{u,D,\alpha_{ij}}\sum_{(i,j)\in\mathcal{U}}\lambda\|u_{ij}-f_{ij}\|_2^2+\sum_{(i,j)\in\mathcal{N}}\beta|u_{ij}-f_{ij}|+\sum_{(i,j)\in\mathcal{P}}\|D\alpha_{ij}-R_{ij}u\|_2^2+\sum_{(i,j)\in\mathcal{P}}\mu_{ij}\|\alpha_{ij}\|_0, \tag{5}$$

where $\lambda,\beta$ are regularization parameters, $\mathcal{P}$ is given in (3) and $u\in\mathbb{R}^N$ is the estimated image. Again, the same as (4), $D\in\mathbb{R}^{n\times K}$ is the dictionary, $R_{i,j}\in\mathbb{R}^{n\times N}$ is the binary matrix to extract small patch from image $u$ at position $(i,j)$, the coefficient $\alpha_{i,j}\in\mathbb{R}^K$ is used to approximate the extracted patch $R_{i,j}u\in\mathbb{R}^n$. Hence, $D\alpha_{i,j}\in\mathbb{R}^n$. Moreover, for each $(i,j)\in\mathcal{P}$, $\mu_{ij}\in\mathbb{R}^K$ is hidden parameter decided by optimization procedure.

A three-phases denoising algorithm will be employed to solve this $l_1$–$l_0$ minimization problem. In the algorithm, the main three steps can be stated as follows:

- Identify the location of impulse noise (the outlier candidate pixels, denoted as set $\mathcal{N}$) using a median-type filter.
- Learn dictionary from the free-outlier pixels and reconstruct the image via the learned dictionary.
- Use an alternating minimization algorithm to solve the proposed $l_1$–$l_0$ denoising model, leading to a further restoration based on the recovered image in the second step.

Before giving the details of the alternating minimization algorithm, let us first present the outlier detection procedure and the Modified K-SVD method which simply combines the outlier detection and K-SVD.

### 3.1. Outlier detection

Let $f$ be a noisy image with Gaussian and impulse noise. The first step of our approach is to detect the outlier candidate pixels which are potentially corrupted by impulse noise. Here, we use adaptive median filter (AMF) [26] for salt-and-pepper noise detection and adaptive center-weighted median filter (ACWMF) [39] for random-valued impulse noise detection since they are simple and effective.

Suppose that $y\in R^{\sqrt{N}\times\sqrt{N}}$ is the filtered result by median-type filter. The candidates of noisy pixels contaminated by impulse noise can be determined as follows:

- For salt-and-pepper noise:
  $\mathcal{N}=\{(i,j)\in\mathcal{A}: y_{ij}\neq f_{ij}\text{ and }f_{ij}\in\{d_{\min},d_{\max}\}\}$,
- For random-valued impulse noise:
  $\mathcal{N}=\{(i,j)\in\mathcal{A}: y_{ij}\neq f_{ij}\}$.

Accordingly, the remained positions are more likely to be uncorrupted by impulse noise which are defined as set $\mathcal{U}$:

$\mathcal{U}\coloneqq\mathcal{A}\backslash\mathcal{N}$.

Let $\mathcal{X}$ be the characteristic matrix of the set $\mathcal{U}$ defined as

$$\mathcal{X}=\begin{cases}1 & \text{if } (i,j)\in\mathcal{U},\\ 0 & \text{otherwise,}\end{cases}$$

then the function (5) can be reformulated as follows:

$$\min_{u,D,\alpha_{i,j}}\lambda\|\mathcal{X}\otimes(u-f)\|_2^2+\beta\|(\mathbf{1}_f-\mathcal{X})\otimes(u-f)\|_1+\sum_{(i,j)\in\mathcal{P}}\|D\alpha_{ij}-R_{ij}u\|_2^2+\sum_{(i,j)\in\mathcal{P}}\mu_{ij}\|\alpha_{ij}\|_0, \tag{6}$$

where $\otimes$ is an entrywise multiplication between two matrices and $\mathbf{1}_f$ is a matrix of constant entries 1 and of the same dimensions with $f$. The first term is a data-fidelity term, which considers the pixels that are probably not corrupted by impulse noise, i.e only with Gaussian noise; the second data-fidelity term is an $\ell_1$ norm covering the outlier candidate pixels, since the outlier candidates are corrupted by impulse noise and $\ell_1$ is more reasonable than $\ell_2$ [35] for impulse noise. The role of $\ell_1$ will be much clear especially in the case of random-valued noise or rather high level salt-and-pepper noise where the detection of noisy position is less accurate. The last term is a sparse representation for image patches via learned dictionary.

### 3.2. Restoration based on the free-outlier data by modified K-SVD

After detection of the pixels corrupted by impulse noise, the remaining pixels in $\mathcal{U}$ are noisy but the noise is almost white Gaussian noise. Hence, a very simple idea is that we use K-SVD to learn dictionary based on the pixels in $\mathcal{U}$, and then reconstruct the image via a simple averaging between patches' approximations and the noisy image. The modified energy function can be formulated as follows:

$$\hat{u}=\arg\min_{u,D,\alpha_{ij}}\lambda\|\mathcal{X}\otimes(u-f)\|_2^2+\sum_{(i,j)\in\mathcal{P}}\mu_{ij}\|\alpha_{ij}\|_0+\sum_{(i,j)\in\mathcal{P}}\|(R_{ij}\mathcal{X})\otimes(D\alpha_{ij}-R_{ij}u)\|_2^2, \tag{7}$$

where the symbols are exactly the same as (6). Indeed, (7) is obtained by adding the outlier characteristic matrix in (4).

The main step of minimizing this modified K-SVD model (7) is similar to the original K-SVD algorithm. More specifically, we aim at solving the energy minimization problem of (7) over the free-outlier candidate pixels. Integrating the location of the outlier

candidate pixels requires the following key modifications to the basic algorithm:

- Sparse coding: All projections in the OMP algorithm included only the free-outlier candidate pixels, and for this purpose, the dictionary elements were normalized so that the free-outlier indices in each dictionary element have a unit norm. Given a fixed $D$,
$$\hat{\alpha}_{ij} = \arg\min_{\alpha} \|(R_{ij}\mathcal{X}) \otimes (R_{ij}u - D\alpha_{ij})\|_2^2 + \mu_{ij}\|\alpha_{ij}\|_0 \quad (8)$$
- Dictionary update: Fix all $\alpha_{ij}$, and for each atom $d_l$, $l \in 1,2,\ldots,K$ in $D$.
  - Select the patches $w_l$ that use this atom, $w_l = \{(i,j)|\alpha_{ij}(l) \neq 0\}$,
  - For each patch $(i,j) \in w_l$, compute its residual
  $$e_{ij}^l = R_{ij}u - D\alpha_{ij} + d_l\alpha_{ij} \quad (9)$$
  and $\mathcal{X}_{ij}^l = R_{ij}\mathcal{X}$ is an index vector of the free-outlier candidate pixels in the small image patch of size $\sqrt{n} \times \sqrt{n}$ from location $(i,j)$ in the image.
  - Set $E_l = (e_{ij}^l)_{(i,j) \in w_l}$ and $\mathcal{X}_l = (\mathcal{X}_{ij}^l)_{(i,j) \in w_l}$. Update $d_l$ by minimizing
  $$\hat{d}_l = \arg\min_{\|d\|_2 = 1} \|\mathcal{X}_l \otimes (E_l - d\alpha^T)\|_2^2. \quad (10)$$
  For this optimal problem, we fix $\alpha$ and solve a simple quadratic term with respect to $d$.
- Reconstruction: Perform a weighted average
$$u = \left(\lambda\mathcal{X} \otimes I + \sum_{(i,j) \in \mathcal{P}} R_{ij}^T R_{ij}\right)^{-1} \left(\lambda\mathcal{X} \otimes f + \sum_{(i,j) \in \mathcal{P}} R_{ij}^T \hat{D}\hat{\alpha}_{ij}\right) \quad (11)$$
which is a modified average expression based on the average step of K-SVD algorithm (see step 3 in Table 1) when $D$ and $\alpha$ are assumed fixed. Note that the gray levels of outlier candidate pixels rely on the reconstruction from the optimal $D$ and $\alpha$, not relate to the noisy values corrupted by the impulse noise.

When the impulse noise level is low, the modified K-SVD (we called it as MK-SVD in the following discussion) can provide satisfactory denoising results. However, when the impulse noise level is high, the MK-SVD may not learn a good dictionary from small number of free-outlier pixels. In order to improve the denoising results, below we consider two aspects: (1) adding a $l_1$-term to allow the false detection of outlier candidates (see (6)); (2) employing an alternating minimization algorithm to enhance restoration via the new dictionary learned from the recovered image. We expect the resulting algorithm can provide better dictionary atoms and denoising results.

### 3.3. Alternating minimization algorithm

In this subsection, we propose an alternating minimization method to solve the minimization problem (6) which iteratively updates $\alpha_{i,j}$, $D$ and $u$ individually by fixing the other two components in the iterative process. Initializing $u$ using the estimated image resulted from the second step, we consider three sub-problems:

- Given $u$, for each $(i,j) \in \mathcal{P}$, update the coefficient $\alpha_{i,j}$ by
$$\hat{\alpha}_{i,j} = \arg\min_{\alpha_{i,j}} \mu_{ij}\|\alpha_{ij}\|_0 + \|D\alpha_{ij} - R_{ij}u\|_2^2. \quad (12)$$
- Given $u, \alpha_{i,j}$, update the dictionary by $D$ by
$$\hat{D} = \arg\min_{D} \sum_{(i,j) \in \mathcal{P}} \|D\alpha_{ij} - R_{ij}u\|_2^2. \quad (13)$$
- Given $D, \alpha_{i,j}$, update $u$ by
$$\hat{u} = \arg\min_{u} \lambda\|\mathcal{X} \otimes (u-f)\|_2^2 + \beta\|(\mathbf{1}_f - \mathcal{X}) \otimes (u-f)\|_1 + \sum_{(i,j) \in \mathcal{P}} \|D\alpha_{ij} - R_{ij}u\|_2^2. \quad (14)$$

By comparison with the original K-SVD algorithm, the first and second steps (i.e., the sparse coding and dictionary updating steps) remain unchanged. The main difference between the proposed method and the original K-SVD is the reconstruction step. We denote
$$W = \sum_{(i,j) \in \mathcal{P}} R_{ij}^T R_{ij}, \quad M = \sum_{(i,j) \in \mathcal{P}} R_{ij}^T D\alpha_{i,j},$$
where $\mathcal{P}$ is defined in (3). Then $W$ and $M$ have the same dimensions with $u$ and $f$. Clearly, (14) is equivalent to
$$\hat{u} = \min_{u} \lambda\|\mathcal{X} \otimes (u-f)\|_2^2 + \beta\|(\mathbf{1}_f - \mathcal{X}) \otimes (u-f)\|_1 + \langle W \otimes u, u\rangle - 2\langle M, u\rangle, \quad (15)$$
where $\langle \cdot, \cdot \rangle$ is the usual Euclidean inner product. As the objective function in (15) is summation of convex terms such as a squares function, an absolute function and a linear function, it is a convex problem for the variable $u$. More precisely, we derive the closed form solution.

For any matrix $\Lambda$, denote $\Lambda_{i,j}$ as the value of position $(i,j)$. Noting that for any $(i,j) \in \mathcal{A}$, the weight $W_{i,j}$ means how many times the pixel at $(i,j)$ is used to construct image patches of size $\sqrt{n} \times \sqrt{n}$, thus we have
$$1 \leq W_{i,j} \leq n.$$

**Proposition 1.** *The minimization problem* (15) *has closed form.*

**Proof.** For any $(i,j) \in \mathcal{A}$, according to the value of $\mathcal{X}_{i,j}$, the solving of (15) boils down to two types of one-dimensional problem:

- if $\mathcal{X}_{i,j} = 1$, consider:
$$\min_{z \in \mathbb{R}} \lambda(z - f_{i,j})^2 + W_{i,j}z^2 - 2M_{i,j}z,$$
- if $\mathcal{X}_{i,j} = 0$, consider:
$$\min_{z \in \mathbb{R}} \beta|z - f_{i,j}| + W_{i,j}z^2 - 2M_{i,j}z.$$

For the first case, let $\mathcal{F}(z) = \lambda(z - f_{i,j})^2 + W_{i,j}z^2 - 2M_{i,j}z$. Since $\mathcal{F}$ is convex, the first case is equivalent to solving $\nabla\mathcal{F}(z) = 0$. The gradient of $\mathcal{F}$ is given by
$$\nabla\mathcal{F}(z) = 2(\lambda z - \lambda f_{i,j} + W_{i,j}z - M_{i,j}). \quad (16)$$
Then the solution for the first case is
$$z = \frac{M_{i,j} + \lambda f_{i,j}}{W_{i,j} + \lambda}. \quad (17)$$
This is indeed the same as the reconstruction step of algorithm K-SVD done on a pixel-by-pixel basis (see Table 1).

Now let $y = z - f_{i,j}$, the second case is then equivalent to solving:
$$\min_{y \in \mathbb{R}} \beta|y| + W_{i,j}(y - b)^2,$$
with $b = M_{i,j}/W_{i,j} - f_{i,j}$. It is easy to verify that this strictly convex problem has a unique minimizer (see [45,21]):
$$y = \text{shrink}\left(b, \frac{\beta}{2W_{i,j}}\right),$$
where for $\tau > 0$, the soft-shrinkage function is defined as
$$\text{shrink}(t,\tau) = \begin{cases} t - \tau & \text{if } t > \tau; \\ 0 & \text{if } |t| \leq \tau; \\ t + \tau & \text{otherwise.} \end{cases}$$

Therefore, the solution for the second case is

$$z = f_{i,j} + \text{shrink}\left(\frac{M_{i,j}}{W_{i,j}} - f_{i,j}, \frac{\beta}{2W_{i,j}}\right). \tag{18}$$

The solution of (15) is concluded as follows:

$$\hat{u}_{ij} = \begin{cases} \dfrac{M_{i,j} + \lambda f_{i,j}}{W_{i,j} + \lambda}, & \mathcal{X}_{i,j} = 1 \\ f_{i,j} + \text{shrink}\left(\dfrac{M_{i,j}}{W_{i,j}} - f_{i,j}, \dfrac{\beta}{2W_{i,j}}\right) & \mathcal{X}_{i,j} = 0 \end{cases}. \tag{19}$$

The above proposition is very useful for the understanding of the impulse noise removal problem. Indeed, from (17), we know that in position without impulse noise ($\mathcal{X}_{i,j} = 1$), then we take a tradeoff between $f_{i,j}$ and $M_{i,j}/W_{i,j}$ where the later is obtained from the information around this pixel. As this is basically Gaussian noise removal problem, the following choice of $\lambda$ which was suggested by the original K-SVD paper [22] will be a good starting point:

$$\lambda = \frac{30}{\sigma},$$

where $\sigma$ is Gaussian noise level (should be known or estimated elsewhere).

On the other side, in position with impulse noise ($\mathcal{X}_{i,j} = 0$), then from (18), the estimated value is just to shrink the neighborhood suggested value toward the direction $f_{i,j}$ by threshold $\beta/2W_{i,j}$. When the impulse noise level is higher, the noise candidate procedure will be less accurate and $f_{i,j}$ is more informative since it is more likely to be a true image pixel value (but wrongly declared as impulse noise position) and the neighborhood suggested value $M_{i,j}/W_{i,j}$ is then less informative, hence, we should take $\beta$ bigger. This analysis will be confirmed by numerical experiments.

Finally, the full algorithm for Gaussian plus impulse noise removal is given in Table 2. Let us remind the reader that this algorithm is rather general since it works for: impulse noise, Gaussian plus salt-and-pepper impulse noise, and Gaussian plus random-valued impulse noise.

**Table 2**
The full algorithm for Gaussian plus impulse noise removal.

**Input**: Noisy image $f$ corrupted by Gaussian plus impulse noise.
**Output**: The reconstruction image $u$.
Parameters: $T$ (maximum number of iterations);
1. Detect the outlier candidates pixels with impulse noise using a median-type filter, and construct the characteristic matrix $\mathcal{X}$.
2. Compute the estimated image $\hat{u}$ by solving formulation (7).
3. Repeat $T$ times.
   - Set the input image $f' = \hat{u}$.
   - Learn dictionary atoms $D$ and the corresponding coefficient matrix $\alpha$ using K-SVD, and then compute $W$ and $M$.
   - Solve $\hat{u}$ as the minimization of formulation (14) via Prop. 1.
4. Set the final recovered image $u = \hat{u}$.

## 4. Experimental results

In this section, experimental results are reported to validate the proposed method. We use eight test images of size $A \times B$: Barbara ($512 \times 512$), Boat ($512 \times 512$), Hill ($512 \times 512$), Lena ($512 \times 512$), Man ($512 \times 512$), Pepper ($512 \times 512$), Cameraman ($256 \times 256$) and House ($256 \times 256$), which are shown in Fig. 1. Peak signal to noise ratio (PSNR) is used to measure the quality of the restored images which is defined as following:

$$PSNR = 20\log_{10}\frac{255}{\frac{1}{AB}\|u^* - u\|_2}, \tag{20}$$

where $u^*$ is the restored image and $u$ is the original image. In all experiments, we set the parameters of K-SVD as following: dictionary size: $64 \times 256$, image patch size: $8 \times 8$ (empirically, we observe that these settings are nearly optimal), $J = 20$ since this is enough to ensure the convergence. We fix the window size $w_{\max} = 19$ and $T = 20$. The remaining parameters $\beta$ and $\lambda$ are tuned empirically to perform well. Roughly speaking, $\lambda \in [10/\sigma, 100/\sigma]$, $\beta \in [50, 200]$ which depend on image and noise levels. Note that currently in this paper, we assume that $\sigma$ is known or can be roughly estimated elsewhere. Further discussions about parameters $\beta, \gamma$ will be addressed later in this section.

In the simulations, images will be corrupted by Gaussian noise with standard deviation $\sigma$, salt-and-pepper noise with density level $s$ or random-valued impulse noise with density level $r$. According to the different types of mixed noise model, we illustrate the experiments in the coming three subsections: purely impulse noise; Gaussian noise with salt-and-pepper impulse noise; and Gaussian noise with random-valued impulse noise.

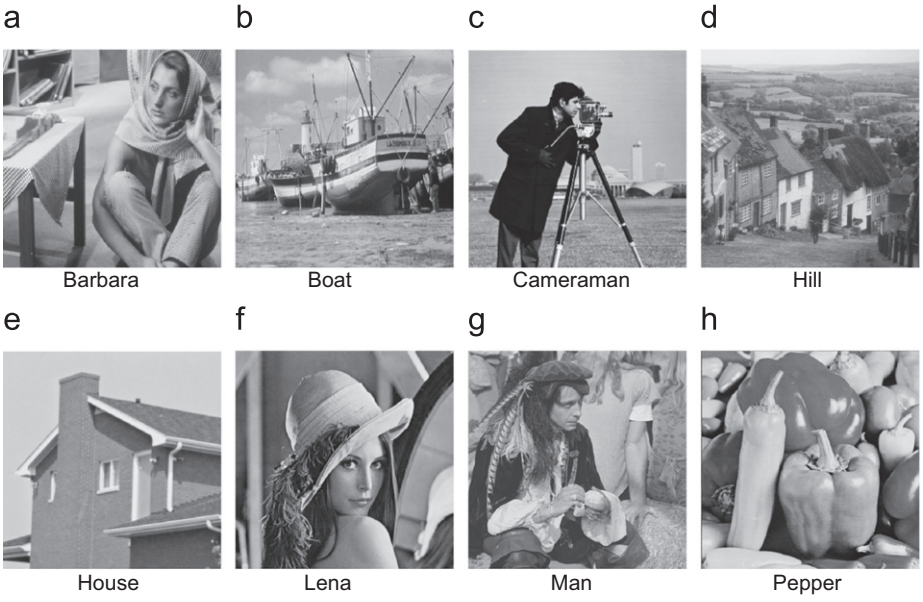


**Fig. 1.** Test images.

### 4.1. Purely impulse noise

Firstly, we consider the purely impulse noise removal problem: salt-and-pepper impulse noise or random-valued impulse noise. Note that though our algorithm is originally proposed to reduce mixed noise, it is natural that it also works for purely impulse noise since one can always use a rather large parameter $\lambda$ to remove a small amount of Gaussian noise. Furthermore, for salt-and-pepper impulse noise, the proposed method is compared with six algorithms including AMF [26], Wang [30], Mila [35], Cai1 [3], Cai2 [4] and MK-SVD. For random-valued impulse noise, the compared methods are ACWMF [39], Mila [35], Cai1 [3], Cai2 [4] and MK-SVD. Two typical images are used to test these algorithms: Lena, the one with homogeneous region; Barbara, the one with texture, and the salt-and-pepper noise levels are varied from 10% to 90% with increments of 10%, the random-valued impulse noise levels are varied from 10% to 40% with increments of 10%. The PSNR of the comparative methods with different noise levels are presented in Tables 3–6. Note that all the methods there are rather stable, the standard deviations of PSNR value of each method is less than 0.1.

It is observed from Tables 3 and 4 that the proposed method achieves higher PSNR than other noise reduction methods at every noise density for salt-and-pepper impulse noise. The PSNR of the proposed method demonstrates much better performance than AMF, Wang, Mila, Cai1 and Cai2 when the salt-and-pepper impulse noise level $s \le 80\%$ for Barbara and $s \le 60\%$ for Lena. Although MK-SVD gains much higher PSNR than the other five compared methods at relative low noise density, our method gains a significant improvement over MK-SVD, since our method can extract enough informative dictionary atoms from the restored images based on iterated study. Moreover, when the noise density level is high, the dictionary learned by MK-SVD cannot preserve well local features of the images while the proposed method can provide a much better dictionary by iteratively learning from the past recovered images. The PSNR of our method is roughly 1–2 dB improvement over MK-SVD when $10\% \le s \le 70\%$. When $s = 90\%$, i.e., almost 90% pixels are corrupted by impulse noise, it seems that neither MK-SVD nor our method can learn dictionary containing enough local details of the ideal images based on the other 10% noiseless pixels, but the PSNR value of our method is still slightly higher than all the other methods.

Tables 5 and 6 illustrate the PSNR value of various methods for random-valued impulse noise, which is more difficult than salt-and-pepper impulse noise. As the detection accuracy rate of random-valued impulse noise is much lower than salt-and-pepper impulse noise, the PSNR of each method for random-valued impulse noise is correspondingly lower than that for salt-and-pepper impulse noise removal. Compared with the other five algorithms, the proposed method gets higher improvement especially for Barbara which contains lots of textures. This is because the $l_1$ term of the proposed minimizing function makes our method less sensitive to the outlier detection accuracy.

**Table 3**
PSNR (dB) for various methods for Lena with salt-and-pepper noise.

| Noise density (%) | Noisy | Method | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | AMF | Wang | Mila | Cai1 | Cai2 | MK-SVD | Ours |
| 10 | 15.12 | 37.24 | 40.35 | 43.28 | 42.86 | 43.79 | 46.12 | **48.11** |
| 20 | 12.19 | 35.07 | 36.58 | 39.91 | 39.33 | 40.45 | 42.21 | **44.05** |
| 30 | 10.37 | 32.71 | 33.64 | 37.15 | 37.00 | 37.74 | 38.73 | **40.43** |
| 40 | 9.14 | 30.98 | 31.81 | 35.36 | 35.38 | 35.93 | 36.28 | **37.92** |
| 50 | 8.15 | 29.22 | 30.19 | 33.39 | 33.73 | 33.88 | 33.68 | **35.27** |
| 60 | 7.35 | 27.43 | 28.92 | 31.46 | 32.21 | 32.31 | 31.56 | **33.06** |
| 70 | 6.70 | 25.75 | 27.75 | 29.47 | 30.53 | 30.32 | 29.50 | **30.75** |
| 80 | 6.13 | 23.88 | 26.25 | 27.70 | 28.29 | 28.28 | 27.20 | **28.40** |
| 90 | 5.60 | 21.04 | 24.13 | 24.41 | 25.13 | 25.29 | 25.00 | **25.70** |

**Table 4**
PSNR (dB) for various methods for Barbara with salt-and-pepper noise.

| Noise density (%) | Noisy | Method | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | AMF | Wang | Mila | Cai1 | Cai2 | MK-SVD | Ours |
| 10 | 15.23 | 28.75 | 33.34 | 35.12 | 35.14 | 35.11 | 43.97 | **45.80** |
| 20 | 12.21 | 27.66 | 29.88 | 31.67 | 31.82 | 31.58 | 39.50 | **41.40** |
| 30 | 10.49 | 26.41 | 27.69 | 29.51 | 29.67 | 29.36 | 36.03 | **37.61** |
| 40 | 9.28 | 25.22 | 26.07 | 27.93 | 28.17 | 27.76 | 33.52 | **34.87** |
| 50 | 8.28 | 24.13 | 24.74 | 26.65 | 26.84 | 26.60 | 31.10 | **32.30** |
| 60 | 7.50 | 22.97 | 23.53 | 25.48 | 25.74 | 25.51 | 28.96 | **29.77** |
| 70 | 6.81 | 22.62 | 21.91 | 24.30 | 24.85 | 24.58 | 26.84 | **27.62** |
| 80 | 6.23 | 20.66 | 21.56 | 22.70 | 23.81 | 23.67 | 24.62 | **25.32** |
| 90 | 5.72 | 18.93 | 20.37 | 19.34 | 22.64 | 22.52 | 22.53 | **22.90** |

**Table 5**
PSNR (dB) for various methods for Lena with random-valued impulse noise.

| Noise density (%) | Noisy | Method | | | | | |
|---|---|---|---|---|---|---|---|
| | | ACWMF | Mila | Cai1 | Cai2 | MK-SVD | Ours |
| 10 | 18.63 | 35.97 | 35.45 | 36.08 | 35.97 | 36.23 | **37.01** |
| 20 | 15.61 | 33.67 | 33.55 | 34.03 | 33.97 | 34.17 | **34.72** |
| 30 | 13.80 | 31.49 | 31.76 | 32.08 | 32.11 | 32.07 | **32.47** |
| 40 | 12.57 | 29.18 | 29.86 | 30.55 | 30.28 | 30.17 | **30.80** |

**Table 6**
PSNR (dB) for various methods for Barbara with random-valued impulse noise.

| Noise density (%) | Noisy | Method | | | | | |
|---|---|---|---|---|---|---|---|
| | | ACWMF | Mila | Cai1 | Cai2 | MK-SVD | Ours |
| 10 | 18.84 | 25.87 | 26.64 | 25.87 | 25.87 | 26.64 | **31.26** |
| 20 | 15.83 | 25.13 | 25.52 | 25.13 | 25.13 | 25.91 | **27.62** |
| 30 | 14.05 | 24.39 | 24.55 | 24.43 | 24.39 | 25.00 | **26.14** |
| 40 | 12.82 | 23.59 | 23.80 | 23.86 | 23.75 | 24.26 | **24.90** |

### 4.2. Gaussian and salt-and-pepper noise

For Gaussian and salt-and-pepper noise removal, we compare the results by the K-SVD with the adaptive median filter for outlier candidate pixels.

From Fig. 2, we know that the proposed method gains much better results than the simple combination of the results from AMF and the K-SVD. The denoising results of the AMF algorithm destroy the original local features, which leads to much smaller improvement by K-SVD due to the poor learned dictionary, especially when the impulse noise level is high.

Here, we select three related algorithms to compare with the proposed algorithm, including Cai1 [3] Cai2 [4] and MK-SVD, since the other three methods are designed for pure impulse noise removal. Extensive experiments are conducted on all the eight test images. The salt-and-pepper noise levels are varied from 30% to 70% with increments of 20%, and Gaussian noise levels varies with $\sigma = 5$, 10 and 15.

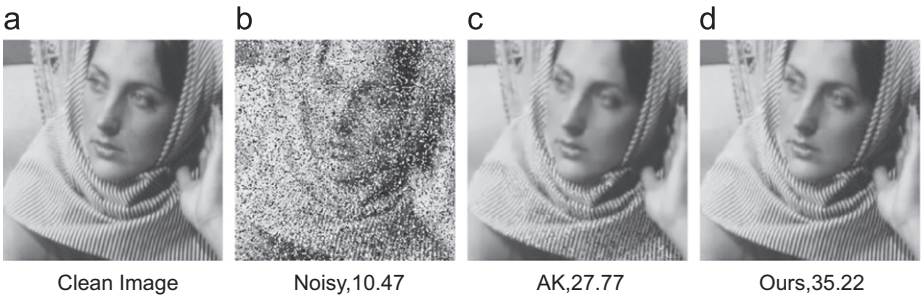


**Fig. 2.** Denoising results of different methods on image Barbara corrupted by Gaussian and salt-and-pepper noise with $\sigma=5$ and $s=30\%$. AK stands for AMF+K-SVD.

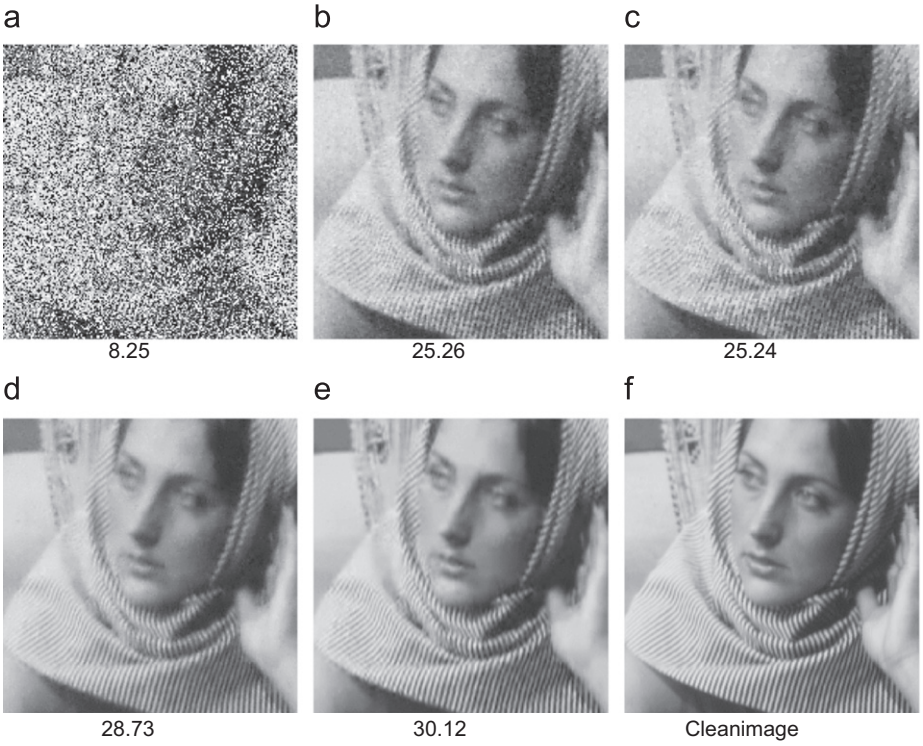


**Fig. 3.** Denoising results of different methods on image Barbara corrupted by Gaussian noise and salt-and-pepper noise with $\sigma=10$ and $s=50\%$: PSNR (dB) values. From (a) to (f): Noisy image, Cai1, Cai2, MK-SVD, our algorithm and clean image.

Figs. 3–6 show the denoising results for Lena and Barbara with Gaussian noise $\sigma=10{,}5$ and salt-and-pepper noise with noise levels 50% and 70%. It can be seen that the proposed method outperforms the other three methods: better visual quality and higher PSNR values. In particular, a notable improvement is observed for Barbara. The texture areas of Barbara are well reconstructed since the image content structured objects (textures) can be well recovered by the learned dictionary. Next, we demonstrate the learned dictionaries of Barbara and Lena with mixed noise $\sigma=5$ and $s=30\%$ and 70%. Seen from Figs. 7 and 8, we can note that there are big differences between the two dictionaries learned from different images. Indeed, much more texture atoms are learned from Barbara. Moreover, the dictionaries learned by our method are much clearer (smooth) and can describe the image features more effectively, especially under the impulse noise level $s=70\%$, which leads to better denoising results.

Table 7 presents the quantitative results of the four denoising algorithms on all test images. Clearly, the proposed algorithm achieves promising denoising results: high improvement for images with much more texture areas, such as Barbara. The PSNR of our method is roughly 4–7 dB improvement over the Cai1 and Cai2 methods on Barbara when the impulse noise level is less than 70%; and 1–2 dB improvement over the Cai1 and Cai2 algorithms on other seven test images. Moreover, MK-SVD and our method show similar level of performance with low level salt-and-pepper impulse noise and high Gaussian noise, while our method gains about 0.5–1.5 dB improvement over MK-SVD with other mixed noise levels.

By comparing with Cai1 and Cai2, we can see clearly that the proposed method successfully suppresses the noise and preserves the image details and textures very accurately. Note that MK-SVD can also get relatively satisfactory denoising results when the salt-and-pepper noise level $s=30\%$ and Gaussian noise $\sigma\neq 0$. However, the PSNR of our method is roughly 1 dB improvement over the MK-SVD when the Gaussian noise level $\sigma\leq 10$.

Moreover, we also test our method on Lena with relative high level of Gaussian noise $\sigma=5{,}10{,}15{,}20{,}25{,}30{,}40$ and impulse noise $s=30\%$, 50% and 70%, the PSNR of different methods are presented in Fig. 9. Clearly, our proposed method also works rather well.

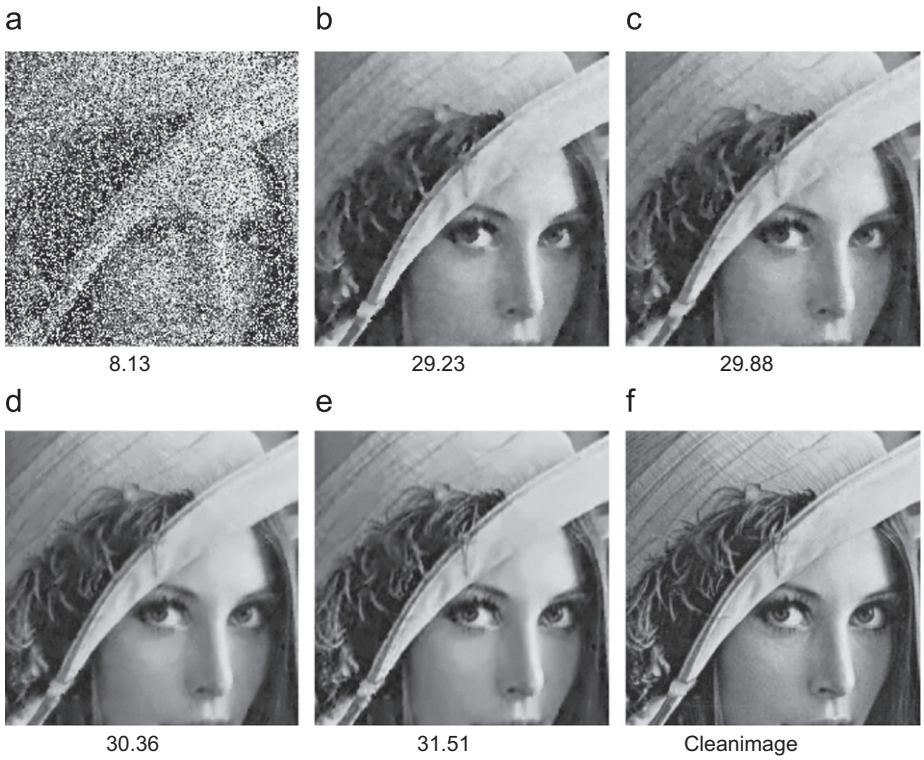


**Fig. 4.** Denoising results of different methods on image Lena corrupted by Gaussian noise and salt-and-pepper noise with $\sigma = 10$ and $s = 50\%$: PSNR(dB) values. From (a) to (f): Noisy image, Cai1, Cai2, MK-SVD, our algorithm and clean image.

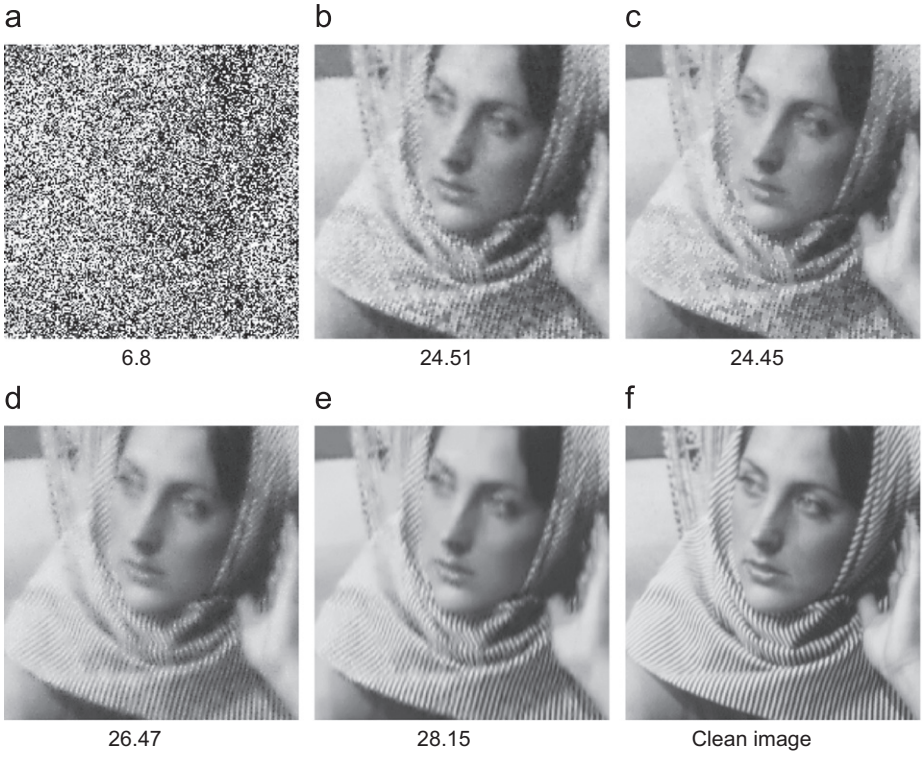


**Fig. 5.** Denoising results of different methods on image Barbara corrupted by Gaussian noise and salt-and-pepper noise with $\sigma = 5$ and $s = 70\%$: PSNR (dB) values. From (a) to (f): Noisy image, Cai1, Cai2, MK-SVD, our algorithm and clean image.

### *4.3. Gaussian and random-valued impulse noise*

For Gaussian and random-valued impulse noise, we also first carry a test on the performance of ACWMF plus K-SVD. The results are reported in Fig. 10. Clearly, the method combining ACWMF and K-SVD directly cannot work well for this kind of mixed noise. Indeed, it is the same as the case of Gaussian and salt-and-pepper impulse noise removal. Next, the proposed

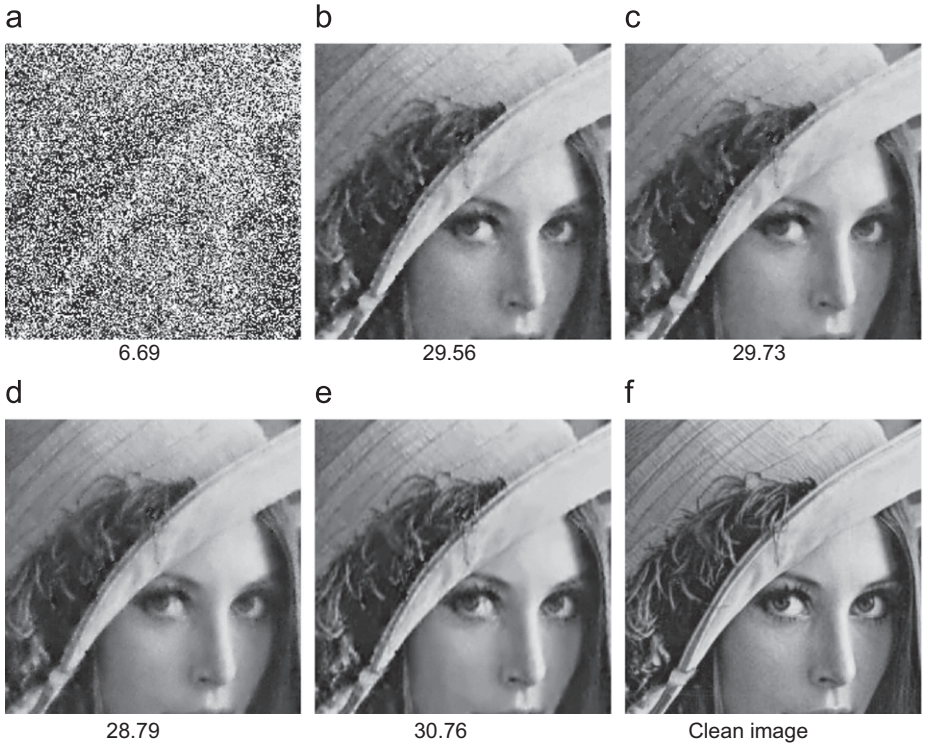


**Fig. 6.** Denoising results of different methods on image Lena corrupted by Gaussian noise and salt-and-pepper noise with $\sigma = 5$ and $s = 70\%$: PSNR (dB) values. From (a) to (f): Noisy image, Cai1, Cai2, MK-SVD, our algorithm and clean image.

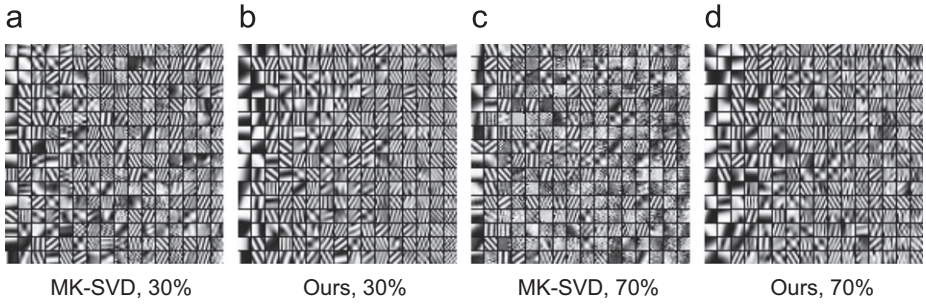


**Fig. 7.** The learned dictionaries of Barbara under impulse noise with levels 30%,70% and Gaussian noise with $\sigma = 5$.

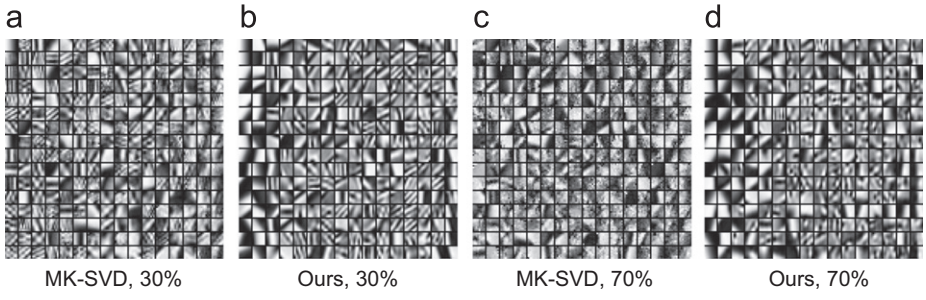


**Fig. 8.** The learned dictionaries of Lena under impulse noise with levels 30%, 70% and Gaussian noise with $\sigma = 5$.

method is tested for Gaussian and random-valued impulse noises removal. Four related algorithms are chosen to compare with the proposed algorithm, including Cai1 [3], Cai2 [4] and MK-SVD. Here, we also use the two typical images: Lena and Barbara to illustrate the performance of our method. The PSNR values of the restoration by four compared methods are given in Tables 8 and 9. The random-valued impulse noise levels varies from 10% to 30% with increments of 10%, and Gaussian noise levels varies with $\sigma = 5,10,15$. It is observed from Tables 8 and 9 that the proposed method achieves much higher PSNR values than other noise reduction methods at every noise density.

**Table 7**
PSNR (dB) for various methods at Gaussian noise and salt-and-pepper impulse noise, MS stands for MK-SVD.

| Image/s | $\sigma=5$ | | | | | $\sigma=10$ | | | | | $\sigma=15$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Noisy | Cai1 | Cai2 | MS | Ours | Noisy | Cai1 | Cai2 | MS | Ours | Noisy | Cai1 | Cai2 | MS | Ours |
| Bar./30% | 10.47 | 27.41 | 28.40 | 33.54 | **35.22** | 10.46 | 25.76 | 26.78 | 31.42 | **32.08** | 10.38 | 25.04 | 25.26 | 29.88 | **30.18** |
| Bar./50% | 8.28 | 26.50 | 26.15 | 30.09 | **32.47** | 8.25 | 25.26 | 25.24 | 28.73 | **30.12** | 8.23 | 24.03 | 24.22 | 27.50 | **28.44** |
| Bar./70% | 6.80 | 24.51 | 24.45 | 26.47 | **28.15** | 6.79 | 23.87 | 23.96 | 25.84 | **26.88** | 6.80 | 23.14 | 23.27 | 25.01 | **26.04** |
| Boa./30% | 10.68 | 31.25 | 32.34 | 33.23 | **34.25** | 10.65 | 29.11 | 29.89 | 31.26 | **31.88** | 10.57 | 27.79 | 28.24 | 29.74 | **30.09** |
| Boa./50% | 8.48 | 29.55 | 30.23 | 30.36 | **32.16** | 8.46 | 27.84 | 28.41 | 28.91 | **29.98** | 8.43 | 26.65 | 27.00 | 27.76 | **28.44** |
| Boa./70% | 7.02 | 27.51 | 27.45 | 27.03 | **28.66** | 7.03 | 26.43 | 26.40 | 26.14 | **27.17** | 7.00 | 25.29 | 25.43 | 25.50 | **26.40** |
| Cam./30% | 10.27 | 29.31 | 29.90 | 30.45 | **31.92** | 10.20 | 26.70 | 27.71 | 29.05 | **29.79** | 10.23 | 25.65 | 26.10 | 28.05 | **28.25** |
| Cam./50% | 8.09 | 26.92 | 27.40 | 27.39 | **29.00** | 8.08 | 25.45 | 25.99 | 26.45 | **27.71** | 8.03 | 24.10 | 24.69 | 25.57 | **26.19** |
| Cam./70% | 6.61 | 24.69 | 24.67 | 24.06 | **25.51** | 6.61 | 23.35 | 23.45 | 23.14 | **24.16** | 6.62 | 22.41 | 22.67 | 22.50 | **23.13** |
| Hou./30% | 10.69 | 33.21 | 34.92 | 35.70 | **37.19** | 10.66 | 31.08 | 32.11 | 33.47 | **33.98** | 10.54 | 29.78 | 30.67 | 32.15 | **32.42** |
| Hou./50% | 8.43 | 32.12 | 33.21 | 32.67 | **35.40** | 8.44 | 29.69 | 30.98 | 31.03 | **32.74** | 8.41 | 28.58 | 29.70 | 30.15 | **31.26** |
| Hou./70% | 7.03 | 29.74 | 30.26 | 28.93 | **32.02** | 7.00 | 28.25 | 29.05 | 28.04 | **29.87** | 6.99 | 27.03 | 27.94 | 27.25 | **29.01** |
| Hill/30% | 10.58 | 32.29 | 33.04 | 33.74 | **34.42** | 10.57 | 29.90 | 30.43 | 31.52 | **31.87** | 10.48 | 28.64 | 28.92 | 30.05 | **30.26** |
| Hill/50% | 8.36 | 30.94 | 31.26 | 31.24 | **32.44** | 8.35 | 28.97 | 29.37 | 29.79 | **30.66** | 8.33 | 27.86 | 28.03 | 28.68 | **29.17** |
| Hill/70% | 6.92 | 29.36 | 29.14 | 28.58 | **29.99** | 6.89 | 27.85 | 27.98 | 27.71 | **28.61** | 6.89 | 26.69 | 26.80 | 26.97 | **27.60** |
| Lena/30% | 10.39 | 33.08 | 34.15 | 35.09 | **36.20** | 10.33 | 30.66 | 31.33 | 32.77 | **33.19** | 10.24 | 29.12 | 29.67 | 31.24 | **31.49** |
| Lena/50% | 8.14 | 31.40 | 32.30 | 32.06 | **33.93** | 8.13 | 29.23 | 29.88 | 30.36 | **31.51** | 8.13 | 27.93 | 28.42 | 29.19 | **29.95** |
| Lena/70% | 6.69 | 29.56 | 29.73 | 28.79 | **30.76** | 6.70 | 27.75 | 28.11 | 27.67 | **28.98** | 6.68 | 25.87 | 26.48 | 26.64 | **27.53** |
| Man/30% | 10.63 | 31.98 | 33.06 | 33.30 | **34.22** | 10.56 | 29.80 | 30.35 | 31.22 | **31.63** | 10.50 | 28.57 | 28.29 | 29.82 | **30.05** |
| Man/50% | 8.40 | 30.91 | 31.08 | 30.50 | **31.91** | 8.40 | 28.74 | 29.21 | 29.20 | **30.14** | 8.35 | 27.64 | 28.01 | 28.22 | **28.81** |
| Man/70% | 6.94 | 28.55 | 28.72 | 27.68 | **29.28** | 6.94 | 27.38 | 27.64 | 26.95 | **28.26** | 6.93 | 26.27 | 26.57 | 26.25 | **27.16** |
| Pep./30% | 10.52 | 32.68 | 33.87 | 34.70 | **35.50** | 10.49 | 30.64 | 31.66 | 32.75 | **33.23** | 10.42 | 29.63 | 30.25 | 31.50 | **31.75** |
| Pep./50% | 8.30 | 30.96 | 31.83 | 32.06 | **33.38** | 8.30 | 29.37 | 30.24 | 30.72 | **31.82** | 8.27 | 28.22 | 28.85 | 29.63 | **30.37** |
| Pep./70% | 6.84 | 29.35 | 29.58 | 29.06 | **30.81** | 6.84 | 27.83 | 28.43 | 28.08 | **29.39** | 6.82 | 26.82 | 27.07 | 27.13 | **27.89** |

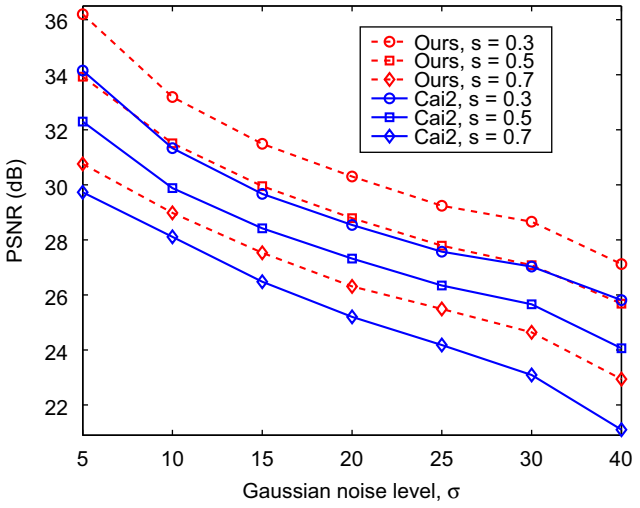


**Fig. 9.** The PSNR of the denoising results for different methods with different noise levels.

### 4.4. Extension to color image denoising

Our method can also be extended to color image denoising. Let a clean color image be represented in the commonly used RGB space and let $x_1, x_2, \ldots, x_n$ be 3-dimensional samples from the image $\mathbf{x}$ with $x_i=(x_{i1}, x_{i2}, x_{i3})^T$. The simple salt-and-pepper impulse noise model for color image used in the literature is defined as

$$\tilde{x}_{iq}=\begin{cases} d_{\min} & \text{with probability } s/2 \\ d_{\max} & \text{with probability } s/2 \\ x_{iq} & \text{with probability } 1-s \end{cases}, \tag{21}$$

where $q=1,2,3$ denote image channels. Moreover, we assume that the observed image $f$ is obtained from the following mixed noise model

$$f=\mathbb{N}_{imp}(x+b).$$

Like for the grayscale noisy image, we can use our method for each RGB channel separately, and then combine the denoising results returned from each color channel together as the final denoising result. In Fig. 11, two $256\times256\times3$ color images are tested with impulse noise $s=30\%$ and Gaussian noise $\sigma=5$. Clearly, the proposed method also achieves satisfactory denoising results for color images. Note that the border problem there for the House image is due to the flaw of the original image (see Fig. 11(d)).

### 4.5. Discussion

In this subsection, we study the effect of the parameters $\lambda$ and $\beta$ in the proposed energy function (6). We test the two parameters separately, i.e., we tune the value of one parameter while the other parameter is fixed. As expected, we found that the choice of these two parameters is dependent on the mixed noise level. The role of parameter $\lambda$ is to adjust the weight of the data-fidelity term which only covers the non-candidate pixels corrupted by impulse noise. As the Gaussian noise level increases, better results are achieved with small values of $\lambda$, and vise versa. Since the pixels with relative low level of Gaussian noise should have a strong effect on the final restored results, while ones with high levels of Gaussian noise should effect the outcome weakly. Fig. 12(a) presents the improvement achieved when increasing the value of $\lambda$ on the test image Lena with two salt-and-pepper impulse noise levels ($s=30\%$, $70\%$) and two Gaussian noise levels ($\sigma=5,15$).

The parameter $\beta$ is the weight coefficient of $l_1$ norm term, which covers the candidate pixels assumed to be corrupted by impulse noise. The $l_1$ term involves an implicit detection of outliers, which will ensure that the pixels (which are wrongly

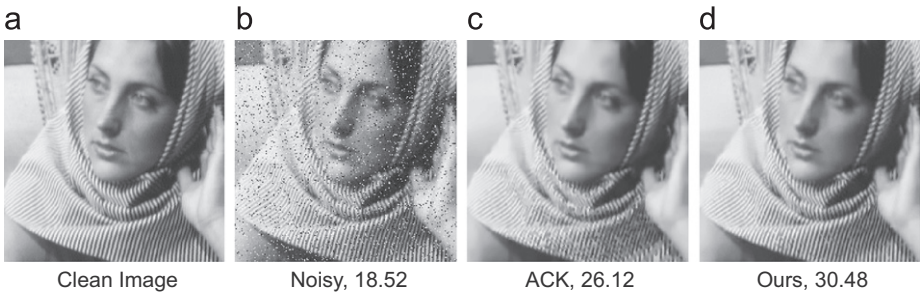


**Fig. 10.** Denoising results of different methods on image Barbara corrupted by Gaussian and random-valued impulse noise with $\sigma=5$ and $r=10\%$. ACK stands for ACWMF+K-SVD.

**Table 8**
PSNR (dB) for various methods for Lena with Gaussian noise and random-valued impulse noise.

| $\sigma$ | $r$ (%) | Noisy | Cai1 | Cai2 | MK-SVD | Ours |
|---|---|---|---|---|---|---|
| 5 | 10 | 18.52 | 32.77 | 33.78 | 33.94 | **34.98** |
| | 20 | 15.52 | 31.78 | 32.37 | 32.74 | **33.64** |
| | 30 | 13.73 | 30.69 | 31.21 | 31.24 | **32.04** |
| 10 | 10 | 18.19 | 30.80 | 31.01 | 31.60 | **32.75** |
| | 20 | 15.38 | 30.02 | 30.33 | 30.80 | **31.66** |
| | 30 | 13.67 | 29.08 | 29.42 | 29.69 | **30.42** |
| 15 | 10 | 17.67 | 29.13 | 29.34 | 29.49 | **30.85** |
| | 20 | 15.14 | 28.59 | 28.78 | 28.88 | **29.98** |
| | 30 | 13.54 | 27.96 | 28.20 | 28.28 | **29.11** |

**Table 9**
PSNR (dB) for various methods for Barbara with Gaussian noise and random-valued impulse noise.

| $\sigma$ | $r$ (%) | Noisy | Cai1 | Cai2 | MK-SVD | Ours |
|---|---|---|---|---|---|---|
| 5 | 10 | 18.78 | 25.33 | 25.52 | 26.34 | **30.48** |
| | 20 | 15.78 | 24.72 | 24.84 | 25.56 | **27.76** |
| | 30 | 14.03 | 24.10 | 24.26 | 24.78 | **25.92** |
| 10 | 10 | 18.36 | 24.46 | 24.68 | 25.43 | **28.42** |
| | 20 | 15.59 | 23.99 | 24.28 | 24.97 | **26.59** |
| | 30 | 13.97 | 23.66 | 23.43 | 24.38 | **25.34** |
| 15 | 10 | 17.86 | 23.80 | 24.12 | 24.44 | **27.31** |
| | 20 | 15.39 | 23.60 | 23.76 | 24.16 | **25.69** |
| | 30 | 13.77 | 23.27 | 23.46 | 23.69 | **24.55** |

considered as outliers by the median filter) are fitted exactly while real outliers are replaced by estimates by the last term in Eq. (6). The value of the parameter $\beta$ is directly dependent on the accuracy of the outlier detection by the median filter, which relates to both the Gaussian noise level and impulse noise level. As expected, low level mixed noise leads to high accuracy of outlier detection, so small values of $\beta$ should be proper, while high level mixed noise need relatively bigger values of $\beta$. Fig. 12(b) presents the improvement achieved when increasing the value of $\beta$ on the test image Lena with two random-valued impulse noise levels ($r=10\%$, 30%) and two Gaussian noise levels ($\sigma=5,15$) for some fixed $\lambda$.

Moreover, the simulations are performed in Matlab 7.6 (R2009b) on a PC equipped with 2.99 GHz CPU and 2 GB RAM memory. The CPU time requirements of the compared methods and the proposed method are represented in Table 10. According to Table 10, MK-SVD is the fastest one among the four compared methods, which also gains relative satisfactory denoising results for most cases in our experiments. Let us remind the reader that as far as we know, MK-SVD has never been clearly addressed for Gaussian and impulse noise removal in the literature before this paper. The proposed alternating method is based on the MK-SVD, which gains better denoising results than MK-SVD and requires much less time than Cai1.

Moreover, the CPU time of our method can still be reduced at least via three possibilities: (1) a computer with better configurations since ours for testing is not too good; (2) including some parallel computing techniques since the most time consuming procedure is the coefficient updating step (12) which can be computed in paralleling way; (3) to stop the outside iterations timely once the satisfactory resulting image has been obtained. Currently we use maximum iteration number $T=20$ which in many cases is largely unnecessary.

## 5. Conclusion

In this paper, we provide a powerful patch-based three-phase denoising method to solve the proposed $l_1$–$l_0$ denoising model. The three-phase algorithm combines a median-type filter with an effective dictionary learning method: K-SVD, to recover images corrupted by Gaussian plus impulse noise. The main contributions of our method are clear: compared with classical impulse-related noise removal algorithm, we propose a double-sparsity approach which clearly outperforms previous works; compared to the usual K-SVD, we consider an energy containing $l_1$–$l_0$ regularization where $l_1$ is proved more suitable for impulse noise removal.

The performance of the proposed method has been compared with some state of the art algorithms for both impulse noise and mixed noise removal tasks. The quantitative and qualitative results on test images demonstrate that our method can remove the noise efficiently while preserving the image local features, even at a rather high impulse noise level with Gaussian noise.

Our results can be further improved by using different noise detectors to increase the accuracy of the outlier detection especially when Gaussian noise density is high and other dictionary learned methods for high level impulse noise. Also the three-phase denoising method proposed in this paper may be further studied, considering the correlation between the image channels, for color image restoration under Gaussian plus impulse noise. Another possible extension is for handling image deblurring with mixed noise, which is also our future research topic.

## Acknowledgements

This work is supported by RGC 203109, RGC 201508, RGC 211710, the FRGs of Hong Kong Baptist University, NSFC (Grant no. 90820013, 61033013, 60875031, 60905029, China) and the


**Fig. 11.** Denoising results of our methods on color images Lena and House in terms of PSNR. From left to right: clean image, noisy image and restored image.

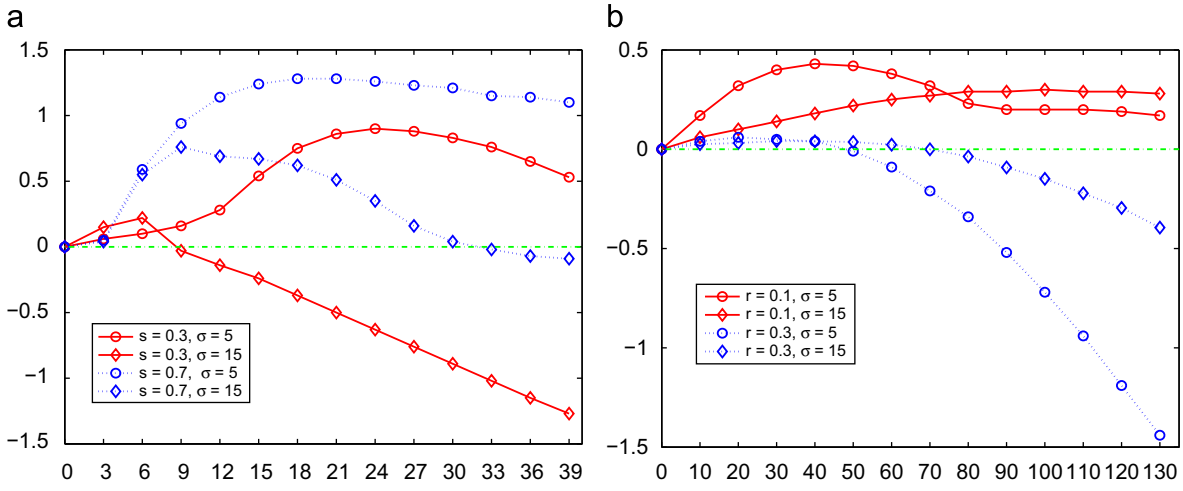


**Fig. 12.** Improvement of the denoising results when increasing the values of $\lambda$ and $\sigma$ for the image Lena. Left: increasing $\lambda$ for salt-and-pepper noise; right: increasing $\beta$ for random-valued noise.

**Table 10**
Time (s) for various methods for Lena and House with different levels of noise where $s+\sigma$ stands for salt-and-pepper noise of density $s$ plus Gaussian noise of std $\sigma$.

| *Image* | $s+\sigma$ | Cai1 | Cai2 | MK-SVD | Ours |
|---|---|---|---|---|---|
| Lena | 30%+5 | 773 | 93 | 64 | 338 |
| | 70%+5 | 283 | 119 | 50 | 247 |
| | 30%+15 | 1556 | 143 | 31 | 215 |
| | 70%+15 | 1084 | 248 | 31 | 176 |
| House | 30%+5 | 78 | 17 | 21 | 159 |
| | 70%+5 | 69 | 24 | 19 | 111 |
| | 30%+15 | 299 | 23 | 9 | 102 |
| | 70%+15 | 321 | 34 | 11 | 102 |

973 Project (Grant no. 2007CB311002, China). The authors would like to thank the helpful comments of the anonymous reviewers.

## References

[1] J. Liu, Z. Huan, H. Huang, An adaptive method for recovering image from mixed noisy data, International Journal of Computer Vision 85 (2009) 182–191.
[2] R. Chan, Y. Dong, M. Hintermüller, An efficient two-phase L1-TV method for restoring blurred images with impulse noise, IEEE Transactions on Image Processing 19 (2010) 1731–1739.
[3] J.-F. Cai, R. Chan, M. Nikolova, Two-phase methods for deblurring images corrupted by impulse plus gaussian noise, Inverse Problem Imaging 2 (2008) 187–204.

[4] J.-F. Cai, R. Chan, M. Nikolova, Fast two-phase image deblurring under impulse noise, Journal of Mathematical Imaging and Vision 36 (2010) 46–53.
[5] B. Zhang, M.-J. Fadili, J.-L. Starck, J.-C. Olivo-Marin, Multiscale variance-stabilizing transform for mixed-Poisson–Gaussian processes and its application in bioimaging, IEEE International Conference on Image Processing (2007) 233–236.
[6] S. Morillas, V. Gregori, A. Hervás, Fuzzy peer groups for reducing mixed Gaussian-impulse noise from color images, IEEE Transactions on Image Processing 18 (2009) 1452–1466.
[7] J.-X. Yang, H.-R. Wu, Mixed Gaussian and uniform impulse noise analysis using robust estimation for digital images, in: Proceedings of the 16th International Conference on Digital Signal Processing, 2009, pp. 468–472.
[8] E. López-Rubio, Restoration of images corrupted by Gaussian and uniform impulsive noise, Pattern Recognition 43 (2010) 1835–1846.
[9] L. Ji, Z. Yi, A mixed noise image filtering method using weighted-linking PCNNs, Neurocomputing 71 (2008) 2986–3000.
[10] Y. Huang, M. Ng, Y. Wen, Fast image restoration methods for impulse and Gaussian noise removal, IEEE Signal Processing Letters 16 (2009) 457–460.
[11] O. Ghita, P.-F. Whelan, A new GVF-based image enhancement formulation for use in the presence of mixed noise, Pattern Recognition 43 (2010) 2646–2658.
[12] C. Lin, J. Tsai, C. Chiu, Switching bilateral filter with a texture/noise detector for universal noise removal, IEEE Transactions on Image Processing 19 (2010) 2307–2320.
[13] H. Kong, L. Guan, A neural network adaptive filter for the removal of impulse noise in digital images, Neural Networks 9 (1996) 993–1003.
[14] L. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithm, Physica D 60 (1992) 259–268.
[15] T. Chan, K. Chen, An optimization-based multilevel algorithm for total variation image denoising, SIAM Journal of Multiscale Modeling and Simulation 5 (2006) 615–645.
[16] T. Chan, S. Esedoglu, F. Park, M. Yip, Recent developments in total variation image restoration, in: Mathematical Models of Computer Vision, 2005.
[17] M. Figueiredo, J. Bioucas-Dias, R. Nowak, Majorization-minimization algorithms for wavelet-based image restoration, IEEE Transactions on Image Processing 16 (2007) 2980–2991.
[18] F. Luisier, T. Blu, M. Unser, A new SURE approach to image denoising: interscale orthonormal wavelet thresholding, IEEE Transactions on Image Processing 16 (2007) 593–606.
[19] A. Buades, B. Coll, J. Morel, A review of image denoising algorithms with a new one, Multiscale Modelling Simulation 4 (2005) 490–530.
[20] T.-F. Chan, H.-M. Zhou, Adaptive ENO-wavelet transforms for discontinuous functions, in: 12th International Conference on Domain Decomposition Methods, 2001, pp. 93–100.
[21] T. Zeng, X. Li, M. Ng, Alternating minimization method for total variation based wavelet shrinkage model, Communications in Computational Physics 8 (5) (2010) 976–994.
[22] M. Elad, M. Aharon, Image denoising via sparse and redundant representations over learned dictionaries, IEEE Transactions on Image Processing 15 (2006) 3736–3745.
[23] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3D transform-domain collaborative filtering, IEEE Transactions on Image Processing 16 (2007) 3736–3745.
[24] V. Katkovnik, A. Foi, K. Egiazarian, J. Astola, From local kernel to nonlocal multiple-model image denoising, International Journal of Computer Vision 86 (2010) 1–32.
[25] J. Astola, P. Kuosmanen, Fundamentals of Nonlinear Digital Filtering, CRC, Boca Raton, United States of America, 1997.
[26] H. Hwang, R. Haddad, Adaptive median filters: new algorithms and results, IEEE Transactions on Image Processing 4 (1995) 499–502.
[27] T. Chen, H.R. Wu, Space variant median filters for the restoration of the impulse noise corrupted images, IEEE Transactions on Circuits and Systems 48 (2001) 784–789.
[28] G. Pok, J.-C. Liu, A.S. Nair, Selective removal of impulse noise based on homogeneity level information, IEEE Transactions on Image Processing 12 (2003) 85–92.
[29] H.-L. Eng, K.-K. Ma, Noise adaptive soft-switching median filter, IEEE Transactions on Image Processing 10 (2001) 242–251.
[30] S.-S. Wang, C.-H. Wu, A new impulse detection and filtering method for removal of wide range impulse noises, Pattern Recognition 42 (2009) 2194–2202.
[31] C. Bouman, K. Sauer, On discontinuity-adaptive smoothness priors in computer vision, IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (1995) 579–586.
[32] T.-F. Chan, H.-M. Zhou, R.-H. Chan, Continuation method for total variation denoising problems, Proceedings of SPIE Symposium on Advanced Signal Processing, vol. 2563, 1995, pp. 314–325.
[33] P. Charbonnier, L. Blanc-Féraud, G. Aubert, M. Barlaud, Deterministic edge-preserving regularization in computed imaging, IEEE Transactions on Image Processing 6 (1997) 298–311.
[34] C. Vogel, M. Oman, Fast, robust total variation-based reconstruction of noisy, blurred images, IEEE Transactions on Image Processing 7 (1998) 813–824.
[35] M. Nikolova, A variational approach to remove outliers and impulse noise, Journal of Mathematical Imaging and Vision 20 (2004) 99–120.
[36] R. Chan, C. Ho, M. Nikolova, Salt-and-pepper noise removal by median-type noise detector and edge-preserving regularization, IEEE Transactions on Image Processing 14 (2005) 1479–1485.
[37] R. Chan, C. Hu, M. Nikolova, An iterative procedure for removing random-valued impulse noise, IEEE Transactions on Signal Processing Letter 11 (2004) 921–924.
[38] S. Chen, X. Yang, G. Cao, Impulse noise suppression with an augmentation of ordered difference noise detector and an adaptive variational method, Pattern Recognition Letters 30 (2009) 460–467.
[39] S.-J. Ko, Y.-H. Lee, Center weighted median filters and their applications to image enhancement, IEEE Transaction on Circuits and Systems 38 (1991) 984–993.
[40] M. Aharon, M. Elad, A. Bruckstein, The K-SVD: an algorithm for designing of overcomplete dictionaries for sparse representation, IEEE Transactions on Image Processing 54 (2006) 4311–4322.
[41] J. Mairal, M. Elad, G. Sapiro, Sparse representation for color image restoration, IEEE Transactions on Image Processing 17 (2008) 53–69.
[42] J. Mairal, G. Sapiro, M. Elad, Learning multiscale sparse representations for image and video restoration, SIAM Multiscale Modeling and Simulation 7 (2008) 214–241.
[43] F. Malgouyres, T. Zeng, A predual proximal point algorithm solving a non negative basis pursuit denoising model, International Journal of Computer Vision 83 (3) (2009) 294–311.
[44] Y.-C. Pati, R. Rezaiifar, P.-S. Krishnaprasad, Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition, Proceedings of the 27th Asilomar Conference on Signals Systems and Computers, vol. 1, 1993, pp. 40–44.
[45] E.-T. Hale, W. Yin, Y. Zhang, Fixed-point continuation for l1-minimization: methodology and convergence, SIAM Journal on Optimization 19 (3) (2008) 1107–1130.

**Yu Xiao** received the B.S. degree in School of Computer Science and Engineering Hebei University of Technology, Tianjin, China, in 2006. She is currently pursuing the Ph.D. degree at the School of Computer and Information Technology Beijing Jiaotong University, Beijing, China. Her current research interests include image processing, machine learning, and data mining.

**Tieyong Zeng** received his B.Sc. degree at Peking University in 2000 and M.Sc. degree at the Ecole Polytechnique in 2004. Awarded the scholarship BDI of CNRS, he obtained his Ph.D. degree with highest honors at University of Paris XIII in 2007. Before joining Hong Kong Baptist University (HKBU) as an Assistant Professor, he worked as a Postdoctoral researcher in the group of Prof. Robert Azencott at the ENS de Cachan, France. His current interests are Image Processing and Statistical Learning.

**Jian Yu** received the B.S. degree in applied mathematics, M.S. degree in mathematics, and Ph.D. degree in applied mathematics from Peking University, Beijing, China, in 1991, 1994, and 2000, respectively. During 1994–1998, he joined the faculty of Beijing Graduate School, China University of Mining and Technology. Currently, he is Professor and Head of Institute of Computer Science Beijing Jiaotong University (previously named Northern Jiaotong University). His current research interests include fuzzy clustering, pattern recognition, and data mining.

**Micheal K. Ng** obtained his B.Sc. degree in 1990 and M.Phil. degree in 1992 at the University of Hong Kong, and Ph.D. degree in 1995 at Chinese University of Hong Kong. He was a Research Fellow of Computer Sciences Laboratory at Australian National University (1995–1997), and an Assistant/Associate Professor (1997–2005) of the University of Hong Kong before joining Hong Kong Baptist University. Michael won the Honourable Mention of Householder Award IX, in 1996 at Switzerland, an excellent young researcher's presentation at Nanjing International Conference on Optimization and Numerical Algebra, 1999, and the Outstanding Young Researcher Award of the University of Hong Kong, 2001. Michael and his collaborators won the outstanding research paper on Artificial Intelligence in the 2007 World Congress in Computer Science, Computer Engineering and Applied Computing. As an applied mathematician, Michael's main research areas include Bioinformatics, Data Mining, Operations Research and Scientific Computing. Michael has published and edited five books, published more than 160 journal papers. He currently serves on the editorial boards of several international journals.