CrossMark

ORIGINAL PAPER

Denoising images corrupted by impulsive noise using projections onto the epigraph set of the total variation function (PES-TV)

Mohammad Tofighi[1] · Kivanc Kose[2] · A. Enis Cetin[1]

Abstract In this article, a novel algorithm for denoising images corrupted by impulsive noise is presented. Impulsive noise generates pixels whose gray level values are not consistent with the neighboring pixels. The proposed denoising algorithm is a two-step procedure. In the first step, image denoising is formulated as a convex optimization problem, whose constraints are defined as limitations on local variations between neighboring pixels. We use Projections onto the Epigraph Set of the TV function (PES-TV) to solve this problem. Unlike other approaches in the literature, the PES-TV method does not require any prior information about the noise variance. It is only capable of utilizing local relations among pixels and does not fully take advantage of correlations between spatially distant areas of an image with similar appearance. In the second step, a Wiener filtering approach is cascaded to the PES-TV-based method to take advantage of global correlations in an image. In this step, the image is first divided into blocks and those with similar content are jointly denoised using a 3D Wiener filter. The denoising performance of the proposed two-step method was compared against three state-of-the-art denoising methods under various impulsive noise models.

✉ Mohammad Tofighi
tofighi@ee.bilkent.edu.tr

Kivanc Kose
kosek@mskcc.org

A. Enis Cetin
cetin@bilkent.edu.tr

[1] Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey

[2] Dermatology Service, Memorial Sloan-Kettering Cancer Center, New York, NY 10022, USA

1 Introduction

In this article, a two-step algorithm for denoising images that are corrupted by impulsive noise is presented. In the first step, local variations among neighboring pixel values are minimized in order to remove the impulsive components of the observed image. The first step does not fully take advantage of the correlation between distant areas of an image with similar appearance, e.g., blue sky region covering all the top portions of an image, cheek of a facial image and even textural regions of a shirt. In the second stage of the denoising method, similar image blocks are determined using a block matching algorithm and they are denoised using Wiener filtering as in [1].

The first step of the proposed algorithm is based on projections onto the epigraph set of the total variation function (PES-TV) [2–4]. In the PES-TV approach, the denoising operation is formulated as an orthogonal projection problem in which the input image is projected onto the epigraph set of the total variation (TV) function.

Dabov et al. [1] proposed block matching 3D filtering (BM3D) denoising method that can utilize the correlation between similar areas of the image by jointly denoising them together. BM3D seems to be the best image denoising method for images corrupted by Gaussian noise [1,5–23]. BM3D is also a two-stage algorithm. However, the first stage of BM3D requires an estimate of the noise variance beforehand to determine the hard thresholding level used in the first stage, which fails to produce a good estimate of the image under impulsive noise. As a result, the second stage of the BM3D does not produce a reliable denoised image when the noise is impulsive. On the other hand, the PES-TV denoising method

🕿 Springer

does not need an accurate estimate of the noise variance in the first step. It does not require any parameter adjustment either. When we combine the second part of BM3D with the PES-TV approach, we get better results than ordinary BM3D approach for images corrupted by impulsive noise. An approximate estimation of the noise variance is enough for the second stage of the PES-TV-based method.

The article is organized as follows. In Sect. 2, the PES-TV framework, is presented. In Sect. 3, block matching and collaborative filtering part of the denoising method is described. Simulation results are presented in Sect. 4.

2 Denoising using projections onto epigraph sets of the TV function (PES-TV)

Let the original image be $\mathbf{v}$, and the noisy image be $\mathbf{v}_0$. Suppose that the observation model is the additive noise model:

$$\mathbf{v}_0 = \mathbf{v} + \boldsymbol{\eta}, \tag{1}$$

where $\boldsymbol{\eta}$ is the additive noise. We define the TV function for a discrete image $\mathbf{w} = [\mathbf{w}^{i,j}]\ 0 \leq i, j \leq M-1\ \in \mathbb{R}^{M \times M} = \mathbb{R}^N$ as follows:

$$\mathrm{TV}(\mathbf{w}) = \sum_{i,j} |\mathbf{w}^{i+1,j} - \mathbf{w}^{i,j}| + \sum_{i,j} |\mathbf{w}^{i,j+1} - \mathbf{w}^{i,j}|. \tag{2}$$

It can be shown that the TV function $\mathrm{TV} : \mathbb{R}^N \to \mathbb{R}$ is a convex cost function [2,3,8,16]. We define the epigraph set of the TV in $\mathbb{R}^{N+1}$ as follows:

$$C_{\mathrm{TV}} = \left\{ \underline{\mathbf{w}} = [\mathbf{w}^T\ y]^T\ :\ y \geq \mathrm{TV}(\mathbf{w}) \right\}, \tag{3}$$

which is the set of $N+1$-dimensional vectors, whose $(N+1)$st component y is greater than $\mathrm{TV}(\mathbf{w})$. We use bold face letters for N-dimensional vectors and underlined bold face letters for $N+1$-dimensional vectors, respectively. A graphical description of the epigraph concept is illustrated in Fig. 1.

The first step of our denoising algorithm consists of making an orthogonal projection onto C_{TV}. Let $\underline{\mathbf{v}}_0 = [\mathbf{v}\ 0]^T$ be an arbitrary vector in $\mathbb{R}^{N+1}$. The projection $\underline{\mathbf{w}}^*$ is determined by minimizing the distance between $\underline{\mathbf{v}}_0$ and C_{TV}, i.e.,

$$\underline{\mathbf{w}}^* = \arg \min_{\underline{\mathbf{w}} \in C_{\mathrm{TV}}} \|\underline{\mathbf{v}}_0 - \underline{\mathbf{w}}\|^2. \tag{4}$$

In this approach, $\mathbf{v}_0$ is projected onto the C_{TV}. This means that we select the nearest vector $\underline{\mathbf{w}}^\star$ on the set C_{TV} to $\mathbf{v}_0$ as illustrated in Fig. 1. Eq. (4) is equivalent to:
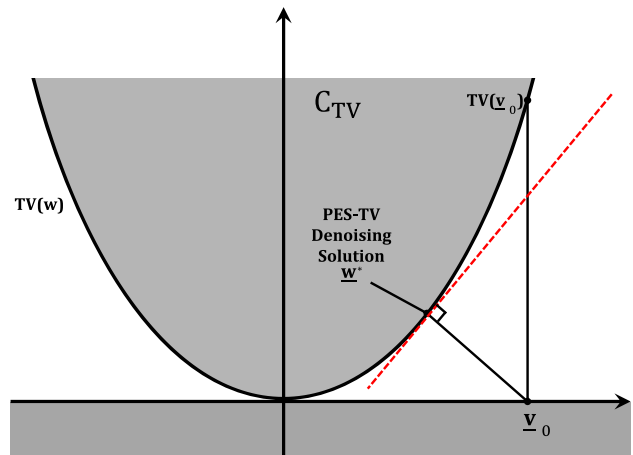

Fig. 1 Graphical representation of the minimization operation in (5) and (6). The corrupted observation vector $\mathbf{v}_0$ is projected onto the set C_{TV}

$$\underline{\mathbf{w}}^\star = \begin{bmatrix} \mathbf{w}^\star \\ \mathrm{TV}(\mathbf{w}^\star) \end{bmatrix} = \arg \min_{\underline{\mathbf{w}} \in C_{\mathrm{TV}}} \left\| \begin{bmatrix} \mathbf{v}_0 \\ 0 \end{bmatrix} - \begin{bmatrix} \mathbf{w} \\ \mathrm{TV}(\mathbf{w}) \end{bmatrix} \right\|^2, \tag{5}$$

where $\underline{\mathbf{w}}^\star = [\mathbf{w}^{\star T}, \mathrm{TV}(\mathbf{w}^\star)]^T$ is the projection of $[\mathbf{v}_0\ 0]^T$ onto the epigraph set. The projection $\underline{\mathbf{w}}^\star$ must be on the boundary of the epigraph set. Therefore, the projection must be of the form $[\mathbf{w}^{\star T}, \mathrm{TV}(\mathbf{w}^\star)]$. Eq. (5) becomes:

$$\underline{\mathbf{w}}^\star = \begin{bmatrix} \mathbf{w}^\star \\ \mathrm{TV}(\mathbf{w}^\star) \end{bmatrix} = \arg \min_{\underline{\mathbf{w}} \in C_{\mathrm{TV}}} \left(\|\mathbf{v}_0 - \mathbf{w}\|^2 + \mathrm{TV}(\mathbf{w})^2 \right). \tag{6}$$

Solution of (6) using projections onto boundary and tangential hyperplanes are described in [24].

It is also possible to use $\phi(\mathbf{w}) = \lambda \mathrm{TV}(\mathbf{w})$ as the convex cost function and Eq. (5) becomes:

$$\begin{bmatrix} \mathbf{w}^\star \\ \mathrm{TV}(\mathbf{w}^\star) \end{bmatrix} = \arg \min_{\underline{\mathbf{w}} \in C_{\mathrm{TV}}} \|\mathbf{v}_0 - \mathbf{w}\|^2 + \lambda^2 \mathrm{TV}(\mathbf{w})^2. \tag{7}$$

In current TV-based denoising methods [16,17] the following cost function is used:

$$f(\mathbf{w}) = \|\mathbf{v}_0 - \mathbf{w}\|^2 + \lambda \mathrm{TV}(\mathbf{w}). \tag{8}$$

In our case, the regularization term is the square of the TV function as shown in (6). Since the noise is impulsive, the contribution of the regularization function (TV) should be higher than usual. The first term in (6) and (8) consists of components $|\mathbf{v}_i - \mathbf{w}_i|$, which are comparable to $|\mathbf{w}_i - \mathbf{w}_{i-1}|$ terms forming the TV function. The ℓ_2-norm dominates the TV function in ordinary LASSO cost function. However, in (6) and (7) the square of $\mathrm{TV}(\mathbf{w})$ increases the effect of the regularization term. It also leads to an efficient computational solution in [24].

Fig. 2 Graphical representation of the proposed two-stage denoising process
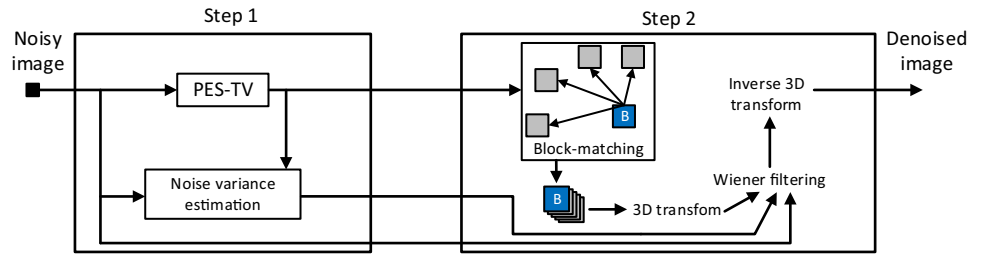
Finding the right regularization parameter is a major problem in LASSO. Unlike LASSO approach [25], where the selection of the λ parameters is determined in an ad hoc manner or inspection, in the proposed PES-TV-based denoising algorithm, it is experimentally observed that $\lambda = 1$ works well [26]. We tried various λ values between 0.2 and 2, and $\lambda = 1$ produced the best results. The PES-TV software is available in [26].

3 Block matching and collaborative filtering

The second step of the proposed denoising method is the three dimensional (3D) approach introduced by Dabov et al. [1]. The output of the PES-TV-based denoising step is fed into the "3D" block matching (BM) step of BM3D.

In natural images, spatially distant areas/blocks are correlated with each other. However, most denoising algorithms do not exploit this fact and only consider local pixel variations in an image. Dabov et al. introduced block matching and collaborative Wiener filtering concepts in a denoising framework to take advantage of similarities between spatially distant blocks in an image. They first group similar looking regions in an image by block matching. Then, they denoise all those regions together using a 3D approach called collaborative Wiener filtering. We borrow this procedure from [1], and use it as the second step of our denoising scheme as shown in Fig. 2. In this section, we briefly review the BM3D denoising method.

3.1 Block matching

First the PES-TV denoised image is divided into non-overlapping regions of fixed size called reference blocks (B$_R$). Then, each reference block is compared against candidate blocks of similar appearance (B$_C$) using the following equation:

$$d(\mathrm{B_R}, \mathrm{B_C}) = \frac{\|\mathrm{B_R} - \mathrm{B_C}\|^2}{N}, \tag{9}$$

where $N = M^2$ is the number of pixels in each block. Blocks satisfying the similarity condition are grouped together to construct 3D arrays of similar blocks (SB). Blocks satisfying

condition of the set $G_{\mathrm{SB_R}}$ are grouped together to construct 3D arrays of similar blocks, as follows:

$$G_{\mathrm{SB_R}} = \{c \in \mathbf{w}^\star : d(\mathrm{B_R}, \mathrm{B_C}) \leq \tau_{th}\} \tag{10}$$

where c represents the coordinate of blocks in the reconstructed image obtained by the PES-TV step, and τ_{th} is the maximum distance whose similarity of blocks is determined according to this threshold. The distance threshold τ_{th} is determined according to deterministic speculations based on the denoised image in the first step [1]. Each set $G_{\mathrm{SB_R}}$ is an $N \times N_{G_{\mathrm{SB}}}$ 3D array of similar blocks, where $N_{G_{\mathrm{SB_R}}}$ is the number of blocks in the set $G_{\mathrm{SB_R}}$.

3.2 Collaborative filtering

The 3D arrays obtained by block matching have both spatial and "temporal" similarity. Therefore, the noise can be efficiently removed by the collaborative 3D Wiener filtering. Wiener shrinkage coefficients for the set of blocks are determined from the 3D transform coefficient as follows:

$$W_{G_{\mathrm{SB_R}}} = \frac{\left|T\left(G_{\mathrm{SB_R}}^{\mathrm{BE}}\right)\right|^2}{\left|T\left(G_{\mathrm{SB_R}}^{\mathrm{BE}}\right)\right|^2 + \sigma^2}, \tag{11}$$

where $G_{\mathrm{SB_R}}^{\mathrm{BE}}$ is the 3D array for similar blocks from basic estimate (BE), which is the output of the PES-TV step, $T(.)$ is the transformation operator, $|T(G_{\mathrm{SB}}^{\mathrm{BE}})|^2$ is the power spectrum of the basic estimate image and σ^2 is the variance of the noise which is estimated using the difference between the noisy image and the denoised image obtained from PES-TV step. After obtaining the coefficients, the collaborative filtering is realized by elementwise multiplication of $W_{G_{\mathrm{SB_R}}}$ by the 3D arrays of noisy image $G_{\mathrm{SB_R}}$, as follows:

$$\mathbf{w}_{\mathrm{rec}}^\star = T^{-1}\left(W_{G_{\mathrm{SB_R}}} T(G_{\mathrm{SB_R}})\right). \tag{12}$$

where $\mathbf{w}_{\mathrm{rec}}^\star$ is the reconstructed 3D array, and $T^{-1}(.)$ is the inverse transformation operator. After filtering the 3D array, inverse transform and aggregation operation [1] is performed to get the final denoised image. The overall process is explained graphically in Fig. 2.

4 Simulation results and conclusions

In this section, simulation examples are presented. The basic estimate, which is obtained in the first step, affects the main denoising process in Wiener filtering step. In BM3D approach, first step requires the knowledge of the variance of the noise; however, for images corrupted by impulsive noise

the exact variance is unknown. Therefore, this step fails to generate an appropriate basic estimate for second step. Using the PES-TV approach [3], we solve this problem.

The impulsive noise changes the pixel values in the image as follows:

$$v_0^{i,j} = \begin{cases} v^{i,j}, & \text{if } x < l \\ i_{\min} + y(i_{\max} - i_{\min}), & \text{if } x > l \end{cases} \quad (13)$$

where $v^{i,j}$ is the (i, j)th pixel in the original image, $x, y \in [0, 1]$ are two uniformly distributed random variables, l is the parameter controlling how much of the image is corrupted, and $i_{\max}$ and $i_{\min}$ are the severity of the noise [27]. The salt and pepper noise and the ϵ-contaminated Gaussian noise are two types of impulsive noises. The ϵ-contaminated Gaussian noise is widely used to represent impulsive noise [18,28]. The PDF of ϵ-contaminated Gaussian noise is given by:

$$v_0^{i,j} = v^{i,j} + \begin{cases} \eta_1^{i,j}, & \text{with probability } 1 - \epsilon \\ \eta_2^{i,j}, & \text{with probability } \epsilon \end{cases} \quad (14)$$

where η_1 and η_2 are independent Gaussian noise sources with variances σ_1^2 and σ_2^2, respectively. We assume that $\sigma_1 \ll \sigma_2$, and ϵ is a small positive number [21]. The reconstruction performance is measured using the signal-to-noise ratio (SNR) and peak-SNR (PSNR) criterions, which are defined as follows:

$$SNR = 20 \times \log_{10}\left(\frac{\|\mathbf{w}_{\text{orig}}\|}{\|\mathbf{w}_{\text{orig}} - \mathbf{w}_{\text{rec}}\|}\right), \quad (15)$$

$$PSNR = 20 \times \log_{10}\left(\frac{\max(\mathbf{w}_{\text{orig}})}{\|\mathbf{w}_{\text{orig}} - \mathbf{w}_{\text{rec}}\|/N}\right), \quad (16)$$

where $\mathbf{w}_{orig}$ is the original signal, $\mathbf{w}_{\text{rec}}$ is the reconstructed signal and N is the total number of pixels in image.

Denoising results for "Note" [3] image with ϵ-contaminated noise are summarized in Table 1. In this toy

Table 1 Comparison of the SNR results for denoising algorithms for ϵ-contaminated Gaussian noise for "Note" image

ϵ	σ_1	σ_2	SNR$_{\text{Input}}$	PES-TV	Chambolle	BM3D
0.1	5	30	14.64	**29.67**	22.26	24.43
0.1	5	40	12.55	**27.84**	20.32	20.75
0.1	5	50	10.75	**25.84**	18.63	17.59
0.1	5	60	9.29	**24.12**	17.37	15.09
0.1	5	70	7.98	**22.52**	16.24	13.14
0.1	5	80	6.89	**21.03**	14.97	11.60
0.1	10	30	12.56	**25.98**	21.71	25.73
0.1	10	40	11.13	**24.74**	19.97	23.83
0.1	10	50	9.85	**23.24**	18.46	21.56
0.1	10	60	8.58	**22.07**	17.10	19.11
0.1	10	70	7.52	**20.49**	16.03	16.71
0.1	10	80	6.46	**18.84**	15.12	14.87
0.05	5	30	16.75	**28.60**	23.78	26.93
0.05	5	40	14.98	**26.04**	21.54	23.10
0.05	5	50	13.41	**23.91**	19.91	19.98
0.05	5	60	12.10	**21.63**	18.63	17.60
0.05	5	70	10.80	**19.50**	17.50	15.87
0.05	5	80	9.76	**17.23**	16.38	14.38
0.05	10	30	13.68	**26.90**	22.62	26.70
0.05	10	40	12.66	**25.68**	21.12	25.46
0.05	10	50	11.71	**24.72**	19.60	23.73
0.05	10	60	10.72	**23.62**	18.30	21.43
0.05	10	70	9.82	**21.77**	17.22	19.33
0.05	10	80	8.92	**20.29**	16.45	17.25

Table 2 PSNR Results for denoising images corrupted with ϵ-contaminated noise with $\epsilon = 0.1$, $\sigma_1 = 5$, with different σ_2's using PES-TV algorithm

Images	$\sigma_2 = 30$	$\sigma_2 = 40$	$\sigma_2 = 50$	$\sigma_2 = 60$	$\sigma_2 = 70$	$\sigma_2 = 80$
House	36.87	34.39	31.87	29.74	28.00	26.53
Lena	34.55	32.85	31.40	29.97	28.55	27.27
Mandrill	28.31	27.86	27.36	26.76	25.33	24.59
Living room	31.61	30.94	29.57	28.41	27.46	26.38
Lake	32.03	31.29	29.71	28.64	27.57	26.58
Jet plane	34.56	32.75	31.20	29.85	28.32	27.05
Peppers	34.64	33.39	32.02	30.56	29.22	27.87
Pirate	31.46	30.80	29.60	28.50	27.49	26.53
Cameraman	35.29	33.45	31.45	29.74	28.14	26.65
Flower	31.17	31.03	29.85	28.78	27.69	26.88
Kodak (ave.)	32.85	31.19	29.88	28.65	27.51	26.48
Average	**32.91**	**31.39**	**30.03**	**28.78**	**27.59**	**26.53**

Table 3 PSNR Results for denoising images corrupted with ϵ-contaminated noise with $\epsilon = 0.1$, $\sigma_1 = 5$, with different σ_2's using BM3D algorithm

Images	$\sigma_2 = 30$	$\sigma_2 = 40$	$\sigma_2 = 50$	$\sigma_2 = 60$	$\sigma_2 = 70$	$\sigma_2 = 80$
House	34.65	30.40	27.59	25.34	23.69	22.40
Lena	33.53	30.13	27.28	25.13	23.55	22.29
Mandrill	31.48	28.88	26.66	24.89	23.36	22.27
Living room	33.06	30.14	27.64	25.56	23.90	22.49
Lake	33.70	30.36	27.63	25.42	23.75	22.46
Jet plane	33.50	30.28	27.67	25.47	24.02	22.68
Peppers	33.66	30.50	27.62	25.48	23.86	22.46
Pirate	32.58	29.74	27.67	25.20	23.69	22.45
Cameraman	33.99	30.32	27.39	25.29	23.69	22.40
Flower	32.72	30.27	27.91	25.76	24.07	22.76
Kodak (ave.)	32.11	30.53	28.10	26.05	24.37	23.04
Average	32.46	30.40	27.93	25.85	24.19	22.87

example, the PES-TV approach produces the best results. The denoising results for a set of 34 images including 10 well-known test images from image processing literature and 24 images from Kodak Database [29], which are corrupted by ϵ-contaminated noise with $\sigma_1 = 5$ and $\epsilon = 0.1$, and $\sigma_2 \in [30, 80]$ are presented in Tables 2 and 3 for PES-TV and BM3D algorithms, respectively. In this case, the noise is the combination of two Gaussian noises with different variances; therefore, it cannot be exactly modeled as a noise with a single variance parameter. The PES-TV algorithm performs better and produces higher PSNR values compared to all other denoising results obtained using [1,16,19], because it does not require knowledge of variance of the noise. We also present an additional illustrative example in Fig. 3. In this figure the "peppers" image is corrupted with ϵ-contaminated Gaussian noise and the denoising results for PES-TV and BM3D methods are presented. The BM3D fails to clear the impulsive noise.

In another set of experiments, images that are corrupted by a mixture of salt and pepper and Gaussian noises are denoised using the PES-TV denoising method and also with BM3D and BM3D with median filtering (BM3D$_M$) for comparison purposes. The salt and pepper impulsive noise model is as follows:

$$v_0^{i,j} = \begin{cases} s_{\min}, & \text{with probability } p \\ s_{\max}, & \text{with probability } q \\ v^{i,j}, & \text{with probability } 1 - p - q \end{cases} \quad (17)$$

where $v^{i,j}$ is the gray level pixel value of the original image, $[s_{\min}, s_{\max}]$ are the dynamic range of the original image, $s_{\min} \leq v^{i,j} \leq s_{\max}$ for all (i, j) values, $v_0^{i,j}$ is the gray level pixel value of the noisy image and $r = p + q$ defines the noise level [30].

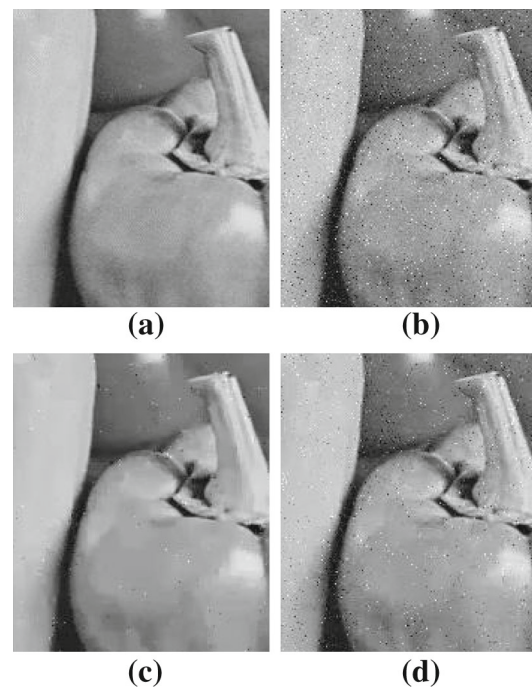The density of the salt and pepper noise is set to 0.02 and 0.05 and Gaussian noise is added with different variances.

(a)   **(b)**

(c)   **(d)**

Fig. 3 **a** A portion of original "Peppers" image, **b** image corrupted by ϵ-contaminated noise with $\epsilon = 0.1$, $\sigma_1 = 5$, and $\sigma_2 = 50$, **c** denoised image, using PES-TV algorithm; PSNR = 32.02 dB, and **d** denoised image, using BM3D; PSNR = 27.62 dB. Standard BM3D algorithm fails to clear impulsive noise

Results for this set of experiments are shown in Tables 4 and 5, respectively. In almost all cases, the PSNR values for PES-TV algorithm are higher than other algorithms. In Tables 4 and 5, the third column refers to median filtering followed by second stage (3D Wiener filtering) of the BM3D algorithm (BM3D$_M$).

An illustrative comparison of the PES-TV vs. BM3D and BM3D$_M$ is presented in Fig. 4 for "Lena" image. In this example, the PES-TV denoising method performs better than

Table 4 PSNR results for denoising images corrupted with "salt and pepper" noise with density $d = 0.02$ and Gaussian noise with variance σ using PES-TV, BM3D, BM3D$_M$ methods

Image	$\sigma = 20$			$\sigma = 30$			$\sigma = 40$			$\sigma = 50$		
	PES-TV	BM3D	BM3D$_M$	PES-TV	BM3D	BM3D$_M$	PES-TV	BM3D	BM3D$_M$	PES-TV	BM3D	BM3D$_M$
Cameraman	**33.33**	27.88	31.62	**31.23**	29.40	29.15	29.75	**30.18**	27.18	28.93	**29.33**	25.63
Note	**35.39**	23.31	29.76	**32.73**	26.18	27.98	**30.72**	28.46	26.05	**28.44**	27.52	24.83
House	**35.64**	29.54	33.14	**33.74**	32.24	30.19	31.93	**32.02**	27.87	30.73	**31.07**	26.08
Flower	**30.20**	27.14	28.40	**28.33**	27.08	26.86	**27.03**	26.44	25.34	**26.12**	25.79	24.12
Lena	**32.67**	30.14	31.08	**30.66**	30.25	28.75	**29.65**	29.28	26.84	28.38	**28.78**	25.24
Jetplane	**32.34**	27.48	30.55	**30.28**	28.76	28.30	**28.84**	28.65	26.49	**27.93**	27.76	24.98
Lake	**30.31**	26.97	28.87	**28.40**	27.20	27.16	**27.08**	26.65	25.53	**26.11**	25.80	24.25
Peppers	**32.91**	29.83	31.14	**31.26**	30.24	28.87	**29.97**	29.68	26.95	**29.19**	28.90	25.37
Livingroom	**30.34**	28.49	28.82	**28.32**	27.81	27.02	**26.91**	26.87	25.48	**26.09**	25.96	24.20
mandrill	**28.63**	26.63	27.51	**26.49**	25.63	25.87	**25.03**	24.72	24.51	**24.11**	23.96	23.36
pirate	**30.32**	27.91	29.11	**28.54**	27.75	27.32	**27.32**	27.14	25.73	**26.61**	26.47	24.43

Table 5 PSNR results for denoising images corrupted with "salt and pepper" noise with density $d = 0.05$ and Gaussian noise with variance σ using PES-TV, BM3D, BM3D$_M$ methods

| Image	$\sigma = 20$			$\sigma = 30$			$\sigma = 40$			$\sigma = 50$		
	PES-TV	BM3D	BM3D$_M$	PES-TV	BM3D	BM3D$_M$	PES-TV	BM3D	BM3D$_M$	PES-TV	BM3D	BM3D$_M$
Cameraman	**33.21**	26.71	30.18	**31.19**	29.55	28.26	**29.73**	29.18	26.60	**28.73**	28.68	25.07
Note	**34.67**	27.10	27.24	**32.36**	27.02	25.85	**30.19**	26.49	24.72	**28.11**	25.78	23.62
House	**35.66**	28.91	31.91	**33.66**	32.67	29.35	**31.80**	32.05	27.35	**30.65**	31.27	25.71
Flower	**30.02**	25.80	27.58	**28.16**	26.26	26.12	**26.95**	25.76	24.86	**25.92**	25.28	23.75
Lena	**32.57**	28.95	30.10	**30.59**	29.55	28.09	**29.19**	28.84	26.37	**28.27**	28.17	24.87
Jetplane	**32.21**	26.40	29.20	**30.14**	28.36	27.36	**28.74**	27.70	25.83	**27.69**	27.10	24.50
Lake	**30.25**	25.81	27.71	**28.35**	26.44	26.27	**27.02**	25.79	24.92	**26.02**	25.19	23.77
Peppers	**32.76**	28.96	30.05	**31.16**	29.67	28.11	**29.83**	29.08	26.37	**28.91**	28.41	24.97
Livingroom	**30.25**	27.01	27.86	**28.19**	26.81	26.36	**26.87**	26.12	25.00	**25.89**	25.53	23.83
Mandrill	**28.50**	25.18	26.50	**26.39**	24.60	25.26	**24.92**	24.00	24.11	**23.98**	23.46	23.02
Pirate	**30.22**	26.81	28.19	**28.44**	26.99	26.66	**27.27**	26.42	25.32	**26.46**	25.95	24.05
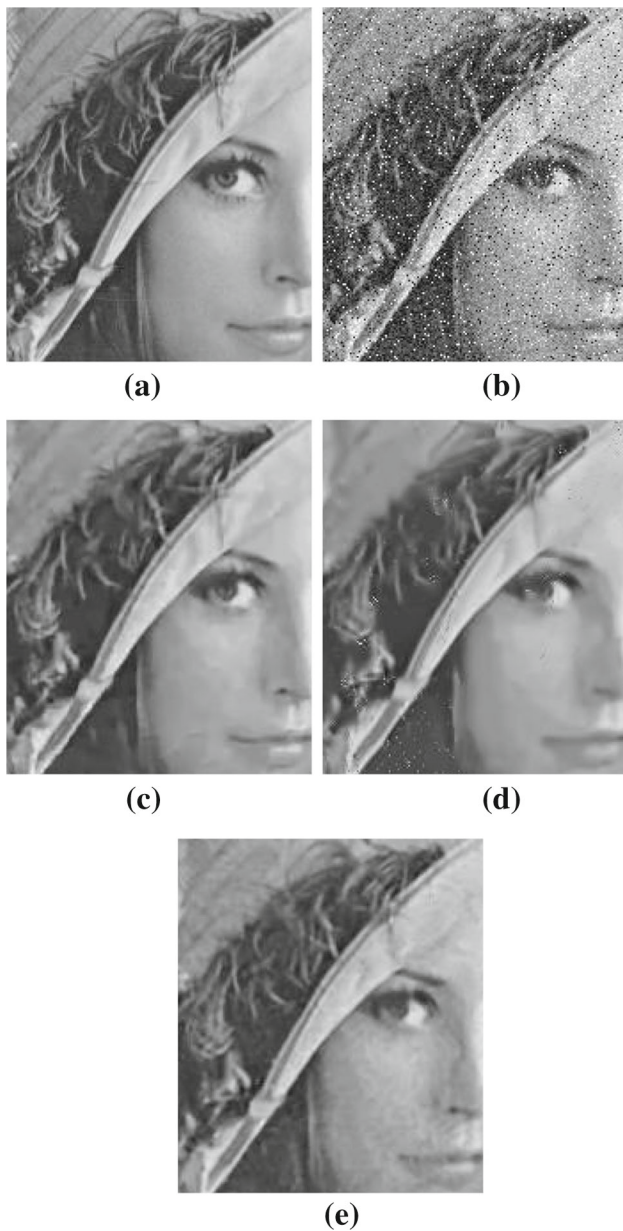
Fig. 4 **a** A portion of original "Lena" image, **b** image corrupted by salt and pepper noise with density 0.05, and additive white Gaussian noise with standard deviation $\sigma = 20$, **c** denoised image, using PES-TV algorithm; PSNR = 32.57 dB, **d** denoised image, using BM3D; PSNR = 28.95 dB, and **e** denoised image, using BM3D-Median; PSNR = 30.10 dB

Fig. 5 **a** Fluorescence microscopic image, **b** image corrupted by salt and pepper noise with density 0.05, and additive white Gaussian noise with standard deviation $\sigma = 30$, **c** denoised image, using PES-TV algorithm; PSNR = 32.98 dB, **d** denoised image, using BM3D; PSNR = 28.61 dB, and **e** denoised image, using BM3D-Median; PSNR = 29.35 dB

the other two denoising methods both visually and in the sense of PSNR. For example in Fig. 4d the artifacts produced by salt and pepper noise are not removed properly by BM3D algorithm and the denoised image by $BM3D_M$ in Fig. 4e has also some artifacts. This issue is solved by PES-TV denoising method in Fig. 4c. This is also apparent in a fluorescence microscopic image in Fig. 5. In this figure, the example fluorescence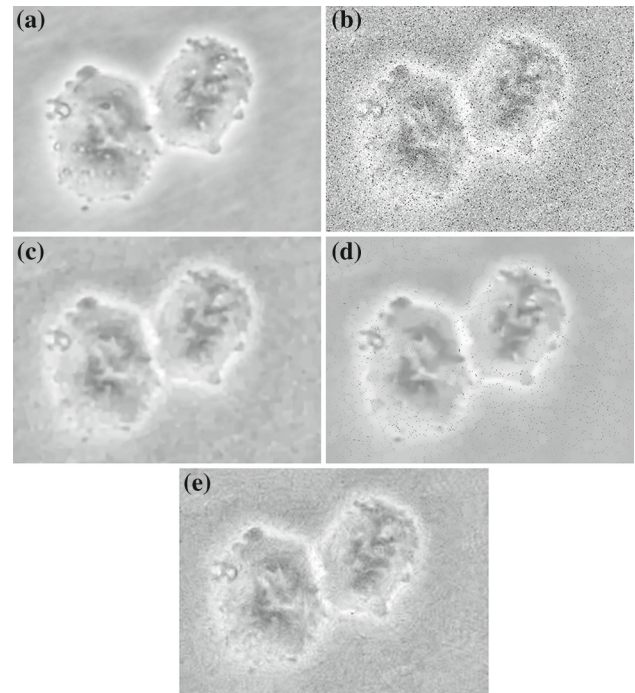 microscopic image is corrupted by salt and pepper noise with density 0.05, and additive white Gaussian noise with standard deviation $\sigma = 30$ and denoised using the PES-TV, BM3D and $BM3D_M$ denoising methods. Both visually and in the sense of PSNR values, PES-TV performs better compared to other two denoising methods.

In [20], the proximity operator-based denoising results for the Cameraman and Lena images are reported for various regularization parameter λ values for Gaussian noise with $\sigma = 15$ and 25 standard deviation levels. Best PSNR values for Lena image for $\sigma = 15$ ($\sigma = 25$) is 32.33 dB (30.13 dB), when the regularization parameter $\lambda = 0.09$ ($\lambda = 0.05$). We obtain PSNR values equal to 32.43 and 30.12 dB, respectively, without any regularization parameter adjustment. For Cameraman our results are much better with PSNR = 33.10 and 30.60 dB compared to 30.39 and 27.77 dB with $\lambda = 0.1$ and $\lambda = 0.07$ for $\sigma = 15$ and 25, respectively.

The first step of the BM3D approach relies on hard thresholding, which cannot remove isolated large amplitude impulsive noise components. On the other hand, the PES-TV approach successfully reduces the impulsive noise and produces better estimate for the second step of the BM3D denoising method. It is experimentally observed that the proposed scheme on images corrupted by impulsive noise results in much better denoising performance compared to both Chambolle's method and standard BM3D denoising.

5 Conclusion

In this article, a novel algorithm for denoising images corrupted by impulsive noise is presented. This algorithm is a two-step algorithm, which in the first step the PES-TV-based denoising algorithm produces basic denoised estimate for the second step. Using this basic estimate, the second stage groups the similar blocks of the noisy image and denoise these 3D arrays of the similar blocks using collaborative 3D Wiener filtering. The PES-TV algorithm does not require the knowledge of the noise variance to denoise the image, then produces better basic estimate for Wiener filtering step in comparison with standard BM3D algorithm. Experimental results indicate that higher SNR and PSNR, and better visual results are obtained using the proposed denoising method compared to other algorithms.

References

1. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3-d transform-domain collaborative filtering. Image Process. IEEE Trans. **16**(8), 2080–2095 (2007)
2. Cetin, A.E., Bozkurt, A., Gunay, O., Habiboglu, Y.H., Kose, K., Onaran, I., Sevimli, R.A., Tofighi, M.: Projections onto convex sets (pocs) based optimization by lifting. IEEE GlobalSIP, Austin, Texas, USA (2013)
3. Tofighi, M., Kose, K., Cetin, A.E.: Denoising using projections onto the epigraph set of convex cost functions. In: Image Processing (ICIP), 2014 IEEE International Conference on, October 2014. pp. 2709–2713, (2014)
4. Cetin, A.E., Tofighi, M.: Projection-based wavelet denoising [lecture notes]. Signal Process. Mag. IEEE., **32**(5), 120–124 (2015)
5. Censor, Y., Chen, W., Combettes, P.L., Davidi, R., Herman, G.: On the effectiveness of projection methods for convex feasibility problems with linear inequality constraints. Comput. Optim. Appl. **51**(3), 1065–1088 (2012)
6. Youla, D., Webb, H.: Image restoration by the method of convex projections: part 1 theory. Med. Imaging IEEE Trans. **1**(2), 81–94 (1982)
7. Trussell, H., Civanlar, M.R.: The Landweber iteration and projection onto convex set. IEEE Trans. Acoust. Speech Signal Process. **33**(6), 1632–1634 (1985)
8. Combettes, P.L., Pesquet, J.: Image restoration subject to a total variation constraint. IEEE Trans. Image Process. **13**, 1213–1222 (2004)
9. Yamada, I., Yukawa, M., Yamagishi, M.: Minimizing the moreau envelope of nonsmooth convex functions over the fixed point set of certain quasi-nonexpansive mappings. In: Fixed-Point Algorithms for Inverse Problems in Science and Engineering. pp. 345–390. Springer, (2011)
10. Censor, Y., Herman, G.T.: On some optimization techniques in image reconstruction from projections. Appl. Numer. Math. **3**(5), 365–391 (1987)
11. Sezan, I., Stark, H.: Image restoration by the method of convex projections: part 2-applications and numerical results. IEEE Trans. Med. Imaging **1**(2), 95–101 (1982)
12. Chierchia, G., Pustelnik, N., Pesquet, J.-C., Pesquet-Popescu, B.: An epigraphical convex optimization approach for multicomponent image restoration using non-local structure tensor. In: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, 2013. pp. 1359–1363
13. Gubin, L., Polyak, B., Raik, E.: The method of projections for finding the common point of convex sets. USSR Comput. Math. Math. Phys. **7**(6), 1–24 (1967)
14. Bregman, L.: The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. USSR Comput. Math. Math. Phys. **7**(3), 200–217 (1967)
15. Kose, K., Cevher, V., Cetin, A.E.: Filtered variation method for denoising and sparse signal processing. IEEE international conference on acoustics, speech and signal processing (ICASSP), pp. 3329–3332. (2012)
16. Chambolle, A.: An algorithm for total variation minimization and applications. J. Math. Imaging Vision **20**(1–2), 89–97 (2004)
17. Combettes, P.L., Pesquet, J.-C.: Proximal splitting methods in signal processing. In: Bauschke, H.H., Burachik, R.S., Combettes, P.L., Elser, V., Luke, D.R., Wolkowicz, H.: (eds.) Fixed-pointalgorithms for inverse problems in science and engineering, ser. Springer optimization and its applications, pp. 185–212. Springer, New York. (2011)
18. Bovik, A., Huang, T., Munson, J.D.C.: A generalization of median filtering using linear combinations of order statistics. Acoust. Speech Signal Process. IEEE Trans. **31**(6), 1342–1350 (1983)
19. Luisier, F., Blu, T., Unser, M.: A new sure approach to image denoising: interscale orthonormal wavelet thresholding. Image Process. IEEE Trans. **16**(3), 593–606 (2007)
20. Micchelli, C.A., Shen, L., Xu, Y.: Proximity algorithms for image models: denoising. Inverse Probl. **27**(4), 045009 (2011)
21. Cetin, A.E., Tekalp, A.: Robust reduced update kalman filtering. Circuits Syst. IEEE Trans. **37**(1), 155–156 (1990)
22. Theodoridis, S., Slavakis, K., Yamada, I.: Adaptive learning in a world of projections. Signal Process. Mag. IEEE **28**(1), 97–123 (2011)
23. Chierchia, G., Pustelnik, N., Pesquet, J.-C., Pesquet-Popescu, B.: Epigraphical projection and proximal tools for solving constrained convex optimization problems. Signal Image Video Process. **9**(8), 1737–1749 (2015)
24. Tofighi, M., Kose, K., Cetin, A.E.: Signal reconstruction framework based on projections onto epigraph set of a convex cost function (PESC). ArXiv e-prints Feberuary 2014
25. Efron, I.J.B., Hastie, T., Tibshirani, R.: Least angle regression. Ann. Stat. **32**(2), 407–499 (2004)
26. PES-TV software. http://signal.ee.bilkent.edu.tr/PES-TV.html
27. Windyga, P.: Fast impulsive noise removal. Image Process. IEEE Trans. **10**(1), 173–179 (2001)
28. Lee, Y.-H., Kassam, S.: Generalized median filtering and related nonlinear filtering techniques. Acousti. Speech Signal Process. IEEE Trans. **33**(3), 672–683 (1985)
29. Kodak lossless true color image suite. http://r0k.us/graphics/kodak/
30. Chan, R., Ho, C.-W., Nikolova, M.: Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. Image Process. IEEE Trans. **14**(10), 1479–1485 (2005)