Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105
Volume **9**, Number **2**, May **2024**

Image Fuzzy Enhancement Based on Membrane Computing Particle Swarm Optimization Algorithm

Xiao-Mao Hou

Automation and Information Engineering College
Hunan Chernical Vocational Technology College
Zhuzhou 412000, P. R. China
houxiaomao_1234@163.com

Liu-Ping Cai

Graduate School Department
Saint Paul University
Tuguegarao 3500, Philippines
2689815@qq.com

Yan-Hui Wang*

School of Computer Science and Engineering
Hunan University of Information Technology
Changsha 410151, P. R. China
yanhuiwang@hnuit.edu.cn

*Corresponding author: Yan-Hui Wang
Received June 26, 2023, revised September 17, 2023, accepted November 9, 2023.

Abstract. *Image fuzzy enhancement is a research hotspot in the field of image processing, which aims to recover enhanced beginning clear images from degraded images. Based on the research of traditional particle swarm optimization algorithm and fuzzy enhancement algorithm, an image fuzzy enhancement method based on membrane computing particle swarm algorithm is proposed. Firstly, in order to make full use of the sparse characteristics of the clear image, the coefficient decomposition under wavelet domain and tightly supported wavelet domain is performed on the image respectively. Then, a joint optimisation model is constructed using L1 parametric constraints to achieve pretzel noise cancellation. Next, the MMH-PSO algorithm is designed by improving the particle swarm algorithm using membrane computing and Metropolis-Hastings sampling. Based on the simulated annealing algorithm temperature drop process, Metropolis-Hastings sampling is used to add randomness to the particle swarm algorithm so that it has the ability to jump out of the local optimum. The use of membrane computing enhances the parallelism of the particle swarm algorithm and can reduce the time complexity in solving complex problems. Finally, MMH-PSO is used to simultaneously search out the magnitude of the two fuzzy parameters in the traditional fuzzy enhancement algorithm in order to improve the accuracy of the algorithm. The experimental results show that the proposed algorithm has better SSIM values than the traditional fuzzy enhancement algorithm, which effectively improves the image quality and makes the image edge information more abundant.*
Keywords: Image enhancement; particle swarm algorithm; membrane computing; simulated annealing algorithm; pretzel noise

1. **Introduction.** In practice, in many cases, due to the ambient light, image sensor quality and other factors, the captured image may suffer from a certain degree of blurry, which affects the clarity and detail of the image, or in more serious cases, may even lead to the loss of image information [1,2]. In such cases, fuzzy enhancement can be used to improve the quality of the image and is also of high research interest.

In the field of aviation, the clarity and detail of the image presentation is crucial. For example, for aerial imagery, if the outline of the target object can be well recovered [3,4], this is important for developing safe flight routes and for performing algorithms such as target tracking and identification. In medical production, accurate display and analysis of information such as lesion areas, blood vessels, and organ edges requires adequate image clarity and detail presentation. Therefore, image fuzzy enhancement techniques can be applied to medical imaging [5,6], such as the enhancement and analysis of X-ray, CT and MRI images, to improve the precision and accuracy of doctors' diagnosis of medical conditions.

Image fuzzy enhancement techniques can effectively improve the clarity and detail information of an image, enhancing the quality and viewing experience of the image. For domain-specific image recognition tasks, such as medical imaging [7,8], military target recognition [9], etc. Fuzzy enhancement techniques can improve recognition efficiency and reduce labour costs and false positive rates while ensuring recognition accuracy. As an important part of image processing technology, image fuzzy enhancement technology can promote the development of artificial intelligence technology [10,11], such as image recognition, target tracking and autonomous driving.

Image fuzzy enhancement is one of the fundamental and core technologies in the field of image processing, and it has important applications and research value in many fields. Although many fuzzy enhancement techniques have emerged, the challenges they face are still relatively serious. With the continuous development of image processing technology, image fuzzy enhancement technology will also be continuously improved in the following aspects [12]: (1) improving clarity and detail presentation; (2) improving computing efficiency and practicality; (3) enhancing edge retention capability; and (4) realising adaptive processing. In the practical application scenario of fuzzy enhancement technology, the computational efficiency and practicality of the algorithm are very important factors. Therefore, in this work, by improving the algorithm framework and optimising the algorithm design, the computational speed and accuracy of the algorithm are improved to make the technology more practical in practical applications.

1.1. **Related Work.** In the research process of image fuzzy enhancement techniques, many fuzzy enhancement techniques based on different algorithm implementations have emerged, and the common techniques include: (1) fuzzy enhancement techniques based on air-domain filtering [13]; (2) fuzzy enhancement techniques based on frequency-domain filtering [14]; (3) fuzzy enhancement techniques based on deep learning [15]; (4) fuzzy enhancement techniques based on (4) image enhancement methods based on optimization theory methods [16].

In contrast to other image enhancement methods, methods based on optimisation theory can optimise the image enhancement process in an adaptive or automated manner and retain the maximum amount of original image features and details. Specifically, optimisation algorithms can improve image quality by adjusting the values of parameters in the enhancement process according to their chosen objective function (e.g. image sharpness, contrast or colour saturation). In addition, these methods can be optimised for different image types and noise models, making them highly applicable and flexible.

Optimisation-based methods are also highly efficient, accurate and easy to automate, making them advantageous in large-scale image processing. As the specific application goals and objectives of image enhancement differ for different operation objects, the methods used in the actual image enhancement process are also somewhat different, so in some applications the best enhancement effect needs to be achieved by adding optimisation algorithms to optimise the relevant functions.

Image enhancement methods based on optimisation theory mainly include the following types of algorithms [17,18]: genetic algorithms, particle swarm optimization (PSO) algorithms, ant colony algorithms, artificial immune algorithms, simulated annealing algorithms, etc.

Genetic algorithms use the principles of inheritance and variation in the natural evolutionary process to perform search and optimisation. For the image enhancement problem, the image can be treated as an individual, and new individual can be generated through crossover and mutation operations, and the performance of the individuals can be evaluated through the fitness function. Deng et al. [19] proposed an indirect image enhancement method based on genetic algorithm, which indirectly transforms image enhancement into a problem of image model parameter optimization to achieve adaptive image enhancement. Anaraki et al. [20] proposed an image enhancement method based on genetic algorithm and neural network theory, which can automatically search for the optimal grayscale domain value of an image to achieve adaptive image enhancement.

Particle swarm algorithms are based on the iterative movement of a large number of small particles in a search space to find the global optimal solution. Compared with genetic algorithm, particle swarm optimization algorithm is faster in calculation and more stable and reliable in image enhancement results. Wan et al. [21] and others put forward an image adaptive enhancement method based on particle swarm optimization algorithm, which can automatically obtain the parameter value of the best gray scale transformation in the beta function and realize the adaptive enhancement of the image. Zhou et al. [22] and others proposed to use particle swarm optimization algorithm to find the adaptive threshold value of the image under the maximum entropy standard, so that the image can be stretched adaptively and the image can be enhanced adaptively.

Through the above analysis, the current research on image enhancement using particle swarm optimisation algorithms, although effective in improving the contrast and detail information of the image itself, does not take into account some of the shortcomings that exist in traditional particle swarm optimisation algorithms. Moreover, the image degradation caused by pretzel noise is not considered in the process of image enhancement.

1.2. **Motivation and contribution.** Membrane computing, also known as P-systems [23], is a computational model abstracted from the structure and function of biological cells, biological tissues, and biological organs with ideal distributed, parallel and non-deterministic characteristics [24].

The combination of membrane computing theory and particle swarm can address some of the shortcomings of traditional particle swarm optimization algorithms to a certain extent. For example, Li et al. [25] proposed a cloud computing method based on particle swarm algorithm and membrane computing theory. The particles within the main membrane perform refined local optimisation search, while the particles within the reference membrane perform global search. Existing algorithms combining membrane computing with particle swarm use a single intra-membrane particle swarm algorithm, resulting in a search mechanism that is still lacking in stochasticity, thus necessitating further research into new combined algorithms with stronger integrated search capability.

The main innovations and contributions of this work include:

(1) Pepper noise is an important factor in image degradation. If these degradation factors are not dealt with properly, the image quality will deteriorate dramatically. Due to the unique impulsive nature of pepper noise, conventional filter methods do not achieve good denoising results. Therefore, this work decomposes the coefficients in the wavelet domain and the tightly supported wavelet domain of the image separately. Then, a joint optimisation model is constructed using the L1 parametric constraint to achieve pepper noise cancellation.

(2) To address the problem that the particle swarm algorithm is prone to fall into local extrema and converge slowly at a later stage, the MMH-PSO algorithm is designed by using membrane computing and Metropolis-Hastings sampling to improve the particle swarm algorithm, which consists of two specific aspects: first, based on the simulated annealing algorithm [26] temperature drop process, Metropolis- Hastings sampling was used to add randomness to the particle swarm algorithm, giving it the ability to jump out of the local optimum. Secondly, the use of membrane computing enhances the parallelism of the particle swarm algorithm and can reduce the time complexity when solving complex problems.

(3) The proposed MMH-PSO algorithm is used to simultaneously search out the magnitude of two fuzzy parameters in the conventional fuzzy enhancement algorithm in order to improve the accuracy of the fuzzy enhancement algorithm.

2. Salt noise cancellation.

2.1. Image noise.

Noise is an unpredictable, random signal that can be understood as a factor that prevents the human sensory organs from understanding the information received from the source. Images are often inevitably affected by noise in the acquisition process, either due to the external environment or to improper personal handling.

As there are various sources of noise, there are different types of noise, such as Gaussian noise, Rayleigh noise, impulse noise and so on. This paper focuses on the most common Gaussian noise as well as pretzel noise for processing and analysis. Gaussian noise, also known as normal noise, has a probability density function that follows a normal distribution. Gaussian noise in images is mainly due to sensor noise caused by poor lighting or high temperatures during information acquisition. For the elimination of Gaussian noise, filtering is often used for processing.

The unique impulsive nature of pepper noise [27], a type of impulsive noise, causes great difficulties in image enhancement. Pepper noise is mainly caused by sensor defects, channel transmission errors, suboptimal transmission media and hardware storage errors. The methods traditionally used to deal with Gaussian noise are no longer effective in dealing with pretzel noise. Image pixels corrupted by pretzel noise can be represented as.

$$\mathbf{y}(i) = \begin{cases} n_{\min}, p/2 \\ n_{\max}, p/2 \\ \mathbf{x}(i), 1-p \end{cases} \tag{1}$$

where p denotes the noise level, $\eta_{\min}$ and $\eta_{\max}$ denote the minimum and maximum value in the dynamic range of the image, respectively [27]. For example, for a grey-scale image their values are 0 and 255 respectively.

2.2. Sparse representation of images.

The sparse nature of images has become a hot topic of research in the field of image restoration. Images have a sparse representation in some transform domain. Therefore, this work decomposes the coefficients of the image in the wavelet domain and in the tightly supported wavelet domain, respectively. Then,

a joint optimisation model is constructed using the L1 parametric constraint to achieve pretzel noise cancellation.

(1) Sparse representation of images in the wavelet domain.

As a successor to the Fourier transform, the wavelet transform is a good reflection of the localisation characteristics in the time and frequency domain. By replacing the infinitely long trigonometric basis of the Fourier transform with a finite length wavelet basis, the wavelet transform is not only able to capture frequency but also to localise to time, while this basis function is scalable and translatable. The wavelet transform of the signal *x(t)* can be expressed as

$$WT_x(a,b) = \frac{1}{\sqrt{a}} \int \mathbf{x}(t)\psi^*(\frac{t-b}{a})dt \tag{2}$$

where $*$ denotes the conjugate complex, a denotes the scale factor, b denotes the displacement factor, and $\psi(t)$ denotes the basic wavelet function or the mother wavelet function.

$$\int_{-\infty}^{\infty} \psi(t)dt = 0 \tag{3}$$

We can observe that the wavelet transform is based on a function of scale and displacement, where the scale factor has a greater influence. The inverse transform of the wavelet transform is defined as

$$\mathbf{x}(t) = \frac{C}{\sqrt{a}} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} WT_{\mathbf{x}}(a,b)\tilde{\psi}(\frac{t-b}{a})db\frac{da}{a^2} \tag{4}$$

where $\tilde{\psi}$ denotes the pairwise function of ψ and C denotes a permissible constant.

We use the matrix $\mathbf{w}$ to denote the wavelet decomposition and $\mathbf{w}^T$ to denote the wavelet reconstruction. Thus, the wavelet coefficients of the image can then be denoted as $\boldsymbol{\alpha}_x = \mathbf{WX}$. The reconstruction of the image can be denoted as $\mathbf{x} = \mathbf{W}^T\boldsymbol{\alpha}_x$. A representation of the image degradation model is shown as follow:

$$\mathbf{y} = \mathbf{H}\mathbf{W}^{\mathrm{T}}\alpha_{\mathbf{x}} \tag{5}$$

(2) Sparse representation of images under tightly supported wavelet domains.

A wavelet transform system is a transform system consisting of orthogonal bases, whereas a tightly supported wavelet transform system is a transform system consisting of non-orthogonal bases. Compared to the wavelet transform, the tightly supported wavelet transform sacrifices regularisation and linear independence for better smoothing properties, tight support and symmetry.

Tightly supported wavelet transform systems usually consist of two parts, the scale function and the tightly supported wavelet function.

$$\phi(t) = \sqrt{2}\sum_k h(k)\phi(2t-k) \tag{6}$$

$$\psi_l(t) = \sqrt{2}\sum_k h(k)\,\phi(2t-k), l = 1,2,\cdots,r \tag{7}$$

where $h(t)$ is a custom function, $\phi(t)$ is a scale function and $\psi(t)$ is a tightly supported wavelet function.

For an arbitrary function f that is squarely integrable, the multiscale representation is shown as follows:

$$f = \sum_{k=-\infty}^{\infty} c_k\phi_k(t) + \sum_{l=1}^{r}\sum_{j=0}^{\infty}\sum_{k=-\infty}^{\infty} d_{k,j,l}\psi_{k,j,l} \tag{8}$$

where c_k is the low-pass tightly supported wavelet coefficients and $d_{j,k,l}$ is the high-pass tightly supported wavelet coefficients.

In this work, the segmented linear tight support wavelet vector $\mathbf{h}$ is defined as shown below:

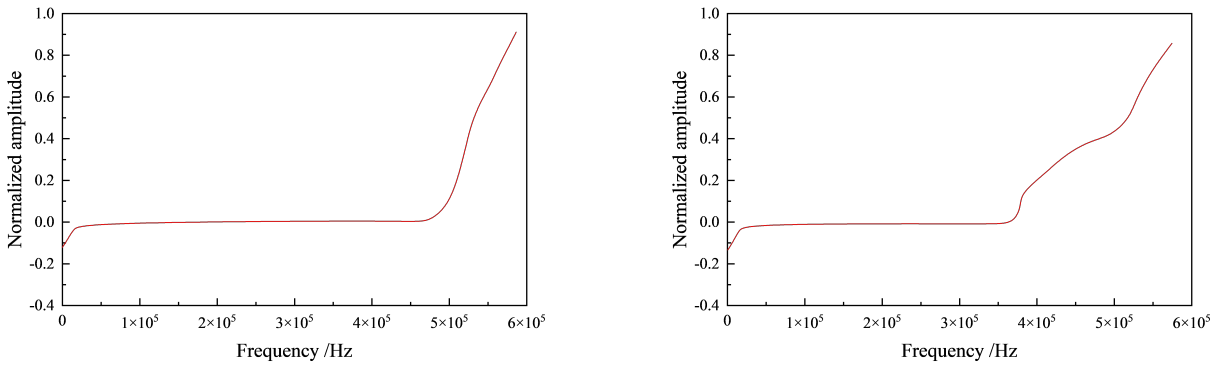
$$\mathbf{h}_0 = \frac{1}{4}[1,2,1]; \mathbf{h}_1 = \frac{\sqrt{2}}{4}[1,0,-1]; \mathbf{h}_2 = \frac{1}{4}[-1,2,-1] \tag{9}$$

Similarly, we use the matrix $\mathbf{w}$ to denote the tightly supported wavelet decomposition and $\mathbf{w}^T$ to denote the tightly supported wavelet reconstruction. Thus, the image decomposition and reconstruction in the tightly supported wavelet domain can be expressed as: $\boldsymbol{\alpha}_x = \mathbf{WX}$ and $\mathbf{x} = \mathbf{W}^T\boldsymbol{\alpha}_x$, respectively. The image degradation model can be rewritten as

$$\mathbf{y} = \mathbf{H}\mathbf{W}^T\boldsymbol{\alpha}_x + \mathbf{n} \tag{10}$$

where $\mathbf{n}$ denotes pretzel noise and $\mathbf{H}$ denotes the degeneracy function.

To further compare the differences between the two transform basis functions, we performed wavelet decomposition and tight support wavelet decomposition on the Lena image respectively. The decomposed coefficients are shown in Figure 1 below.



(a) Sorted tightly supported wavelet coefficients

(b) Sorted wavelet coefficients

Figure 1. Comparison of coefficient sparsity after wavelet decomposition

It is observed that the coefficients after tightly supported wavelet decomposition have better sparse characteristics compared to those after wavelet decomposition. Therefore, this work uses a method based on tightly supported wavelet decomposition to obtain better fuzzyred image enhancement. It can be seen that $\alpha_{\mathbf{x}}$ has sparse properties, while the pepper noise $\mathbf{n}$ itself also has sparse properties. In order to make full use of the sparse property of the degenerate function, we transform the degenerate function model into a degenerate matrix consisting of zeros and ones. Then, a joint optimisation equation is specially constructed to achieve pepper noise cancellation while obtaining an estimate of the clear image.

$$\begin{aligned} &\underset{\mathbf{a_x},\mathbf{H},\mathbf{n}}{\text{minimize}} \|\alpha_{\mathbf{x}}\|_1 + \lambda_{11}\|\mathbf{H}\|_1 + \lambda_2\|\mathbf{n}\|_1 \\ &\text{subject to } \left\|\mathbf{y} - \mathbf{H}\left(\mathbf{W}^T\alpha_{\mathbf{x}}\right) - \mathbf{n}\right\|_2^2 \leq \varepsilon \end{aligned} \tag{11}$$

where ε ¡ 0 is an infinitely small constant.

3. **Image fuzzy enhancement based on membrane computing particle swarm algorithm.**

3.1. **Particle swarm algorithms.** The earliest particle swarm algorithm is a bionic mathematical model that simulates the activity of a flock of birds [28]. A vector (X_i, V_i, P_i) is used to represent the state of existence of each bird in an n-dimensional space. Each bird is viewed as a particle and the activity of the flock is abstracted into a particle swarm optimization model.

The global optimal position that each bird of the entire flock will find over the course of successive iterations. For the problem of solving the minimum of the objective function $f(X_i)$, the model for particle swarm optimisation is shown below:

$$P_i(t+1) = \begin{cases} X_i(t+1) & f\left(X_i(t+1) < f\left(P_i(t)\right)\right. \\ P_i(t) & f\left(X_i(t+1) \geq f\left(P_i(t)\right)\right. \end{cases} \tag{12}$$

where $P_i(t)$ denotes the historical optimal position of the i-th individual particle in generation t and X_i denotes the position of the i-th individual particle.

The global optimal position of the particle corresponding to each bird in the flock is shown as follow:

$$P_g(t) = \min\{P_1(t), P_2(t), \ldots, P_N(t)\} \tag{13}$$

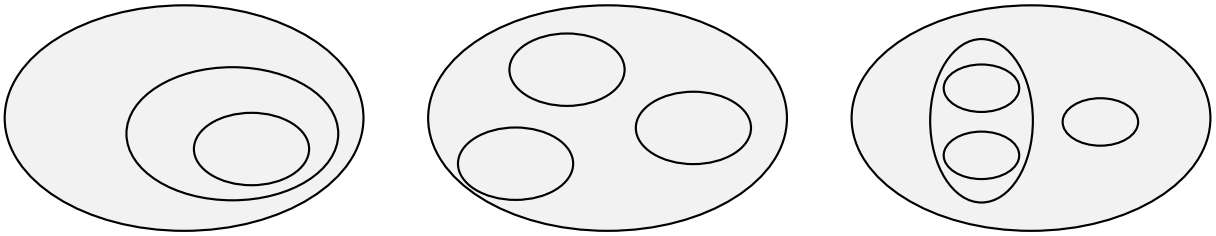
The velocity update method and the position update method for the ith individual particle, which are shown as follow:

$$v_{id}(t+1) = v_{id}(t) + c_1 r_1[p_{id}(t) - X_{id}(t)] + c_2 r_2[p_{gd} - X_{id}(t)] \tag{14}$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \tag{15}$$

where $v_{id}(t+1)$ denotes the velocity of the i-th individual particle in dimension d, $X_{id}(t)$ denotes the position of the i-th individual particle in dimension d, r denotes the random number, c denotes the acceleration constant, p_{id} denotes the position of the individual optimal particle, and p_{gd} denotes the position of the global optimal particle.

3.2. **Membrane calculation principles.** The three membrane structures for membrane calculations are shown in Figure 2.



(a) Nested structure (b) Monolayer structure (c) Hybrid structure

Figure 2. Diagram of the three membrane structures

The objects of membrane calculation theory are denoted by letters.

$$\prod = (V, T, C, \mu, \omega_1, \cdots, \omega_m, (R_1, \rho_1), \cdots, (R_m, \rho_m), i_0) \tag{16}$$

where V is the representation character, T is the output character, C is the catalyst, μ is a membrane structure with m membranes, ω is the multiset of elements inside space i, R is the evolutionary mode, ρ is the order of execution of the evolution, and i_0 is the external space of the system.

The P system consists of four membranes: the surface membrane 1, the basic membrane 3, the basic membrane 4 and membrane 2. Each membrane encloses a different space and the elements corresponding to the way they evolve exist within the corresponding space. By having a computer-like computing style, the P-system is extremely parallel, distributed and non-deterministic.

3.3. **Design of the proposed MMH-PSO algorithm.** By simulating the temperature descent process of the annealing algorithm, this paper uses the number of iterations of the particle swarm algorithm to design the acceptance probability of Metropolis-Hastings. The probability is used to determine whether to accept the individual optimal position and the global optimal position generated by a new iteration, thus adding randomness to the particle swarm algorithm. The use of Metropolis-Hastings sampling to add randomness to the particle swarm algorithm enhances the ability of the particle swarm algorithm to obtain a globally optimal solution.

The main idea of this work is to combine the Metropolis-Hastings sampling in the simulated annealing algorithm with the PSO algorithm. The acceptance rules for Metropolis-Hastings sampling in the simulated annealing algorithm are based on temperature to design the acceptance probabilities. However, this paper does not have the concept of temperature and therefore the Metropolis-Hastings acceptance probability needs to be redesigned. Considering that the acceptance probability will become smaller as the number of iterations increases, the number of iterations n is used as one of the indicators for the new probability. The new Metropolis-Hastings sampling probability is also designed based on the iteration formula for temperature in the simulated annealing algorithm, which is shown as follows:

$$\Delta C' = C(S') - C(S) \tag{17}$$

$$p = \begin{cases} 1 & \Delta c' \leq 0 \\ e^{(-\Delta c' n/k)} & \Delta c' > 0 \end{cases} \tag{18}$$

where S denotes the optimal state generated in the previous step, S' denotes the optimal state generated in the current iteration step, C denotes the fitness function, whose smaller value indicates better results, $\Delta C'$ denotes the difference between the current fitness value and the previous step, n denotes the number of iterations, and k denotes a constant to be determined experimentally.

Set k to 100 based on the setting of the constants in the simulated degradation algorithm. When $\Delta c'$ is less than 0, the probability p is 1 and at this time, the individual optimal position and the global optimal position generated by a new round of iteration can be received.

By adding the Metropolis-Hastings sampling of the simulated annealing algorithm to the PSO algorithm, the problem that PSO tends to fall into local optimum solutions is effectively solved. However, the addition of Metropolis-Hastings sampling leads to an increase in the time complexity of the PSO algorithm, while the larger data volume of the pixel points of the image is time demanding. Therefore the idea of adding a single-cell P-system to the above improvements. the greatest advantage of the P-system is the parallelism, which can increase the computational efficiency.

Therefore, the inclusion of membrane computing can help the PSO algorithm to achieve parallel computing and can effectively reduce the time complexity of the PSO algorithm. This work introduces the Metropolis-Hastings sampling PSO algorithm into the evolutionary mechanism of membrane computing to evolve membrane objects, thus realising the MMH-PSO algorithm. The communication rules in membrane computing are also used to facilitate co-evolution between membranes.

The membrane structure is designed according to a tissue-like membrane system, where different membranes can communicate with each other in the same direction or in the opposite direction. The environment is the output cell, and the initial state is empty. In the process of evolution, the optimal object enters the environment, which in turn transmits the optimal object to the individual base membranes, helping them to complete their evolution. The membrane structure is shown in Figure 3.

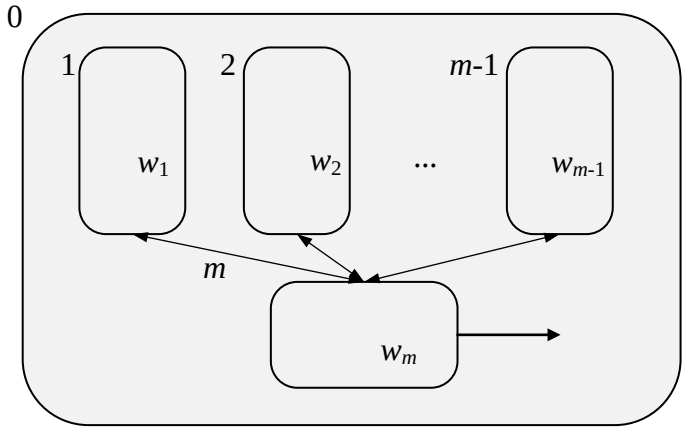


Figure 3. MMH-PSO algorithm for membrane structure

Typically, a tissue-like membrane system of degree $m \geq 1$ is represented as shown below:

$$\prod = (w_1, \ldots, w_m, R_1, \ldots, R_{m-1}, R_m, R^n, i_0) \tag{19}$$

where w_m denotes the initial object in the basic membrane labelled m, R_m denotes the rule in membrane m and i_0 denotes the final result of the optimal particle output from membrane m into the environment after the conditions have been met.

The tissue-like membrane system contains a total of two types of objects: one is the particle object; the other is the fitness value object PBEST and GBEST. PBEST is the fitness value of the particle at the beginning and is the same as the number of particle objects; GBEST is the largest PBEST in the basic membrane and there is one and only one. The fitness rule is the transformation of particle objects into fitness objects. When the fitness object has executed the fitness rule, it transfers the GBEST object and its corresponding particle object to the membrane m. The GBEST object is the optimal fitness value and the particle object is the most eligible input.

3.4. **MMH-PSO-based image fuzzy enhancement algorithm.** This work combines the MMH-PSO algorithm with the traditional fuzzy enhancement algorithm and is able to automatically find the optimal values of the fuzzy parameters F_p, F_e by setting an effective target fitness function and objective function constraints to determine the enhanced image with the maximum fuzzy sharpness function $H(P)$.

The MMH-PSO based image fuzzy enhancement algorithm adaptively searches for two fuzzy parameter values, effectively improving the visual effect of the original image, and its execution steps are shown below:

Step 1: Let the number of particles in the population be N and the search space be D-dimensional. Randomly initialize each particle in the population with its own parameters.

Step 2: Determine if the end condition is met. If it does, go to Step 9, if not, continue to Step 3.

Step 3: Update the position vector and velocity vector equations for each particle. The velocity and position are updated at each iteration of the algorithm until the end condition is met or the optimal solution is obtained.

Step 4: Determine the size of the two fuzzy parameters F_p and F_e, and use the fuzzy feature function $P_{u,v}$ to extract the image fuzzy features and generate the fuzzy feature plane P.

$$P_{u,v} = F(u, v) = [1 + \frac{(x_{\max} - x_{u,v})}{F_p}]^{F_e} \tag{20}$$

where $x_{\max}$ is the maximum grey value of the image and $x_{u,v}$ is a grey value of the pixel at point (u, v). The fuzzy feature function $P_{\mu,\nu}$ is mainly used to represent the subordinate level of the maximum grey value of the pixel at point (u, v).

Step 5: Calculate the fitness value of each particle in MMH-PSO and determine the candidate locations (solutions). In order to quantitatively determine the quality of the enhanced image, the fuzzy sharpness function $H(P)$ is used in the MMH-PSO algorithm to evaluate the quality of the image enhancement. $H(P)$ is used as the fitness function, i.e., the objective function. The optimal fuzzy parameters F_p, F_e are determined by finding their maximum values.

$$H(P) = \lg[(Sd(i) + 0.1 \cdot Fjd(i)/Fcs(i)) \cdot Gl(i) \cdot E(i)] \tag{21}$$

where $Sd(i)$ is the standard deviation of the fuzzy enhanced image, $Fjd(i)$ is the fuzzy feature definition of the fuzzy feature plane, $F_{cs}(i)$ is the fuzzy compactly supported feature plane of the fuzzy enhanced image, $GI(i)$ is the entropy of the fuzzy enhanced image, and $E(i)$ is the Entropy value of the fuzzy enhanced image.

Step 6: Count the number of particles on either side of the global optimal solution at this moment.

Step 7: Determine if the particle needs to select a new unsearched space to continue the search, if so go directly to Step 2.

Step 8: Feature reduction is performed on the new fuzzy feature plane P' using the inverse transform of Equation (20) to obtain the enhanced image $F(u, v)$.

Step 9: Output the enhanced image and the program ends.

4. **Experimental results and analysis.**

4.1. **Verification of the MMH-PSO algorithm.**

To verify the performance of the proposed MMH-PSO algorithm for finding the best performance, four standard test functions were selected for the finding experiments and compared with the standard PSO, MPSO [30] and EPSO [31].

(1) Sphere function.

$$f_1(x) = \sum_{i=1}^{n} x_i^2, \quad -100 \le x_i \le 100, \quad \min f_1(x) = f_1(0, 0, \cdots, 0) = 0 \tag{22}$$

(2) Rosenbrock function.

$$f_2(x) = \sum_{i=1}^{n-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right], \quad -200 \le x_i \le 200, \quad \min f_2(x) = f_2(1, 1, \cdots, 1) = 0 \tag{23}$$

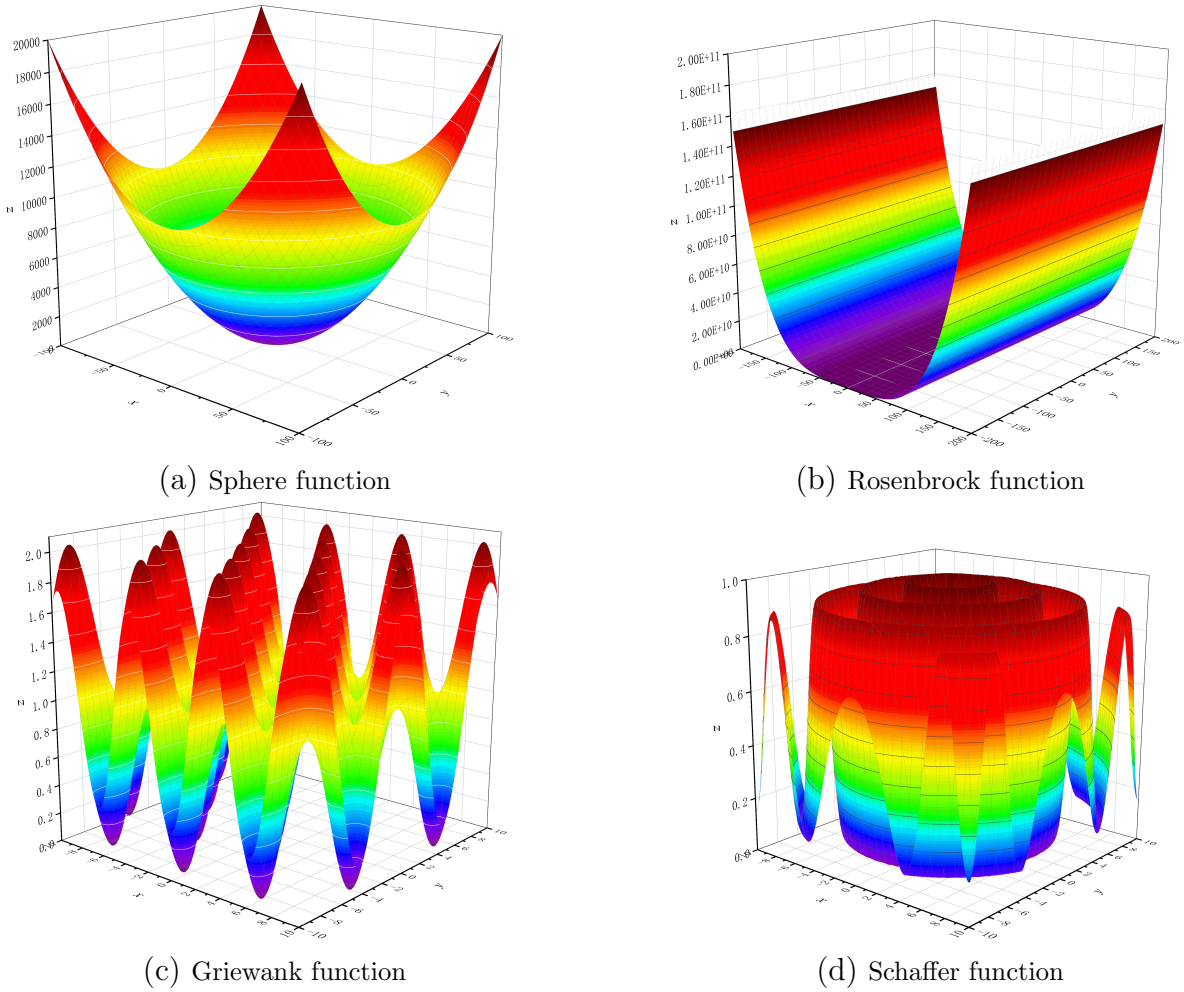
(3) Griewank function.

$$f_3(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\frac{x_i}{\sqrt{i}} + 1, \quad -600 \le x_i \le 600, \quad \min f_3(x) = f_3(0, 0, \cdots, 0) = 0 \tag{24}$$

(4) Schaffer function.

$$f_4(x, y) = 0.5 - \frac{\sin^2\sqrt{x^2 + y^2} - 0.5}{\left(1 + 0.001\left(x^2 + y^2\right)\right)^2}, \quad -10 \le x, y \le 10, \ \min f_4(x, y) = f_4(0, 0) = 1 \tag{25}$$

The geometric curve characteristics of the four functions are shown in Figure 4.



(a) Sphere function (b) Rosenbrock function

(c) Griewank function (d) Schaffer function

Figure 4. Geometric curve properties of 4 functions

Table 1. Optimisation results of the four algorithms.

Algorithms	**Assessment indicators**	**Sphere**	**Rosenbrock**	**Griewank**	**Schaffer**
PSO	Average	4.0×10-15	1.3×10-6	1.3×10-7	2.8×10-16
	Standard deviation	2.5×10-15	1.5×10-6	3.2×10-7	2.5×10-16
	Time/s	0.81	0.15	0.70	0.19
MPSO	Average	8.7×10-8	0.1×10-1	3.1×10-8	0.8×10-2
	Standard deviation	1.7×10-7	0.9×10-2	3.6×10-8	0.5×10-2
	Time/s	10.26	11.20	11.01	0.78
EPSO	Average	2.9×10-35	1.9×10-5	2.8×10-28	1.1×10-3
	Standard deviation	9.3×10-35	4.8×10-5	3.5×10-28	2.2×10-3
	Time/s	12.88	11.52	11.06	0.96
MMH-PSO	Average	0	0	3.1×10-30	0
	Standard deviation	0	0	5.5×10-30	0
	Time/s	2.16	2.21	2.41	0.46

The number of particles within each fundamental membrane in the MMH-PSO algorithm is 8 and the overall number of particles is 48. The acceleration constants are all 1.49, the sizes of the populations are all 100 and the number of iterations is 200.

The average optimisation results of the four algorithms run 10 times are shown in Table 1. The MMH-PSO algorithm has the highest overall optimisation accuracy as seen by the results of the test function. From the standard deviation, the MMH-PSO algorithm has less volatility. In terms of running time, the MMH-PSO algorithm is slightly inferior to the standard PSO algorithm, but significantly better than the MPSO algorithm and EPSO algorithm. The MMH-PSO algorithm outperforms the other three algorithms in terms of the algorithm's search accuracy, search speed and robustness, and thus the MMH-PSO algorithm is adapted to solve a wider range of optimisation problems.

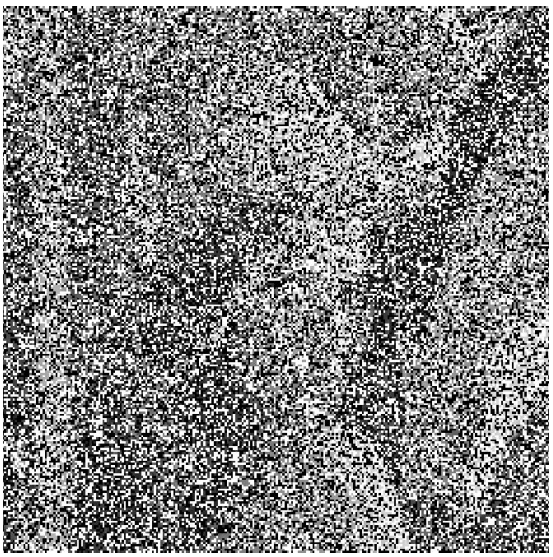
4.2. **Experimental results for the pretzel noise scene.** Finally, the degraded image due to pepper noise is simulated. When the noise contamination reaches 70%, a comparison of the results of conventional grayscale transformation, fuzzy enhancement, PSO fuzzy enhancement [32] and MMH-PSO fuzzy enhancement is shown in Figure 5.

It can be seen that the MMH-PSO fuzzy enhancement algorithm does not have any significant noise residue in the enhancement of the Lena image. The MMH-PSO fuzzy enhancement algorithm reveals the foreground details to a large extent. However, in the other figures only the face and hat outline can be faintly seen. A comparison of the SSIM of each algorithm under different levels of noise pollution is shown in Figure 6.

It can be seen that the MMH-PSO fuzzy enhancement algorithm has the best objective performance metrics, fully validating its superior performance, in line with the above findings. While the SSIM metrics of the other algorithms drop rapidly when the noise level increases, the SSIM values of the MMH-PSO fuzzy enhancement algorithm are consistently very, very high and close to 1.

5. **Conclusion.** Due to the unique impulsive nature of the pepper noise, conventional filter methods do not achieve good denoising results. Therefore, this work decomposes the coefficients in the wavelet domain and the tightly supported wavelet domain for the image separately. Then, a joint optimisation model is constructed using the L1 parametric constraint to achieve pepper noise cancellation. To address the problem that the particle swarm algorithm is prone to fall into local extrema and converge slowly in the later stage, the MMH-PSO algorithm is designed by improving the particle swarm algorithm using membrane computing and Metropolis-Hastings sampling, which contains two specific aspects: firstly, based on the simulated annealing algorithm temperature drop process, Metropolis-Hastings sampling is used as Firstly, based on the simulated annealing algorithm temperature drop process, Metropolis-Hastings sampling is used to provide the particle swarm algorithm with the ability to jump out of the local optimum. Secondly, the use of membrane computing enhances the parallelism of the swarm algorithm and can reduce the time complexity in solving complex problems. The proposed MMH-PSO algorithm is used to simultaneously search out the magnitude of the two fuzzy parameters in the traditional fuzzy enhancement algorithm in order to improve the accuracy of the fuzzy enhancement algorithm. Simulation experimental results verify the effectiveness of the proposed method. The current research object of this paper on image enhancement processing is two-dimensional images, and the next step is to try to apply the proposed MMH-PSO kind of algorithm to the field of three-dimensional image or video processing.

(a) Original image



(b) 70% pretzel noise image

(c) Conventional grayscale transformation

(d) Fuzzy enhancement

(e) PSO fuzzy enhancement

(f) MMH-PSO fuzzy enhancement

Figure 5. Enhancement of Lena images under the influence of pretzel noise

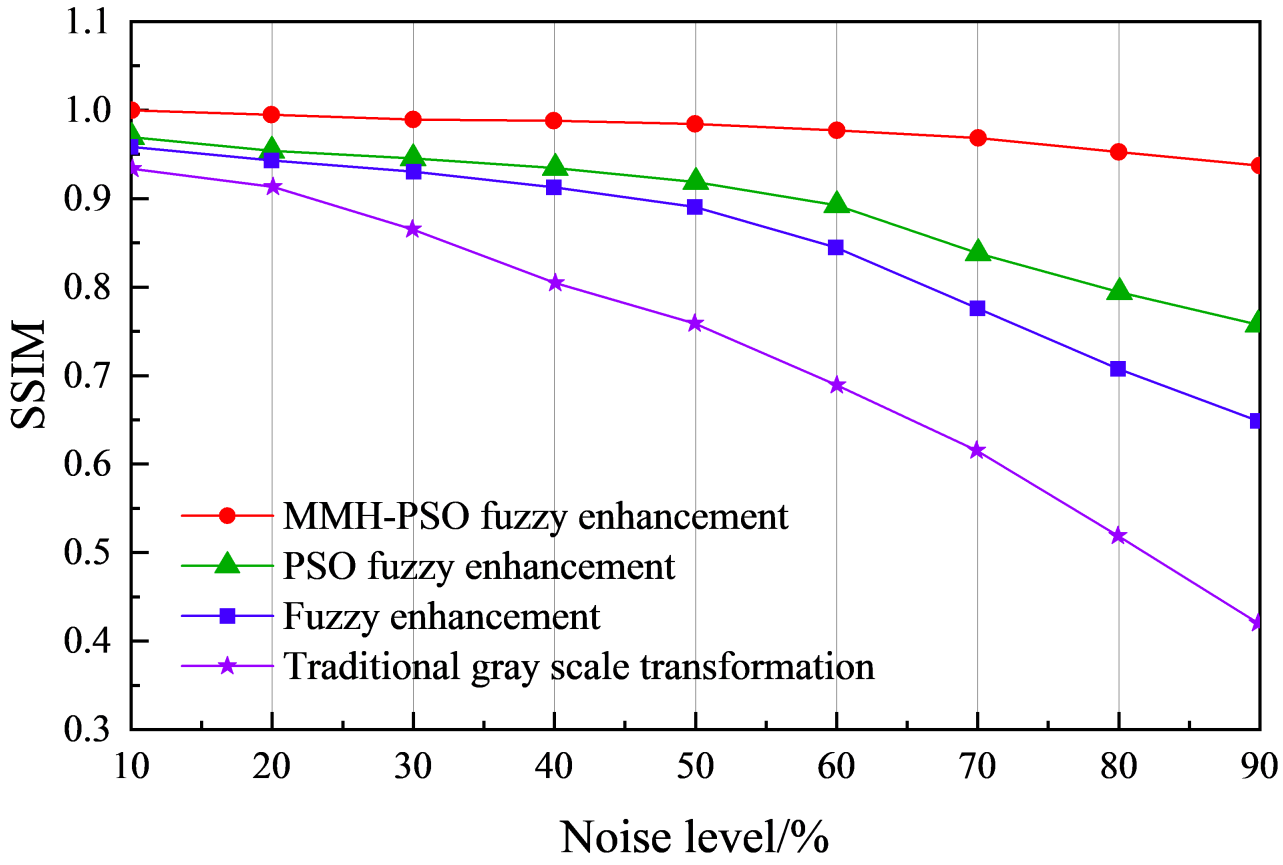


Figure 6. Comparison of the SSIM of each algorithm for different levels of noise pollution

REFERENCES

[1] F. Russo, "Recent advances in fuzzy techniques for image enhancement," *IEEE Transactions on Instrumentation and Measurement*, vol. 47, no. 6, pp. 1428-1434, 1998.

[2] T.-Y. Wu, A. Shao, and J.-S. Pan, "CTOA: Toward a Chaotic-Based Tumbleweed Optimization Algorithm," *Mathematics*, vol. 11, no. 10, 2339, 2023.

[3] T.-Y. Wu, H. Li, and S.-C. Chu, "CPPE: An Improved Phasmatodea Population Evolution Algorithm with Chaotic Maps," *Mathematics*, vol. 11, no. 9, pp. 1977, 2023.

[4] F. Zhang, T.-Y. Wu, J.-S. Pan, G. Ding, and Z. Li, "Human motion recognition based on SVM in VR art media interaction environment," *Human-centric Computing and Information Sciences*, vol. 9, 40, 2019.

[5] B. Subramani, and M. Veluchamy, "MRI brain image enhancement using brightness preserving adaptive fuzzy histogram equalization," *International Journal of Imaging Systems and Technology*, vol. 28, no. 3, pp. 217-222, 2018.

[6] Z. Zhuang, N. Lei, A. N. Joseph Raj, and S. Qiu, "Application of fractal theory and fuzzy enhancement in ultrasound image segmentation," *Medical & Biological Engineering & Computing*, vol. 57, pp. 623-632, 2019.

[7] K. Mayathevar, M. Veluchamy, and B. Subramani, "Fuzzy color histogram equalization with weighted distribution for image enhancement," *Optik*, vol. 216, 164927, 2020.

[8] A. L. H. P. Shaik, M. K. Manoharan, A. K. Pani, R. R. Avala, and C.-M. Chen, "Gaussian Mutation–Spider Monkey Optimization (GM-SMO) Model for Remote Sensing Scene Classification," *Remote Sensing*, vol. 14, no. 24, 6279, 2022.

[9] L. Kang, R.-S. Chen, N. Xiong, Y.-C. Chen, Y.-X. Hu, and C.-M. Chen, "Selecting Hyper-Parameters of Gaussian Process Regression Based on Non-Inertial Particle Swarm Optimization in Internet of Things," *IEEE Access*, vol. 7, pp. 59504-59513, 2019.

[10] C.-M. Chen, S. Lv, J. Ning, and J. M.-T. Wu, "A Genetic Algorithm for the Waitable Time-Varying Multi-Depot Green Vehicle Routing Problem," *Symmetry*, vol. 15, no. 1, 124, 2023.
[11] M. Liu, Z. Zhou, P. Shang, and D. Xu, "Fuzzified image enhancement for deep learning in iris recognition," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 1, pp. 92-99, 2019.
[12] J. Arnal, M. Chillarón, E. Parcero, L. B. Súcar, and V. Vidal, "A parallel fuzzy algorithm for real-time medical image enhancement," *International Journal of Fuzzy Systems*, vol. 22, pp. 2599-2612, 2020.
[13] S. Mandal, S. Mitra, and B. U. Shankar, "FuzzyCIE: fuzzy colour image enhancement for low-exposure images," *Soft Computing*, vol. 24, no. 3, pp. 2151-2167, 2020.
[14] R. Chandrasekharan, and M. Sasikumar, "Fuzzy transform for contrast enhancement of nonuniform illumination images," *IEEE Signal Processing Letters*, vol. 25, no. 6, pp. 813-817, 2018.
[15] M. Veluchamy, and B. Subramani, "Fuzzy dissimilarity color histogram equalization for contrast enhancement and color correction," *Applied Soft Computing*, vol. 89, 106077, 2020.
[16] A. Slowik, and H. Kwasnicka, "Evolutionary algorithms and their applications to engineering problems," *Neural Computing and Applications*, vol. 32, pp. 12363-12379, 2020.
[17] K. Li, R. Chen, G. Fu, and X. Yao, "Two-archive evolutionary algorithm for constrained multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 2, pp. 303-315, 2018.
[18] L. Deng, H. Zhu, Q. Zhou, and Y. Li, "Adaptive top-hat filter based on quantum genetic algorithm for infrared small target detection," *Multimedia Tools and Applications*, vol. 77, pp. 10539-10551, 2018.
[19] A. K. Anaraki, M. Ayati, and F. Kazemi, "Magnetic resonance imaging-based brain tumor grades classification and grading via convolutional neural networks and genetic algorithms," *Biocybernetics and Biomedical Engineering*, vol. 39, no. 1, pp. 63-74, 2019.
[20] M. Wan, G. Gu, W. Qian, K. Ren, Q. Chen, and X. Maldague, "Particle swarm optimization-based local entropy weighted histogram equalization for infrared image enhancement," *Infrared Physics & Technology*, vol. 91, pp. 164-181, 2018.
[21] N. R. Zhou, A. W. Luo, and W. P. Zou, "Secure and robust watermark scheme based on multiple transforms and particle swarm optimization algorithm," *Multimedia Tools and Applications*, vol. 78, pp. 2507-2523, 2019.
[22] D. A. Greene, R. Qi, R. Nguyen, T. Qiu, and R. Luo, "Heterogeneous dielectric implicit membrane model for the calculation of MMPBSA binding free energies," *Journal of Chemical Information and Modeling*, vol. 59, no. 6, pp. 3041-3056, 2019.
[23] D. Orellana-Martín, L. Valencia-Cabrera, A. Riscos-Núñez, and M. J. Pérez-Jiménez, "Minimal cooperation as a way to achieve the efficiency in cell-like membrane systems," *Journal of Membrane Computing*, vol. 1, pp. 85-92, 2019.
[24] K. Li, L. Jia, and X. Shi, "IPSOMC: An Improved Particle Swarm Optimization and Membrane Computing based Algorithm for Cloud Computing," *International Journal of Performability Engineering*, vol. 17, no. 1, pp. 56-71, 2021.
[25] B. Morales-Castañeda, D. Zaldívar, E. Cuevas, O. Maciel-Castillo, I. Aranguren, and F. Fausto, "An improved Simulated Annealing algorithm based on ancient metallurgy techniques," *Applied Soft Computing*, vol. 84, 105761, 2019.
[26] U. Erkan, L. Gökrem, and S. Enginoğlu, "Different applied median filter in salt and pepper noise," *Computers & Electrical Engineering*, vol. 70, pp. 789-798, 2018.
[27] U. Erkan, and L. Gökrem, "A new method based on pixel density in salt and pepper noise removal," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 26, no. 1, pp. 162-171, 2018.
[28] Y. Zhu, G. Li, R. Wang, S. Tang, H. Su, and K. Cao, "Intelligent fault diagnosis of hydraulic piston pump combining improved LeNet-5 and PSO hyperparameter optimization," *Applied Acoustics*, vol. 183, 108336, 2021.
[29] A. Tharwat, and W. Schenck, "A conceptual and practical comparison of PSO-style optimization algorithms," *Expert Systems with Applications*, vol. 167, 114430, 2021.
[30] A. J. Telmoudi, M. Soltani, Y. Ben Belgacem, and A. Chaari, "Modeling and state of health estimation of nickel-metal hydride battery using an EPSO-based fuzzy c-regression model," *Soft Computing*, vol. 24, pp. 7265-7279, 2020.
[31] S. K. Ghosh, B. Biswas, and A. Ghosh, "A novel approach of retinal image enhancement using PSO system and measure of fuzziness," *Procedia Computer Science*, vol. 167, pp. 1300-1311, 2020.