INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

ORIGINAL ARTICLE

OPEN ACCESS

Received: 03/02/2025

Accepted: 16/05/2025

Published: 30/05/2025

Citation: Hareesh S, Rethinavalli S (2025) Bone Fracture Image Denoising using a Novel Filter with Hybrid Optimization. Indian Journal of Science and Technology 18(21): 1636-1650. https://doi.org/10.17485/IJST/v18i21.899

* **Corresponding author**.

hareeshphd0@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Bone Fracture Image Denoising using a Novel Filter with Hybrid Optimization

S. Hareesh[1]*, **S. Rethinavalli**[2]

1 Research Scholar, PG and Research Department of Computer Science, Shrimati Indira Gandhi College (Affiliated to Bharathidasan University), Tiruchirappalli, Tamil Nadu, India
2 Research Supervisor, PG and Research Department of Computer Science, Shrimati Indira Gandhi College (Affiliated to Bharathidasan University), Tiruchirappalli, Tamil Nadu, India

Abstract

Objective: The prime objective of this particular research is to enhance the quality of Bone fracture medical images by noise removal in the preprocessing stage, ensuring much better detection and analysis of anomalies in the human body. At the same time, it presents a solution to some of the deficits of traditional digital filters with regard to the optimization robustness and time efficiency. **Method:** Introduces a Decimation Based Guided Box Filter novelty used in the first time for Bone fracture Medical Image Denoising. For performance optimization, a hybrid algorithm based on the Particle Swarm optimization with Ant Colony Optimization (PSOACO) is adopted. The hybrid method ensures that the selection of optimal parameters for the filter is across various noise types. **Findings:** The experimental PSOACO model proved to be superior performance in removing any of the four considered kinds of noise, since it gave the marked PSNR of 38.6 dB, SSIM of 0.99, and MSE of 0.01, which are higher than those produced by PSO, CS, or FF algorithms in any way. The method was found to be effective in terms of perceived quality and numerical measures alike. **Novelty:** This study reports on the first-time use of a decimation box filter specifically aimed at medical image denoising, combining this innovative filter together with a hybrid bio-inspired optimization method for higher perform ability. The approach is thus highly innovative with respect to both the filtering technique as well as the optimization strategies. **Application:** The method can be used in any medical imaging systems, especially in diagnostic and pre-processing tools, and it can also be integrated into AI platforms for an improved image interpretation and clinical accuracy. Other possibilities of applications include tumor detection, organ segmentation, and automated diagnostic support.

Keywords: Digital filters; Bone Fracture medical images; PSO; ACO; FF; Bio-inspired optimization

1 Introduction

The medical field has accepted the digital age, with a sharp rise in the last two decades, which also includes the processing of digital images. To date, this technology can be found in various medical processes, including image acquisition, transfer, noise reduction, pre-processing, channel equalization, restoration, interference detection, and disease diagnostics. Digital filters are needed in radiation modalities for biomedical applications, such as CT scans, X-rays, and magnetic resonance imaging (MRI), to improve image quality and eventually help in the analysis process better[1]. Some of the denoising application areas of digital filters are their adoption in super-resolution[2], image enhancement[3] and electroencephalography (EEG) signal processing[4]. Medical images taken with X-rays, MRIs and CT scans as well as ultrasound are used to define diseases and their severity. Knowledge garnered from these images is critical to tracing the proper prediction and behaviour of the subject. The process of image capture, however, could introduce unwanted noise that could ruin the quality of the diagnostic results. Sources of noise in medical images vary, from environmental influences and outpoints in the image acquisition system to changes in sensor sensitivity caused by temperature fluctuations. The above makes noise an issue to be dealt with on captured images, mostly when the type of noise is unknown. The common noises are Gaussian noise, Poisson noise, salt-and-pepper noise (impulse noise), and speckle noise, which generally render quite an impact on images both in their quality and diagnostics.

There are different approaches to digital filter design; each has its pros and cons. Rather conventional filter design approaches such as least squares (LS)-which normally minimize error in filter design-yield discontinuities in the filter response. The Chebyshev approximation-based filters were proposed to solve that[5]. Digital filter architectures usually generate multimodal error surfaces regarding its filter coefficients, which complicates optimization further. Thus, traditional derivative-based algorithms like least mean squares become trapped in local minima in such kind of optimization work. Moreover, those recursive algorithms are rather unstable, converging slowly, and suffer reduced performance with the passage of time. Bio-inspired algorithms promise to be the best alternative solutions in that they offer improved convergence both with local and global minima.

This makes bio-inspired algorithms quite effective in attaining convergence towards both local and global minima[6]. An image denoising algorithm with a convolutional neural network-based denoising autoencoder was proffered by Farooq et al. This process involves partitioning the images into training and testing data sessions, training the autoencoder, and tuning it further by using convolutional networks. Advantages include better noise reduction and feature preservation, particularly for the medical images; however, it involves heavy computation and large datasets for training[7].

V Karthikeyan et al. suggested an image denoising and contrast enhancement hybrid DWT-DnCNN model under low-light conditions. It utilizes SSIM-based loss to retain the texture as well as employing modern deep learning techniques. This method outperforms RIDNet by a margin of 2% in SSIM. The advantages of the method include better quality images and retention of textures, while the disadvantage is higher complexity of the model[8].

Torun et al proposed SM-CNN, a new hyperspectral image denoising model that uses a novel spectral self-modulating residual block (SSMRB) to adaptively deal with complicated noise. It uses spectral and spatial features to give an improved denoising performance. Compared to current models, it gives a slight edge but has higher complexity and computation costs[9].

Sridhar et al.[10] proposed an adaptive bilateral filter, designed with the backtracking optimized search algorithm in which sharpness of edges is taken into consideration as optimization parameter. Bhonsle et al.[11] used enhanced grasshopper optimization for denoising medical images. Nevertheless, this method consumed a lot of resources compared to edge-preserving smoothing.

An innovative multi-scale denoising network known as MSDNet, was proposed by Jibin Deng et al. to overcome some of the disadvantages of conventional CNNs that denote their heavy reliance on training data resulting in overfitting and poor preservation of details. The three ingredients necessary for the proposed MSDNet are three modules-namely: multi-scale progressive fusion block (MSPFB), pixel-wise attention block (PWAB), and residual learning (RL)-throughout the proposed MSDNet, which in turn acts to improve noise removal and detail retention. The proposed method shows better denoising and single-image deraining capabilities than existing models. While higher performance and detail handling is a plus, it does add to architectural complexity since the models in question are more complex[12].

Murad proposed real-time deep learning-based image denoising tailored for cybersecurity purposes with an overall accuracy of 92% concerning noise versus signal classification. The proposed method effectively removes noise but preserves the critical information needed, thus promising a reliable solution for enhancing cybersecurity. One major limitation could be its dependence on a curated dataset that may hinder generalizability[13].

Qi Zhang suggested TDCNN, a texture-oriented CNN for image denoising impressing on preserving texture by means of texture-extraction, refinement, and transformation blocks. TDCNN shows a better performance than other methods on both synthetic and real images. The advantages are better texture handling, while the complexity of TDCNN could be a potential limitation[14].

Soniya S proposed MNRU-Net, being a newly proposed model for image denoising that enhances U-Net through hand-crafted features and noise approximations, gets better results at higher noise with lesser computational complexity. It does better than complex models such as RDDCNN, although it relies on using hand-crafted features [15].

The Lili Han et al.'s hybrid CNN-Transformer architecture for denoising wild horseshoe crab images utilizes concepts like multi-head transposed attention, gating mechanisms, and depth-wise separable convolution. While restoring relevant image features that assist tracking and localization, this architecture's strong point is actually feature reconstruction, increasing complexity due to hybridization [16].

Isma Batool et al. proposed the dual residual dense network for image denoising, distinguished for its efficient design features taken from residual learning and dense connectivity. The authors validate that it consistently outperforms existing methods on the Kodak24 and BSDS300 datasets with significantly high PSNR and SSIM scores. The merit of the method lies in efficiency and good performance, while a shortcoming relates to dependence on dataset-specific performance measures [17].

Wang Tiantian et al. have proposed a lightweight network for the denoising of images combining the residual and attention mechanisms, thereby reducing the computational burden and retaining image features. Its PSNR, SSIM, and FSIMc scores on most datasets outperform over 20 methods, while its advantages are great denoising performance with preserved details, and since it could be a potential downside, its complexity [18].

Jinwoong Chae proposed a deep learning model for denoising TEM images based on CNNs which were trained on simulated images with different types of noise. While the model is successful in reducing noise, it is having difficulty with the preservation of circular shapes avoiding patch artifacts. Possible future methods and research have been proposed as solutions for addressing these limitations [19]. Built on the earlier contributions of existing research, this paper also studies bio-inspired optimization algorithms as a reasonable approach for edge-preserved denoising of medical images. The key contributions of this paper are as follows:

- This paper introduces A Novel Decimation-Based Guided Box Filter for Image Denoising, which is the first application of the so-called decimation box filter, so increasing the novelty of the approach.
- The designed PSOACO is application-independent and thus works on various noise types such as Gaussian, Poisson, Speckle, and salt-and-pepper noise.
- The performance would therefore be extremely boosted by adopting noise reduction through PSOACO, as the PSOACO optimizes the parameters of the filter.
- The study includes an ablation experiment on the converging themes of various bio-inspired optimization algorithms, and also includes the results analyzed against varying levels of noise and the different decimation box sizes.
- The proposed algorithm is computationally effective, as it only includes simple arithmetic processes on the image pixels.

The focus of this paper is to develop a Bone fracture medical image denoising filter using a novel decimation box filter, with guidance from a hybrid bio-inspired optimization algorithm to minimize the error introduced by different noises. Filter design and the optimization algorithm are implemented in Python, discussed thoroughly in this framework. The proposed filter is tested against the FracAtlas dataset and various types of noise, outperforming other existing denoising filters of the state of the art.

2 Methodology

2.1 Applying Bio-Inspired Techniques to Digital Filter Engineering

Multi-rate systems functioning normally consist of data processed at different sampling rates at different stages of the system. These systems allow for an on-the-fly change in sampling frequency during the signal processing [20]. The basic operations involved in the multi-rate processing are decimation and interpolation. Decimation simply refers to the decrease in sampling frequency to some lower rate which is an integer fraction of the original frequency. This process is also commonly called down-sampling.

A p-fold decimation filter is shown in Figure 1. The structure of this filter includes a low-pass filter (like FIR filter) to avoid aliasing, followed by p-fold decimation, which rejects frequencies greater than $\frac{n_s}{l}$ referred to as the Nyquist frequency. At this stage, only every p-th sample is retained while the rest are discarded. Thus, the effective sampling frequency is decreased to $\frac{n_s}{l}$, and the output signal, n_x, is generated. Here, n_s is the original sampling frequency of the input signal m_x. The goal of a bio-inspired optimization algorithm is to identify a set of coefficients.

The bio-inspired optimization algorithm is devoted to finding an ideal number of coefficients (c), $c \in RM$ that allows the filter to perform in the way that the designer wants. The main goal of these optimization-based filters is to reduce the error

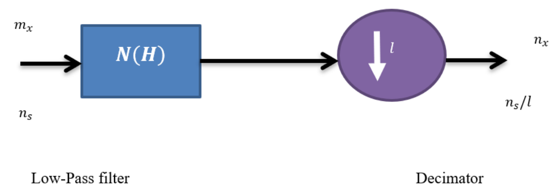


Fig 1. Decimation Filter

between the ideal frequency response) and the desired frequency response (G_I). This will be done iteratively, by tuning the filter coefficients to minimize this difference.

$$Err_z\left(e^{jw}, d\right) = G_I\left(e^{jw}\right) - G_D\left(e^{jw}\right) \tag{1}$$

Err_z, shows that this is complicated. Several conventional methods, such as windowing techniques for digital decimation filter design, can be applied; however, these approaches are incapable of accurately controlling transition bandwidths and cutoff frequencies.

2.1.1. Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is a computational optimization that imitates the social behavior of flocks of birds or schools of fish. It is a population-based algorithm in which each solution (called a particle) moves through the space of solutions to find an optimal solution. In PSO, each particle is a potential solution to the problem under consideration and moves through a process designed by the two most important components- Its own best-known position (local best, *p*best). The best-known position of the swarm (global best, *g*best). Particles move through the search space by updating their velocity and position in every iteration according to these two components.

The performance of each particle in PSO is administered by the following equations:

1. Particle Position Update:

$$Pos_i(t+1) = Pos_i(t) + vel_i(t+1) \tag{2}$$

Where

$Pos_i(t)$ is the position of the i^{th} particle at time step t.

$vel_i(t+1)$ is the velocity of the i^{th} particle at the next time step t+1.

2. Particle Velocity Update:

$$Vel_i(t+1) = w.Vel_i(t) + c_1.rand_1.\left(p_{best,i} - pos_i(t)\right) + c_2.rand_2.\left(g_{best} - pos_i(t)\right) \tag{3}$$

Where

The inertia weight w regulates the influence exerted by previous velocity upon the current velocity. It attempts to balance exploration and exploitation. The cognitive coefficient $c1$ determines the extent to which the particle is influenced by its own best position (*p*best). The social coefficient *c*2 determines the extent to which the particle is influenced by the global best position (*g*best). rand1 and rand2a re stochastic random numbers between 0 and 1 for providing stochastic behavior. The *p*best is the best position achieved so far by the *i*th particle. The *g*best is the best position found by the total swarm.

3. Update of Personal Best Position:

$$p_{best,i} = \begin{cases} pos_i(t) if\ f\left(pos_i(t)\right) < f\left(p_{best,i}\right) \\ p_{best,i} \qquad\qquad otherwise \end{cases} \tag{4}$$

Where

f is the objective function value for the particle at position pos*i*(*t*).

The current position is considered the particle's best personal best.

4. Update of Global Best Position:

$$g_{best} = \begin{cases} p_{best,i}(t) if\ f\left(p_{best,i}(t)\right) < f\left(g_{best}\right) \\ g_{best} \qquad\qquad otherwise \end{cases} \tag{5}$$

Where

The global best is updated if any particle's personal best is better than the current global best.

Algorithm 1. **Particle Swarm Optimization (PSO)**

Input: Minimization objective function (X) Search space is the range in which the sampling interval Δ and the shift δ exist.

Output: Δ, δ Initialize positions of particles pos_0 randomly within the search space of Δ and δ.

Initialize the velocities of the particles vel_0 randomly.

Set the best-known position of each particle $pbest$ to its initial position pos_0.

Evaluate the objective function at $pbest$, and

set the global best position $gbest = argmin\,(f(pbest))$.

For $i = 1, 2, \ldots, Ti = 1, 2, \ldots, T$ (where T is the maximum number of iterations):

If no change happens to gbest for $i_{\max}$ iterations, return Δ, δ, and $gbest$ as the solution.

Using the velocity update equation, update the velocity of each particle $vel._n$

Using the position update equation update the position of each particle pos_n.

Evaluate the objective function at the new position (Xi).

If the new position is better than previous best position, update est.

If any particle's best position is better than the global best, update $gbest$.

End For.

2.1.2. ***Ant Colony Optimization (ACO)***

The ACO is a probabilistic metaheuristic algorithm that draws inspiration from the foraging behavior of real ants. It was first introduced by Marco Dorigo in the early 1990s to address classically formulated combinatorial problems such as the Traveling Salesman Problem (TSP), scheduling, and routing. Real ants lay down pheromone trails on the various paths from their colony to the food sources. Over time, the shorter paths receive more pheromone (because the ants have made it quicker to go to these sources and return) and thus attract other ants which follow and reinforce these shorter and better paths. ACO utilizes this mechanism with artificial "ant-like agents" exploring the solution candidates in a computational space.

a. Probability of Moving from Node *i* to Node *j*:

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{\left[\tau_{ij(t)}\right]^{\alpha} \cdot \left[\eta_{ij}\right]^{\beta}}{\sum_{I \epsilon allowed_k} \left[\tau_{ij(t)}\right]^{\alpha} \cdot \left[\eta_{ij}\right]^{\beta}} & If\, j \in allowed_k \\ 0, \qquad otherwise \end{cases} \tag{6}$$

Where

(t) is the pheromone value on edge (i), at time is the heuristic desirability, often $\eta_{ij} = \frac{1}{d_{ij}}$, where d_{ij} is the distance. α and β control the relative importance of pheromone vs. heuristic. allowed: the set of cities not yet visited by ant k.

Pheromone Update Rule:

$$\tau_{ij}(t+1) = (1-\rho).\tau_{ij}(t) + \Delta\tau_{ij}(t) \tag{7}$$

Where:

$\rho \in$ is the evaporation rate.

$\Delta(t)$ is the amount of pheromone added by ants:

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t) \tag{8}$$

And for each ant k:

$$\Delta\tau_{ij}^{k}(t) = \begin{cases} \dfrac{Q}{L_k}, & if\ and\ k\ used\ edge(i,j) \\ 0, & otherwise \end{cases} \tag{9}$$

Where:

Q is a constant.

Lk is the total cost (e.g., distance) of the tour constructed by ant k.

Algorithm 2: Ant Colony Optimization (ACO)

Initialization:
Set $(0) = \tau_0$ for all edges (i, j)
Set iteration $t = 0$
Repeat until convergence or $t = T$:
a. Construct Ant Solutions:
For each ant $k = 1m$:
Place the ant at a randomly chosen starting node.
Construct a complete solution (e.g., tour) by moving probabilistically using (t)
b. Evaluate Solutions:
Compute the cost Lk of each ant's tour.
c. Update pheromones:
Evaporation: reduce all pheromones by factor $(1-\rho)$
Deposit: for each ant, deposit pheromone on used edges using $\Delta\tau_{ij}^{k}(t) = \frac{Q}{L_k}$
d. Update Best Solution (optionally use elitism to reinforce best-so-far tour more strongly)
e. $t \leftarrow t+1$
Return the best solution found

2.1.3. Firefly Optimization (FF)

Three fundamental ideas comprise the firefly algorithm:

- There exist no male or female fireflies; all of them can attract mates.
- The fireflies are attracted to bright objects, and this attraction reduces with increasing distance. Among two fireflies, the brighter one attracts the other.
- The brightness of an individual firefly depends on its performance concerning an objective function.

Brightness Y_{ji} and attraction X_{ji} can mathematically then be expressed as:

$$Y_{ji} = Y_0 * e^{-\gamma * \alpha i j^2} \tag{10}$$

$$X_{ji} = X_0 * e^{-\gamma * \alpha j j^2} \tag{11}$$

In the equation, Y_0 denotes the initial brightness, and X_0 denotes the initial attraction, γ is the light absorption coefficient, and α is the distance between firefly j_{th} and i_{th}. In this case, fireflies with great brightness (ff_i) attract fireflies with lesser brightness $\left(ff_j\right)$. mathematically, this is represented as:

$$x_i = x_i + \beta o e^{-\gamma r_{ij}^2}\left(x_j - x_j\right) + \alpha.(rand - 0.5) \tag{12}$$

Where:

The first term holds the position. The second term moves i towards j based on attractiveness. The third term randomizes the moves to escape from the local optimum. randUniform(0,1) is an actual random vector. The firefly algorithm is described in-

Algorithm 3. Firefly Optimization (FF)
Initialization:
The position of the n fireflies $x_i \in Rd$ is randomly initialized.
For all fireflies, the light intensity $I_i = (x_i)$ is calculated.
Main Optimization Loop (for $t = 1$ to T):
For each firefly $i = 1n$:
For each firefly $j = 1n$:
If $Ij > I$:
- Move firefly i toward firefly j using:

$$x_i = x_i + \beta o e^{-\gamma r_{ij}^2}\left(x_j - x_j\right) + \alpha.\epsilon$$

Evaluate new fitness (x_i)
Track the best solution found so far:

$$x_{best} = \arg\min_i f(x_i)$$

2.2 Proposed Methodology

2.2.1. Hybrid Optimization Based Image Denoising Filter

This study presents a filter specifically developed for denoising images of bone fractures. The proposed model works through a digital filter that utilizes a hybrid bio-inspired optimization algorithm to estimate the optimal coefficients. A block diagram depicting the configuration of the medical image filter is shown in Figure 2.

The main objective is the guided decimation box filter optimized with a hybrid ACO swarm optimization (PSOACO) algorithm. The process starts with the ingestion of the noisy image, after which the red, green, and blue (RGB) channels is treated independently. Processing of each channel proceeds with low-pass filtering. The Butterworth low-pass filter has been utilized here to eliminate the effects of high-frequency noise while still maintaining the critical low-frequency attributes of the digital image.

LPF of order n is mathematically represented as

$$H(i,j) = \frac{1}{1 + \left[\sqrt[2]{\frac{(i^2 + j^2)}{D_0}} \right]^{2n}} \tag{13}$$

The term $\sqrt[2]{(i^2 + j^2)}$ represents the Euclidean distance from the origin to any point (i, j) in the frequency domain of the input image, where D_0 the cutoff frequency is. Following the filtering with a low-pass filter (LPF), a two-level 2-dimensional discrete wavelet transform (2-DWT) is performed on each of the RGB sub-components of the image. The relevant coefficients were selected from the resulting sub-bands and processed by a decimation box filter of size $n \times n$ with a down-sampling factor of n. Padding was applied to each sub-band before this operation to ensure proper filtering.

The noisy coefficients will then be further enhanced by the proposed Hybrid Particle Swarm Optimizer combined with Ant Colony Optimization (PSOACO). This is followed by applying thresholding functions to yield the Optimized Filter Coefficients (OFC), refined further into coefficients: R_{coeff}, G_{coeff}, and B_{coeff}, which refer to the red, green, and blue channels, respectively. These coefficients will then suppress noise in each of the components. Each optimized solution's performance is assessed so as to obtain the best-performing coefficient set. Coefficients related to zero-padding shall now be dropped, with others subjecting to inverse discrete wavelet transform (IDWT) for final construction of the denoised image.

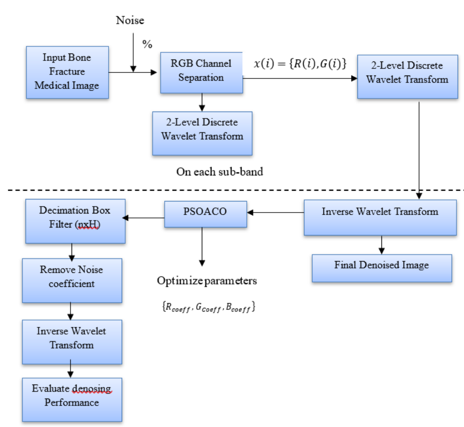


Fig 2. Diagram of medical image filtering using proposed novel filter

Algorithm 4. Proposed Hybrid optimization based image filter

1. Extract the RGB channels from the input image $P_{m,n}$ Yielding $\{P_R, P_G, P_B\}$
2. For each channel $P \epsilon \{P_R, P_G, P_B\}$ perform the following:

   (a) Initialize an $n \times m$ decimation box filter with given decimation factor.
   (b) Apply 2 level Discreate wavelet Transform (DWT) to obtain wallet coefficient P_{coeff}
   (c) Filter P_{coeff} using the initialized decimation box to obtain coefficients Db_I.

(d) Optimize the decimation box filter coefficient Db_I using Hybrid Particle Swarm Optimizer with Ant Colony Optimization (PSOACO), resulting in optimized coefficients $Db_{opt.}$

(e) Apply inverse 2 level DWT on $Db_{opt.}$ To reconstruct the filtered channel O_I.

3. After processing all 3 channels, combine the filtered channels O_R, O_G, O_B to produce the final denoised RGB image O_{xy}.
4. Return the denoised image O_{xy}.

It first takes an input image $P_c(m,n)$, which is then artificially corrupted with random noise according to a noise factor $F_{m,n}$. The resultant corrupted image, therefore, can be expressed mathematically as follows:

$$P_c(m,n) = P_{m,n} + F_{m,n} \tag{14}$$

Different levels of noise that picture denotions get attributed to differences between noise variances. Some filters are specifically designed to minimize these distortions. Bilateral filters are most commonly used for image denoising; however, they usually suffer from significant computation burden and produce undesirable artifacts of radiant reversals, resulting in image distortion. Instead, guided filters serve as digital filters for denoising purposes. The denoised image is denoted by $P_{m,n}$. However, these guided filters are linear filters[21] and can be mathematically expressed as follows:

$$P_{m,n} = \sum Wt_{m,n} * G * P_{c(m,n)} \tag{15}$$

In this context, (m,n) denotes kernel weight associated with the decimation box filter while G stands for guidance image for preserving structural information during the filtering process. In coefficient form, the denoised images can be expressed as follows:

$$P_{(m,n)} = a_k * G_y * b_k \tag{16}$$

In this case, a_k and b_k are the linear coefficients at pixel k corresponding to the guided image G_x; thus, the purpose of the filter is to minimize a cost function that measures the difference between the original and denoised image alongside regularization imposing on the parameters. In other words, (a_k,b_k), the cost function, is defined as:

$$\phi(a_k, b_k) = \sum\nolimits_{(m,n)\epsilon w_k} a_k * G_k + b_k - P_{(m,n)}{}^2 + \epsilon a_k^2 \tag{17}$$

Here,

$$a_k = \frac{\frac{1}{|w_k|}\sum (m,n)\epsilon w_k \left(G_x . I_{m,n}\right) - \mu G \mu I}{\alpha_G^2 + \epsilon} \tag{18}$$

$$b_k = I\ - a_k \mu G$$

The estimation in this scheme is based on fixing linear coefficients to constant values within each decimation box region. The parameter ϵ measures the amount of smoothing considered for regularization; w is defined as the decimation box area; and σ and μk represent the variance and mean value of the decimation box, respectively, used as discussed previously. In every decimation box, the variance in the denoised output may differ pixel by pixel; therefore, a constant result is obtained by averaging the pixel values within the considered box. Guided filtering is especially useful for edge preservation and smoothing, working together as a feature-aware and noise-suppressive filter. The filter examines regions in an image and distinguishes between the ones labeled semi-automatically as "flat," with low variance, and those considered "high variance" because they contain textures or edges. For flat regions, pixels located centrally within uniform patches are averaged and replaced with that mean value to reduce noise. In high variance areas (edges or textured areas), pixels retain their original values to preserve structural details. The degree of smoothing stems from a regularization parameter that controls the shaping of low-variance (to be smoothed) and high-variance (to be preserved). Smoothing is performed in a patch as long as its variance surpasses certain regularization threshold; otherwise, it remains unchanged. Furthermore, pixels nearby edges are rendered negligible by the kernel of the guided filter-therefore avoiding distortion of the edge representation. Most importantly, the filter counteracts gradient reversal artifacts, inherent within traditional filters. However, the method also tends to manifest high computational complexity as the kernel size expands. For better efficiency, one way to evaluate the output of a box filter would be to use the integral image method where every box-filtering summation could work off of one summation operation.

Filters the optimum smoothness into a defined neighborhood region to effectively filter out the noise of an image. The smoothness degree will serve as the flexible threshold for the standard deviation of the neighborhood, making a fit balance between denoising and detail.

If the input image has a size of $P \times Q$, it is split into $n \times n$ kernels, and the mean of each kernel is taken as the output of the decimation box. This hybrid bio-inspired optimization method in this study identifies the optimal filter coefficients depending on the level of noise present in the input image, while the guided filter itself is structured as an $n \times n$ downsampling box where n is also the downsampling factor. This is then optimized by the PSOACO algorithm.

The time complexity of the computation for the filter size is given by $(m \times N)$ while m is the size of the filter and N is the number of pixels in the full image. For large filter kernels, this may significantly increase the computational time. The integral image technique allows the computation of box filters to be reduced to linear time complexity (N), which remarkably improves the efficiency.

This Hybrid Particle Swarm Optimizercum Ant Colony Optimization (PSOACO) algorithm is the decimation filter co-efficient optimization method applied in this study. This hybrid technique composes the strengths of Ant Colony Optimization method and Particle Swarm Optimization in denoising non-linear challenges, which are effective. Furthermore, the motivation behind hybrid model development was to upgrade the capabilities of global search while counteracting the limitations of the two algorithms-Pso and ACO.

Although it is simple and efficient, the drawback of PSO is that it gets caught up in premature convergence in higher dimensions, and consequently, most of the time, it converges into a local optimum. Despite being costlier in terms of computation with respect to iteration, ACO performs better with respect to global exploration. The proposed PSOACO leads to the amalgamation of behavior of nest building of ACO and refinement of standard PSO particles, thereby improving the convergence speed and solution quality.

First, the parameters such as the number of particles, swarm size, acceleration constants, inertia weights, and boundary limits for each particle are initialized. This setup forms the basis for an iterative optimization process to accurately tweak the decimation filter coefficients for efficient noise removal in medical images. The initial population is randomly generated with positions denoted by: $\Theta_i = \{\theta_1, \theta_2, \ldots, \theta_N\}$ where Θ_i represents the position of the i-th particle in the search space. The fitness of each particle is evaluated using the function:

$$f(\Theta_i) = \emptyset_i \tag{19}$$

Based on this, the personal best $\left(Best_i^{(t+1)}\right)$ and the global best $\left(Global_i^{(t+1)}\right)$) are updated as follows:

Personal Best Update Rule:

$$Best_i^{(t+1)} = \begin{cases} \Theta_i^{(t+1)} iff \left(\Theta_i^{(t+1)}\right) < f\left(Best_i^{(t)}\right) \\ Best_i^{(t)} otherwise \end{cases} \tag{20}$$

Global Best Update Rule:

$$Globel^{(t+1)} = \arg main\left(Best_1, Best_2, \ldots, Best_N\right) \tag{21}$$

The algorithm updates the nest populations by excluding deflection on the Lévy flight mechanism in PSOACO. Instead, the personal best positions obtained from the PSO algorithm are considered for this purpose. Therefore, some of the worst nests (denoted as Wnest) are discarded and better nests are considered for further iteration (as shown in Algorithm 5). Further, PSOACO introduces an additional adaptive thresholding function for further refinement of the selection criteria, mathematically defined as follows:

$$\Omega(Best, \kappa) = \begin{cases} Best - \dfrac{\kappa^2}{2 * Best} & Best > \kappa \\ 0.5 * \dfrac{Best^3}{\kappa^2} & Best \leq \kappa \end{cases} \tag{22}$$

Here, $Best$ indicates the optimal selected nest value or filter coefficient, while the symbol κ represents the adaptive threshold value. This threshold gets updated dynamically in each iteration, assigning it minimum value among the swarm parameters of all the generated populations. Lastly, the following equations are executed to update the velocity and position of particles.:

$$vel_n(t+1) = \omega * vel_n(t) + C_1 Levy(\varphi) * \left(Best_n^t - \theta_n^t\right) + C_2 \oplus Levy(\varphi) * \left(Global^t - \theta_n^t\right) \tag{23}$$

$$\theta_n^{t+1} = \theta_n^t + vel_n^{(t+1)} \quad (24)$$

To empirically establish the optimal value of κ for adaptive thresholding function, optimization techniques are employed for minimizing mean square error (MSE) through identification of κ values, during this optimization process, where the very lowest MSE is intended to be produced.

Algorithm 5: PSOACO Optimization
Input Parameters:
Np: Number of particles (filter coefficients)
Nv: Number of variables
Itrtmax: The maximum number of iterations
Domain: The search space for initialization
Output:
nest_best: The most optimal solution found
Initialization Phase:
Randomly initialize particle positions posθ within the defined domain for Δ and δ.
Assign the initial velocities velθ randomly.
Set each particle's personal best $Best \leftarrow \theta_o$.
Identify the global best solution $Globel \leftarrow Best\left[argmin\left(f(Globel)\right)\right]$.
Adaptive Control:
Compute hybrid control parameters using an adaptive thresholding function (as defined in **Equation 22**).
This threshold will be used to influence the solution selection during the optimization.
Iterative Optimization Process:
For each iteration until $itr = Itrt_{\max}$:
a. Update the fitness values for all particles using the objective function $f(X)$.
b. From $Global$ select the top-performing solution as the best ACO candidate.
c. Choose a nest randomly at the first iteration $(itr = 0)$.
d. If the fitness of the current nest is better than its previous state:
– Replace the older nest with a new solution generated.
e. Remove a percentage of the poorest nests (worst solutions).
f. Re-evaluate and rank all solutions.
g. Store the best-performing nest as $nest_{best}$.
Output the final best solution.

3 Result Analysis and Discussions

Furthermore, the effectiveness of the suggested approach will be compared to the outcomes of different filtering strategies used on the same collection of photos. The purpose of this comparison is to highlight the benefits and drawbacks of the suggested strategy. A laptop running Windows 10 and equipped with an Intel Core i7 2.2 GHz processor and 16 GB of RAM was used for the implementation. This setup provides the best conditions for managing high-resolution photos and effectively executing intricate algorithms.

Numerical metrics such as Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and the Structural Similarity Index Method (SSIM) will be used to evaluate how well the filtering systems are presented. By evaluating elements like structural similarity and fidelity to the original image, these metrics provide a thorough evaluation of image quality. The effectiveness of the suggested approach can be thoroughly examined in relation to well-established filtering techniques by applying these evaluation criteria to the examination images. This will ultimately shed light on the method's possible advantages in image processing applications. Measures of image quality evaluation are crucial for determining how faithfully processed images compare to their original counterparts. Three commonly used metrics are as follows: Mean Squared Error (MSE), Structural Similarity Index (SSIM), and Peak Signal-to-Noise Ratio (PSNR).

3.1 Mean Squared Error (MSE)

Definition: Definition: MSE quantifies the average formed difference among the pixel values of the original and the handled image.

$$MSE = \frac{1}{N}\sum_{j=1}^{N}(J(j) - L(j))^2 \tag{25}$$

Where:

J is the original image.

L is the processed image.

N is the total number of pixels.

Interpretation: A lower MSE value designates improved quality. However, it doesn't account for perceived visual differences.

3.2 Peak Signal-To-Noise Ratio (PSNR)

Definition: PSNR is a logarithmic metric used to assess the ratio between the determined likely signal power of an image and the power of the noise present in it. It is derived from MSE.

$$PSNR = 10\log_{10}\left(\frac{MAX^2}{MSE}\right) \tag{26}$$

Where: MAX is the maximum possible pixel value (e.g., 255 for an 8-bit image).

Interpretation: Higher PSNR values designate improved image quality. PSNR values above 30 dB are typically considered good.

3.3 Structural Similarity Index (SSIM)

Definition: SSIM assesses the similarity among two pictures by associating luminance, contrast, and structure. It's designed to model human visual perception more closely than MSE or PSNR.

$$SSIM(x,y) = \frac{\left(2\mu_x\mu_y + C_1\right)\left(2\sigma_{xy} + C_2\right)}{\left(\mu_x^2 + \mu_y^2 + C_1\right)\left(\sigma_x^2 + \sigma_y^2 + C_2\right)} \tag{27}$$

Where:

x and y are local image patches.

μ represents the mean.

σ represents the variance and covariance.

$C1$ and $C2$ are small constants to avoid instability.

Interpretation: SSIM values choice from -1 to 1, where 1 designates faultless comparison. It's more sensitive to perceptual differences than MSE and PSNR.

3.4 Dataset

The FracAtlas data set is found at https://figshare.com/articles/dataset/The_dataset/22363012 and https://figshare.com/articles/dataset/The_dataset/22363012. By offering a sizable, meticulously annotated collection of digital radiographs, the FracAtlas dataset aims to address the difficulties in computer-aided bone fracture diagnosis. Although digital radiography is frequently used to diagnose fractures, it necessitates professional intervention, which takes time and requires a high level of training. While the lack of extensive, annotated X-ray datasets has impeded progress in fracture classification, localization, and segmentation, the quick development of computer vision algorithms has generated interest in automated diagnosis. FracAtlas was created to close this gap by utilizing 4,083 X-ray pictures that were gathered from three significant Bangladeshi hospitals. Two skilled radiologists and an orthopedist used the makesense.ai labeling platform to manually annotate these images. Two skilled radiologists and an orthopedist used the makesense.ai labeling platform to manually annotate these images. The dataset comprises 717 images with 922 fracture instances, each labeled with global classification labels and pixel-level masks and bounding boxes.

A comparative ablation study on the proposed hybrid optimization technique against other bio-inspired algorithms. The analysis focuses on the performance of Hybrid Particle Swarm Optimizer with Ant Colony Optimization (PSOACO) combined with decimation box filters for medical image processing. The learning parameters of each optimization method are detailed in Table 1.

Figure 3 shows the convergence behavior of the various optimization-based filtering algorithms like PSO, Firefly, ACO, and proposed PSOACO. The evaluation was done for 100 iterations, and the graph plotted the fitness value concerning the epochs. Among all these methods, PSOACO was found to possess the fastest convergence as seen in its sharper and more stabilized fitness curve. This hybrid approach thus perfectly retrieves the optimal value of cost function, proving its better optimization capability.

Table 1. Parameters for bio-inspired optimization algorithms.

Algorithm	Parameter
PSO	Size of particle = 20 No. of iterations = 100 Max inertia wt. = 0.94 Min inertia wt. = 0.39
FF Algorithm	Size of fireflies = 15 No. of iterations = 100 Absorption coefficient = 1 Light absorption coefficient = 1 Mutation value = 0.19
ACO	Size of nests = 15 No. of iterations = 100 Mutation probability = 0.1 Scaling factor = 1.48
PSOACO	Size of particle = 15 Size of nests = 10 Max inertia wt. = 0.88 Min inertia wt. = 0.39 No. of iterations = 100 Mutation probability = 0.11 Scaling factor = 1.48

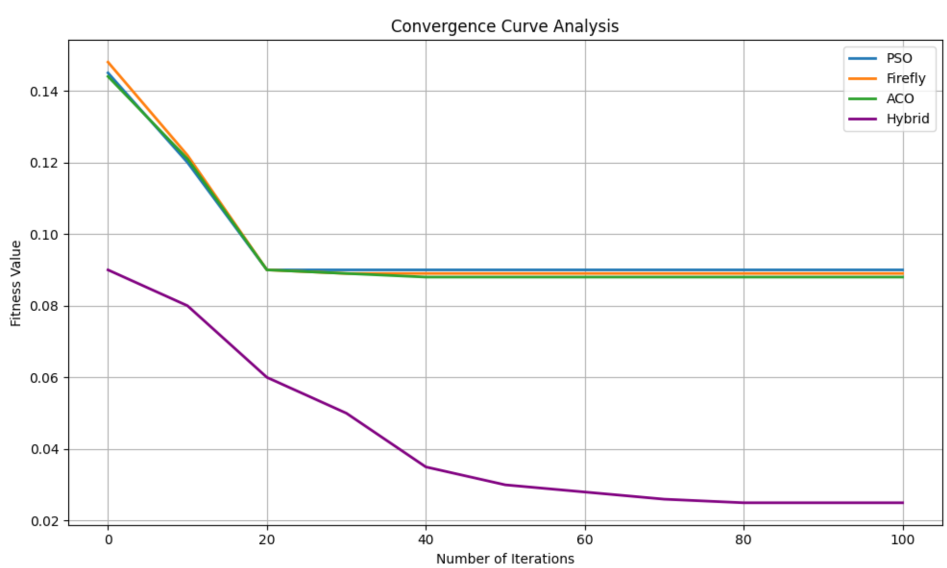


Fig 3. Convergence analysis of optimization-based filters

Table 2 depicts the performance metrics comparison of various bio-inspired optimization techniques, namely: PSO, Firefly, ACO, and the proposed PSOACO model, at different low-pass filter orders. Cells have been analyzed for the three image quality metrics: PSNR, MSE, and SSIM. The bone fracture image is 192 × 192 pixels and has been degraded by noise with a variance of 0.01. Overall, the PSOACO model outperformed other models in each metric and filter variant, showcasing its robustness and utility in medical applications for image filtering purposes.

Table 2. Performance analysis of bio-inspired optimization algorithms with respect to filter order.

	Pixel density	PSO	FF	ACO	PSOACO
	10%	33.3987	36.1323	31.4397	37.9439
	20%	38.9276	35.2459	32.9292	35.3943
PSNR	40%	32.1624	34.1418	33.1863	36.9162
	60%	33.4565	33.5689	34.7992	35.2992
	70%	32.7391	33.4413	31.4713	38.6565
	10%	1.3706	0.31701	1.1703	0.0305
	20%	2.6422	0.84014	2.6465	0.0601
MSE	40%	3.0106	1.07643	3.1042	0.1021
	60%	2.6176	0.84601	5.6162	0.0623
	70%	1.0714	0.02360	1.0141	0.0171
	10%	0.9841	0.9799	0.9776	0.9987
	20%	0.9591	0.9829	0.9599	0.9975
SSIM	40%	0.9682	0.9874	0.9698	0.9976
	60%	0.9568	0.9869	0.9638	0.9979
	70%	0.9896	0.9899	0.9889	0.9985

In Figure 5, the comparative illustration of bio-inspired optimization algorithms is demonstrated with respect to Particle Swarm Optimization (PSO), Firefly Algorithm (FF), and Ant Colony Optimization (ACO) against the proposed Hybrid Particle Swarm Optimizer with Ant Colony Optimization (PSOACO) on all three datasets, as previously described.

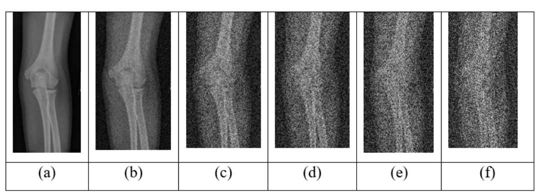


Fig 4. (a) Standard Bone Fracture image. (b) Image adds 10% noise density. (c) Image adds 20% noise density. (d) Image adds 40% noise density. (e) Image adds 60% noise density. (f) Image adds 70% noise density

Figure 6 shows the complexity analysis of the presented PSOACO model subjected to different noise conditions. On average, the model portrays low computational complexity with execution taking about one second. In examining performance, Figure 7 deals with digital filters applied in the spatial domain for image processing tasks. To provide a comprehensive overview, many state-of-the-art techniques were incorporated into the analysis such as: Deep Convolutional Neural Network (DNCNN) [22], Enhancement novel Cuckoo Search Optimization (ECSO) [23], Adaptive Modified Cuckoo Search Optimization (AD-MCSO) [24], Deep Image Prior with Regularization by Denoising (DeepRED) [25], Enhanced Grey Wolf Optimization (GWO) [26], Unsupervised Deep Neural Network (DNN) [27], Ant Colony Optimization (ACO) [28], Bi-dimensional Empirical Mode Decomposition (BEMD) [29], and Hybrid model Genetic Algorithm (HGA) [30]. Approximate PSNR and SSIM value comparatives of the PSOACO model with existing state-of-the-art techniques are given in Figure 7. From the comparative graphs, it is evident that PSOACO has consistently outshined the other models in the PSNR and SSIM metrics. PSNR acts as a quantitative measure of quality assessment for an image, whereas SSIM provides a qualitative measure. The results here suggest that PSOACO algorithm preserves the quality of images in the denoising process.

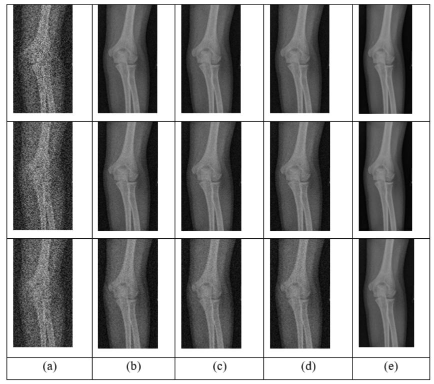


Fig 5. Results of various filters applied to a bone fracture image: (a) Image with different noise densities. (b) Output of the PSO Filter. (c) Output of the FF. (d) Output of the ACO. (e) Output of the PSOACO. Rows 1 to 3 display the processed results from different filters on a bone fracture image corrupted by 20%, 40%, and 60% noise densities, respectively

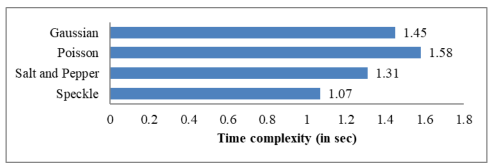
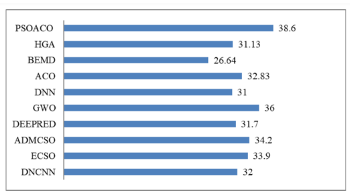
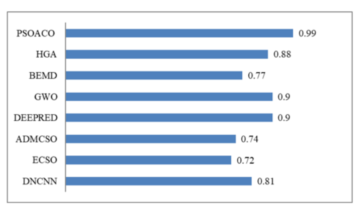


Fig 6. Time Complexity



(a) PSNR comparison



(b) SSIM Comparison

Fig 7. Comparative state-of-the-art

4 Conclusion

Noise in bone fracture images makes it difficult to accurately detect and analyze fractures. Most denoising processes are information-lossy at best and over-smoothed at worst; they also tend to demand more computational resources. This underlines the need for better optimized denoising. After all, denoising is essentially a pre-processing step for all other diagnostic tasks. This paper presents a new digital filtering approach for bone fracture medical image denoising. It is based on the Hybrid Particle Swarm Optimizer with Ant Colony Optimization (PSOACO) coupled with a guided decimation box filter to restore noisy images. An ablation study was conducted to evaluate the performance of different optimization algorithms. For the evaluation of results, input bone fracture medical images were artificially corrupted with various types of noise and further processed with the proposed hybrid-optimization-based filter. The method has been successful in establishing optimal filter coefficients and smoothing specifics that adapt according to noise image resolution and decimation parameters. To test the efficacy of the proposed filter, its performance was compared against several of the existing denoising techniques. The results reveal that the proposed model performs significantly better than others in terms of quantification. Future work involves extending this approach to real time medical imaging applications.

References

1) Hu J, Fan S, Caballero-Águila R, Zhu M, Zhang G. Distributed fusion filtering for multi-rate nonlinear systems with binary measurements under encryption and decryption scheme. *Information Fusion*. 2025;117:102900. https://doi.org/10.1016/j.inffus.2024.102900.
2) Abbass MY, Kwon KC, Alam MS, Piao YL, Lee KY, Kim N. Image super resolution based on residual dense CNN and guided filters. *Multimedia Tools and Applications*. 2021;80:5403–5421. https://doi.org/10.1007/s11042-020-09824-3.
3) Deeplpf: Deep local parametric filters for image enhancement. 2020. https://doi.org/10.1109/CVPR42600.2020.01284.
4) Andreev A, Cattan G, Congedo M. The Riemannian Means Field Classifier for EEG-Based BCI Data. *Sensors*. 2025;25(7):2305. https://doi.org/10.3390/s25072305.
5) James BP, Leung MF, Vaithiyanathan D, Mariammal K. Optimal Realization of Distributed Arithmetic-Based MAC Adaptive FIR Filter Architecture Incorporating Radix-4 and Radix-8 Computation. *Electronics*. 2024;13(17):3551. https://doi.org/10.3390/electronics13173551.
6) Hamici H, Kanan A, Al-hammuri K. Optimized fir filter using genetic algorithms: A case study of ecg signals filter optimization. *BioMedInformatics*. 2023;3(4):1197–1215. https://doi.org/10.3390/digital3040071.
7) Farooq Y, Savaş S. Noise removal from the image using convolutional neural networks-based denoising auto encoder. *Journal of Emerging Computer Technologies*. 2024;3(1):21–28. https://doi.org/10.57020/ject.1390428.
8) Karthikeyan V, Raja E, Pradeep D. Energy based denoising convolutional neural network for image enhancement. *The Imaging Science Journal*. 2024;72(1):105–120. https://doi.org/10.1080/13682199.2023.2198350.
9) Torun O, Yuksel SE, Erdem E, Imamoglu N, Erdem A. Hyperspectral image denoising via self-modulating convolutional neural networks. *Signal Processing*. 2024;214:109248. https://doi.org/10.1016/j.sigpro.2023.109248.
10) Cluster medical image segmentation using morphological adaptive bilateral filter based BSA algorithm. IEEE. 2021. https://doi.org/10.1109/ICOEI51242.2021.9452816.
11) Reduction of Gaussian noise from Computed Tomography Images using Optimized Bilateral Filter by Enhanced Grasshopper Algorithm. IEEE. 2022. https://doi.org/10.1109/ICOEI51242.2021.9452816.
12) Deng J, Hu C. A new multi-scale CNN with pixel-wise attention for image denoising. *Signal, Image and Video Processing*. 2024;18(3):2733–2741. https://doi.org/10.1007/s11760-023-02944-x.
13) Real-Time Image Denoising Using Deep Learning for Cybersecurity Applications. 2024. https://doi.org/10.1145/3660853.3660936.
14) Zhang Q, Xiao J, Zhang S, Lin JC, Tian C, Zhang C. Texture-guided CNN for image denoising. *Multimedia Tools and Applications*. 2024;83(23):63949–63973. https://doi.org/10.1007/s11042-023-17670-2.
15) Soniya S, Sriharipriya KC. Edge and texture aware image denoising using median noise residue U-net with hand-crafted features. *PeerJ Computer Science*. 2025;11:e2449. https://doi.org/10.7717/peerj-cs.2449.
16) Han L, Liu X, Wang Q, Xu T. Wild horseshoe crab image denoising based on CNN-transformer architecture. *Scientific Reports*. 2025;15(1):11622. https://doi.org/10.1038/s41598-025-96218-w.
17) Batool I, Imran M. A dual residual dense network for image denoising. *Engineering Applications of Artificial Intelligence*. 2025;147:110275. https://doi.org/10.1016/j.engappai.2025.110275.
18) Tiantian W, Hu Z, Guan Y. An efficient lightweight network for image denoising using progressive residual and convolutional attention feature fusion. *Scientific Reports*. 2024;14(1):9554. https://doi.org/10.1038/s41598-024-60139-x.
19) Chae J, Hong S, Kim S, Yoon S, Kim G. CNN-based TEM image denoising from first principles. *arXiv preprint arXiv:250111225*. 2025. https://doi.org/10.48550/arXiv.2501.11225.
20) Bao F, Lei Y, Jia Y, Du H, Gao C, Zhang Y. An image denoising method based on the nonlinear Schrödinger equation and spectral subband decomposition. *Computer Vision and Image Understanding*. 2023;237:103840. https://doi.org/10.1016/j.cviu.2023.103840.
21) Jia H, Yin Q, Lu M. Weighted guided image filtering with entropy evaluation weighting. *Computers & Graphics*. 2023;117:114–123. https://doi.org/10.1016/j.cag.2023.10.022.
22) Chinnaiyan AM, Sylam BWA. Deep demosaicking convolution neural network and quantum wavelet transform-based image denoising. *Network: Computation in Neural Systems*. 2024;p. 1–25. https://doi.org/10.1080/0954898X.2024.2358950.
23) Kamoona AM, Patra JC. A novel enhanced cuckoo search algorithm for contrast enhancement of gray scale images. *Applied Soft Computing*. 2019;85:105749. https://doi.org/10.1016/j.asoc.2019.105749.
24) Asokan A, Anitha J. Adaptive Cuckoo Search based optimal bilateral filtering for denoising of satellite images. *ISA transactions*. 2020;100:308–321. https://doi.org/10.1016/j.isatra.2019.11.008.
25) Sun H, Peng L, Zhang H, He Y, Cao S, Lu L. Dynamic PET image denoising using deep image prior combined with regularization by denoising. *IEEE Access*. 2021;9:52378–52392. https://doi.org/10.1109/ACCESS.2021.3069236.
26) Reduction of Gaussian noise from Computed Tomography Images using Optimized Bilateral Filter by Enhanced Grasshopper Algorithm. IEEE. 2022. https://doi.org/10.1109/ICOEI51242.2021.9452816.
27) Gong K, Han P, Fakhri GE, Ma C, Li Q. Arterial spin labeling MR image denoising and reconstruction using unsupervised deep learning. *NMR in Biomedicine*. 2022;35(4):e4224. https://doi.org/10.1002/nbm.4224.
28) Yadava PC, Srivastava S. Denoising of poisson-corrupted microscopic biopsy images using fourth-order partial differential equation with ant colony optimization. *Biomedical Signal Processing and Control*. 2024;93:106207. https://doi.org/10.1016/j.bspc.2024.106207.
29) Satapathy LM, Das P. A New Biomedical Image Denoising method using an adaptive multi-resolution technique. *WSEAS Transactions on Systems and Control*. 2022;17:16–24. https://doi.org/10.37394/23203.2022.17.2.
30) Ibiere C, Anireh B, Matthias V, Daniel. A Hybrid Model for denoising in Data Mining and Exploration. *International Journal of Computer Sciences and Engineering*. 2024;12(6):1–12. https://doi.org/10.26438/ijcse/v12i6.112.