Multimedia Tools and Applications (2024) 83:43685–43703
https://doi.org/10.1007/s11042-023-17352-z

Two efficient three-term conjugate gradient methods for impulse noise removal from medical images

Ali Mousavi[1] · Mansour Esmaeilpour[2] · Amir Sheikhahmadi[1]

Received: 27 August 2021 / Revised: 6 March 2022 / Accepted: 29 September 2023 /
Published online: 23 October 2023

Abstract
In this paper, we discuss two efficient three-term conjugate gradient methods (**ECG**) for impulse noise removal. The directions of **ECG** are first the direction of steepest descent and then spanned by the three terms: The steepest descent direction, the previous direction, and the gradient differences at the previous and current points. The second and third terms are scaled by two different step sizes called conjugate gradient parameters. Our goal is to generate and control these parameters such that they do not jointly dominate while preserving the effect of all terms, except near the optimizer where the first term dominates the other two terms. They are independent of the line search method and useful for finite precision arithmetic. The global convergence of **ECG** is proved. The efficiency (the lowest relative cost of function evaluations) and robustness (highest number of solved problems) of **ECG** compared to known conjugate gradient methods are shown in terms of PSNR (peak signal noise ratio) and time in seconds.

Keywords Image processing · Impulse noise removal · Unconstrained optimization · Conjugate gradient method · Wolfe line search method

Mathematics Subject Classification 90C30 · 90C25 · 65K05

1 Introduction

Image denoising is a fundamental problem in image processing that occurs in the acquisition, transmission, storage, and processing of images. Image denoising is about removing noise from a noisy image and restoring the true image. In denoising, it is not easy to detect edges and textures that have high-frequency components due to noise, so some details may be lost. Therefore, it is important to develop new algorithms that can recover meaningful information

✉ Mansour Esmaeilpour
esmaeilpour@iauh.ac.ir

[1] Department of Computer Engineering, Sanandaj Branch, Islamic Azad University, Sanandaj, Iran

[2] Computer Engineering Department, Hamedan Branch, Islamic Azad University, Hamedan, Iran

from noisy images and produce high-quality images. The development of two-phase algorithms whose first phase is denoising may be more effective than denoising methods, e.g., see [6–9, 21, 22]. The second phases of these algorithms can be any optimization methods with small memory requirements, such as conjugate gradient methods and limited memory quasi-Newton methods. These optimization methods can recover the images found by denoising. Therefore, two-phase methods are developed as a version of denoising. There are different types of noise such as Gaussian, Poisson, Rician, and salt and pepper impulse noise in medical images [1, 6, 7]. In medicine, many methods are used to remove impulse noise, such as X-ray, computed tomography, magnetic resonance imaging, ultrasound, and dermoscopy.

2 Related work

There are many methods for removing salt and pepper impulse noise, such as the descent spectral conjugate gradient method of Yu et al. [34], nonmonotone adaptive gradient method of Yu et al. [35] with a low-complexity, generalized trimmed mean filter by Rytsar & Ivasenko [28], wavelet filter of Karthikeyan et al. [20] and Wang et al. [31], fuzzy algorithms of Anisha et al. [1] and Russo et al. [27], trust region method of Kimiaei & Rahpeymaii [21], conjugate gradient method of Kimiaei & Rostami [22] and Zhou et al. [37], and gradient method of Liu et al. [23]. For impulse noise removal methods, other recent references are Chen et al. [11], Halder & Choudhuri [17], Nadeem et al. [24], Shah et al. [29], and Wang et al. [32].
Among all methods, conjugate gradient methods (CG) are popular because they require little memory. CG methods try line search methods along CG directions to speed up reaching an optimal point. Line search methods find step sizes that force a reduction in objective function values. Directions of CG methods are a kind of subspace spanned by two terms (the steepest descent gradient at the current point and the previous direction) or three terms (the steepest descent gradient at the current point and the previous direction, the gradient differences at the old and current points). These terms are scaled by step sizes called CG parameters. These parameters play a key role in the efficiency of CG methods. To ensure that CG methods find an optimal point, CG directions should satisfy the descent condition, i.e., the product of the gradient at the current point and the CG direction should be negative. If CG parameters become too large, the descent condition may be violated and a saddle point or a maximum point will be found instead of a minimum point. On the other hand, if these parameters become small, the corresponding term will be dominated by the other terms and the subspace will become smaller. Therefore, it is very important to generate and control CG parameters so that they do not become too large or too small, unless CG iterations are near an optimal point. In this case, the first term (the steepest descent direction) should dominate the other two terms. When CG iterations are near a minimum point, the gradient is small and the CG direction (which in this case is the steepest descent direction) has a small magnitude. In other words, the risk of skipping the minimum point due to large steps is reduced and CG methods have a high chance of finding the minimum point when noise is present. To achieve this, CG parameters should be slowly reduced by a decreasing factor. So far, there are no known studies on these points. Therefore, if it is possible to improve CG methods, this would be interesting and could significantly improve the efficiency and robustness of CG methods. Then such CG methods can be used in the second phase of two-phase methods (e.g., [6–9, 21, 22]).

2.1 Variant Median filter methods

Shukla et al. [30] proposed the median filter (MF) method to remove salt and pepper noise by using the median of the window to replace the central pixels with the window. If the central pixels are salt and pepper, they are replaced by the median of the window. MF cannot distinguish true pixels from noise.
The adaptive median filter (AMF) method is a widely used adaptation of MF (cf. [19]). AMF performs spatial processing to determine whether or not pixels in an image are affected by impulse noise. In addition, this method classifies pixels as noise by comparing each pixel in the image with the surrounding neighboring pixels. With AMF, the size of the neighborhood and the threshold for the comparison are adjustable. Impulse noise is a pixel that is different from most of its neighbors but is not matched with similar pixels. In this case, this noisy pixel is replaced by the neighboring pixels.

Notation Let $x \in \mathbb{R}^{M\times N}$ be the original image with $M \times N$ pixels. Then we consider the set

$$\mathcal{A} = \Big\{1, 2, ..., M\Big\} \times \Big\{1, 2, ..., N\Big\}$$

of indices. We denote by $x_{i,j}$ the component of the original image at position (i, j). We form the four neighborhoods

$$\mathcal{V}_{i,j} = \Big\{x_{i-1,j}, x_{i,j-1}, x_{i,j+1}, x_{i+1,j}\Big\}$$

of $x_{i,j}$. Let $[d_{\min}, d_{\max}]$ be the dynamic range of the original image x, i.e., for all $i = 1, 2, \cdots, M$ and $j = 1, 2, \cdots, N$, $x_{i,j} \in [d_{\min}, d_{\max}]$. Then we define

$$y_{i,j} = \begin{cases} x_{i,j} & \text{with probability } 1 - p - q, \\ d_{\min} & \text{with probability } p, \\ d_{\max} & \text{with probability } q, \end{cases}$$

since the noisy image contains salt and pepper noise. Applying AMF to the noisy image y, we obtain the repaired image $\tilde{y}$ and the candidate set

$$\mathcal{N} = \Big\{(i, j) \in \mathcal{A} \mid \;\; \tilde{y}_{i,j} \neq y_{i,j}, \;\; y_{i,j} = d_{\min} \text{ or } d_{\max}\Big\}$$

of noise.

2.2 Two-phase methods

In this section, two-phase methods are described. In the first phase, damaged noises are identified. Then, in the second phase, damage noises are recovered by solving an unconstrained optimization problem with a continuously differentiable objective function (e.g., see [6–9, 21, 22]).
Let us consider now how to work two-phase methods. In first phase, noisy image pixels are identified. If $(i, j) \notin \mathcal{N}$, then the pixel $x_{i,j}$ is not damaged and is kept without any change. But for $(i, j) \in \mathcal{N}$ the pixel $y_{i,j}$ should be denoise. For this, let u^* be a denoised image. As in [6, 8, 9], in the second phase, if $(m, n) \in \mathcal{V}_{i,j} \setminus \mathcal{N}$, we set $u^*_{m,n} = y_{m,n}$ and

for $(m,n) \in \mathcal{V}_{i,j} \cap \mathcal{N}$ we obtain $y_{m,n}$ by solving the following problem with the smooth objective function

$$\min_{u \in \mathbb{R}^{|\mathcal{N}|}} \mathbf{F}(u) := \sum_{(i,j)\in\mathcal{N}} \Big\{ |u_{i,j} - y_{i,j}| + \eta \Big[\sum_{(m,n)\in\mathcal{V}_{i,j}\setminus\mathcal{N}} \phi_\alpha(u_{i,j} - y_{m,n}) + \frac{1}{2} \sum_{(m,n)\in\mathcal{V}_{i,j}\cap\mathcal{N}} \phi_\alpha(u_{i,j} - u_{m,n}) \Big] \Big\}. \tag{1}$$

Here η is a regularization parameter and ϕ_α is an even edge-preserving potential function. Some of the most popular edge-preserving potential functions [4, 5, 7, 10, 15] are

$$\phi_\alpha(t) = \sqrt{t^2 + \alpha}, \quad \alpha > 0, \tag{2}$$

$$\phi_\alpha(t) = |t|^\alpha, \quad 1 < \alpha \leq 2, \tag{3}$$

$$\phi_\alpha(t) = \log\Big(\cosh(\alpha t)\Big), \quad \alpha > 0, \tag{4}$$

$$\phi_\alpha(t) = \frac{|t|}{\alpha} - \log\Big(1 + \frac{|t|}{\alpha}\Big), \quad \alpha > 0, \tag{5}$$

$$\phi_\alpha(t) = \begin{cases} \dfrac{t^2}{2\alpha}, & |t| \leq \alpha, \\ |t| - \dfrac{\alpha}{2} & |t| > \alpha, \end{cases} \quad \alpha > 0. \tag{6}$$

There are many ways to solve the problem (1), such as Newton method, quasi-Newton methods, trust region methods, etc. Since this problem is a large problem, only low-memory methods such as CG methods (e.g., see [6–9, 22]) and limited memory quasi-Newton methods (cf. [21]) can be used to solve these problems.

2.3 Known CG methods

To solve the problem (1), starting from an initial guess $u_0 \in \mathbb{R}^{|\mathcal{N}|}$, CG methods generate a sequence $\{u_k\}_{k\geq 0}$ by

$$u_{k+1} := u_k + \alpha_k d_k. \tag{7}$$

Here α_k is a step size, obtained by inexact line search methods, and d_k is a descent search direction satisfying the descent condition

$$\nabla \mathbf{F}_k^T d_k < 0, \tag{8}$$

computed by

$$d_k := \begin{cases} -\nabla \mathbf{F}_k, & k = 0, \\ -\nabla \mathbf{F}_k + \beta_k d_{k-1}, & k \geq 1. \end{cases} \tag{9}$$

Here $\nabla \mathbf{F}_k := \nabla \mathbf{F}(u_k)$ is the gradient of $\mathbf{F}$ at u_k and $\beta_k \in \mathbb{R}$ is the CG parameter with known choices

$$\beta_k^{HS} := \frac{\nabla \mathbf{F}_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, \qquad \text{Hestenes \& Stiefel [18]} \tag{10}$$

$$\beta_k^{FR} := \frac{\|\nabla \mathbf{F}_k\|^2}{\|\nabla \mathbf{F}_{k-1}\|^2}, \qquad \text{Fletcher \& Reeves [14]} \tag{11}$$

$$\beta_k^{DY} := \frac{\|\nabla \mathbf{F}_k\|^2}{d_{k-1}^T y_{k-1}}, \qquad \text{Dai \& Yuan [12]} \tag{12}$$

$$\beta_k^{HZ} := \beta_k^{HS} - 2\|y_{k-1}\|^2 \frac{\nabla \mathbf{F}_k^T d_{k-1}}{(d_{k-1}^T y_{k-1})^2}, \qquad \text{Hager \& Zhang [16]} \tag{13}$$

$$\beta_k^{PR} := \frac{\nabla \mathbf{F}_k^T y_{k-1}}{\|\nabla \mathbf{F}_{k-1}\|^2}, \qquad \text{Polak \& Ribiere [26]} \tag{14}$$

where $y_{k-1} := \nabla \mathbf{F}_k - \nabla \mathbf{F}_{k-1}$ and $\|\cdot\|$ denotes the Euclidean norm.
A generic version of three-term CG (TTCG) methods was proposed by Beale [3] to solve unconstrained optimization problems whose search direction

$$d_k := -\nabla \mathbf{F}_k + \beta_k d_{k-1} + \gamma_k d_t \tag{15}$$

is calculated. Here $\beta_k = \beta_k^{FR}, \beta_k^{HS}, \beta_k^{DY}$, d_t is a restarted direction, a direction with a restart procedure, and

$$\gamma_k := \begin{cases} 0, & k = t+1, \\ \dfrac{\nabla \mathbf{F}_k^T y_t}{d_t^T y_t}, & k > t+1. \end{cases}$$

In fact, the goal of TTCG methods is to improve two-term CG methods. As in [2], a comparison among TTCG methods is made for solving unconstrained optimization problems, showing that TTCG methods are numerically efficient and robust. To obtain the step size α_k, we use the strong Wolfe line search [33]

$$\mathbf{F}(u_k + \alpha_k d_k) - \mathbf{F}(u_k) \le c_1 \alpha_k \nabla \mathbf{F}_k^T d_k, \tag{16}$$

$$\left| \nabla \mathbf{F}_{k+1}^T d_k \right| \le -c_2 \nabla \mathbf{F}_k^T d_k, \tag{17}$$

where $0 < c_1 < c_2 < 1$. The inequality (16) is called Armijo which tries to get a decrease in the function value. The inequality (17) is called curvature which lies a step size in the broad neighborhood of a local minimum point of $\mathbf{F}$ with respect to α.

As discussed in Section 2, CG methods have no plan to control CG parameters. For example, if β_k in (15) is close to zero before CG iterations are near a minimum point, then d_k will be the steepest descent $-\nabla \mathbf{F}_k$, which has a zigzagging behavior, one of the sources of inefficiency of optimization algorithms. However, if CG iterations are close to a minimum point, the steepest descent direction $-\nabla \mathbf{F}_k$ can be used effectively. In (17), if one of β_k or γ_k or both are near zero before CG iterations are near an optimal point, the subspace is reduced and d_k cannot take advantage of d_{k-1} and d_t. On the other hand, if at least one β_k or γ_k is large, then the descent direction (8) can be violated. In this case, the Wolfe line search method cannot be used the descent condition (8) does not hold.

3 Our contribution and organization

In this work, we develop a two-phase methods to remove noise. First, AMF is used to detect damaged pixels. Second, two efficient TTCG methods are proposed based on the steepest descent direction, the old direction, and the gradient differences at the old and current points.

The new phase of our algorithm is indeed the second phase. The first phase is not new and not very important for us, since it deals only with the detection of the damaged pixels. In practice, the second phase will be very efficient and will be able to recover the noisy pixels with high quality, regardless of how efficient the first phase is or how it can be updated by new techniques.

The parameters of our CG directions are generated and controlled in a new way:

- None of the terms of the subspace form is large, which does not violate the descent condition (8), which is a prerequisite for using the Wolfe line search method to speed up our CG methods.
- None of the terms of the subspace form dominates the other in general. In fact, our CG directions use their three terms when CG iterations are not near the optimal point.
- When CG iterations are near a minimum point, the first term (steepest descent direction) dominates the other two terms because CG iterations should have small gradients in this case and accordingly CG directions has a small size and the risk of significantly skipping the minimum point is reduced.

If the second and third terms dominate the first term in cases where CG iterations are near a minimum point, CG directions may be large, violating the descent condition (8) and causing saddle points or maximum points to be found.

The descent property and global convergence of the proposed algorithms are proved. Numerical results show that our method is competitive compared to the state-of-the-art CG methods. In fact, in our comparison, the first phases of all CG methods are the same; only the second phases use different CG directions along which the Wolfe line search method is tried.

Section 2 describes two efficient CG methods to remove salt and pepper noise. Section 3 gives numerical results to illustrate the efficiency and robustness of the new algorithms. Some conclusions are given in Section 4. Finally, an additional material is reported in Section 5.

4 New efficient CG directions

In this section, we construct two new efficient TTCG methods for impulse noise removal from medical images. Our new directions are spanned by the steepest descent direction $-\nabla \mathbf{F}_k$, the previous direction d_{k-1}, and the difference of the gradients y_{k-1} at the old point u_{k-1} and at the current point u_k. Moreover, CG parameters are used to improve the numerical results of the noise removal. In practice, each component of d_{k-1} and y_{k-1} is adjusted based on the corresponding component of g_k such that each component of the new direction is no larger than that of $\nabla \mathbf{F}_k$. Since in finite precision arithmetic the descent condition cannot be guaranteed numerically without such an adjustment, although this condition holds theoretically, the search direction is almost always orthogonal to the gradient. Hence this adjustment is essential to generate an efficient direction in finite precision arithmetic.

- The first efficient CG method (**ECG1**). We first define CG parameters based on our discussed adjustment as follows:

$$\beta_k^1 := \frac{\eta_1}{(1+k)^{\eta_2}} \max_{i=1,2,\dots,|\mathcal{N}|} \left\{ \left| \frac{(\nabla \mathbf{F}_k)_i}{(d_{k-1})_i} \right| \;\middle|\; (d_{k-1})_i \neq 0 \right\}, \tag{18}$$

$$\beta_k^2 := \frac{\eta_1}{(1+k)^{\eta_2}} \max_{i=1,2,\dots,|\mathcal{N}|} \left\{ \left| \frac{(\nabla \mathbf{F}_k)_i}{(y_{k-1})_i} \right| \;\middle|\; (y_{k-1})_i \neq 0 \right\}, \tag{19}$$

  in which $\eta_1, \eta_2 \in (0, 1)$ and k counts the number of iterations. Hence, the new iterate is updated by (7) whose direction d_k is computed by

$$d_k := \begin{cases} -\nabla \mathbf{F}_k + \beta_k^1 d_{k-1} + \beta_k^2 y_{k-1}, & \text{if } \nabla \mathbf{F}_k^T d_{k-1} \geq 0, \\ -\nabla \mathbf{F}_k, & \text{otherwise.} \end{cases} \tag{20}$$

Here the step size α_k satisfies (16)-(17). Whenever the directions generated by (20) are not descent, they are replaced by $d_k = -\nabla \mathbf{F}_k$.

- The second efficient CG method (**ECG2**). Before our new direction is introduced, four Boolean variables are defined as follows:

$$\texttt{Dec1} := \Big(\nabla \mathbf{F}_k^T d_{k-1} > 0 \text{ and } \nabla \mathbf{F}_k^T y_{k-1} < 0\Big),$$
$$\texttt{Dec2} := \Big(\nabla \mathbf{F}_k^T d_{k-1} < 0 \text{ and } \nabla \mathbf{F}_k^T y_{k-1} > 0\Big),$$
$$\texttt{Dec3} := \Big(\nabla \mathbf{F}_k^T d_{k-1} < 0 \text{ and } \nabla \mathbf{F}_k^T y_{k-1} < 0\Big),$$
$$\texttt{Dec4} := \Big(\nabla \mathbf{F}_k^T d_{k-1} > 0 \text{ and } \nabla \mathbf{F}_k^T y_{k-1} > 0\Big).$$

Therefore the new direction

$$d_k := \begin{cases} -\nabla \mathbf{F}_k - \beta_k^1 d_{k-1} + \beta_k^2 y_{k-1}, & \text{if } \texttt{Dec1} \text{ is true,} \\ -\nabla \mathbf{F}_k + \beta_k^1 d_{k-1} - \beta_k^2 y_{k-1}, & \text{if } \texttt{Dec2} \text{ is true,} \\ -\nabla \mathbf{F}_k + \beta_k^1 d_{k-1} + \beta_k^2 y_{k-1}, & \text{if } \texttt{Dec3} \text{ is true,} \\ -\nabla \mathbf{F}_k - \beta_k^1 d_{k-1} - \beta_k^2 y_{k-1}, & \text{if } \texttt{Dec4} \text{ is true,} \end{cases} \tag{21}$$

of **ECG2** is computed while satisfying the descent condition (8). The new iterate of **ECG2** is updated by (7) while computing the search direction by (21).

Our CG parameters (18) are slowly reduced by a factor $\dfrac{\eta_1}{(1+k)^{\eta_2}}$, so that when CG iterations are near an optimal point (k becomes large), the steepest descent direction $-\nabla \mathbf{F}_k$ is used, whose size is small in such a case. On the other hand, two terms

$$\max_{i=1,2,\ldots,|\mathcal{N}|} \left\{ \left| \frac{(\nabla \mathbf{F}_k)_i}{(d_{k-1})_i} \right| \;\middle|\; (d_{k-1})_i \neq 0 \right\}$$

of (18) and

$$\max_{i=1,2,\ldots,|\mathcal{N}|} \left\{ \left| \frac{(\nabla \mathbf{F}_k)_i}{(y_{k-1})_i} \right| \;\middle|\; (y_{k-1})_i \neq 0 \right\}$$

of (19) do not allow β_k^1 and β_k^2 to quickly become too small. Indeed, these two terms allow none of $-\nabla \mathbf{F}_k$, d_{k-1}, and y_{k-1} to dominate until CG iterations are not near an optimal point.

5 Global convergence

To investigate the descent property and the global convergence of **ECG1** and **ECG2**, we assume the following standard assumptions:

Assumption 3.1 The level set $L(u_0) := \left\{ u \in \mathbb{R}^{\mathcal{N}} \;\middle|\; \mathbf{F}(u) \leq \mathbf{F}(u_0) \right\}$ is bounded, i.e., there exists a constant $M > 0$ such that

$$\|u\| \leq M, \qquad \text{for all } u \in L(u_0). \tag{22}$$

Assumption 3.2 In some neighbourhood $\Omega \subseteq L(u_0)$, the gradient of the objective function $\mathbf{F}$ has Lipschitz continuous gradient, i.e., there exists a constant $L > 0$ such that

$$\|\nabla \mathbf{F}(u) - \nabla \mathbf{F}(v)\| \leq L\|u - v\|, \qquad \text{for all } u, v \in \Omega. \tag{23}$$

Assumptions 3.1 and 3.2 imply that there exists a constant $\epsilon > 0$ such that

$$\|\nabla \mathbf{F}_k\| \leq \epsilon, \qquad \text{for all } k \geq 0.$$

The following lemma ensures that the descent property of our algorithms is satisfied, resulting in the angle between the gradient and the search direction is far from 90°.

Lemma 1 *Let d_k be the direction generated by* ***ECG1*** *or* ***ECG2****. Then, d_k is a descent direction, i.e., the condition* (8) *holds.*

The following lemma is called Zoutendijk condition which is used to prove the global convergence of CG methods.

Lemma 2 *Let d_k be a descent direction and the step size α_k satisfies the strong Wolfe line search* (16)-(17)*. Then, under Assumptions 3.1 and 3.2, the condition*

$$\sum_{k=0}^{+\infty} \frac{(\nabla \boldsymbol{F}_k^T d_k)^2}{\|d_k\|^2} < +\infty \tag{24}$$

holds.

Proof See [38].

The following lemma [36] has a key role in the proof of the next theorem below.

Lemma 3 *Let d_k be the direction generated by* ***ECG1*** *or* ***ECG2*** *and Assumptions 3.1 and 3.2 hold. If*

$$\sum_{k=1}^{\infty} \frac{1}{\|d_k\|^2} = +\infty, \tag{25}$$

then

$$\liminf_{k \to \infty} \|\nabla \boldsymbol{F}_k\| = 0.$$

The following theorem is the main global convergence of our new algorithms whose proof is discussed in Section 8.2.

Theorem 1 *Let d_k be the descent direction and $\{u_k\}_{k \geq 0}$ be the sequence generated by* ***ECG1*** *or* ***ECG2****. Then*

$$\liminf_{k \to \infty} \|\nabla \boldsymbol{F}_k\| = 0. \tag{26}$$

6 Numerical results

We compare two versions of our methods **ECG1** and **ECG2** with two-phase methods, whose first phases are AMF and whose second phases are known CG methods, on several standard images. AMF detects damaged pixels and restores them to some extent and then CG methods (CG directions along which Wolfe line search method is used) uniquely restores the image restored by AMF. As mentioned earlier, any denoising method can be used in the first phase. In fact, these two-phase methods are an evolution of denoising and their efficiency comes from the second phase.

Code compared. The following known CG algorithms are used in our comparison:
CGFR: This algorithm uses $d_k := -\nabla \mathbf{F}_k + \beta_k^{FR} d_{k-1}$.
CGHS: This algorithm uses $d_k := -\nabla \mathbf{F}_k + \beta_k^{HS} d_{k-1}$.
CGPR: This algorithm uses $d_k := -\nabla \mathbf{F}_k + \beta_k^{PR} d_{k-1}$.
CGDY: This algorithm uses $d_k := -\nabla \mathbf{F}_k + \beta_k^{DY} d_{k-1}$.
CGHZ: This algorithm uses $d_k := -\nabla \mathbf{F}_k + \beta_k^{HZ} d_{k-1}$.

NPR1: This algorithm uses $d_k := -\nabla \mathbf{F}_k + \beta_k^{NPR1} d_{k-1}$, $\beta_k^{NPR1} = \dfrac{\beta_k^{PR}}{y_{k-1}^T d_{k-1} + \|y_{k-1}\| \|d_{k-1}\|}$.

NPR2: This algorithm uses $d_k := -\nabla \mathbf{F}_k + \beta_k^{NPR2} d_{k-1}$, $\beta_k^{NPR2} = \dfrac{\beta_k^{PR}}{\nabla \mathbf{F}_k^T d_{k-1} + \|\nabla \mathbf{F}_k\| \|d_{k-1}\|}$.

NPR3: This algorithm uses $d_k := -\nabla \mathbf{F}_k + \beta_k^{NPR3} d_{k-1}$, $\beta_k^{NPR3} = \dfrac{\beta_k^{PR}}{\nabla \mathbf{F}_k^T \nabla \mathbf{F}_{k-1} + \|\nabla \mathbf{F}_k\| \|\nabla \mathbf{F}_{k-1}\|}$.

NPR4: This algorithm uses $d_k := -\dfrac{\|y_{k-1}\|^2}{\|s_{k-1}\|^2} \nabla \mathbf{F}_k + \beta_k^{NPR4} d_{k-1}$, and

$$\beta_k^{NPR4} := \begin{cases} \dfrac{\nabla \mathbf{F}_k^T y_{k-1}}{\|\nabla \mathbf{F}_{k-1}\|^2}, & \text{if } \nabla \mathbf{F}_k^T y_{k-1} < 0, \\ -\dfrac{\|\nabla \mathbf{F}_k\|^2}{\|\nabla \mathbf{F}_{k-1}\|^2}, & \text{if } \nabla \mathbf{F}_k^T d_{k-1} > 0, \\ 0, & \text{otherwise.} \end{cases}$$

Test images. Five standard images are used:

- Lena has been widely used in image processing since 1973. It is an image of Swedish model Lena Forsé cropped from the centerfold of the November 1972 issue of Playboy magazine. This image has a fixed resolution of 100 lines per inch and 256 × 256 pixels.
- House's test image is in many versions due to cropping, scanning, resizing, compression, or conversion from color to grayscale. Here we use the version with 256 × 256 pixels.
- Cameraman image is one of the most popular standard grayscale test images of size 256 × 256 belonging to MIT. By using standard test images, it is possible to test different image processing and compression algorithms and compare the results both visually and quantitatively. This image presents a number of challenges to image algorithms, such as image enhancements, image compression, etc. The image has a dynamic range of pixels between 0 and 255 (8- bits). The minimum value is 7 and the maximum value is 253. Here we use both versions of 256 × 256 and 512 × 512 pixels.
- Computed tomography of the head (CT) uses X-rays to produce cross-sectional images of the body. This is possible because different tissues interact with X-rays in different ways. Some tissues allow X-rays to pass through without affecting them much, while other tissues exert a stronger effect. This image has 512 × 512 pixels.
- The brain sagittal (CerebSagE) MRI study looks at the brain with 24 sagittal (vertical - front to back) slices starting on the right side of the brain and moving to the left. This image consists of 512 × 512 pixels.

Tuning parameters. ECG1 and **ECG2** use the tuning parameters $\eta_1 = 0.1$ and $\eta_2 = 0.85$ while all algorithms in the Wolfe line search use the tuning parameters $c_1 = 10^{-4}$ and $c_2 = 0.5$.

Stopping tests. All algorithms are stopped whenever $\|\nabla \mathbf{F}_k\| \leq 10^{-4}$.

Performance profile. A popular tool for identifying the efficiency of algorithms is the performance profile of DOLAN & MORÉ [13]. This profile uses two cost measures: time in seconds and the peak signal noise ratio

$$PSNR = 10 \log_{10} \frac{255^2}{\frac{1}{M \times N} \sum_{i,j} (u^r_{i,j} - u^*_{i,j})^2}.$$

Here $M \times N$ is the size of the image, $u^r_{i,j}$ is the noisy image, $u^*_{i,j}$ is restored the image and

$$\overline{u} = \frac{1}{M \times N} \sum_{i,j} (u^*_{i,j})^2,$$

is the mean of the corrupted image. Denote by $\mathcal{S}$ the list of compared solvers, by $\mathcal{P}$ the list of problems, and by $c_{p,s}$ the cost measure of the solver s to solve the problem p. The performance profile of the solver $s \in S$

$$\rho_s(\tau) := \frac{1}{|\mathcal{P}|} \left| \left\{ p \in \mathcal{P} \;\middle|\; pr_{p,s} := \frac{c_{p,s}}{\min(c_{p,s_0} \mid s_0 \in S)} \leq \tau \right\} \right|.$$

is the fraction of problems that the performance ratio $pr_{p,s}$ is at most τ. In particular, the fraction of problems that the solver s wins compared to the other solvers is $\rho_s(1)$ and the fraction of problems for sufficiently large τ that the solver s can solve is $\rho_s(\tau)$.

Our finding. We say that a method is competitive with others if it is significantly more efficient and robust than others in terms of cost measures: PNSR, time in seconds, relative cost of function and gradient evaluations, and the number of iterations. This can be done by plotting performance profiles of all methods with respect to these cost measures:

- First, we compare all known CG with respect to PNSR, as shown in Fig. 1. The result is that **NPR4** and **NPR1** are the first and second best methods.
- Then, we compare only **NPR1** and **NPR4** to find out which of the two methods has the lowest relative cost of function and gradient evaluations, the lowest iterations, and the lowest time in seconds (see Fig. 2), which confirms that **NPR4** is competitive compared to **NPR1**.
- Next, two versions of our methods **ECG1** and **ECG2** were compared (see Fig. 3). The result is that **ECG2** is competitive compared to **ECG1**.
- Finally, we compare our best method **ECG2** with the best method **NPR4** of the known CG methods shown in Fig. 4, with the result that **ECG2** is so competitive compared to **NPR4**. Figures 5, 6, 7, 8 and 9 show that, after AMF identifies damaged pixels, **ECG2** can significantly recover noisy images, even in the presence of high noise. After applying our CG methods, PSNR is significantly increased. Numerical results show that **ECG2** is competitive with other known methods in terms of time in seconds and PSNR.

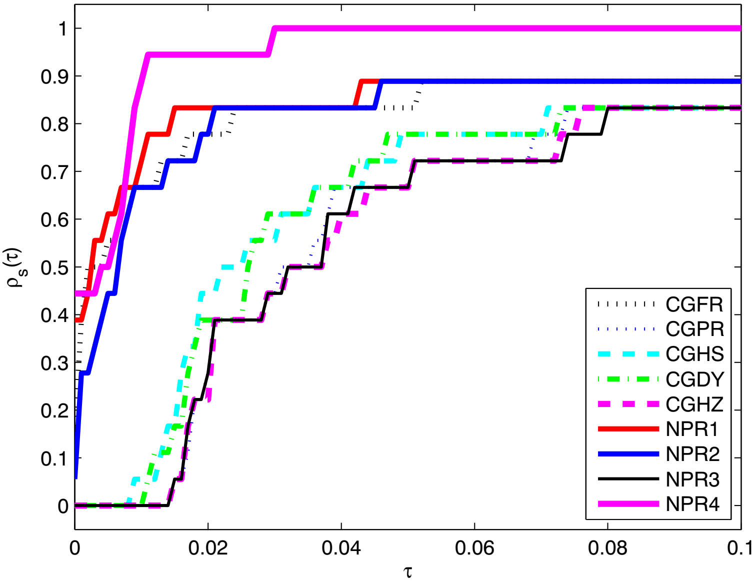
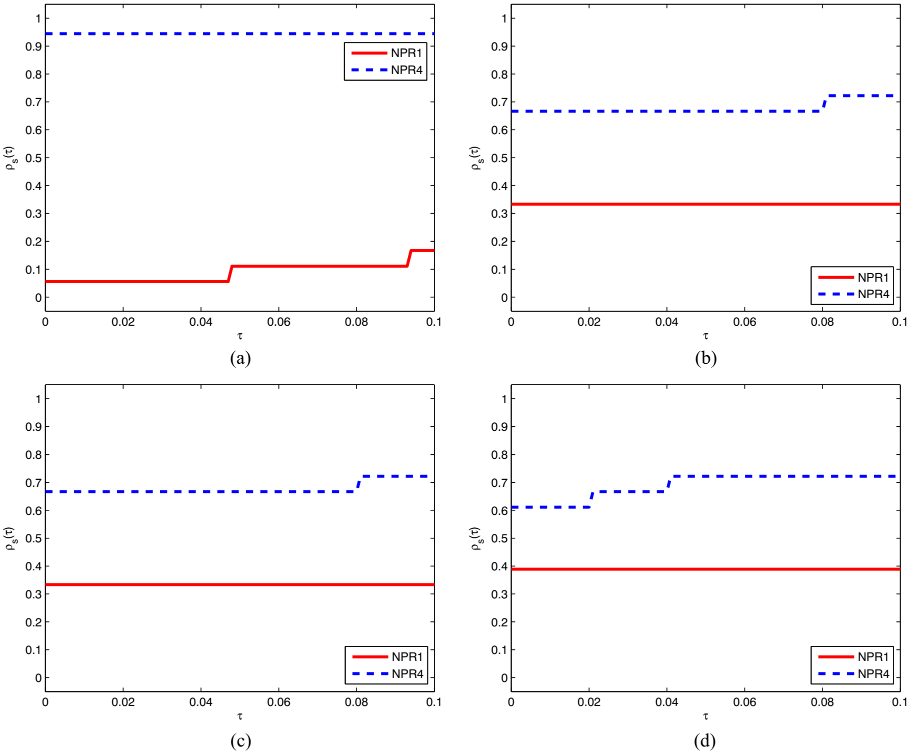


Fig. 1 Performance profiles of known CG methods in terms of PSNR



(a)

(b)

(c)

(d)

Fig. 2 Performance profiles of **NPR1** and **NPR4** in terms of: (a) the total number of iterations; (b) the total number of function evaluations; (c) the total number of gradient evaluations; (d) time in seconds

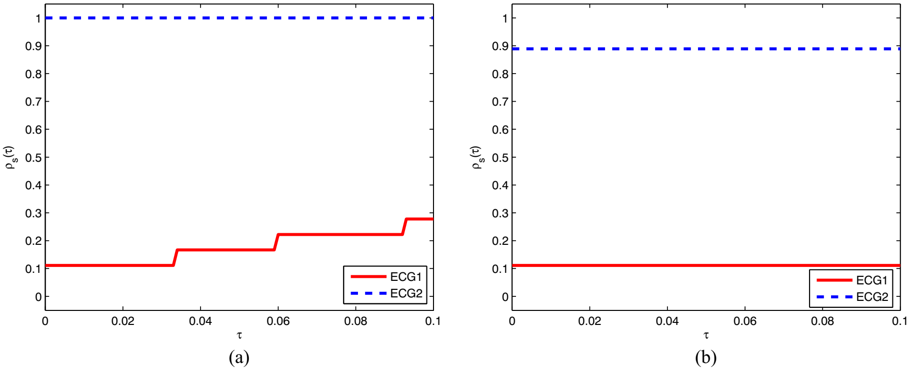


Fig. 3 (a) Performance profiles of **ECG1** and **ECG2** in terms of PSNR; (b) Performance profiles of **ECG1** and **ECG2** in terms of time in seconds

7 Conclusion

In this work, a two stages method for removing impulse noise, especially from medical images, was presented. The first stage used AMF to detect damaged pixels. Every new technique welcomes the use of the first stage, but this is not our case because this stage does not affect our method. The second stage included two new three-term CG directions, a kind of subspace spanned by the direction of steepest descent, the old direction and the gradient differences at the old and current points. The second and third terms of this subspace were scaled by two new CG parameters whose goal was to obtain three terms (no dominance among the three terms), guarantee the descent condition, and use the steepest descent directions when CG iterations were near an optimal point.

Our CG methods are independent of the type of line search method and useful in finite precision arithmetic. The global convergence of CG methods are proved. Several standard images were used to show that the new method is robust and efficient compared to other known CG methods for removing impulse noise.

Future work may involve constructing larger subspace techniques that can use other types of directions as a new component of the subspace. For example, the conjugate gradient

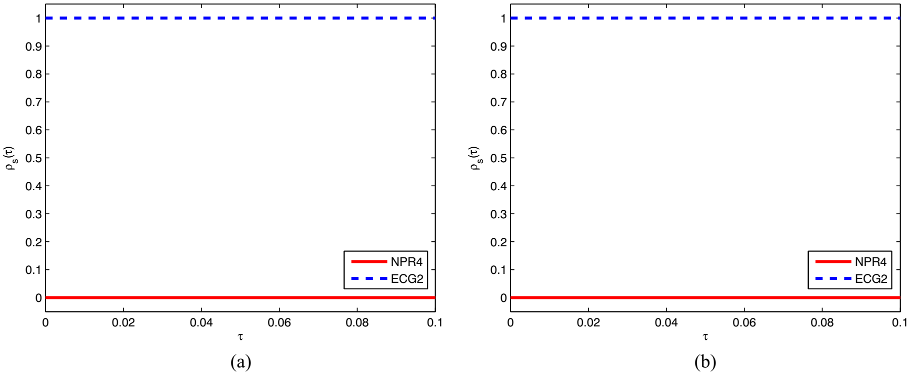
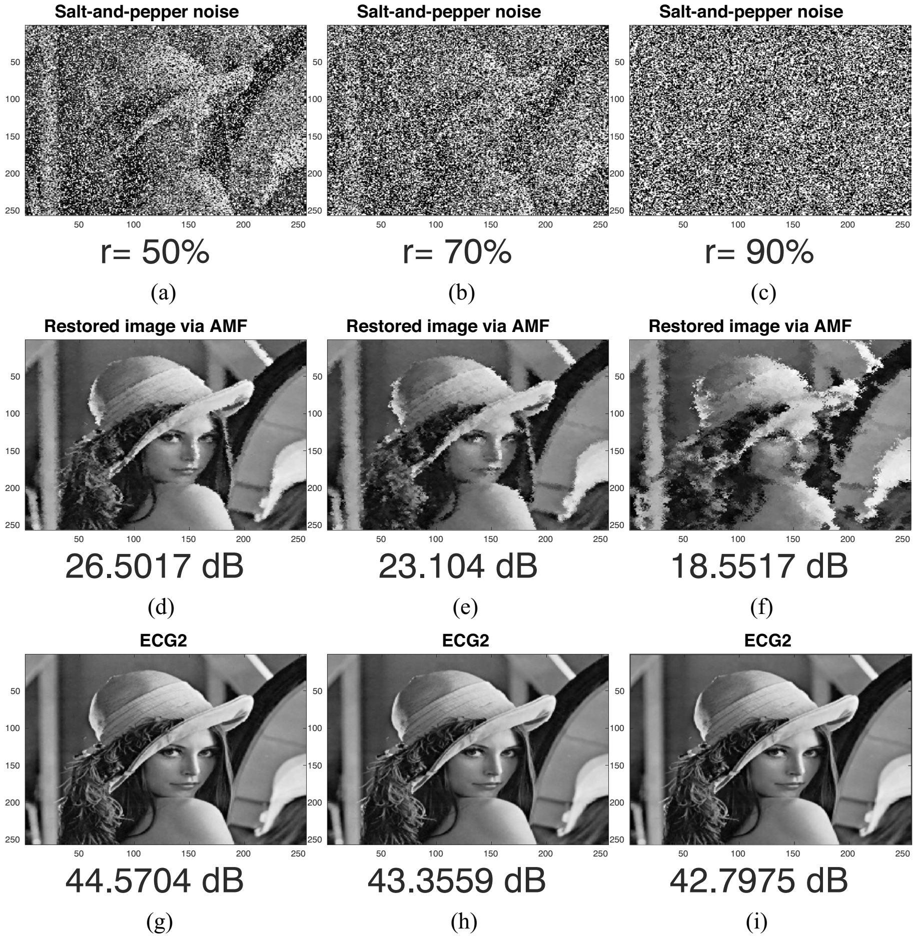


Fig. 4 (a) Performance profiles of **NPR4** and **ECG2** in terms of PSNR; (b) Performance profiles of **NPR4** and **ECG2** in terms of time in seconds



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 5 Recovering the 256 × 256 noisy Lena image via **ECG2**. (a)-(c) are the noisy images with 50%, 70% and 90% of noises, (d)-(f) are the restored images via adaptive median filter (AMF), (g)-(i) are the restored images via **ECG2**

direction can be enriched and combined with other directions such as the limited memory quasi-Newton method or other low-memory techniques.

8 Tools for ECG

8.1 The proof of Lemma 1

Let d_k be the direction by **ECG1**, computed by (20). If d_k does not satisfy the descent condition, then we have

$$\nabla \mathbf{F}_k^T d_k = -\|\nabla \mathbf{F}_k\|^2 < 0.$$

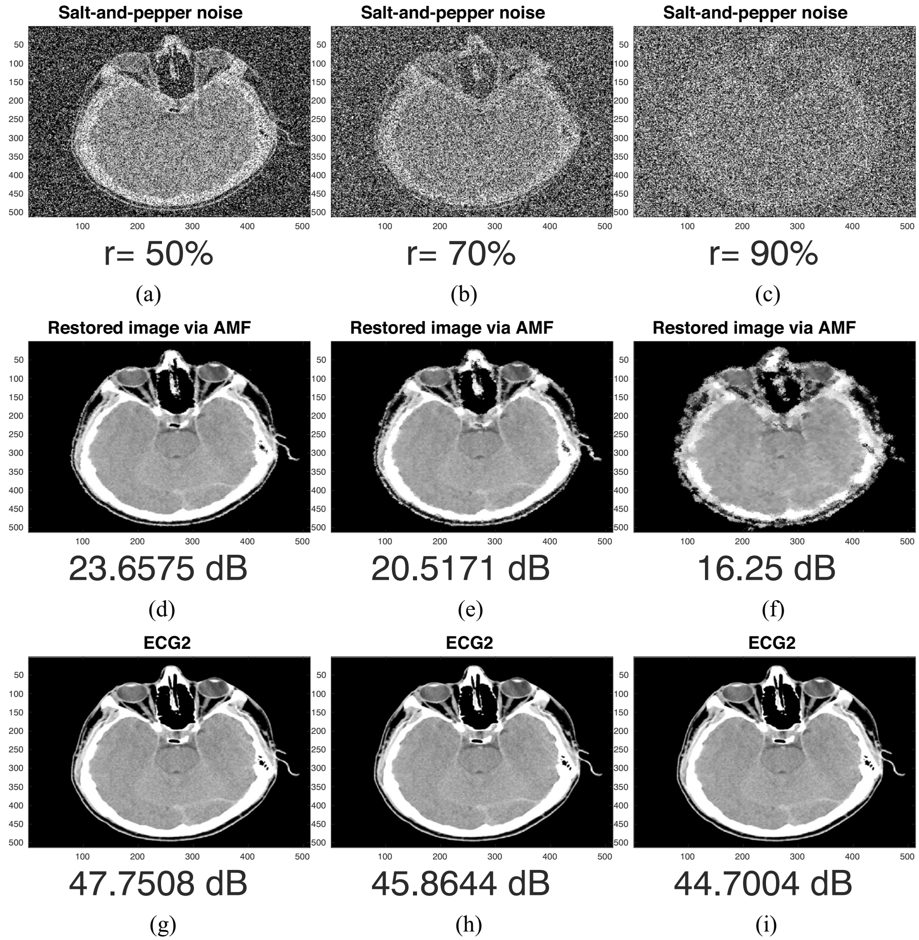


Fig. 6 Recovering the 512×512 noisy HeadCT image via **ECG2**. (a)-(c) are the noisy images with 50%, 70% and 90% of noises, (d)-(f) are the restored images via adaptive median filter (AMF), (g)-(i) are the restored images via **ECG2**

Suppose that d_k is generated by **ECG2**. We consider the following four cases.

CASE 1. `Dec1` results in that d_k is descent, i.e.,

$$\nabla \mathbf{F}_k^T d_k = -\|\nabla \mathbf{F}_k\|^2 - \beta_k^1 \nabla \mathbf{F}_k^T d_{k-1} + \beta_k^2 \nabla \mathbf{F}_k^T y_{k-1} < 0.$$

CASE 2. `Dec2` results in that d_k is descent, i.e.,

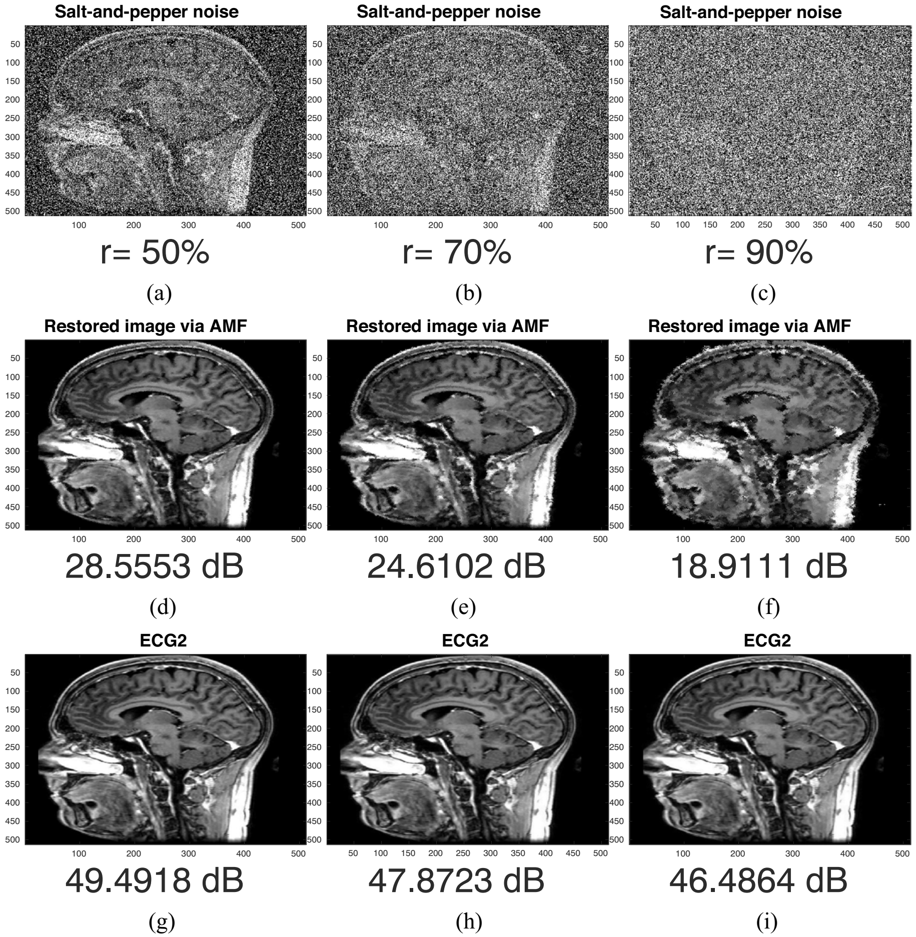
$$\nabla \mathbf{F}_k^T d_k = -\|\nabla \mathbf{F}_k\|^2 + \beta_k^1 \nabla \mathbf{F}_k^T d_{k-1} - \beta_k^2 \nabla \mathbf{F}_k^T y_{k-1} < 0.$$

CASE 3. `Dec3` results in that d_k is descent, i.e.,

$$\nabla \mathbf{F}_k^T d_k = -\|\nabla \mathbf{F}_k\|^2 + \beta_k^1 \nabla \mathbf{F}_k^T d_{k-1} + \beta_k^2 \nabla \mathbf{F}_k^T y_{k-1} < 0.$$

CASE 4. `Dec4` results in that d_k is descent, i.e.,

$$\nabla \mathbf{F}_k^T d_k = -\|\nabla \mathbf{F}_k\|^2 - \beta_k^1 \nabla \mathbf{F}_k^T d_{k-1} - \beta_k^2 \nabla \mathbf{F}_k^T y_{k-1} < 0.$$



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 7 Recovering the 512 × 512 noisy CerebSagE image via **ECG2**. (a)-(c) are the noisy images with 50%, 70% and 90% of noises, (d)-(f) are the restored images via adaptive median filter (AMF), (g)-(i) are the restored images via **ECG2**

As a result, the direction by **ECG2** is descent. □

8.2 The proof of Theorem 1

Let

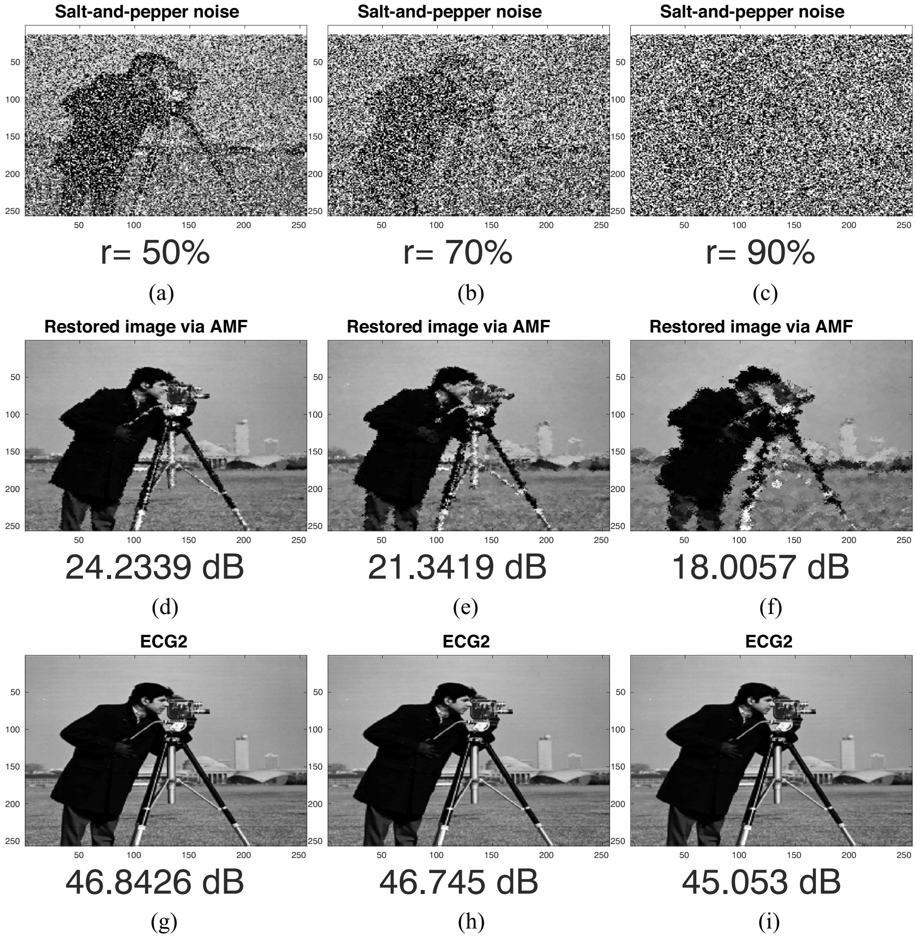
$$\gamma_k = \max\left\{|\beta_k^1|, |\beta_k^2|\right\} \leq M^*. \tag{27}$$

Then either (20) or (21) results in

$$\|d_k\| \leq \|\nabla \mathbf{F}_k\| + |\beta_k^1|\|d_{k-1}\| + |\beta_k^2|\|y_{k-1}\|. \tag{28}$$

Assumption 3.2 implies

$$\|y_{k-1}\| \leq \alpha_k\|d_{k-1}\|. \tag{29}$$



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 8 Recovering the 256 × 256 noisy Cameraman image via **ECG2**. (a)-(c) are the noisy images with 50%, 70% and 90% of noises, (d)-(f) are the restored images via adaptive median filter (AMF), (g)-(i) are the restored images via **ECG2**

Therefore, (27)-(29) lead to

$$\|d_k\| \leq \epsilon + M^*\|d_{k-1}\| + M^*L\alpha_k\|d_{k-1}\|$$
$$= \epsilon + M^*\Big(1 + \alpha_k L)\|d_{k-1}\|.$$

In the same way as the proof of Lemma 3.1 in [36], there exists a positive constant M such that

$$\|d_k\| \leq M, \qquad \text{for all } k \geq 0,$$

so that

$$\sum_{k=1}^{\infty} \frac{1}{\|d_k\|^2} \geq \sum_{k=1}^{\infty} \frac{1}{M^2} = +\infty.$$

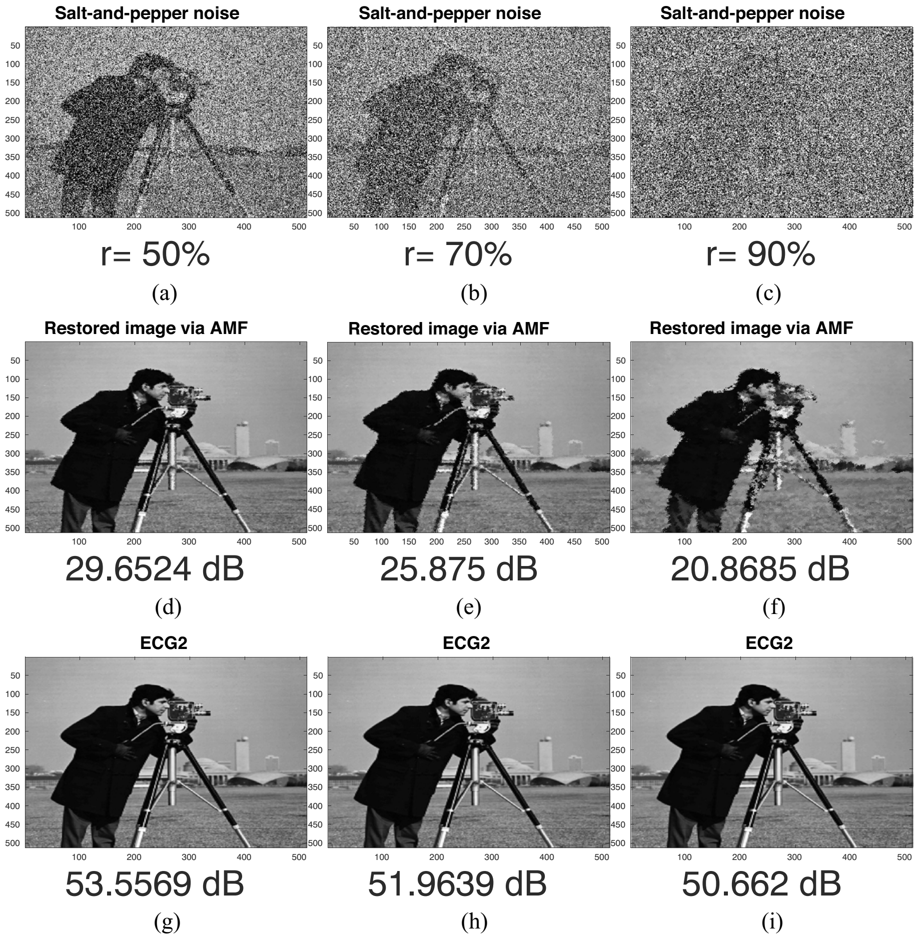


Fig. 9 Recovering the 512 × 512 noisy Cameraman image via **ECG2**. (a)-(c) are the noisy images with 50%, 70% and 90% of noises, (d)-(f) are the restored images via adaptive median filter (AMF), (g)-(i) are the restored images via **ECG2**

Finally, Lemma 3 implies that

$$\liminf_{k\to\infty} \|\nabla \mathbf{F}_k\| = 0.$$

□

Funding There is no any Funding

Declarations

Conflicts of interests None

Competing interests None

References

1. Anisha K, Wilscy M (2011) Impulse noise removal from medical images using fuzzy genetic algorithm. Int J Multimed Appl 3:93–106
2. L. Arman Y, Xu M, Rostami F (2020) Rahpeymaii, Some three-term conjugate gradient methods for solving unconstrained optimization problems, Pacific Journal of Optimization
3. Beale EML (1972) A derivative of conjugate gradients In: Lootsma, F.A (ed.) Numerical Methods for Nonlinear Optimization, Academic, London. 39–43
4. Black MA (1996) Rangarajan, On the unification of line processes, outlier rejection, and robust statistics with applications to early vision. Int J Comput Vis 19:57–91
5. Bouman C, Sauer K (1995) On discontinuity-adaptive smoothness priors in computer vision. IEEE Trans Pattern Anal Mach Intell 17:576–586
6. Cai JF, Chan RH, Morini B (2007) Minimization of an edge-preserving regularization functional by conjugate gradient type methods, image processing based on partial differential equations. In: Mathematics and Visualization, Springer, Berlin Heidelberg, pp. 109–122. https://doi.org/10.1007/978-3-540-33267-1_7
7. Chan RH, Ho CW, Nikolova M (2005) Salt-and-pepper noise removal by median-type noise detector and edge-preserving regularization. IEEE Trans Image Process 14:1479–1485
8. Chan R, Hu C, Nikolova M (2004) Iterative procedure for removing random-valued impulse noise. IEEE Signal Process. Lett 11(12):921–924. https://doi.org/10.1109/lsp.2004.838190
9. Chan TF, Shen J, Zhou H (2006) Total variation wavelet inpainting. J Math Imaging Vision 25:107–125. https://doi.org/10.1007/s10851-006-5257-3
10. Charbonnier P, Blanc-Féraud L, Aubert G, Barlaud M (1997) Deterministic edge-preserving regularization in computed imaging. IEEE Trans Image Process 6:298–311
11. Chen J, Zhan Y, Cao H (2020) Iterative deviation filter for fixed-valued impulse noise removal. Multimed Tools Appl 79:23695–23710
12. Dai YH, Yuan Y (1999) A nonlinear conjugate gradient method with a strong global convergence property. SIAM J Optim 10:177–182
13. Dolan ED, Moré JJ (2002) Benchmarking optimization software with performance profiles. Math Program 91:201–213
14. Fletcher R, Reeves C (1964) Function minimization by conjugate gradients. Comput J 7(2):149–154
15. Green PJ (1990) Bayesian reconstructions from emission tomography data using a modified EM algorithm, IEEE Transactions on Medical Imaging, MI-9, 84–93
16. Hager WW, Zhang H (2005) A new conjugate gradient method with guaranteed descent and an efficient line search. SIAM J Optim 16:170–192
17. Halder A, Choudhuri R (2022) Impulse Noise Removal Algorithm Using Modified Adaptive Distance-Related Weighted Mean Filter. In: Das A.K., Nayak J., Naik B., Dutta S., Pelusi D. (eds) Computational Intelligence in Pattern Recognition. Advances in Intelligent Systems and Computing, vol 1349. Springer, Singapore
18. Hestenes MR, Stiefel EL (1952) Methods of conjugate gradients for solving linear systems. J Res Natl Bur Stand 49:409–436
19. Hwang H, Haddad RA (1995) Adaptive medianfilters: new algorithms and results. IEEE Trans Image Process 4:499–502
20. Karthikeyan K, Chandrasekar C (2011) Speckle Noise Reduction of Medical Ultrasound Images using Bayesshrink Wavelet Threshold. Int J Comput Appl 22:8–14
21. Kimiaei M, Rahpeymaii F (2019) Impulse noise removal by an adaptive trust-region method. Soft Comput 23:11901–11923
22. Kimiaei M, Rostami M (2016) Impulse noise removal based on new hybrid spectral conjugate gradient approach. KYBERNETIKA 52(5):791–823
23. Liu J, Cao H, Zhao Y, Zhang L (2020) A gradient-type iterative method for impulse noise removal. Numer Linear Algebra Appl 28(4)
24. Nadeem M, Hussain A, Munir A, Habib M, Tahir Naseem M (2020) Removal of random valued impulse noise from grayscale images using quadrant based spatially adaptive fuzzy filter. Signal Process 169:107403
25. Nikolova M (2004) A variational approach to remove outliers and impulse noise. Journal of Math Imaging Vis 20:99–120
26. Polak E, Ribière G (1969) Note sur la convergence de directions conjugées. Rev Francaise Informat Recherche Opertionelle 3e Année. 16:35–43
27. Russo F, Ramponi F (2000) A Fuzzy filter for images corrupted by impulse noise. IEEE Trans Image Process 3:168–170

28. Rytsar YB, Ivasenko IB (1997) Application of (alpha, beta)-trimmed mean filtering for removal of additive noise from images. SPIE Proceeding. Optoelectronic and Hybrid Optical/Digital Systems for Image Processing 45–52
29. Shah A, Bangash JI, Khan AW, Ahmed I, Khan A, Khan A, Khan A (2020) Comparative analysis of median filter and its variants for removal of impulse noise from gray scale images. Journal of King Saud University - Computer and Information Sciences
30. Shukla HS, Kumar N, Tripathi RP (2014) Median Filter based Wavelet Transform for Multilevel Noise. Int J Comput Appl 107(14):11–14
31. Wang L, Lu J, Li Y, Yahagi T, Okamoto T (2005) Noise reduction using wavelet with application to medical X-ray image. Int Conf Ind Technol 20:33–38
32. Wang L, Xiao D, Hou WS, Wu XY, Chen L (2021) Weighted Schatten p-norm minimization for impulse noise removal with TV regularization and its application to medical images. Biomed Signal Process Control 66:102123
33. P. Wolfe (1971) Convergence conditions for ascent methods. II: some corrections. SIAM Rev 13(2):185–188
34. Yu G, Huang J, Zhou Y (2010) A descent spectral conjugate gradient method for impulse noise removal. Appl Math Lett 23:555–560
35. Yu G, Qi L, Sun Y, Zhou Y (2010) Impulse noise removal by a nonmonotone adaptive gradient method. Signal Process 90:2891–2897
36. Zhang L, Zhou W, Li DH (2006) A descent modified Polak-Ribiére-Polyak conjugate gradient method and its global convergence. IMA J Numer Anal 26(4):629–640
37. Zhou EL, Xia BY, Li E, Wang TT (2022) An efficient algorithm for impulsive active noise control using maximum correntropy with conjugate gradient. Appl Acoust 188:108511
38. Zoutendijk G (1970) Nonlinear programming, computational methods. In: Abadie J (ed) Integer and nonlinear programming. North-holland, Amsterdam, pp 37–86

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.