International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.6, No.6 (2013), pp.41-50
http://dx.doi.org/10.14257/ijsip.2013.6.6.04

BM3D Image Denoising Algorithm with Adaptive Distance Hard-threshold

Li Dai, Yousai Zhang and Yuanjiang Li

Institute of Electronic and Information, Jiangsu University of Science and Technology, Zhenjiang 212003, China
daidai234@126.com

Abstract

Block-matching and 3D filtering (BM3D) denoising algorithm [1] proposed recently has a problem of computational burden especially for low noise level and a sharp performance drop for high noise level. To solve it, an improved version of BM3D is proposed. The solution combines the digital image characteristic with added noise pollution levels, and adaptively selects block-matching threshold in grouping stage. Experimental results demonstrate it outperforms not only in terms of objective criteria of PSNR and running time, but also in visual quality.

Keywords: *Image denoising, BM3D, adaptive threshold, PSNR*

1. Introduction

Recently, a new image denoising algorithm, block-matching and 3D filtering (BM3D) which is considered to be current state-of-the-art is introduced in [1]. The core idea is grouping and collaborative filtering. The noisy image is divided into blocks in a sliding manner. Each block is processed by searching similar blocks with fixed threshold. These matched blocks are stacked together to form a 3D array. Because of the similarity in a group, the data exhibits high level of correlation. Collaborative filtering and weighted averaging are then performed. These steps turn out to achieve an excellent denoising result.

As the algorithm uses fixed threshold in grouping step, however, when noise level is low, the maximum block-matching distance is overlarge, resulting in too much time consumption, as well as lots of unnecessary similar block. On the other hand, for high noise level, the threshold value is too small, therefore BM3D can't get enough similar blocks that leads to a sharp drop in denoising result and that "block effect" appears [2]. The authors of [1] put forward some improvement, which improved the denoising effect to a certain extent. But they destroyed the algorithm in the consistency of mathematical expression and the continuity of the algorithm [2]. Dabov, *et al.*, also developed other methods [3-4]. In the case of lower noise, [3] had a better denoising effect; but for relatively larger noise intensity, the denoising performance was even worse than the original. Paper [4] really promoted the quality of denoised image, but it caused a big problem of the computational burden as the time was about fifty times than before. Besides, when noise deviation reached 40, a few modifications are made in [5]. These changes achieved better denoising performance, nevertheless they failed to make the algorithm continuous. BM3D based on adaptive threshold was proposed in 2012 which avoided modifying many parameters and was proved to be effective [6-7].

In this paper, an improved version of BM3D based on adaptive distance hard-threshold is proposed whose hard-threshold is set differently according to the features of image (the ratio of the mean and standard deviation and the estimated noise intensity). Experimental results show that it is more effective than the original BM3D in terms of both PSNR and visual quality. What's more, when the noise is low, our method reduces the execution time. On the other hand, although the denoising effect is slightly worse than BM3D-SAPCA [4], our proposed method requires much less time consumption.

ISSN: 2005-4254 IJSIP

2. Proposed Technique

Firstly we recall the grouping processing in step one of BM3D. For a noisy image z which is denoted as

$$z(x) = y(x) + \eta(x) \tag{1}$$

where x is a 2D spatial coordinate that belongs to image domain, y is the true image, and η is i.i.d. zero-mean Gaussian noise with variance σ^2. In the processing of block-matching, the block-distance is measured by using a coarse prefiltering.

$$d(Z_{x_R}, Z_x) = \frac{\| \gamma'(\mathrm{T}_{2D}^{ht}(Z_{x_R})) - \gamma'(\mathrm{T}_{2D}^{ht}(Z_x)) \|_2^2}{(N_1^{ht})^2} \tag{2}$$

where $\| \bullet \|_2$ is the ℓ^2-norm, γ' denotes the hard-threshold operator with threshold $\lambda_{2D}\sigma$, T_{2D}^{ht} is the normalized 2D linear transform and the blocks Z_{x_R} and Z_x are respectively located at x_R and $x \in X$ in z. With the formula (2) of d-distance, a set $S_{x_R}^{ht}$ can be obtained whose elements are the coordinates of the blocks that are similar to Z_{x_R},

$$S_{x_R}^{ht} = \{x \in X : d(Z_{x_R}, Z_x) \le \tau_{match}^{ht}\} \tag{3}$$

where τ_{match}^{ht} is the maximum d-distance for which some blocks are considered similar to the currently being processed one. The fixed parameter τ_{match}^{ht} is chosen from deterministic speculations about the acceptable value of the ideal difference, mainly ignoring the noisy components of the image [1].

In general, the similar degree of two images or image blocks is judged mainly from the following two aspects: similarity based on pixels and structure similarity. In the procedure of block-matching, a single pixel gray value, as well as its neighborhood, is compared with another according to the gray distribution. When image details or edges are strong, however, only to measure similarity with gray scale distribution will cause erroneous grouping. In this paper, similarity based on pixels is combined with structure similarity [8] to find the most similar blocks.

Suppose a pixel is denoted as $v(r,c)$, whose horizontal and vertical gradients as follows respectively,

$$v_x = v(r,c) - v(r,c-1) \tag{4}$$

$$v_y = v(r,c) - v(r-1,c) \tag{5}$$

Then a symmetric, nonnegative matrix of the image of size $M \times N$ can be obtained,

$$D = \begin{pmatrix} \sum_{k=1}^{M\times N} v_x(k)^2 & \sum_{k=1}^{M\times N} v_x(k)v_y(k) \\ \sum_{k=1}^{M\times N} v_x(k)v_y(k) & \sum_{k=1}^{M\times N} v_y(k)^2 \end{pmatrix} = \begin{pmatrix} E & F \\ F & G \end{pmatrix} \tag{6}$$

D has two eigenvalues. The bigger value is gradient module, denoted as λ, while its eigenvector is gradient direction, denoted as $(\cos\theta, \sin\theta)$.

$$\lambda = \frac{(E+G) \pm \sqrt{(E-G)^2 + 4F^2}}{2} \tag{7}$$

$$\cos\theta = \frac{\lambda - F}{\sqrt{(\lambda - F)^2 + G^2}} \tag{8}$$

$$\sin\theta = \frac{G}{\sqrt{(\lambda - F)^2 + G^2}} \tag{9}$$

Gradient vector of the reference image (g_x, g_y) is defined as,

$$g_x = \lambda \cos\theta \tag{10}$$

$$g_y = \lambda \sin\theta \tag{11}$$

Using the gradient vector (g_x, g_y), structure similarity (SS) of two images can be calculated according to the following formula,

$$SS(i,j) = \sqrt{(g_x(i) - g_x(j))^2 + (g_y(i) - g_y(j))^2 + (\mu(i) - \mu(j))^2} \tag{12}$$

where μ is average pixel value of the reference image. The smaller the SS value is, the more similar two images are. When two images are exactly the same, the SS values 0. For two image blocks, using the formula (12), the similarity of them can be obtained.

The following subsections present the steps of the proposed method.

1. Divide the noisy image into blocks. Given the reference block Z_{x_R}, estimate the added noise $\sigma(n)$, then compute the ratio of its mean and standard deviation $\frac{\mu(n)}{\delta(n)}$.

2. For all candidate blocks Z_x (Z_{x_R} excluded) of the noisy image, find out the relationship between $SS(Z_{x_R}, Z_x)$ and $d(Z_{x_R}, Z_x)$. Figure 1 shows the results. With the different block-matching distances, SS values are changing. From the figure, one may note that SS value achieves its minimum when the block-matching distance reaches a certain value (assumed to be $d'(n)$). The smallest SS value embodies the best similarity between two image blocks in structure. And from the perspective based on pixels, all distance values less than $d'(n)$ result from those candidate blocks which are most similar to the reference one. In view of the above analysis, select the block-matching distance $d'(n)$ whose corresponding $SS'(n)$ is the smallest one.

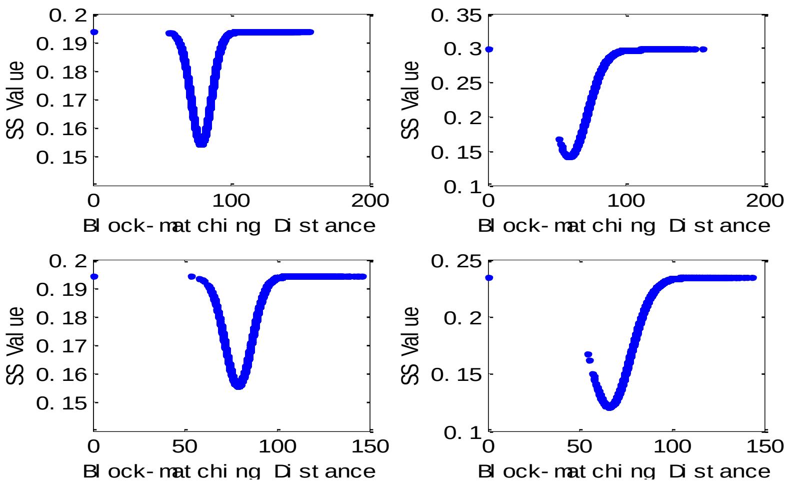
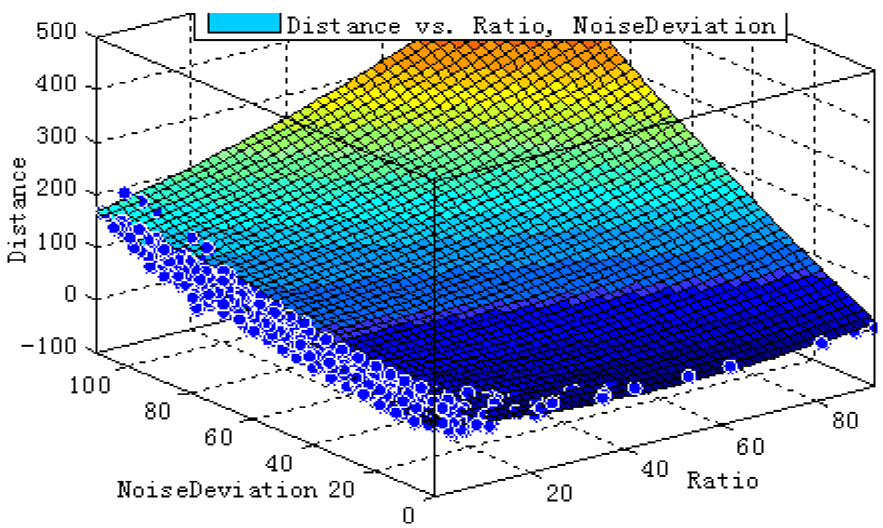


Figure 1. Relationship between SS Value and Block-matching Distance

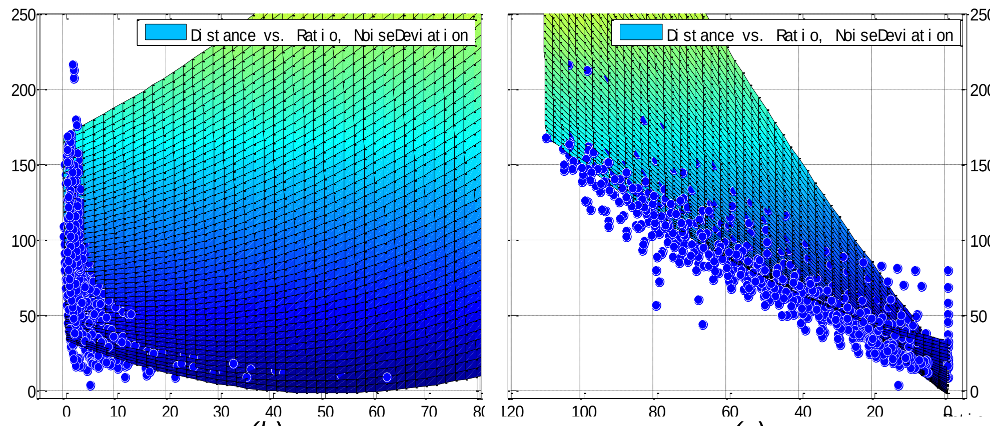
3. Respectively to different image blocks of the same standard images polluted by the same noise level, different noise standard deviation of the same image, and different standard images, repeat the above operations cyclically.

4. Build a function of the relationship between $d'(n)$, $\frac{\mu(n)}{\delta(n)}$ and $\sigma(n)$: $d'(n) = f\left(\frac{\mu(n)}{\delta(n)}, \sigma(n)\right)$. Then fit the data obtained in the above steps. The selection of degree of $\frac{\mu(n)}{\delta(n)}$ and $\sigma(n)$ is empirical. Here we chose quadratic polynomial.

$$
\begin{aligned}
d'(n) = \; & 33 - 1.391 \cdot \frac{\mu(n)}{\delta(n)} + 0.48 \cdot \sigma(n) + 0.02764 \cdot \frac{\mu(n)}{\delta(n)} \cdot \sigma(n) \\
& + 0.0136 \cdot \left(\frac{\mu(n)}{\delta(n)}\right)^2 + 0.0078 \cdot (\sigma(n))^2
\end{aligned}
\tag{13}
$$



(a)



(b) (c)

Figure 2. The Results of Data Fitting about the Relationship between $d'(n)$, $\mu(n)/\delta(n)$ and $\sigma(n)$. (b) and (c) Describe the Relationship Respectively: $d'(n)$-$\mu(n)/\delta(n)$, $d'(n)$-$\sigma(n)$

Figure 2 shows the results of data fitting about the relationship between block-matching threshold, the ratio of block's mean and standard deviation and noise level. In the algorithm of BM3D, for an image to be denoised, the formula of block-matching threshold can be written as,

$$\tau_{match}^{ht} = 33 - 1.391\frac{\mu}{\delta} + 0.48\sigma + 0.02764\frac{\mu}{\delta}\cdot\sigma + 0.0136(\frac{\mu}{\delta})^2 + 0.0078\sigma^2 \tag{14}$$

where μ, δ and σ are values of the whole image's mean, standard deviation and noise level respectively.

3. Experimental Results

The performance of the proposed image denoising approach is compared to the original BM3D, BM3D-SAPCA and improved BM3D in [5] respectively. These algorithms are applied on the images from the BM3D website. All experiments are performed on a Core(TM) 2 2.00 GHz Duo CPU and 2 GB Rom computer.

In Figures 3 and 4, compared with the original BM3D, the proposed method introduces less artifacts, as well as reserves more details. An objective PSNR evaluation is presented in Table 1. Obviously, our method significantly increases the PSNR values especially in the case of high noise level. Table 2 gives the running time comparison between the original method and the proposed one. For the latter, due to adopting adaptive threshold, the maximum distance of block-matching is relatively small in a small noise, and the corresponding execution time is short. When the noise level is high, the maximum distance increases in order to get enough similar blocks, which makes the time complexity increased. However, in Table 3, one can see that the processing time is much less than the one by BM3D-SAPCA, although the quality of denoised images by proposed method is slightly worse.

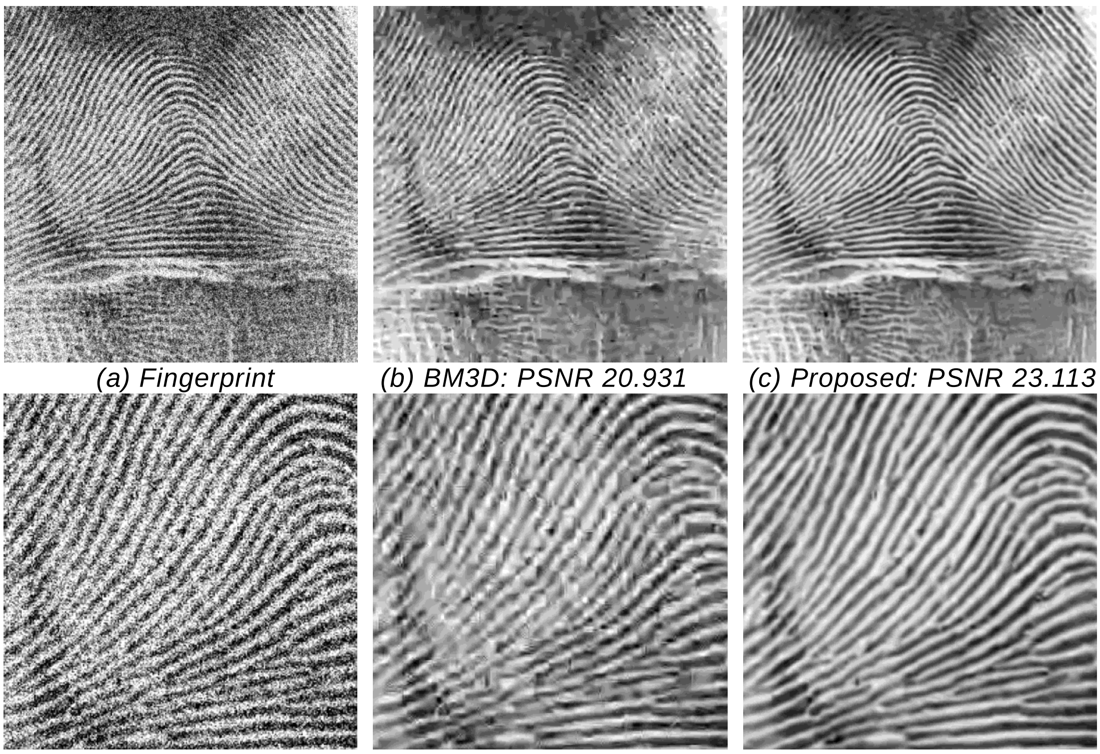
In addition, Figures 5, 6 and Table 4 demonstrate the different performance obtained by method in [5] and the proposed one respectively. Figure 5 and Table 4 show that our method performs slightly better than the method presented in paper [5] in terms of visual quality and PSNR value. What's more, the running time is shorter than that of the algorithm in [5].

Overall, compared with original BM3D, the proposed denoising algorithm with adaptive distance hand-threshold achieves better visual quality and outperforms in terms of objective PSNR criterion in the same class of running time. And for strongly noisy images, it is also superior to the method proposed in [5], whether in respect of objective criteria of PSNR and running time, or in visual effect.

(a) Barbara (b) BM3D: PSNR 28.055 (c) Proposed: PSNR 28.317

(d) Fragment of (a) (e) Fragment of (b) (f) Fragment of (c)

Figure 3. Noisy (σ=40) Image, Denoised Results by BM3D and Proposed Method, and Corresponding Fragments



(a) Fingerprint (b) BM3D: PSNR 20.931 (c) Proposed: PSNR 23.113

(d) Fragment of (a) (e) Fragment of (b) (f) Fragment of (c)

Figure 4. Noisy (σ=70) Image, Denoised Results by BM3D and Proposed Method, and Corresponding Fragments

Table 1. Output PSNR Comparison of BM3D and Proposed Method

σ /PSNR	C.man 256^2	House 256^2	Barbara 512^2	Boat 512^2	Couple 512^2	Man 512^2	Lena 512^2	Hill 512^2
10	34.165	36.614	34.843	33.884	**34.008**	33.936	35.902	**33.585**
	34.189	**36.622**	**34.861**	**33.886**	**34.008**	**33.939**	**35.907**	**33.585**
20	30.457	33.703	31.655	30.828	30.720	30.528	33.031	30.651
	30.471	**33.712**	**31.677**	**30.834**	**30.727**	**30.535**	**33.034**	**30.656**
30	28.636	32.051	29.753	29.066	28.829	28.826	**31.280**	29.104
	28.639	**32.052**	**29.760**	**29.075**	**28.832**	**28.831**	**31.280**	**29.105**
40	27.194	30.759	28.055	27.763	27.500	27.670	29.934	28.007
	27.366	**30.790**	**28.317**	**27.819**	**27.506**	**27.698**	**30.015**	**28.027**
50	25.445	28.379	24.699	26.207	25.909	26.260	28.020	26.636
	26.373	**29.679**	**27.156**	**26.849**	**26.481**	**26.848**	**29.027**	**27.197**
60	24.281	26.780	23.507	25.093	24.792	25.284	26.695	25.685
	25.549	**28.712**	**26.202**	**26.056**	**25.648**	**26.151**	**28.203**	**26.503**
70	23.435	25.824	22.852	24.321	24.041	24.572	25.803	24.981
	24.801	**27.816**	**25.388**	**25.407**	**24.958**	**25.564**	**27.494**	**25.904**
80	22.727	24.998	22.308	23.641	23.400	23.957	25.052	24.333
	24.144	**27.010**	**24.671**	**24.843**	**24.387**	**25.044**	**26.881**	**25.381**
90	22.105	24.213	21.855	23.059	22.837	23.395	24.366	23.780
	23.604	**26.328**	**24.034**	**24.359**	**23.899**	**24.593**	**26.333**	**24.920**
100	21.582	23.536	21.434	22.541	22.328	22.877	23.761	23.232
	23.095	**25.714**	**23.465**	**23.921**	**23.466**	**24.184**	**25.837**	**24.505**

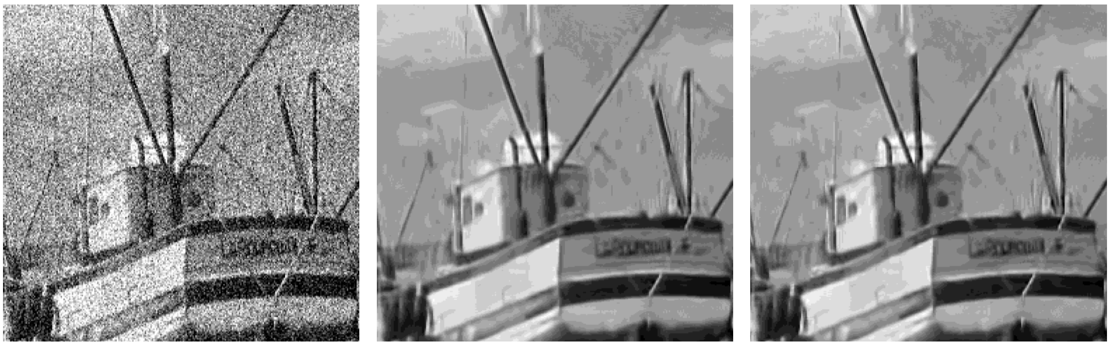
Table 2. Running Time Comparison of BM3D and Proposed Method

σ /time	10	20	30	40	50	60	70	80	90	100
House	5.92	5.99	6.03	**5.61**	**3.22**	**2.71**	**2.51**	**2.36**	**2.22**	**2.11**
	5.70	**5.85**	**5.98**	6.19	6.23	6.37	6.50	6.64	6.72	6.79
Barbara	41.08	52.12	53.20	**47.53**	**27.84**	**24.64**	**23.13**	**21.99**	**20.90**	**20.09**
	20.17	**50.32**	**52.81**	54.96	56.99	58.99	60.82	62.38	63.59	64.62

Table 3. Performance Comparison of BM3D, BM3D-SAPCA and Proposed Method

Peppers256	prefprmance	BM3D	BM3D-SAPCA	proposed method
σ = 10	PSNR	34.650	**34.938**	34.665
	time	4.67	199.52	**4.40**
σ = 60	PSNR	24.785	**26.060**	25.898
	time	**2.23**	191.56	5.45

(a) C.man *(b) [5]: PSNR 26.588* *(c) Proposed: PSNR 26.662*



(d) Boat *(e) [5]: PSNR 27.235* *(f) Proposed: PSNR 27.241*

Figure 5. Fragments of Noisy (σ=45) C.man (top row) and Boat (bottom row), and the Corresponding Denoised Results by [5] and Proposed Method

Table 4. Output PSNR Comparison of Method in [5] and Proposed Method

(a)

σ /PSNR	45	50	55	60	65	70
C.man	26.588	26.120	25.712	25.316	24.950	24.613
	26.662	**26.181**	**25.756**	**25.357**	**24.985**	**24.636**
Boat	27.235	26.781	26.381	26.021	25.696	**25.403**
	27.241	**26.791**	**26.383**	**26.025**	**25.700**	**25.403**
Hill	27.585	27.191	**26.848**	26.520	26.226	**25.934**
	27.588	**27.194**	26.847	**26.521**	**26.227**	**25.934**

(b)

σ /PSNR	75	80	85	90	95	100
C.man	24.330	24.044	23.793	23.526	23.297	**23.074**
	24.344	**24.066**	**23.810**	**23.532**	**23.301**	23.071
Boat	**25.120**	**24.862**	24.622	24.387	**24.176**	**23.970**
	25.119	24.861	**24.623**	**24.388**	**24.176**	**23.970**
Hill	**25.676**	**25.429**	25.197	**24.985**	**24.780**	**24.584**
	25.676	**25.429**	**25.198**	**24.985**	**24.780**	**24.584**

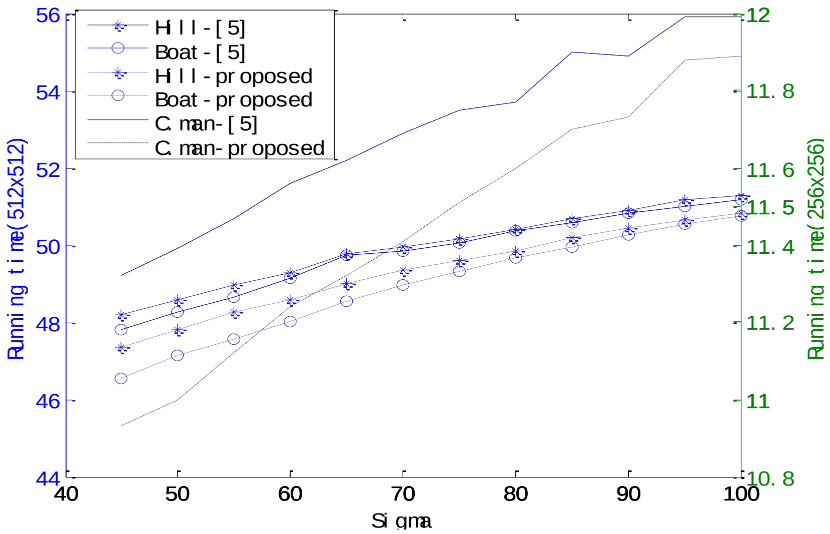


Figure 6. Running Time Comparison between Method in [5] and Proposed Method: '-[5]' Denotes using the Method given in Paper [5], '-Proposed' Denotes using the proposed Method. Both Image Hill and Boat are size of 512x512, whose Corresponding Running Time Scale are Marked on the Coordinates of the Left. While the size of C.man is 256x256, and its two kinds of Execution Time Scale are Labeled on the Coordinates of the Right

4. Conclusion

Taking the image features (the ratio of mean and standard deviation) and noise pollution levels into consideration, feasible improvement to the original BM3D algorithm is provided in this paper. The improved algorithm performs more stably and satisfactory in image denoising. Firstly, it not only removes more noise from noisy image, but also introduces less blocking artifacts. Secondly, for low noise, the proposed method consumes less time than BM3D. Last but not least, it makes the BM3D algorithm more consistent.

References

[1] K. Dabov, A. Foi, V. Katkovnik and K. Egiazarian, "Image Denoising by Sparse 3D Transform-Domain Collaborative Filtering", IEEE Trans. Image Processing, vol. 16, no. 8, **(2007)**, pp. 2080-2095.
[2] S. J. Zhu, "The Research on Non-Local Means Filtering Algorithm", Jiangsu University of Science and Technology, Zhenjiang, China, Chapter 1, **(2013)**, pp. 1-6.
[3] K. Dabov, A. Foi, V. Katkovnik and K. Egiazarian, "A Nonlocal and Shapeadaptive Transform-Domain Collaborative Filtering", Proc. Int. Workshop on Local and Non-Local Approx. in Image Process, LNLA 2008, Lausanne, Switzerland, **(2008)**.
[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "BM3D Image Denoising with Shape-Adaptive Principal Component Analysis", Proc. Workshop on Signal Processing with Adaptive Sparse Structured Representations (SPARS'09), Saint-Malo, France, **(2009)**.
[5] Y. K. Hou, C. X. Zhao, D.Y. Yang, and Y. Chen, "Comment on 'Image Denoising by Sparse 3D Transform-Domain Collaborative Filtering", IEEE Trans. Image Processing, vol. 20, no. 1, **(2011)**, pp. 268-270.
[6] Y. S. Zhang, S. J. Zhu and Y. J. Li, "BM3D Denoising Algorithm with Adaptive Block-Match Thresholds", Applied Mechanics and Materials, vol. 229-231, **(2012)**, pp. 1715-1720.
[7] Y. S. Zhang, S. J. Zhu and Y. J. Li, "Image Denoising based on Adaptive BM3D and Singular Value Decomposition", International Journal of Advancements in Computing Technology, vol. 5, no. 6, **(2013)**, pp. 859-865.
[8] M. Wang, H. J. Wang, G. Y. Sun, J. H. Guo, H. Q. Wu, L. Li and H. Y. Zhu, "Three-Dimensional Image Denoising Algorithm Based on Non-Local Means", Infrared Technology, vol. 35, no. 4, **(2013)**, pp. 238-241.