ADAPTIVE THRESHOLDING HOSVD ALGORITHM WITH ITERATIVE REGULARIZATION FOR IMAGE DENOISING

*Rodion Movchan, Zhengwei Shen**

School of Mathematics and Physics, University of Science and Technology Beijing, Beijing 100083, China

ABSTRACT

In this paper, we propose a very simple 3D patch stack based image denoising method by Higher Order Singular Value Decomposition (HOSVD). We used the idea of iterative regularization from spatially adaptive iterative singular-value thresholding(SAIST) to design our algorithm, which indicates more faster convergence speed than some other methods. By using the parallel computing technique for implementing the algorithm, the computational complexity is highly reduced. The experiments also show good PNSR result with different noise levels.

***Index Terms*—** Nonlocal self-similarity, HOSVD, Adaptive iterative singular-value thresholding, Image denoising

1. INTRODUCTION

Nonlocal self-similarity(NSS)[1] prior of natural images become an issue of increasing interest in a wide range of applications, such as image restoration, computer vision and pattern recognition. The successful applications to image denoising of NSS including nonlocal means filter denoising[15, 17, 18], block-matching 3D(BM3D)[10] and the learned simultaneous sparse coding(LSSC). Following their insight, more recent denoising methods were proposed[2, 3, 4, 5]based on NSS prior.

More recently, HOSVD as an extension of the matrix SVD to high order tensors with arbitrary number of indices has been successfully applied in image processing and big data analysis, e.g. in image denoising [2, 4, 5], face recognition[7] and high dimensional data analysis[8]. In[2]authors proposed a patch-based machine learning methods for image denoising using HOSVD and NSS. They grouped image patches to similar patches stacks by NSS and then applied HOSVD to perform hard thresholding of their coefficients. Very recently, in [5] was proposed two-step grouped-based image denoising method which use some ideas of HOSVD to perform denoising. Similar to the techniques in[2], authors applied an adaptive hard-thresholding to image patches stacks decomposed by HOSVD. As the second step, authors used 2D group-based framework with adaptive iterative singular-value thresholding to improve denoising result.

In this paper, following the weighted nuclear norm minimization(WNNW)[4] and utilizing the merits of HOSVD to 3D tensors grouped by NSS prior of natural image, we propose an adaptive thresholding HOSVD image denoising algorithm with iterative regularization which performance improvement has been shown for wavelet-based models[12]. The main idea of denoising in this paper is to obtain the clean desired image X by group similar patches from noisy observed image Y into 3D patches stacks y, and then perform iterative procedure to threshold coefficients of their HOSVD . However, the differences of the proposed algorithm in this paper to that in[2, 5] are HOSVD to NSS 3D tensor is utilized in contrast to implementing SVD of 2D NSS matrix in the second step of[5], and an adaptive iterative thresholding to the HOSVD coefficients is performed that is different from that in[2]. So, iterative regularization is desired to us for improving denoisng result based on HOSVD.

2. HIGH ORDER SINGULAR VALUE DECOMPOSITION

Singular Value Decomposition of a matrix $A \in R^{n\times m}$ is the factorization of the form $A_{n\times m} = USV^T$ where matrices U and V is orthogonal matrix satisfying $UU^T = I_n$, $VV^T = I_m$, and $S = \operatorname{diag}(\lambda_1, \lambda_2, \dots, \lambda_k)$ is the diagonal singular values matrix satisfying $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_k \geq 0$.

Let the formed 3D tensor stack be $M \in R^{p\times p\times k}$ with p is patch size and k is stack size. In this paper, we consider to split the observed image Y to patches of a chosen size L and put the similar patches to one stack. For the best result, patches should overlapping each other [13]. To increase of computation speed, we choose distance between left bound (upper bound) of one patch and the left bound (upper bound) of the next patch is equal to $\frac{L}{2}$. Let G be the number of stacks. So, we have G stacks of similar patches with k patches in each stack, but this requires a choice of k which may not be the same for each stack discussed in the next section.

Then, the HOSVD of one stack M given as follows[11]

$$M = S \times_1 U^{(1)} \times_2 V^{(2)} \times_3 W^{(3)} \tag{1}$$

where $U^{(1)} \in R^{p\times p}$, $V^{(2)} \in R^{p\times p}$, $W^{(3)} \in R^{k\times k}$, in tensor

Corresponding author* email: ijwmip@ustb.edu.cn.

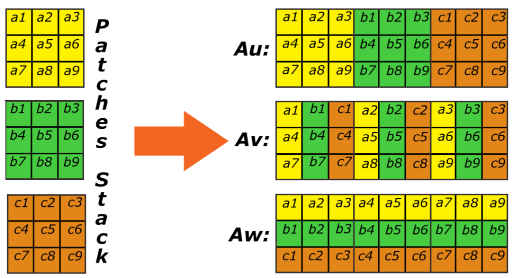

Fig. 1: *n-mode(n=1,2, 3) unfolding of a stack tensor* $M \in R^{3\times3\times3}$

notation, are orthonormal matrices; and S is 3D coefficients core tensor of size $p \times p \times k$; operator $\times_i (i = 1, 2, 3)$ is called n-mode multiplies defined in the following. Because matrices $U^{(1)}$, $V^{(2)}$, $W^{(3)}$ are of eigenvectors of row-row correlation matrix of a stack tensor M unfolding $A_U \in R^{p\times pk}$, $A_V \in R^{p\times pk}$, $A_W \in R^{k\times pp}$(shown in Fig.1) to each coordinate respectively. Specifically, $U^{(1)}$,$V^{(2)}$ and $W^{(3)}$ can be calculated by performing a matrix SVD on the matrix A_U, A_V and A_W respectively. In tensor notation, the n-mode multiplies is defined by $A \times_n U = fold_n(U * unfold_n(A))$. From Fig.1, we can see n-mode unfolding procedure of a tensor $M \in R^{3\times3\times3}$ (shown on the left) into n-mode matrices. $A \times_n U$ multiplication is defined by[7]: $(A \times_1 U)(j, i_2, i_3) = \sum_{k=1}^{p} u_{j,k} a_{k,i_2,i_3}$, $(A \times_2 U)(i_1, j, i_3) = \sum_{k=1}^{p} u_{j,k} a_{i_1,k,i_3}$ and $(A \times_3 U)(i_2, i_2, j) = \sum_{k=1}^{k} u_{j,k} a_{i_1,i_2,k}$ where i-mode and j-mode multiplication commute if $i \neq j$, $i, j \in \{1, 2, 3\}$. Then, the core tensor S, not directly related to eigenvalues, can be calculated as:

$$S = M \times_1 U^{(1)T} \times_2 V^{(2)T} \times_3 W^{(3)T} \tag{2}$$

and the tensor stack M^* of the constructed patches can be obtained by performing inverse transform via Eq. (1).

3. PROPOSED MODEL AND ADAPTIVE ITERATIVE REGULARIZATION ALGORITHM

3.1. Model for the image denoising

Motivated by ideas proposed in [4, 9] to perform image denoising by nuclear norm minimization(NNM), in our paper, given the formed tensor stacks y(i.e. patches stack tensor of the observed image Y), we adopt the following denoising model[4]

$$\min_{S,U^{(1)},V^{(2)},W^{(3)}} \|y - S \times_1 U^{(1)} \times_2 V^{(2)} \times_3 W^{(3)}\|_F^2 + \sum_i |\tau_i \lambda_i(x)|_1 \tag{3}$$

where S is the core tensor of x following Eq.(1) and $\lambda_i(x)$ is the value in core tensor S(singular value in SVD). The fidelity F-norm $\|\cdot\|_F$ is the matrix norm defined as the square root of the sum of the absolute squares of its elements, i.e. $\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{i,j}|^2} = \sqrt{trace(A^*A)}$, with matrix $A \in R^{m\times n}$. The distinctive features of this norm is that it always at least as large as the spectral radius of matrix A and also it has the very useful property of being invariant under any rotation matrix R as $\|A\|_F^2 = \|AR\|_F^2 = \|AY\|_F^2$, which applicability for solving our problem is shown in[4]. The regularization term actually is the adaptive weighted nuclear norm regularizer with the adaptive weight $\tau = [\tau_1, \ldots, \tau_n]$, $\tau_i > 0$ assigned to singular values $\lambda_i(x)$. The selection of the adaptive weight will be shown later. The main idea of this model is to approximate noise part Y by X while minimizing the nuclear norm of X, where model is non-convex due to the varying weight τ.

3.2. Algorithm and parameters setting

A: Choosing stack size k and patch size p.

In ideal situation, bigger k (stack size) performs better denoisng result, because in this situation all patches in one stack differ only by noised pixels. But it is a dilemma choice, we have two problems need to be considered:

For best result, patches in one stack should be differing only for noise, so the patches in one stack should be similar as soon as possible without considering the noise. HOSVD use nonlocal techniques that estimate denoised patch stack by minimizing a penalty term on their average weighted distance between patches. Therefore, if we just pursuit bigger stack size instead of considering the more different details between the patches in one stack then, after denoisng, patches in this stack can be over-smoothed, which means the more details of edges or the textures in patches will probably be smoothed out. So, the similar patches choosing process, which result mainly depends by chosen patch size, should be performed very accurate. On the other hand, if the stack is too big, it will be computationally very hard to implement finding similar patches process; meanwhile, computational complexity of decomposition process by HOSVD, of which is the major disadvantage of the HOSVD, also will increase dramatically. To strike a balance between the denoising effect and the algorithm's computational complexity, we find from experiments that the stack size k can be set in the interval of between 20 and 30.

The size of the patches also influence the denoising result, so it should be adapted for noise level of current image. In [14], the authors tested different patch sizes for different noise level and estimated that, for low noise level the patch size must be relatively small to be well adapted to the details, however, for high noise level lager patch size is a better choice.

B: Searching similar patches.

To construct a stack of similar patches, we choose a reference patch P_{ref} from the noisy image and then to search all

Fig. 2: *Result on the noisy ($\sigma = 10$) 512×512 image Boats. (a) original, (b) noisy image, (c) denoised result.*

the similar patches that satisfy the distance threshold τ_s:

$$\|P_{ref} - P_i\|^2 < \tau_s \tag{4}$$

where $\tau_s = 3\sigma^2 p^2$ for known noise model $N(0, \sigma^2)$. This threshold will consider them to be similar, with a very high probability [2].

C: Iterative regularization

To obtain solution for our model (3), we adopted the iterative-thresholding algorithm[9], of which the threshold τ_i is adaptive since it is tuned following the noise level (i.e. noise variance σ_N) and the values λ_i of core tensor Σ. The main idea of adaptive-thresholding is that the larger the coefficients have the less shrinkage level, so only strong signals(i.e. the carton, geometrical configuration and textures) can be survived in thresholding. Under the assumption that the singular value λ_i of core tensor observe i.i.d Laplacian distribution[3] and the thumb rule for choosing the threshold[6], in this paper, the adaptive threshold is picked to be:

Algorithm 1 Adaptive HOSVD iterative regularization

Require: Noise Image Y, patch size p, stack size k, Iterations I and parameters σ, β, γ;
Ensure: Clean Image X;

```
Initialize: X^0 ← Y, Y^0 ← Y;
for n=1 in 1:I do
    Re-estimate Y^n by (6);
    Re-estimate noise variance σ_N^n of Y^n (7);
    for each patch y_j in Y^n do
        Find similar patches by (4).
        Estimate U^(1), V^(2) and W^(3);
        Find tensor core S by (2);
        For each λ_i^(n) in S compute τ_i by (5)
        Apply threshold τ_i to each λ_i^(n) in S;
        Estimate denoised patch stack by (1);
    end for
    Obtain the denoised image X^n by (9);
end for
```

$$\tau_i = \frac{2\sqrt{2}\sqrt{k_i}\sigma_N^2}{\lambda_i + \epsilon} \tag{5}$$

where noise standard deviation σ_N will be denoted below, k_i is the patches number in the current stack, ϵ is a small positive number for avoiding dividing by zero.

The basic idea of iterative regularization algorithm is to add filtered noise back to the denoised image, so we use the following form:

$$Y^{n+1} = \hat{X}^n + \beta(Y - \hat{X}^n) \tag{6}$$

where n denotes the iteration number, β is relaxation parameter(i.e. step size), and Y is the noised group of image patches. Because of adding the residual(i.e. the filtered noise or details of image) back to the denoised image, the noise variance in each iterate need to be updated as follows:

$$\sigma_N^{(n)} = \gamma\sqrt{\sigma^2 - \frac{1}{M \times N}\|Y - Y^{(n+1)}\|_F^2} \tag{7}$$

where σ_N is noise variance of Y, γ is a scaling factor controlling the re-estimation of noise variance, M and N is image height and width.

D: Aggregation

After filtering by using the preceding iterative algorithm, patches stacks should be put back in to the image. Each patches stack may have different amount of noise, so it should be taken into account. The main idea of weighted averaging is to give more weight to patches stacks with less noise and less weight to those with much noise.

We can definite weight of i-th patch stack (for each patch in patch stack) as:

$$w_i = \begin{cases} \frac{k \times p^2}{k \times p^2 + N}, & \text{if } N \geq 1 \\ 1, & \text{otherwise} \end{cases} \tag{8}$$

where k is the number of patches in current stack, p is patch size and N is the number of thresholded element in core tensor of this patch stack. Consider the overlap between the patches of one patches stack and the others patch stacks, weighted averages needs to be normalized by dividing the result with the sum of the weights. The result image can be calculated by:

$$X(x,y) = \frac{\sum_{i \in I_{x,y}} \sum_{j \in J(i)_{x,y}} w_{i,j}\Omega_{i,j}}{\sum_{i \in I_{x,y}} \sum_{j \in J(i)_{x,y}} w_{i,j}} \tag{9}$$

where $I_{x,y}$ are all patch stacks overlapping in position (x, y), $J(i)_{x,y}$ are all patches in i-th patch stack that overlap in position (x, y), and $\Omega_{i,j}$ is pixel in patch stack i, patch j in position (x, y) of image.

4. EXPERIMENT RESULTS

In this section, we will conduct extensive experiments to highlight the superiority of our proposed models and algorithms by comparing with the state-of-the-art method, such

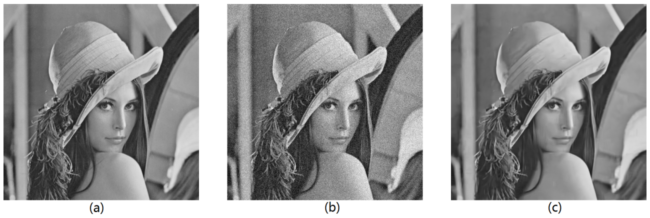

Fig. 3: *Result on the noisy ($\sigma = 20$) 512×512 image Lena. (a) original, (b) noisy image, (c) denoised result.*

Table 1: *Comparison PSNR denoising result of four different algorithms.* ***First****: This paper,* ***second****: LASSC[3],* ***third****: BM3D[15],* ***fourth****: exemplar-based method[16]*

	Noise level		
	$\sigma = 10$	$\sigma = 20$	$\sigma = 50$
Barbara	34.51	31.38	25.92
	25.23	32.10	27.54
	34.87	31.77	27.51
	33.79	30.37	24.09
Lena	35.44	32.88	28.40
	35.90	33.08	29.01
	35.83	33.03	29.08
	35.18	32.64	28.38
House	35.96	33.64	28.89
	36.67	33.90	30.20
	36.37	33.54	29.65
	35.26	32.90	28.67
Boats	33.40	30.62	26.12
	33.91	30.81	26.66
	33.79	30.65	26.71
	33.09	30.12	25.93

as LASSC[3], BM3D[15], exemplar-based method[16]. We exploit four common international standard images as our test images, which further corrupted by the additional Gaussian white noise with standard variation $\sigma = 10, 20, 50$. The variety of the parameters setting in Algorithm 1 is used to assure the high quality performance for our methods. For additional noise with $\sigma = 10$ to the images, we used parameters as follows: patch size: $L = 6$, stack size: $K = 25$, iteration count: 5, $\beta = 0.32$ and $\gamma = 0.45$. For $\sigma = 20$ to the images, we set the parameters as $L = 6$(8 for Barbara and Lena), $K = 30$, iteration count: 7(10 for Barbara and Lena), $\beta = 0.28$(0.20 for Barbara and Lena) and $\gamma = 0.50$(0.39 for Barbara and Lena), and $L = 8$, $K = 35$, iteration count: 10, $\beta = 0.23$, $\gamma = 0.50$ for the $\sigma = 50$ to the images. The search area G for similar patches is set to be 36×36 (i.e. 15 pixels from each side of the patch). Finally, the PNSR values of our methods compared with LASSC, BM3D, exemplar-based method are shown in Table 1. We also illustrate the experimental results in Fig. 2, 3, 4 for Boats, Lena and House images corrupted

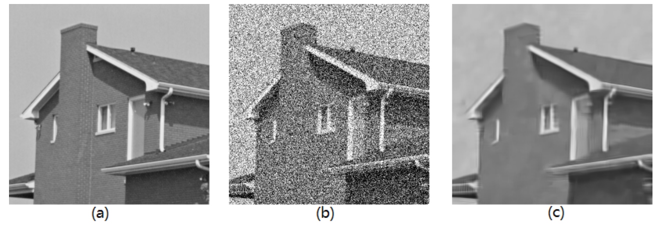

Fig. 4: *Result on the noisy ($\sigma = 50$) 256×256 image House. (a) original, (b) noisy image, (c) denoised result.*

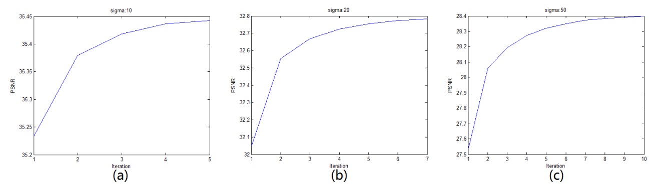

Fig. 5: *Dependence of PSNR to iteration count for Lena image with (a) is $\sigma = 10$ noise corrupted, (b) is $\sigma = 20$ noise corrupted and (c) is $\sigma = 50$ noise corrupted, where the beginning value is equal to HOSVD method without iterative regularization.*

by Gaussian white noise with $\sigma = 10, 20, 50$ respectively. In order to improve computational speed and to estimate result in shorter time, the algorithm is implemented using C# and parallel computing that used to compute each patches stack in own thread.

The experimental results show that the desirable denoised images can be obtained in a small iteration count, which indicates well convergence speed. Although our method doesn't show the best denoising result as shown in Table 1, it outperform other methods in computation speed, with very close denoisng result to them. In fact, the experiments show good PSNR result of filtering images with different noise levels, and from the Fig. 5 we can see that iterative regularization is really improve HOSVD result, which equal to the proposed method in this paper at the first iteration. The denoising result also can be further improved using Wiener filter step in HOSVD as recommends authors in [2]

5. CONCLUSION

In this article, we proposed an adaptive thresholding HOSVD method with iterative regularization which combining ideas about HOSVD low-rank decomposition with SAIST iterative regularization. We show what does mean of SVD in applicability for image denoisng, and how it related with HOSVD. We show how better about grouping image patches to stacks, and then how to apply HOSVD to it. Some suggestions of how better to choose patches size and how much patches should be in patch stack also are considered. The C# Code of this algorithm can be found at `https://github.com/stargox/Iterative_HOSVD_denoising.git`.

6. REFERENCES

[1] A.Buades, B.Coll and J.M. Morel, "A non-local algorithm for image denoising," in Proc. Conf. Comput. Vis. Pattern Recognit., vol. 2, pp. 60-65, 2005.

[2] A.Rajwade, A. Rangarajan. and A.Banerjee, "Image denoising using the higher order singular value decomposition," IEEE Trans. Pattern Anal. Mach. Intell., vol. 35 no.4, pp. 849-862, 2013.

[3] W. Dong, G.Shi, and X. Li, "Nonlocal Image Restoration With Bilateral Variance Estimation: A Low-Rank Approach," IEEE Trans. Image Process., vol. 22, no.2, pp. 700-711, 2013.

[4] S.Gu, L. Zhang, X.Feng, and W. Zuo, "Weighted nuclear norm minimization with application to image denoising," in Proc. Conf. Comput. Vis. Pattern Recognit., pp. 2862-2869, 2014.

[5] H.Pan, T. Huang, and T. Ma, "Two-step group-based adaptive soft-thresholding algorithm for image denoising," Optik, vol.127, no.1, pp.503-509, 2016.

[6] S.Chang, B.Yu, and M.Vetterli, "Adaptive wavelet threshold for image denoising and compression," IEEE Trans. Image Process., vol.9, no.9, pp.1532-1546, 2000.

[7] L. Lita, E.Pelican, "A low-rank tensor-based algorithm for face recognition," Appl. Math. Model. vol.39, no.(3-4),pp.1266-1274, 2015.

[8] D.Hatch,D. Negrete, and P.Terry, "Analysis and compression of six-dimensional gyrokinetic datasets using higher order singular value decomposition, "J. Comput. Phys., vol. 231, no.11, pp. 4234-4256, 2012.

[9] J.Cai, E.Candes, and Z.Shen, "A singular value thresholding algorithm for matrix completion," SIAM J. Optim., vol. 20, no.4, pp.1956-1982, 2010.

[10] K.Dabov, A.Foi A,V. Katkovnik, and K.Egiazarian, "Image Denoising by Sparse 3D Transform-Domain Collaborative Filtering," IEEE Trans. Image Process., vol.16, no.8, pp.2080-2095, 2007.

[11] L.Lathauwer, B. Moor, and J.Vandewalle, "A multilinear singular value decomposition," SIAM J. Matrix Anal. Appl., vol.21,no.4, pp.1252-1278, 2000.

[12] J. Xu, S.Osher, "Iterative regularization and nonlinear inverse scale space applied to wavelet-based denoising," IEEE Trans. Image Process., vol.16, no.2, pp.534-544, 2007.

[13] R.Coifman, D.Donoho, "Translation-Invariant Denoising," Technical report, Yale Univ., 1995.

[14] M.Lebrun, "An Analysis and Implementation of the BM3D Image Denoising Method," Image Process On Line, vol.2, pp.175-213, 2012.

[15] K.Dabov, A.Foi, V.Katkovnik, and K.Egiazarian, "Image denoising with block-matching and 3D filtering," in Electronic Imaging'06, Proc. SPIE 2006, vol.6064, 2006.

[16] C.Kervrann, J. Boulanger, "Local adaptivity to variable smoothness for exemplar-based image denoising and representation," Int. J. Compt. Vis., vol.79,no.1 pp.45-69, 2008.

[17] S. Lefkimmiatis, A. Roussos, P. Maragos, and M. Unser, "A Structure Tensor Total Variation," SIAM J. Imaging Sci., vol. 8, no. 2, pp. 1090-1122, 2015.

[18] G. Chierchia, N. Pustelnik, B. Pesquet-Popescu, and J.-C. Pesquet. "A Nonlocal Structure Tensor Based Approach for Multicomponent Image Recovery Problems." IEEE Trans. Image Process., vol. 23, no. 12, pp. 5531-5544, 2014.