

# A MULTI-STAGE ALGORITHM FOR IMAGE DENOISING BASED ON PCA AND ADAPTIVE TV-REGULARIZATION

**Tran Dang Khoa Phan**

Faculty of Electronics and Telecommunication Engineering  
University of Science and Technology - The University of Danang  
Vietnam  
ptdkhoa@dut.udn.vn

Article history:

Received 03.09.2021, Accepted 25.11.2021

## Abstract

In this paper, we present an image denoising algorithm comprising three stages. In the first stage, Principal Component Analysis (PCA) is used to suppress the noise. PCA is applied to image blocks to characterize localized features and rare image patches. In the second stage, we use the Gaussian curvature to develop an adaptive total-variation-based (TV) denoising model to effectively remove visual artifacts and noise residual generated by the first stage. Finally, the denoised image is sharpened in order to enhance the contrast of the denoising result. Experimental results on natural images and computed tomography (CT) images demonstrated that the proposed algorithm yields denoising results better than competing algorithms in terms of both qualitative and quantitative aspects.

## Key words

Image denoising, total variation, Principal Component Analysis, split Bregman method, sharpening

## 1 Introduction

Imaging devices always produce some noise during the image acquisition process, and denoising is an essential step in most of image processing pipelines. Let  $u : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$  be a clean image, and let  $f$  be a noisy image (i.e., the degraded version of  $u$ ). The commonly used model of image degradation assumes that the observed image  $f$  is perturbed by a Gaussian noise of the form

$$f = u + \eta(0, \sigma^2), \quad (1)$$

where  $\eta(0, \sigma^2)$  denotes an additive white Gaussian noise with zero mean and a variance  $\sigma^2$ .

Reconstruction of the clean image  $u$  is an inverse problem, which is generally ill-posed. Adding a regulariza-

tion term, which provides a priori information, enforces a unique approximation of the solution. Rudin-Osher-Fatemi (ROF) model is one of the most famous works in this direction that uses the TV functional as a regularization energy [Rudin et al., 1992]. The main drawback of the ROF model is to produce the staircase effect, which appears at the extrema and at the boundary of an image. To improve the denoising performance of the ROF model, various types of weighted functions are incorporated to the TV regularization in order to control the amount of smoothing [Zhou and Li, 2013; Liu et al., 2016; Wang and Qi, 2018; Bazhanov et al., 2018; Lai et al., 2019; Thang et al., 2019; Pham et al., 2020].

Among filter-based algorithms, non-local means filtering (NLMF) was proven to be asymptotically optimal under a generic statistical image model [Buades et al., 2005b]. NLMF takes advantage of the redundancy of natural images. By this, some image blocks at different positions in a natural image have self-similar features, which means they have a certain correlation. NLMF estimates the denoised image by computing the average of all pixels in the image, weighted by the similarity between these pixels and the target pixel. The similarity is measured as a decreasing function of the weighted Euclidean distance [Buades et al., 2005a].

The image denoising problem can also be solved by exploiting the observation that natural images can be approximated by images with sparsity [Deledalle et al., 2011; Zhang et al., 2012; Ji et al., 2010; Gu et al., 2014; Gu et al., 2017]. In the work of Zhang et al. [Zhang et al., 2010], the authors presented an image denoising scheme comprising two stages. In the first stage, the noise is reduced in the PCA domain. In the second stage, the denoising procedure is iterated one more time to further enhance the denoising performance of the first stage. Experimental results showed that the proposed scheme

achieves very competitive denoising performance compared with state-of-the-art algorithms. However, this approach also produces visual artifacts due to thresholding in the transformed domain.

Recently, deep learning networks have achieved great success on tasks of image restoration such as [Zhang et al., 2017; Lefkimmiatis, 2018; Zhang et al., 2018]. These denoising models yield significant gain of denoising capability, especially for noisy images which have low SNR. Yet, it requires a large amount of resources to train and deploy models. This is a crucial limitation of these learning models, especially when they are integrated in the pre-processing block of a computer vision system.

In this paper, we investigate a denoising algorithm for images corrupted by Gaussian noise. Our contribution is threefold. First, we propose a multi-stage denoising algorithm which comprises a PCA-based denoiser, an adaptive TV-based denoiser and a sharpening procedure. The proposed algorithm can achieve the denoising performance better than that of widely used algorithms. Second, we propose an adaptive TV-based model using Gaussian curvature for the second stage that allows the model to adjust the edge-preservation and the noise reduction abilities automatically. Third, we modify the thresholding technique of the PCA-based denoiser to reduce resulting artifacts.

The rest of the paper is organized as follows. The proposed algorithm is presented in Sections 2. Section 3 discusses experimental results.

## 2 Proposed algorithm

The proposed algorithm comprises three stages as shown in Fig.1. The first stage yields the initial estimation of the denoised image by using a modified patch-based PCA method with the Stein thresholding. The choice of PCA-based denoising is based on the fact that it can produce denoising results competitive with the state-of-the-art denoising algorithms [Deledalle et al., 2011; Zhang et al., 2010]. Most of the noise is suppressed in this stage. Yet, visual artifacts are produced due to thresholding in the PCA domain and the influence of the noise. In the second stage, the output of the first stage will be further refined by using an adaptive TV-based denoiser to reduce visual artifacts while preserving edges. Since the TV-based denoiser reduces the contrast of the denoised image, a sharpening procedure is applied in the third stage to enhance the contrast of the image. Details of the proposed algorithm are described as follows.

### 2.1 First stage: Patch-based PCA denoising

The first stage of the proposed algorithm relies on the observation that natural images can be approximated by a sparse representation. PCA is applied to decompose an image into uncorrelated variables called principal components. In the PCA domain, the energy of the noiseless image concentrates on a subspace spanned by a small

subset of components, while the energy of the noise distributes uniformly over the whole space. Thus, the shrinkage approaches can be applied to achieve a sparse representation in the transformed domain.

In order to perform the PCA transformation, we need a set of training samples of the noisy image  $f$ . For this purpose, we use a sliding window with the size of  $L \times L$  to extract image patches, which are training examples for PCA. To reduce the computational complexity, the sliding window is applied with a stride  $S$ . Image patches inside a small image region are grouped into a block to characterize the image locally. Then, PCA is performed for each block. In this way, we can find resulting bases adapted to the local image regions as well as the whole image. A thresholding technique can be applied in the PCA domain to remove the noise. Details of the first stage are described as follows.

We denote by  $\mathbf{F}$  the sample matrix for a block, which has the following form:

$$\mathbf{F} = \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1n} \\ f_{21} & f_{22} & \cdots & f_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ f_{m1} & f_{m2} & \cdots & f_{mn} \end{bmatrix}. \quad (2)$$

where each row comprises  $n = L \times L$  pixel values of a patch. Each block has  $m$  patches. Let  $F_i = [f_{i1} \ f_{i2} \ \cdots \ f_{in}]$  be the  $i$ -th row of the sample matrix  $\mathbf{F}$ , which corresponds to a patch of the block.

The mean value of each column ( $j = 1, \dots, n$ ) is computed, then placed into the mean vector  $M$  as

$$M = [\mu_1 \ \mu_2 \ \cdots \ \mu_n]^T, \quad (3)$$

where

$$\mu_j = \frac{1}{m} \sum_{i=1}^m f_{ij}. \quad (4)$$

The sample matrix  $\mathbf{F}$  is centralized by subtracting the mean vector  $M$  from each row of  $\mathbf{F}$ :

$$\bar{\mathbf{F}} = \mathbf{F} - HM^T, \quad (5)$$

where  $H = [1 \ 1 \ \cdots \ 1]^T$  is an  $m \times 1$  column vector.

The covariance of the centralized sample matrix  $\bar{\mathbf{F}}$  is computed as

$$\Sigma = \frac{1}{m} \bar{\mathbf{F}} \bar{\mathbf{F}}^T. \quad (6)$$

According to the PCA method, the singular value decomposition of the covariance matrix  $\Sigma$  is computed. Let  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n \geq 0$  be the eigenvalues of  $\Sigma$  and  $X_1, X_2, \dots, X_n \in \mathbb{R}^n$  be the corresponding eigenvectors. Eigenvectors are also called principal components

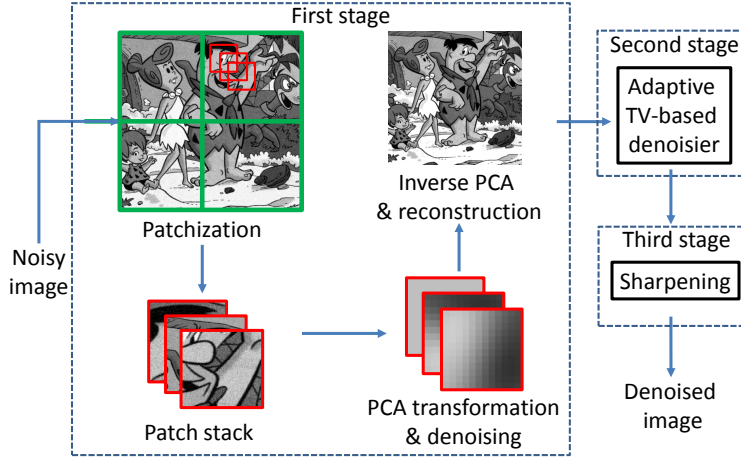


Figure 1. Flowchart of the proposed algorithm.

and form an orthogonal basis. A patch  $F_i$  of the block can be decomposed as

$$F_i = M^T + \sum_{k=1}^n \langle F_i - M^T, X_k^T \rangle X_k^T. \quad (7)$$

It is clear that for a given patch, only a small subset of components are relevant, but they may vary between different patches. Thus, we denoise patches according to the magnitude of the projection onto the PCA space instead of the variances of principal components. For this purpose, we use the Stein thresholding  $S_\lambda(\cdot)$  to reduce resulting artifacts as:

$$S_\lambda(t) = \max\left(1 - \frac{\lambda^2}{|t|^2}, 0\right) t, \quad (8)$$

where  $\lambda$  is a threshold parameter.

The thresholding operator is applied for patches of each block as

$$\hat{F}_i = M^T + \sum_{k=1}^n S_\lambda(\langle F_i - M^T, X_k^T \rangle) X_k^T. \quad (9)$$

The overlap of patches creates redundancy of patch estimation. Therefore, for all pixels, the final estimate is the uniform average of candidate estimates.

## 2.2 Second stage: Adaptive TV-based denoising

Most of the noise is suppressed in the first stage but the output image contains visually unpleasant artifacts and noise residual. The main reason is due to the fact that the covariance matrix  $\Sigma$  is corrupted by noise, causing the estimation bias of the PCA transformation. In addition, only few components represent an image block so that it is unable to explain rare patches. Therefore, we propose to use a second stage to further refine the denoising performance. Among denoising methods, the ROF model is effective at simultaneously maintaining edges

whilst removing noise in flat regions due to the TV-based regularizer. The ROF model is formulated as

$$\min_u \frac{1}{2} \int_{\Omega} (u - f)^2 dx + \alpha \int_{\Omega} |\nabla u| dx, \quad (10)$$

where  $\nabla$  stands for the gradient operator;  $\alpha$  is a positive regularization parameter;  $|\cdot|$  denotes the  $L^1$ -norm. The first term in (10) measures the fidelity to the noisy image  $f$ . The second one, called the TV of the image  $u$ , is the regularization term. The TV term allows the existence of edges and is also a smoothing term.

The ROF model is not adaptive since the regularization parameter is the same for all pixels. To improve the denoising performance of the ROF model, we modify it by introducing a weighted function into the TV-based regularizer in order to adjust the strength of smoothing for each pixel of the image.

To this end, we propose an adaptive TV-based denoising model as

$$\min_u \frac{1}{2} \int_{\Omega} (u - f)^2 dx + \int_{\Omega} \alpha(x) |\nabla u| dx, \quad (11)$$

where  $\alpha(\cdot)$  is a weighted function, which is expressed by

$$\alpha(x) = \frac{\alpha_0}{1 + \left(\frac{\Gamma_{G_\theta * f}}{\gamma}\right)^2}, \quad (12)$$

where  $\alpha_0$  - an initial regularization parameter;  $\gamma$  - a contrast parameter;  $\Gamma_{G_\theta * f}$  is the Gaussian curvature of the smoothed image of  $f$ , which is defined as

$$\Gamma_v = \frac{v_{xx}v_{yy} - v_{xy}v_{yx}}{(v_x^2 + v_y^2 + 1)^2}, \quad (13)$$

where  $v_x, v_y$  and  $v_{xx}, v_{yy}, v_{xy}, v_{yx}$  denote the first-order and the second-order derivatives of  $v$  with respect to  $x, y$ ;

$G_\theta(x) = \frac{1}{\theta\sqrt{2\pi}} \exp\left(-\frac{|x|^2}{2\theta^2}\right)$  denoting the Gaussian filter with a standard deviation  $\theta$ . The Gaussian filter is used to reduce the noise, which affects the computation of the Gaussian curvature in (13). As can be seen from (11), (12), and (13), the proposed model has the strength of smoothing larger for homogeneous regions compared with that for edges. This property allows the model preserve edges, producing better denoising results compared to the original ROF model.

We use the split Bregman method [Goldstein and Osher, 2009] to solve the problem (11). According to the split Bregman method, we minimize the following discrete unconstrained problem of (11):

$$\min_{u,d} \frac{\lambda}{2} \|u - f\|_2^2 + \alpha(x) \|d\|_1 + \frac{\mu}{2} \|d - \nabla u - b\|_2^2, \quad (14)$$

where  $d$  is an auxiliary variable;  $b$  is a variable chosen through each iteration;  $\mu$  is a positive penalty parameter.

The problem (14) is solved by iteratively minimizing with respect to  $u$  and  $d$  separately. We obtain the  $u$  and the  $d$  subproblems as

$$u^{k+1} = \min_u \frac{\lambda}{2} \|u - f\|_2^2 + \frac{\mu}{2} \|d^k - \nabla u - b^k\|_2^2. \quad (15)$$

$$d^{k+1} = \min_d \alpha(x) \|d\|_1 + \frac{\mu}{2} \|d - \nabla u^{k+1} - b^k\|_2^2. \quad (16)$$

The  $u$  subproblem can be solved via the optimality condition using the Fourier transform as

$$u^{k+1} = \mathcal{F}^{-1} \left( \frac{\mathcal{F}(\lambda f - \mu \operatorname{div}(d^k - b^k))}{\lambda I - 2\mu \left( \cos \frac{2\pi r}{M} + \cos \frac{2\pi s}{N} - 2 \right)} \right), \quad (17)$$

where  $r \in [0, M-1]$  and  $s \in [0, N-1]$  are the frequencies in the discrete frequency domain;  $\mathcal{F}$  and  $\mathcal{F}^{-1}$  stand for the forward and the inverse Fourier transforms; “—” stands for the point-wise division of matrices.

We solve the  $d$  subproblem by using the shrinkage operator [Setzer, 2011] as

$$d^{k+1} = \frac{\nabla u^{k+1} + b^k}{|\nabla u^{k+1} + b^k|} \max \left( |\nabla u^{k+1} + b^k| - \frac{\alpha(x)}{\mu}, 0 \right). \quad (18)$$

The variable  $b$  is updated after each iteration

$$b^{k+1} = b^k + \nabla u^{k+1} - d^{k+1}. \quad (19)$$

By iteratively computing the sequence of equations (17), (18) and (19), we can find the solution of the problem (14). Note that instead of computing the weighted function  $\alpha(\cdot)$  only once using the observed noisy image  $f$ , we iteratively refine it using the denoised image at each iteration. This step ensures that the weighted function is updated with the denoised image, resulting in more pleasant denoising results.

### 2.3 Third stage: Sharpening

Although the second stage can reduce noise while preserving edges, the TV term causes the loss of contrast in the denoised image [Strong and Chan, 2003; Li et al., 2014]. Since the noise and the artifacts are suppressed significantly after the first two stages, we can utilize the unsharp masking technique to enhance the denoising result in terms of contrast. Without the second stage, the sharpening procedure could not enhance the denoising results due to the noise and artifacts.

Let  $u(x, y)$  be the original image. Unsharp masking is created by subtracting the blurred image from the original image as

$$g_{mask}(x, y) = u(x, y) - G_\rho * u(x, y), \quad (20)$$

where  $G_\rho$  is a Gaussian filter with the parameter  $\rho$ .

Then a weighted portion of the mask is added to the original image to obtain a sharper image

$$g(x, y) = u(x, y) + \beta \cdot g_{mask}(x, y), \quad (21)$$

where  $\beta$  is a scaling constant.

## 3 Experimental results

In this section, we compare the proposed algorithm (PCATVS) with related algorithms, including ROF [Rudin et al., 1992], PLPCA [Zhang et al., 2010], and NLMF [Buades et al., 2005a]. The original source codes of competing algorithms are obtained, and we use the default parameters. For PCATVS, the parameter values are fixed as:  $L = 7$ ,  $S = 3$ ,  $\lambda = 2.5$ ,  $\alpha_0 = 1$ ,  $\gamma = 0.9$ ,  $\theta = 0.5$ ,  $\mu = 5$ ,  $\rho = 1.5$ ,  $\beta = 0.8$ .

We perform experiments on 14 widely used images (Fig. 2). Images are perturbed by additive white Gaussian noise with zero mean and a standard deviation  $\sigma$ . We consider three noise levels, including  $\sigma = 10, 20$  and  $40$ . We evaluate the performance of algorithms by using three criteria: PSNR, SSIM and visual quality.



Figure 2. Testing images. First row: Barbara, Boat, Cameraman, Couple, Fingerprint, Flinstones, Hill; second row: House, Jetplane, Lena, Man, Mandril, Monarch, Peppers.

We, first, demonstrate the denoising performance after each stage of the proposed algorithm. Figure 3 shows the denoised images by each stage for the image *Monarch*.

The quantitative results are also shown for each denoising result. It can be seen that the first stage produces an image with a large amount of visual artifacts. Although the second stage nearly does not improve the PSNR measure, the SSIM value is gained significantly. One can see from the Fig. 3d that most of visual artifacts are reduced by the second stage while maintaining edges. Yet, its denoised result has lower contrast compared with that of the first stage. The sharpening stage significantly improves the denoising result in terms of both quantitative and qualitative criteria.

Table 1 reports the PSNR and SSIM results of different methods. From Table 1 we have the following observations. First, on all noise levels, PCATVS achieves the best results for almost all images. By PSNR, the proposed algorithm surpasses PLPCA, ROF, and NLMF by 0.42dB, 1.74db, and 1.3dB, respectively; by SSIM, PCATVS outperforms them by 0.0131, 0.0439, and 0.0448, respectively. Second, PCATVS outperforms competing algorithms by a large margin on images such as *Barbara*. This is because the image *Barbara* is dominated by a large amount of repetitive structures, which can be effectively exploited by PCA-based method. Besides, the second stage of the proposed algorithm enhances the denoising performance of the PCA-based stage by reducing visual artifacts.

Figure 4 shows denoised images of *Barbara* by different methods. The original image is corrupted by a Gaussian noise with the standard deviation  $\sigma = 10$ . A portion of images is enlarged for visual evaluation. From Fig. 4, one can see that PCATVS and NLMF yield denoising results, which are more visually pleasing than that of PLPCA and ROF. Compared with PCATVS, low-contrast textures are smoothed out by NLMF. PLPCA produces visual artifacts, whereas ROF generates blocky image regions and reduces the contrast of the image.

Figures 5 and 6 demonstrate denoised images of *Jetplane* and *Monarch* for the noise levels  $\sigma = 20$  and 40, respectively. It can be seen that features, which are characterized by long and small edges such as characters, are well preserved by PCATVS. NLMF produces noisy pixels along edges. As the noise level increases, PLPCA and NLMF generate more visual artifacts, creating visually unpleasant results. The denoised images by ROF suffers the staircase effect.

We further evaluate methods on a computed tomography (CT) image. CT is one of the major tools in medical diagnostics. The quality of CT images depends on the amount of X-ray. As increasing the X-ray dose, the quality of CT images becomes better. However, high-dose of X-ray makes negative impacts on patients. Due to the statistical uncertainty in physical measurements, low-dose CT images are noisy and difficult to observe [Erofeeva et al., 2018]. Thus, denoising low-dose CT images improves the quality of images without increasing the X-ray dose. CT images are often corrupted by speckle noise [Duan et al., 2016; Duan et al., 2015]. Fig. 7 shows denoising results by different methods on the

CT image corrupted by speckle noise. It can be seen that artifacts generated by PLPCA and ROF are significant, whereas NLMF produce over-smooth image. The proposed algorithm can remove noise while preserving edges and fine details well.

#### 4 Conclusion

In this paper, we have presented a denoising algorithm for images corrupted by Gaussian noise. The proposed algorithm consists of three stages. First, the noise is suppressed in the PCA domain using the Stein thresholding. The output of the first stage is further refined by a TV-based denoising model, which is adaptive with the reconstructed image. Finally, the denoised image is sharpened using the unsharp masking technique. Experimental results demonstrated that the proposed algorithm outperforms competing algorithm in terms of both qualitative and quantitative aspects, especially for images which have a large amount of repetitive structures. Experimental results on CT images showed the flexibility of the proposed algorithm.

#### Acknowledgments

This work was supported by The Murata Science Foundation and The University of Danang, University of Science and Technology, code number of Project: T2020-02-03MSF.

#### References

- Bazhanov, P., Kotina, E., Ovsyannikov, D., and Ploskikh, V. (2018). Optimization algorithm of the velocity field determining in image processing. *Cybernetics and Physics*, **7** (4), pp. 174–181.
- Buades, A., Coll, B., and Morel, J.-M. (2005a). A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, IEEE, pp. 60–65.
- Buades, A., Coll, B., and Morel, J.-M. (2005b). A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, **4** (2), pp. 490–530.
- Deledalle, C.-A., Salmon, J., Dalalyan, A. S., et al. (2011). Image denoising with patch based pca: local versus global. In *BMVC*, vol. 81, pp. 425–455.
- Duan, J., Lu, W., Tench, C., Gottlob, I., Proudlock, F., Samani, N. N., and Bai, L. (2016). Denoising optical coherence tomography using second order total generalized variation decomposition. *Biomedical Signal Processing and Control*, **24**, pp. 120–127.
- Duan, J., Tench, C., Gottlob, I., Proudlock, F., and Bai, L. (2015). New variational image decomposition model for simultaneously denoising and segmenting optical coherence tomography images. *Physics in Medicine & Biology*, **60** (22), pp. 8901.
- Erofeeva, V., Granichin, O., and Senin, I. (2018). Modifying the physical process of ultrasound tomography

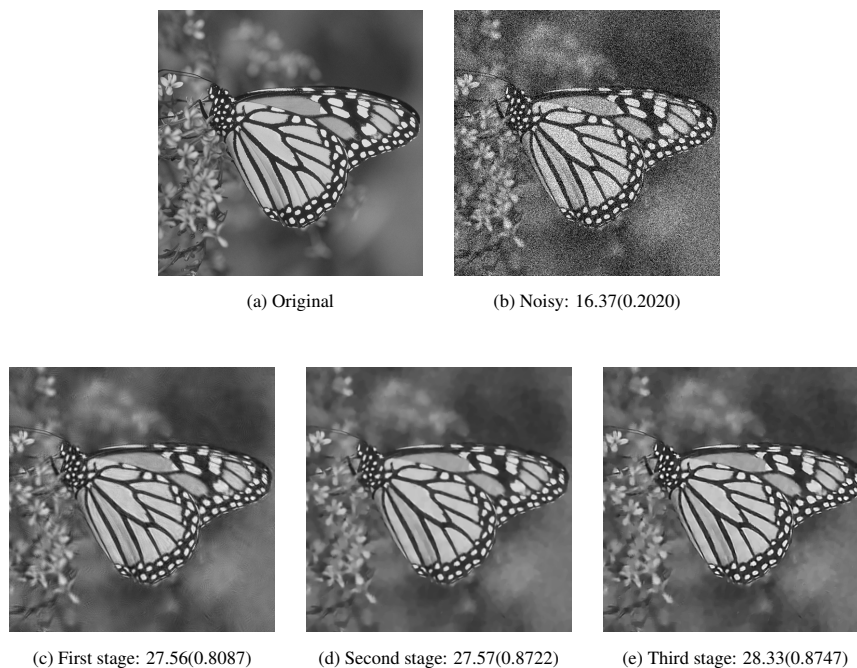


Figure 3. Denoised images by each stage of the proposed algorithm for the image *Monarch*.

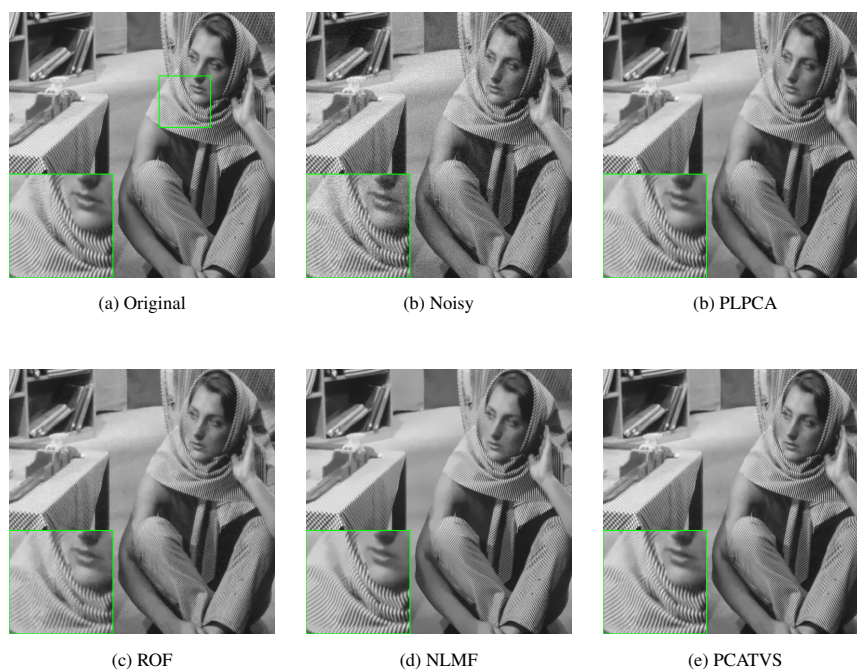


Figure 4. Denoised images of the image *Barbara* with the noise level  $\sigma = 10$ .

scanning through compressive sensing. *Cybernetics and Physics*, **7** (2), pp. 66–71.

Goldstein, T. and Osher, S. (2009). The split bregman method for  $l_1$ -regularized problems. *SIAM journal on imaging sciences*, **2** (2), pp. 323–343.

Gu, S., Xie, Q., Meng, D., Zuo, W., Feng, X., and Zhang, L. (2017). Weighted nuclear norm minimization and its applications to low level vision. *International journal of computer vision*, **121** (2), pp. 183–208.

Gu, S., Zhang, L., Zuo, W., and Feng, X. (2014).

Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2862–2869.

Ji, H., Liu, C., Shen, Z., and Xu, Y. (2010). Robust video denoising using low rank matrix completion. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 1791–1798.

Lai, R., Mo, Y., Liu, Z., and Guan, J. (2019). Local and nonlocal steering kernel weighted total variation model

Table 1. PSNR and SSIM results of different methods

| Methods       | PLPCA                  | ROF           | NLMF                   | PCATVS               |
|---------------|------------------------|---------------|------------------------|----------------------|
| $\sigma = 10$ |                        |               |                        |                      |
| Barbara       | 33.59(0.9225)          | 30.71(0.8522) | 33.24(0.9178)          | <b>34.70(0.9354)</b> |
| Boat          | 33.48(0.8817)          | 32.24(0.8617) | 32.28(0.8519)          | <b>33.75(0.8864)</b> |
| Cameraman     | 33.33(0.9138)          | 31.49(0.8937) | 32.57(0.8977)          | <b>33.61(0.9187)</b> |
| Couple        | 33.45(0.8959)          | 32.07(0.8737) | 32.33(0.8686)          | <b>33.67(0.9009)</b> |
| Fingerprint   | 32.24(0.9659)          | 31.07(0.9550) | 30.21(0.9488)          | <b>32.44(0.9674)</b> |
| Flinstones    | 32.02(0.9052)          | 31.12(0.8989) | 31.65(0.8939)          | <b>32.16(0.9061)</b> |
| Hill          | 31.49(0.8773)          | 30.72(0.8532) | 30.49(0.8239)          | <b>31.74(0.8800)</b> |
| House         | 35.08(0.8971)          | 30.72(0.8532) | 34.87(0.8907)          | <b>35.85(0.9066)</b> |
| Jetplane      | 35.46(0.9223)          | 34.45(0.9177) | 34.12(0.9170)          | <b>35.80(0.9289)</b> |
| Lena          | 35.20(0.9028)          | 34.17(0.8900) | 34.41(0.8965)          | <b>35.64(0.9109)</b> |
| Man           | 33.42(0.8953)          | 32.64(0.8812) | 32.48(0.8728)          | <b>33.71(0.9025)</b> |
| Mandrill      | 33.33(0.9308)          | 31.09(0.8990) | 30.74(0.8912)          | <b>33.40(0.9328)</b> |
| Monarch       | 35.12(0.9385)          | 34.20(0.9278) | 34.59( <b>0.9461</b> ) | <b>35.45(0.9457)</b> |
| Peppers       | 33.90(0.9179)          | 32.95(0.9111) | 33.31(0.9138)          | <b>34.21(0.9237)</b> |
| AVERAGE       | 33.65(0.9119)          | 32.12(0.8906) | 32.66(0.8950)          | <b>34.01(0.9176)</b> |
| $\sigma = 20$ |                        |               |                        |                      |
| Barbara       | 29.70(0.8539)          | 26.82(0.7526) | 29.47(0.8468)          | <b>31.03(0.8837)</b> |
| Boat          | 30.01(0.7961)          | 29.07(0.7758) | 29.00(0.7654)          | <b>30.34(0.8073)</b> |
| Cameraman     | 29.25(0.8371)          | 28.11(0.8233) | 29.14(0.8259)          | <b>30.30(0.8665)</b> |
| Couple        | 29.88(0.8116)          | 28.65(0.7765) | 28.49(0.7658)          | <b>30.01(0.8180)</b> |
| Fingerprint   | 28.25(0.9154)          | 26.68(0.8852) | 27.16(0.8869)          | <b>28.41(0.9181)</b> |
| Flinstones    | 28.33(0.8478)          | 26.99(0.8339) | 28.07(0.8499)          | <b>28.56(0.8512)</b> |
| Hill          | 28.06(0.7583)          | 27.60(0.7366) | 27.10(0.7084)          | <b>28.22(0.7608)</b> |
| House         | 32.05(0.8437)          | 27.60(0.7366) | 31.86(0.8398)          | <b>32.65(0.8558)</b> |
| Jetplane      | 31.96(0.8759)          | 31.15(0.8701) | 30.96(0.8589)          | <b>32.25(0.8891)</b> |
| Lena          | 32.12(0.8506)          | 31.16(0.8332) | 31.23(0.8338)          | <b>32.53(0.8636)</b> |
| Man           | 29.93(0.8074)          | 29.50(0.7951) | 29.09(0.7773)          | <b>30.17(0.8184)</b> |
| Mandrill      | 28.78(0.8306)          | 27.05(0.7747) | 27.00(0.7637)          | <b>28.84(0.8344)</b> |
| Monarch       | 31.26(0.9007)          | 30.81(0.8899) | 30.87(0.8966)          | <b>31.64(0.9177)</b> |
| Peppers       | 30.13(0.8636)          | 29.43(0.8533) | 29.73(0.8505)          | <b>30.55(0.8749)</b> |
| AVERAGE       | 29.98(0.8423)          | 28.62(0.8098) | 29.23(0.8192)          | <b>30.39(0.8543)</b> |
| $\sigma = 40$ |                        |               |                        |                      |
| Barbara       | 26.34(0.7572)          | 23.66(0.6483) | 25.32(0.7060)          | <b>29.33(0.8121)</b> |
| Boat          | 26.63(0.6914)          | 26.00(0.6726) | 25.48(0.6358)          | <b>26.77(0.7039)</b> |
| Cameraman     | 25.11(0.7283)          | 24.57(0.7307) | 25.13(0.6959)          | <b>25.23(0.7537)</b> |
| Couple        | <b>26.34(0.6862)</b>   | 25.56(0.6529) | 24.78(0.6035)          | <b>26.33(0.6934)</b> |
| Fingerprint   | 24.72( <b>0.8270</b> ) | 22.53(0.7393) | 23.01(0.7514)          | <b>24.77(0.8253)</b> |
| Flinstones    | 23.63(0.7371)          | 22.55(0.7209) | 23.70(0.7321)          | <b>23.94(0.7540)</b> |
| Hill          | 25.37( <b>0.6289</b> ) | 25.03(0.6014) | 24.27(0.5659)          | <b>25.43(0.6283)</b> |
| House         | 28.85(0.7822)          | 25.03(0.6014) | 27.41(0.7267)          | <b>29.43(0.8159)</b> |
| Jetplane      | 28.27(0.8226)          | 27.72(0.8194) | 27.50(0.7603)          | <b>28.48(0.8413)</b> |
| Lena          | 28.92(0.7806)          | 28.20(0.7679) | 27.70(0.7188)          | <b>29.33(0.8121)</b> |
| Man           | 26.84(0.6953)          | 26.75(0.6979) | 25.99(0.6482)          | <b>26.99(0.7107)</b> |
| Mandrill      | 24.84(0.6596)          | 23.84(0.5896) | 23.41(0.5613)          | <b>24.99(0.6736)</b> |
| Monarch       | 27.19(0.8312)          | 27.12(0.8439) | 27.26(0.7867)          | <b>28.42(0.8790)</b> |
| Peppers       | 25.96(0.7675)          | 25.64(0.7677) | 25.58(0.7278)          | <b>26.50(0.7929)</b> |
| AVERAGE       | 26.36(0.7425)          | 25.30(0.7039) | 25.47(0.6872)          | <b>26.85(0.7640)</b> |

for image denoising. *Symmetry*, **11** (3), pp. 329.  
 Lefkimmatis, S. (2018). Universal denoising networks:  
 a novel cnn architecture for image denoising. In *Pro-*

*ceedings of the IEEE conference on computer vision  
 and pattern recognition*, pp. 3204–3213.



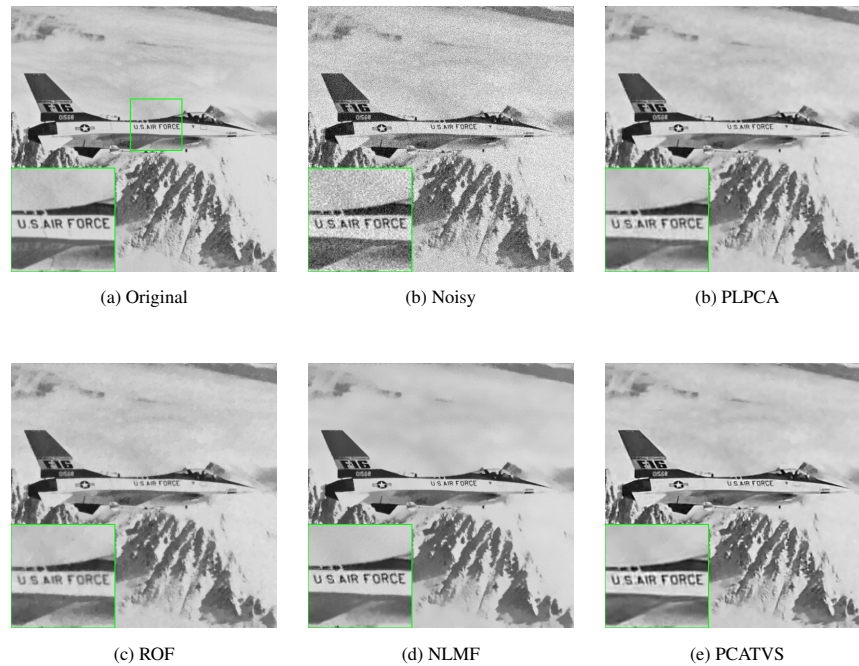


Figure 5. Denoised images of the image *Jetplane* with the noise level  $\sigma = 20$ .

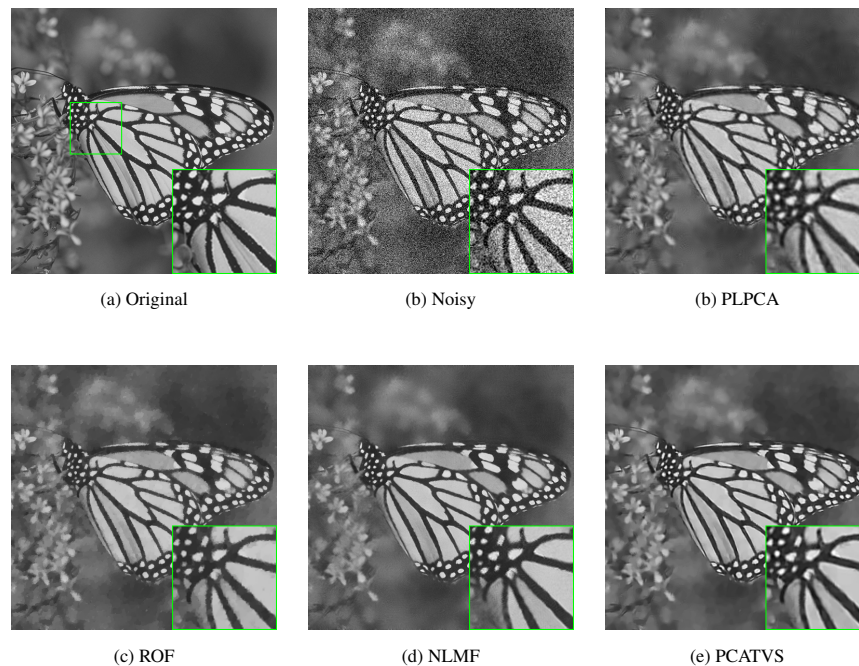


Figure 6. Denoised images of the image *Monarch* with the noise level  $\sigma = 40$ .

Li, L., Feng, W., and Zhang, J. (2014). Contrast enhancement based single image dehazing via tv-l1 minimization. In *2014 IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, pp. 1–6.

Liu, H., Gu, J., Meng, M. Q.-H., and Lu, W.-S. (2016). Fast weighted total variation regularization algorithm for blur identification and image restoration. *IEEE Access*, **4**, pp. 6792–6801.

Pham, C. T., Tran, T. T. T., and Gamard, G. (2020). An efficient total variation minimization method for image

restoration. *Informatica*, **31** (3), pp. 539–560.

Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, **60** (1-4), pp. 259–268.

Setzer, S. (2011). Operator splittings, bregman methods and frame shrinkage in image processing. *International Journal of Computer Vision*, **92** (3), pp. 265–280.

Strong, D. and Chan, T. (2003). Edge-preserving and scale-dependent properties of total variation regular-



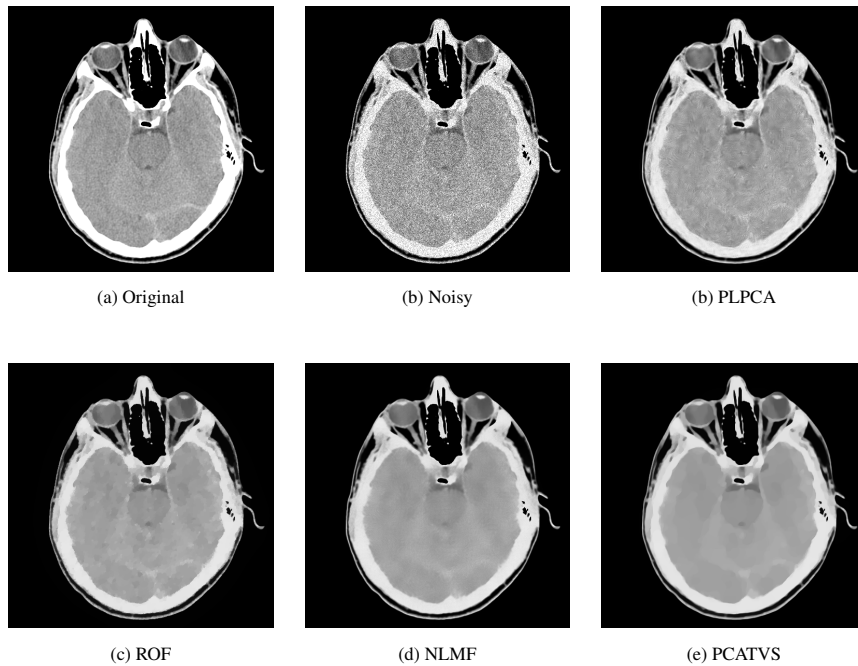


Figure 7. Denoised images by different methods on the CT image corrupted by speckle noise.

- ization. *Inverse problems*, **19**(6), pp. S165.
- Thang, P. C., Thao, T. T. T., Khoa, P. T. D., Sang, D. V., Tuan, P. M., and Hieu, N. M. (2019). An adaptive algorithm for restoring image corrupted by mixed noise. *Cybernetics and Physics*, **8**(2), pp. 73–82.
- Wang, Y. and Qi, Z. (2018). A new adaptive-weighted total variation sparse-view computed tomography image reconstruction with local improved gradient information. *Journal of X-ray Science and Technology*, **26**(6), pp. 957–975.
- Zhang, D., Hu, Y., Ye, J., Li, X., and He, X. (2012). Matrix completion by truncated nuclear norm regularization. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 2192–2199.
- Zhang, F., Cai, N., Wu, J., Cen, G., Wang, H., and Chen, X. (2018). Image denoising method based on a deep convolution neural network. *IET Image Processing*, **12**(4), pp. 485–493.
- Zhang, K., Zuo, W., Chen, Y., Meng, D., and Zhang, L. (2017). Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, **26**(7), pp. 3142–3155.
- Zhang, L., Dong, W., Zhang, D., and Shi, G. (2010). Two-stage image denoising by principal component analysis with local pixel grouping. *Pattern recognition*, **43**(4), pp. 1531–1549.
- Zhou, W. and Li, Q. (2013). Adaptive total variation regularization based scheme for poisson noise removal. *Mathematical Methods in the Applied Sciences*, **36**(3), pp. 290–299.