2024 IEEE 16th International Conference on Computational Intelligence and Communication Networks (ICCN) | 979-8-3315-0526-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/CICN63059.2024.10847356

Optimized Adaptive Based Method for MR Image Denoising

Noorbakhsh Amiri Golilarz
Department of Computer Science and Engineering
Mississippi State University
Mississippi State, MS, USA
amiri@cse.msstate.edu

Iman Dehzangi
Department of Computer Science
Center for Computational and Integrative Biology (CCIB)
Rutgers University
Camden, NJ, USA
i.dehzangi@rutgers.edu

Keyan Rahimi
Department of Computer Science
Brown University
Providence, RI, USA
keyan_rahimi@brown.edu

Abstract— **In this paper, we aim to utilize the orca optimization algorithm (OOA) as a computational intelligence method combined with a non-linear data-driven function in the optimization process for brain image de-noising. This meta-heuristic population-based optimizer has recently been proposed to solve complex optimization problems and has achieved promising results in various domains. Here, it is used to obtain the optimum threshold value and other parameters for the activation function. We investigate the impact of this novel algorithm compared to other state-of-the-art optimizers and alternative approaches such as Cuckoo Search (CS), Particle Swarm Optimization (PSO), Differential Evolution (DE), Thresholding Neural Network (TNN), Neighboring Wavelet Coefficients (Neishrink), Hard and Soft thresholds. Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE) are also used to measure the performance analysis of various de-noising approaches. The results show that the proposed method performs better than other alternative optimization techniques.**

Keywords-Brain image de-noising; computational intelligence; optimum threshold value; orca optimization algorithm; PSNR; MSE

I. INTRODUCTION

In image and signal processing, noise refers to unwanted signals that cause undesirable effects, such as blurred objects, obscured lines, and distorted scenes. Noise can be generated during the capturing and transmission processes. Common types of noise include additive noise, speckle noise, and salt-and-pepper noise that distort and degrade the visual quality of images. To be able to analyze and process the images effectively and accurately, it is essential to eliminate these noises that, in turn, enhance the image quality and resolution.

De-noising as a preprocessing step is considered a critical task in image processing before any further analysis. Although conventional filtering-based noise suppression methods can remove noise from images, they may blur parts of the images and reduce the resolution. The primary goal of de-noising and noise removal approaches is to lessen the effect of noise to consequently enhance the visual quality of images. Various de-noising approaches have been presented for this purpose. Liu et al., introduced a fast/robust approach based on integrating the Laplacian Pyramid and non-local means algorithm for reducing the noise [1]. The method noise, as a new measure for performance evaluation of the noise suppression techniques, was developed by Buades et al. [2] using local smoothing filters and non-local means based on the image pixels' non-local averaging. Bilateral filter-based noise removal was developed in [3]. This method can reduce the noises efficiently due to spatial averaging without smoothing edges [3].

There are also several studies found in literature of the wavelet domain. Wang et al., introduced an undecimated-DWT combined with least squares Support Vector Machine (SVM) for noise removal [4]. To do this, they applied un-decimated wavelet transform (UWT) to the noisy image (noisy coefficients) to decompose it into various frequency sub-bands. Then, they used wavelet-based spatial regularity method to form the noisy pixel's feature vector. Thereafter, they used a Least Square Support Vector Machine to generate two types of wavelet components, namely, noisy and noise-free constituents. These noisy components can be denoised using adaptive threshold.

De-noising using Wiener filter is an approach that can effectively reduce the noise. The wavelet image de-noising algorithm based on Wiener filtering using local covariance was done in [5] to obtain coefficients with high visual quality and to preserve the edges, effectively. The traditional mean filters for noise removal cannot only retain the significant features and characteristics of images so that the image will be blurry. Later, Song et al. proposed a mean filter combined with wavelet transform to achieve better performance in image de-noising [6].

In addition, to distinguish the noisy constituents from noise-free components, Bayesian wavelet-based image de-noising was proposed in [7] as a combined model using three main strategies, namely, the magnitudes of the components, the evolution of coefficients across the scales and large components' spatial clustering close to the edges. Several noise

979-8-3315-0526-4/24/$31.00 ©2024 IEEE
DOI: 10.1109/CICN.2024.161

removal methods have been developed with respect to the dependencies of wavelet constituents [8]. De-noising using Bivariate shrinkage in the wavelet domain was introduced in [9]. They proposed Bivariate non-Gaussian distributions, and then, based on the theory of Bayesian estimation, they extracted the shrinkage function. They proved that the coefficients are not independent and there is a dependency between the wavelet components and their parents which they called parent-child dependency [10, 9].

It was discussed in [11] that a de-noised image can be influenced by the type of the employed wavelet as well as the adopted threshold value. To obtain customized filters and wavelet threshold simultaneously, they used simulated annealing with respect to the noisy components. Additionally, a neighborhood near the customized wavelet constituents was considered for thresholding for noise removal purposes [11]. Wavelet domain-based noise reduction utilizing support vector regression (SVR) was also proposed in [12] for Gaussian noise reduction. Another method derived from statistical distributions based on sub-bands' modeling of wavelet detail coefficients was proposed in [10].

A complex wavelet-based bivariate alpha-stable distributions for noise reduction was also proposed in [13]. Wavelet-based noise reduction using Gaussian scale mixture was introduced in [14, 15, 16]. In a different study, [17] focused on curvelet-based image de-noising with combined local parameters and mixture Gaussian model for wavelet detail coefficients' distribution. Additionally, noise removal utilizing complex Gaussian scale mixture and Laplace distributions mixture was proposed in [18] and [19], respectively.

This study uses a newly developed optimization technique called Orca Optimization Algorithm (OOA) [32] for image denoising to improve the results of conventional methods. This optimizer works based on the unique wave-washing technique of orcas and has shown promising results on some benchmark functions as compared to other state-of-the-art metaheuristic optimizers [32]. We utilize this algorithm to learn the parameters of the adaptive non-linear data-driven function and also for acquiring the optimum threshold value. Hence, there is no need to use any gradient descent LMS learning algorithm to find the optimum values. The de-noising results obtained show the superiority of the proposed model over other alternative de-noising approaches.

The rest of the paper is organized as follows. In section II we discuss the definition of noise and image de-noising in the wavelet domain. Section III covers the de-noising process with optimization algorithms. In section IV, we present our proposed approach, and briefly describe Orca Optimization Algorithm. In Section V, we present our results and discuss them. Finally, we conclude this article in Section VI.

II. DE-NOISING

A. Noise Definition

The noisy data vector (influenced by additive white Gaussian noise) is defined as: $z = [z_0, z_1, \dots, z_{N-1}]^T$:

$$z_k = f_k + n_k \qquad k = 0,1,2, \dots, N-1 \tag{1}$$

where, f_k is the k^{th} wavelet coefficients (without noise) of noise-free signal and n_k denotes the Gaussian noise in the wavelet transform domain.

Note that the data vector of coefficients of a noise-free signal is $f = [f_0, f_1, \dots, f_{N-1}]^T$ and the estimate of the real signal is $\hat{f} = [\hat{f}_0, \hat{f}_1, \dots, \hat{f}_{N-1}]^T$. The main aim of image de-noising is to remove the noise from z_k and also to acquire the estimate $\hat{f}$ of the true image f to minimize the mean square error (making the estimated value of $\hat{f}$ close to the real value of f) [20] [10]. The mean square risk is formulated below.

$$J(\hat{f}, f) = \frac{1}{2} E\left\|\hat{f} - f\right\|^2 = \frac{1}{2}\sum_{k=0}^{N-1}(\widehat{f_k} - f_k)^2 \tag{2}$$

where, N denotes the sub-band size, f_k is the noise-free coefficients and $\widehat{f_k}$ is the threshold coefficients.

In VisuShrink technique, the universal threshold (t_u) is unique for all the noisy components [10]. This value can be obtained as:

$$t_u = \sigma\sqrt{2\ln(n)} \tag{3}$$

where, n is the sample size, σ is the noise's standard deviation and also it is called as robust median estimator [21] in which it can be acquired as:

$$\hat{\sigma} = Median\left(\left|D_{k,m}\right|\right)/0.6745 \tag{4}$$

where $D_{k,m}$ is the coefficient in high frequency sub-band of HH_1 sub-band [21].

B. Wavelet

We can write $f(t)$ based on the scale and wavelet functions [22]. 1D-DWT can be written as follows [22]:

$$f(t) = \sum_k a_{g0,p}\ \Gamma_{g0,p}(t) + \sum_{g \le g_0}\sum_k d_{g,p}\ \Lambda_{g,p}(t) \tag{5}$$

where $\Gamma_{g,p}(t) = 2^g\Gamma(2^g t - p)$ denotes the scale function, $\Lambda_{g,p}(t) = 2^{g/2}\Lambda(2^g t - p)$ is the wavelet function, the inner product $a_{g,p} = \langle F, \Gamma_{g,p} \rangle$ is the approximation constituents and the inner product $d_{g,p} = \langle F, \Lambda_{g,p} \rangle$ is the detail coefficients.

Basically, in discrete wavelet transform (DWT) for the first level, the input signal will pass through both the low pass filter and the high pass filter and then it is followed by the decimation. The input signal can be decomposed in detail and approximation coefficients by DWT. To discard all the high frequencies, the signal needs to pass through low pass filter

and then it is followed by down sampling. Doing these procedures (filtering and sub-sampling) can discard half of the frequencies. As a result of this process, the quality of image/signal will be halved. In the second level, we continue the same procedure considering half of the former cut off frequencies. For the higher levels of decomposition, we have the same scenario as in [22]. Figures 1 and 2, show the general block diagram of DWT for first and higher level of decomposition, respectively. In these figures, *L* represents low frequency, and H is for high frequency.

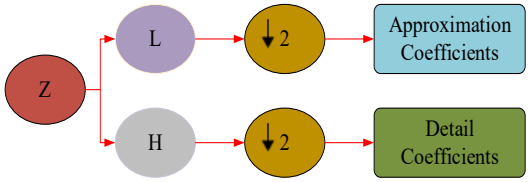

Figure 1. First level of decomposition for DWT

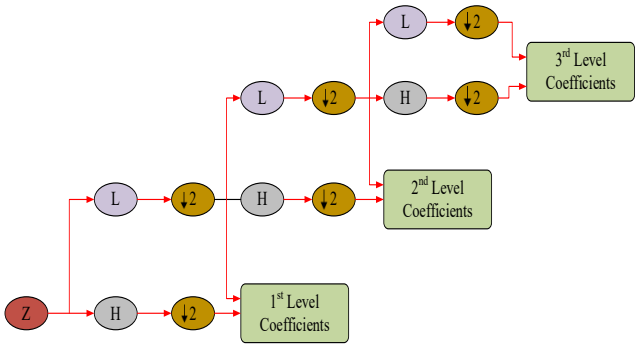

Figure 2. DWT for higher level of decomposition.

C. *Image De-noising in the Wavelet Domain*

Additive White Gaussian Noise (AWGN) with zero mean and standard deviation of σ gives the input noisy image. This noise need to be discarded so that we have a high resolution image. The process of image de-noising using wavelet transform is as follows:

- Apply Discrete Wavelet Transform
- Obtain wavelet coefficients.
- Apply the threshold function.
- Obtain the thresholded wavelet coefficients.
- Apply inverse DWT.
- Obtain output de-noised image.

Hard, soft, and garrote are the basic threshold functions that have been utilized in the past. Recently, TNN-based approaches have become more common among researchers. In these approaches, the aim is to use a network combined with an activation function, which is an improved and enhanced version of the conventional standard threshold function.

In thresholding Neural Networks (NN), instead of using standard hard and soft thresholds, we can use the non-linear, data driven and differentiable functions [23]. These functions can tune the small noisy coefficients by a polynomial instead of setting them to zero [24]. The Least mean square (LMS) algorithm as an iterative learning approach is one of the important methods in NN learning. In this case, utilizing gradient-based least mean square, we can calculate the optimal threshold value [31, 10].

III. Denoising with Optimization Algorithm

Although using LMS learning in TNN to find the optimum threshold value has some advantages, it is still very time-consuming and slow. Thus, different studies have been trying to find a way with shorter processing time and with more efficiency. In this regard, optimized adaptive-based de-noising was proposed in [24]. In this approach, instead of using gradient descent LMS learning, they utilized nature-inspired optimizers such as Differential Evolution (DE) [25], Particle Swarm Optimization (PSO) [26], Wind Driven Optimization (WDO) [27], Firefly Algorithm (FA) [28], Cuckoo Search (CS) Algorithm [29], and JADE algorithm [30] for acquiring the optimized threshold components.

The procedure of the optimized-based de-noising is as follows [24]:

- Obtain Noise coefficients by applying a DWT on the noisy image and setting the optimizer's parameters (e.g. scale parameter, solutions' number and iterations' number of iterations, number of solutions, scale parameters, etc.).
- Finding the solutions of the algorithm by passing these noisy components through the combined nature-inspired optimizer and adaptive data-driven threshold.
- Finding the best fitness value for each solution by computing it through the threshold function.
- Obtaining optimized wavelet components by passing these parameters through adaptive function.
- Applying inverse discrete wavelet transform on these coefficients.
- Get output de-noised image.

IV. Proposed Approach

A. *Orca Optimization Algorithm (OOA)*

Particular kinds of orcas that live in Antarctic region use the cunning tactic of repeatedly hunting in a crowd and forming strong waves to wash seals off drifting ice blocks. The orcas' population produces strong waves in various directions and repeats this treat many times until the target (seal in the orca optimization algorithm) falls from the ice block into the ocean.

The orca optimization algorithm models this behavior, and artificial orcas represent the answers in the algorithm [32]. In the orca optimization algorithm, each orca (potential candidate) represents the unknown variables of the problem and has a similar role to the bee in the artificial bee colony (ABC) optimization algorithm, chromosome in the genetic algorithm (GA), cuckoo in the cuckoo optimization algorithm, or particle in the particle swarm algorithm (PSO).

The optimization process begins with a set of random orcas in the search space. Each orca must represent the unknown variables of the problem that we want to find their optimal values. In the orca optimization algorithm, different orcas knock the ice block from various orientation and angles which each orca has its specific energy depending on its velocity. The value of this energy determines the orca's capacity to wash the ice block and get closer to the target or global point. In the outlined manner, orcas will be in the predefined zone by radius R, at irregular direction and positions.

The orcas that are located farther away from the ice block have a quicker speed when reaching the ice block and wash it with more energy and higher severity. At the same time, the orcas that are located near ice blocks will possess smaller energy when they smash the piece of ice. To determine the new position of artificial orcas in the orca optimization algorithm, the energy of artificial orcas is implemented. The equation below is defined for computing the orca's energy:

$$Energy_i = Fitness_i - Fitness_s \tag{6}$$

In Eq. (6), $Fitness_i$ and $Fitness_s$ denote the fitness of orcas at P_i and P_s respectively. The orca with upper and lowest energy can be determined by sorting the energy vector $[Energy]_{N\times 1}$. Based on the arranged energy vector, we compute the normalized number of energies by Eq. (7):

$$E_i = \frac{Energy_i - Energy_{min}}{Energy_{max} - Energy_{min}} \tag{7}$$

The population of orca will move approaching the piece of ice (where the target is in that area) and lie somewhere within the two circles as illustrated in Figure 3:

$$d_i = Energy_i \times \text{Radius} \tag{8}$$

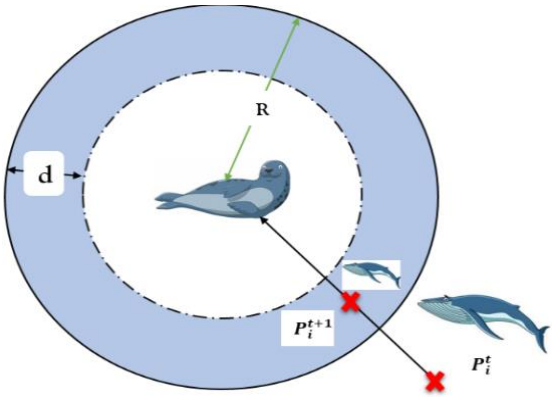

Figure 3. Orcas moving toward the target in OOA [32]

At each iteration of algorithm, K percent of population that have the worst fitness according to defined fitness function will be wiped out. Instead of eliminating orcas, new orcas will be placed randomly. Eliminating weak orcas and generating new orcas allows the orca optimization algorithm to escape from local minima.

B. *Denoising using OOA*

As can be seen in Figure 4, to process optimized based image de-noising, applying discrete wavelet transform provides us with noisy coefficients. These coefficients can pass through a channel containing the orca optimization algorithm and adaptive wavelet threshold [10] to get optimized threshold components. The orca algorithm is used to find the optimum threshold value and other parameters. Then, applying inverse discrete wavelet transform on the optimized constituents gives the output de-noised image. Here the cost function is considered as the mean square error risk between original and estimated images.

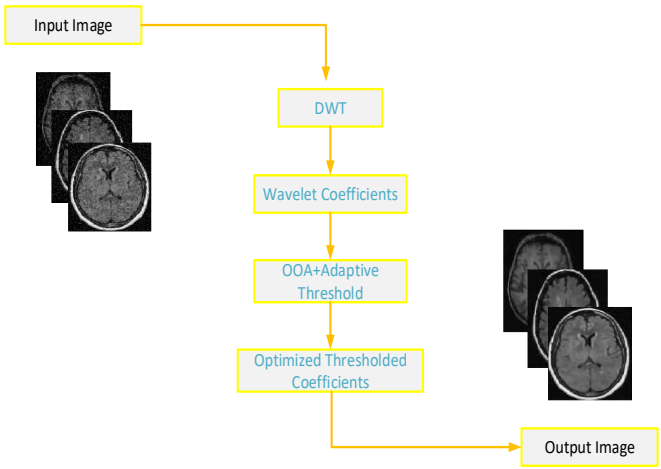

Figure 4. The procedure of optimized based de-noising

V. Experimental Results and Discussions

A. *Results*

We performed several experiments to validate the superiority of the proposed method over other alternative approaches. We used PSNR (dB) and Mean Square Error (MSE) as two metrics to evaluate the performance of the de-noising methods. In the experimental parts we have utilized several brain images (with the size of 128×128) as can be seen in Figure 5. The "Db4" wavelet with one decomposition level is used in all the experiments. The data sets are available in [33, 34]. Note that the images have been corrupted by AWGN with zero mean and various variances (var) as can be seen in the below experiments.

In the first experiment, we compared the OOA based de-noising with Zhang [35], Nasri-TNN [10], and Zhang-TNN [20] methods both visually and quantitatively in Figure 6. Here we utilized test images 1-4. As is shown in this figure, both quality and PSNR values of OOA de-noising method are higher than Zhang's method followed by Nasri-TNN and Zhang-TNN techniques.

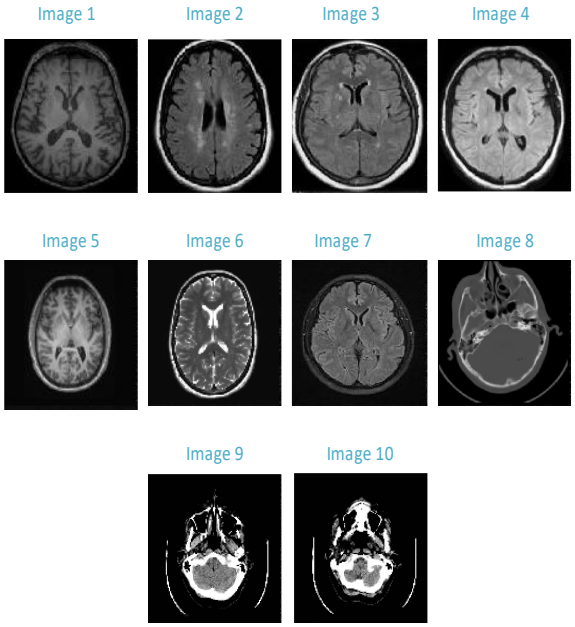

Figure 5. Test images with the size of 128×128.

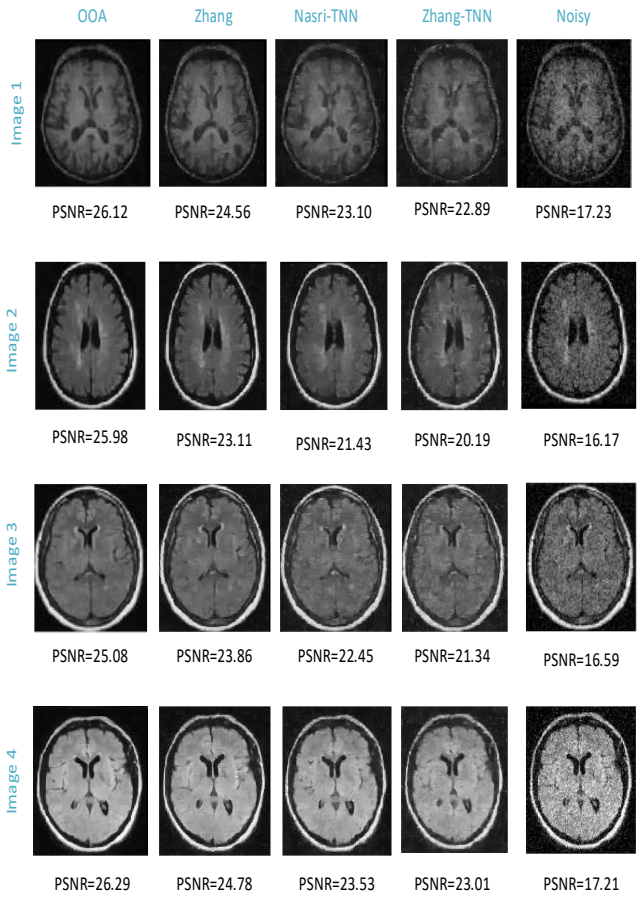

Figure 6. Qualitative and quantitative comparison of different de-noising methods for variance of 0.01 (the values are in dB).

In the second experiment we compared the proposed de-noising method with other alternative approaches such as HHO [36], JADE [24], CS [29], Bayes [37], Neishrink [38], Soft [21] and Hard [21]. As can be seen in Table 1, the PSNR values obtained by OOA are higher than other methods. Images 5-9 have been utilized in this experiment. Additionally, the MSE results which are indicated in Table 2 show that the proposed method generates lower MSE values.

In the last experiment, as can be seen from Figure 7, we used test image 10 to make a comparison between proposed method and DE [25], Wiener, and Median methods. The result indicates that the PSNR values obtained by OOA stays higher than DE followed by Wiener and Median techniques. For a standard deviation of 0.03, the PSNR of Median, Wiener and DE is 21.6 dB, 23.02 dB, and 23.89 dB, respectively. On the other hand, OOA achives the highest value which is equal to 24.9 dB.

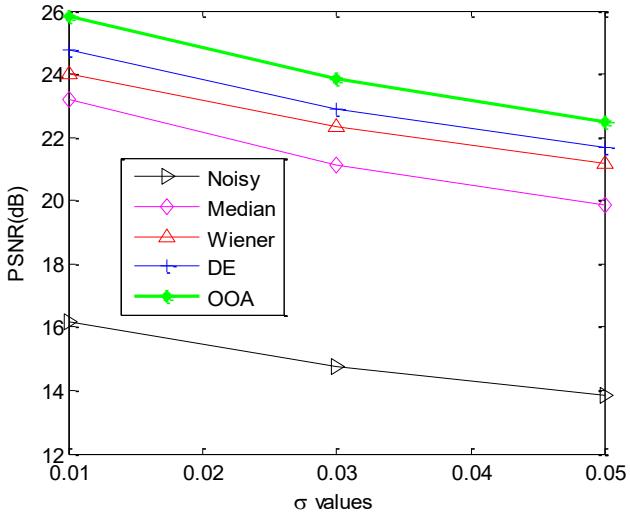

Figure 7. Performance analysis of OOA based de-noising with other methods in terms of PSNR values for various variances.

B. *Discussion*

This study proposes an optimized based de-noising structure containing evolutionary algorithms and an adaptive threshold function. Orca optimizer has been utilized in the optimization procedure for searching and finding unknown parameters. There are several facts we need to consider here. The exploration phase and having a more-in-depth scanning ability are categorized as important factors of a dominant optimizer. Among the evolutionary algorithms available in the literature, the proposed optimizer performs well in the initial phase. How about the second phase which is called shifting process from exploration and exploitation? Can exploitation process affect the de-noising results as well? Some of these optimizers may also act well in the second phase or even in the initial phase. However, the PSNR and MSE results obtained by OOA are better than others. Here we need to look more in depth in the searching strategy to confirm how better exploration, shifting, and exploitation processes affect the PSNR results? Can we strongly say that a much better exploration phase results in much better PSNR?

Or does the much better exploitation phase result in much better PSNR? These may be some crucial reasons why OOA achieves better quantitative and qualitative results compared to other optimizers.

It is important to highlight that swarm intelligence-based de-noising approaches achieve more efficient and reliable results when the metaheuristic optimization method can make a more apropos equilibrium among the exploration and exploitation trends. The surprising thing is how the proposed optimizer combines computational intelligence with image de-noising. Based on the qualitative and quantitative results, metaheuristic approaches perform better compared to the TNN and thresholding-based methods. There are a few questions in which their answers help us to have a deep understanding about choosing appropriate de-noising approaches. As discussed, an image may be contaminated by various types of noises.

TABLE I. PSNR COMPARISON BETWEEN DU=IFFERENT DE-NOISING METHODS FOR AVERAGE OF 10 EXPERIMENTS

Image	var	Hard	Soft	NeiShrink	Bayes	CS	JADE	HHO	**OOA**
Image 5	0.01	21.2980	22.8210	23.1645	24.0514	24.8925	25.8111	27.2659	**27.6029**
	0.03	19.8793	21.6450	21.8897	22.7713	23.6737	24.5523	25.6101	**26.2003**
	0.05	17.2987	18.7982	19.4987	20.5009	21.2114	21.8218	23.2219	**23.4218**
Image 6	0.01	21.4986	22.8902	23.2098	24.1187	24.7983	25.7789	27.1179	**27.4512**
	0.03	18.3012	19.8129	20.1197	21.0983	21.9760	22.8112	24.1098	**24.4967**
	0.05	16.5108	17.7987	18.3654	19.2981	20.1238	20.7689	22.3102	**22.7981**
Image 7	0.01	22.0918	23.8971	24.3987	25.5110	26.4568	27.1115	28.5812	**28.7642**
	0.03	20.8982	22.4532	22.9871	23.8654	24.7765	25.5429	26.5185	**26.9997**
	0.05	19.4009	21.0312	21.5190	22.3918	23.2101	23.8120	25.2912	**25.6823**
Image 8	0.01	22.1128	23.2986	23.6239	24.4118	25.3112	25.8136	27.3918	**27.7112**
	0.03	20.7983	21.8762	22.3982	23.2761	24.0989	24.7910	26.3712	**26.6783**
	0.05	18.1102	19.4892	19.9984	20.8920	21.8793	22.6980	24.2526	**24.6129**
Image 9	0.01	20.2918	21.5987	22.0125	22.9081	23.7770	24.6072	26.0514	**26.4128**
	0.03	19.1764	20.4598	20.9670	21.7098	22.6680	23.4182	24.5538	**24.9123**
	0.05	17.6908	19.0981	19.6121	20.3128	21.1021	21.8993	23.2197	**23.7012**

TABLE II. MSE COMPARISON BETWEEN DIFFERENT DE-NOISING METHODS FOR AVERAGE OF 10 EXPERIMENTS

Image	var	Hard	Soft	NeiShrink	Bayes	CS	JADE	HHO	**OOA**
image 5	0.01	482.2590	339.6099	313.7836	255.8234	210.7804	170.5965	122.0366	**112.9250**
	0.03	668.5753	455.2258	420.8335	343.5186	279.0682	227.9556	178.6777	**155.9733**
	0.05	1211.1846	857.5517	729.8109	579.4158	491.9719	427.4648	309.6637	**295.7333**
image 6	0.01	460.4901	334.2414	310.5277	251.8896	215.4023	171.8661	126.2671	**116.9392**
	0.03	961.5245	678.8758	632.5725	504.9523	412.5535	340.3771	252.4063	**230.8927**
	0.05	1452.1128	1079.4694	947.4152	764.3114	631.9756	544.7413	381.9973	**341.4053**
image 7	0.01	401.6986	265.0759	236.1621	182.8017	147.0281	126.4533	90.1488	**86.4291**
	0.03	528.7621	369.6241	326.8664	267.0178	216.4862	181.4639	144.9540	**129.7508**
	0.05	746.4321	512.8145	458.3321	374.8869	310.5062	270.3213	192.2916	**175.7318**
image 8	0.01	399.7609	304.2428	282.2866	235.4509	191.4081	170.4984	118.5496	**110.1438**
	0.03	541.0661	422.1437	374.3349	305.8231	253.0406	215.7646	149.9547	**139.7173**
	0.05	1004.7554	731.4091	650.4896	529.5175	421.8425	349.3657	244.2420	**224.7969**
image 9	0.01	607.9954	449.9977	409.1008	332.8667	272.5086	225.0921	161.4136	**148.5252**
	0.03	786.0141	584.9252	520.4516	438.6320	351.7874	295.9786	227.8769	**209.8216**
	0.05	1106.6247	800.3323	711.0012	605.0626	504.5106	419.9043	309.8206	**277.3066**

These noises have a wide variety of features and characteristics. According to the obtained results, metaheuristics as computational intelligence approaches can handle these noisy characteristics better than other alternative methods available in the literature. It may be due to the fact that the swarm intelligence methods have strong searching ability (in exploration and exploitation procedures) to find the optimum parameters leading to better de-noising results. Do metaheuristics optimizers perform well even if the images are corrupted by other types of noise? If not, it is one of the drawbacks of computational intelligence-based methods in de-noising fields.

The proposed method may have some shortcomings as well. First, it may not work well if we consider more images and more datasets. There may be room to enhance the efficiency, robustness, and accuracy of the OOA by working on the initial phase. Computational time and speed are also other important factors which need to be taken into

consideration. Second, the utilized adaptive threshold function may not work well for large images or highly noisy images. There is also room to enhance and improve the threshold function since it deals with noisy components, and it needs to be proficient enough. The proper threshold function may also affect the de-noising speed as well.

VI. Conclusion

In this paper we have introduced a new optimized based approach for brain image denoising using a newly proposed population-based metaheuristic optimizer called the orca optimization algorithm, combined with an adaptive data driven threshold function. This optimizer has demonstrated promising results for solving various complex optimization problems. The OOA was utilized here in the optimization process for acquiring the optimal values of threshold function. Its performance has been compared to various approaches like TNN, Neishrink, and other optimized based methods. We have used two metrics (PSNR and MSE) to measure the performance of the proposed de-noising technique. The results indicate the superiority of our proposed method over other techniques. As a direction for future work, we aim at extending the OOA by having more-in-depth investigation on the exploration phase to improve the searching ability. We also aim at improving the threshold function to enhance the de-noising results.

Acknowledgment

This work was supported by PATENT Lab (Predictive Analytics and TEchnology iNTegration Laboratory) at the Department of Computer Science and Engineering, Mississippi State University.

References

[1] Y. L. Liu, J. Wang, X. Chen, et al, "A robust and fast non-local means algorithm for image denoising," Journal of Computer Science and Technology, vol. 23, pp. 270–279, 2008

[2] A. Buades, B.Coll, J. M. Morel, "A non-local algorithm for image denoising," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR2005), 2005, 60–65

[3] M. Zhang, B. K. Gunturk, "Multi resolution bilateral filtering for image denoising," IEEE Transactions on Image Processing, vol. 17, pp. 2324–2333, 2008.

[4] X. Y. Wang, H. Y. Yang, Z. K. Fu, "A new wavelet-based image denoising using undecimated discrete wavelet transform and least squares support vector machine," Expert System with Applications, vol. 37, pp. 7040-7049, 2010

[5] L. Dan, W. Yan, F. Ting, "Wavelet image denoising algorithm based on local adaptive wiener filtering," Proc. MEC, Jilin, China, 2011, pp. 2305–2307

[6] Q. Song, L. Ma, J. Cao, et al, "Image denoising based on mean filter and wavelet transform," Proc. of AITS, Harbin, China, 2015, pp. 39–42.

[7] A. Pižurica, W. Philips, I. Lemahieu, et al, "A joint inter- and intrascale statistical model for Bayesian wavelet based image denoising," IEEE Transactions on Image Processing, vol. 11 pp. 545-557, 2002.

[8] W. Ding, F. Wu, X. Wu, et al, "Adaptive directional lifting based wavelet transform for image coding," IEEE Transactions on Image Processing, vol. 16, pp. 416–427, 2007.

[9] L. Sendur and I. W. Selesnick, "Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency," IEEE Transactions on Signal Processing, vol. 50, pp. 2744–2756, 2002.

[10] M. Nasri, H. Nezamabadi-pour, "Image denoising in the wavelet domain using a new adaptive thresholding function," Neurocomputing, vol. 72, pp. 1012-1025, 2009.

[11] G.Y. Chen, T.D. Bui, A. Krzyzak, "Image denoising with neighbor dependency and customized wavelet and threshold," Pattern Recognition, vol. 38, pp. 115–124, 2005.

[12] H. Cheng, J. W. Tian, J. Liu, et al, "Wavelet domain image denoising via support vector regression," Electronics Letters, vol. 40, pp. 1479-1481, 2004.

[13] A. Achim, E. Kuruoghlu, "Image denoising using bivariate alpha-stable distributions in the complex wavelet domain," IEEE Signal Processing Letters, vol. 12, pp. 17–20, 2005.

[14] J. Portilla, V. Strela, M. J. Wainwright, et al, "Image denoising using scale mixtures of Gaussians in the wavelet domain," IEEE Transactions on Image Processing, vol. 12, pp. 1338-1351, 2003

[15] J. Portilla, V. Strela, M.J. Wainwright, et al, "Adaptive wiener denoising using a Gaussian scale mixture model in the wavelet domain," Proc. International Conference on Image Processing (Cat. No.01CH37205), Thessaloniki, Greece, Oct. 2001

[16] J. Portilla, "Full blind denoising through noise covariance estimation using Gaussian scale mixtures in the wavelet domain," Proc. International Conference on Image Processing, 2004. ICIP 04, Singapore.

[17] H. Rabbani, M. Vafadust, S. Gazor, "Image denoising in curvelet transform domain using gaussian mixture model with local parameters for distribution of noise-free coefficients," Proc. 4th IEEE/EMBS International Summer School and Symposium on Medical Devices and Biosensors, Cambridge, UK, Aug. 2007

[18] L. Srinivasan, Y. Rakvongthai, S. Oraintara, "Microarray image denoising using complex gaussian scale mixtures of complex wavelets," IEEE Journal of Biomedical and Health Informastics, vol. 18, pp. 1423-1430, 2014.

[19] H. Rabbani, M. Vafadoost, "Wavelet based image denoising based on a mixture of Laplace distributions," Iranian Journal of Science and Technology Transactions B-Engineering, 30 (B6), pp. 711–733, 2006.

[20] X. P. Zhang, "Thresholding neural network for adaptive noise reduction," IEEE Transactions Neural Networks, vol. 12, pp. 567–584, 2001.

[21] D.L. Donoho, I.M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," Biometrika, vol. 81, pp. 425–455, 1993.

[22] B. Rasti, J. R. Sveinsson, M. O. Ulfarsson, et al, "Hyper-spectral image de-noising using 3D wavelets," Proc. International Conference on Geoscience and Remote Sensing Symposium (IGARSS), 2012, pp. 1349-1352.

[23] X. P. Zhang, M. D. Desai, "Adaptive denoising based on SURE risk," IEEE Signal Processing Letters, vol. 5, pp. 265–267, 1998.

[24] A. K. Bhandari, D. Kumar, A. Kumar, et al, ''Optimal sub-band adaptive thresholding based edge preserved satellite image denoising using adaptive differential evolution algorithm," Neurocomputing, vol. 174, pp. 698-721, 2016.

[25] R. Storn, K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," Journal of Global Optimization, vol. 11, pp. 341–359, 1997.

[26] R. Poli, J. Kennedy, T. Blackwell, "Particle swarm optimization: an overview," Swarm Intelligence, vol. 1, pp. 33–57, 2007.

[27] Z. Bayraktar, J. P. Turpin, D. H. Werner, "Nature-inspired optimization of high-impedance metasurfaces with ultra-small interwoven unit cells," IEEE Antennas Wireless Propagation Letters, vol. 10, pp. 1563–1566, 2011.

[28] X. S. Yang, “Firefly algorithm, Levy flights and global optimization,” Max Bramer, Richard Ellis, Miltos Petridis (Eds.), Research and Development in Intelligent Systems XXVI, London, Springer, 2010, pp. 209–218

[29] X. S. Yang, S. Deb, “Cuckoo search via levey flights [C]. Proceedings of Nature and Biologically Inspired Computing (NaBIC), Coimbatore, 2009, pp. 210–214

[30] J. Zhang, A. C. Sanderson, “JADE: adaptive differential evolution with optional external archive,” IEEE Transactions on Evolutionary Computation, vol. 13, pp. 945–958, 2009.

[31] S. M. E. Sahraeian, F. Marvasti, N. Sadati, “Wavelet image denoising based on improved thresholding neural network and cycle spinning,” Proceedings of International Conference on Acoustics, Speech and Signal Processing, USA, 2007

[32] N. A. Golilarz, H. Gao, A. Addeh and S. Pirasteh, “ORCA Optimization Algorithm: A New Meta-Heuristic Tool for Complex Optimization Problems,” 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 2020, pp. 198-204

[33] Helwan, A., El-Fakhri, G., Sasani, H., & Uzun Ozsahin, D. (2018) , “Deep networks in identifying CT brain hemorrhage,” Journal of Intelligent & Fuzzy Systems, 35(2), 2215-2228

[34] Available: https://www.kaggle.com/datasets?datasetsOnly=true

[35] Y. Zhang, W. Ding, Z. Pan, J. Qin, “Improved wavelet threshold for image de-noising,” Frontiers in Neuroscience, 2019

[36] A. A. Heidari, S. Mirjalili, H. Faris, et al, “Harris hawks optimization: Algorithm and applications,” Future Generation Computer Systems, vol. 97, pp. 849–872, 2019

[37] S.G. Chang, B. Yu, M. Vetterli, “Adaptive wavelet thresholding for image de-noising and compression,” IEEE Trans. Image Process, vol. 9, pp. 1532–1546, 2000.

[38] G. Y. Chen, T. D. Bui, and A. Krzyzak, ‘‘Image denoising using neighbouring wavelet coefficients,’’ Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., Oct. 1994, pp. 917–920.