

Intelligent Optimization for Three-dimensional Visualization Based on Hadoop System

Pan-Pan Jia

School of Electronic Information and Artificial Intelligence
Yibin Vocational and Technical College, Sichuan 644100, P. R. China
jiapp@hpu.edu.cn

Yan-Yang Zeng*

School of Electronic Information and Artificial Intelligence
Yibin Vocational and Technical College, Sichuan 644100, P. R. China
zengyy@hpu.edu.cn

*Corresponding author: Yan-Yang Zeng

Received December 26, 2023, revised March 13, 2024, accepted May 8, 2024.

ABSTRACT. *To achieve the best rendering effect of three-dimensional visualization is usually by artificial selection, which leads to the increase of iteration number and poor design efficiency, while choosing the appropriate level models requires high-performance processing power. Therefore, an intelligent optimization algorithm based on Hadoop system is proposed. In this algorithm, meta-modeling is introduced to meet the spectral multi-resolution model's transformation in three-dimensional visualization. Searching for optimal matching of multi-resolution model is reformulated as a global optimization problem. Based on the improved Particle Swarm Optimization (PSO), multiple iterations are utilized to obtain optimal rendering effect. A lot of different resolution three-dimensional models of reading and conversion process, the Hadoop is used to store and process model files. The experimental results are evaluated on image information entropy and draw rate, showing that the algorithm can achieve result in model matching quickly and improve the visual effect effectively, also consider the real-time.*

Keywords: intelligent optimization; multi-resolution model; image information entropy; Hadoop; Particle Swarm Optimization; three-dimensional visualization

1. Introduction. With the development of computer technology, the concept of data visualization has been greatly expanded, which includes not only data visualization for scientific computing, but also the visualization for engineering data and measurement data. Spatial data visualization is often called volume visualization, it is also known as three-dimensional visualization technology, which was originally used in medical imaging, and has become one of the basic techniques for many related disciplines [1-3]. Currently, it is used for the depiction of various phenomena, such as clouds, water, molecular structures and biological structures, and has become an indispensable technology method. In [4], the authors give an augmented reality-based approach for three-dimensional optical visualization and depth map retrieval of a scene using multi-focus sensing, it can be optically displayed in smart glasses, allowing the user to visualize the real three-dimensional scene along with synthesized perspectives of it. [5] proposes a single input data format that allows for the use of the same algorithm for 3D visualization of various types, and provided application examples. [6] proposes a three-dimensional visualization platform for road infrastructure through object detection and data augmentation methods, which

was used for intelligent and fast maintenance of expert highways. [7-9] describe some other fields which were used in three-dimensional visualization technology. Overall, it is important and useful for us to understand and describe the scenes.

In recent years, multi-objective optimization has become a focus and has been applied widely in some science fields [10-12]. It refers to finding a solution to meet all the optimization objectives and the solution usually exists in an uncertain point set form. Many scholars have studied the method of solving multi-objective optimization, and have obtained a lot of research results. Some algorithms have been proposed to solve the problems. For example, [13] proposes a migration-based method of enriching the algorithm library for constrained multi-objective optimization problems. In [14] a multi-preference-based constrained multi-objective optimization algorithm is proposed. The evolutionary algorithms (e.g. genetic algorithm, particle swarm optimization algorithm and ant colony algorithm) [15-17] are one of the evolutionary techniques that have been successfully used as an optimization tool. In these algorithms, the main methods are the evaluation function method and the goal programming method.

The paper integrates the above two ideas, namely, the multi-objective optimization and the evolutionary algorithm. To balance the distance between the vector of the objective function and the ideal objective vector, it needs to construct one cooperative solving model. We propose a combination optimization approach, which is a helpful guidance on achieving automation of the visualization process, it can avoid excessive human-computer interaction. Therefore, we can obtain the optimal rendering effect based on multi-objective optimization, and an improved particle swarm optimization algorithm is adopted in the optimization process [18,19]. Particle Swarm Optimization (PSO) is one of the important branches of evolutionary computing [20,21]. It is a heuristic search optimization algorithm proposed by simulating the swarm intelligence behavior of natural biological activities and drawing on artificial life, bird foraging, fish learning and other swarm intelligence behaviors. Compared with other heuristic search algorithms, PSO has been successfully applied in many real-world optimization problems such as machine learning, parameter recognition, and resource scheduling due to its advantages of easy implementation and simple operation [22-26]. However, PSO has the problems of premature convergence, slow convergence speed, and low solution accuracy. Different scholars have made many improvements in particle activity behavior, particle performance evaluation, and adaptive adjustment of inertia weights [27-29].

In the scene rendering process, a lot of different resolution three-dimensional models of reading and conversion cause the query process to be a bottleneck. Therefore, we discuss how to use the Hadoop system to store and handle different resolution files. The files are generally hundreds of KB, which is small relative to HDFS (Hadoop Distributed File System). This will cause a great burden on "Namenode" node, while the presence of large numbers of files can also cause frequent readers of "Namenode" node, which seriously affects the performance of the system. In order to solve the system handle massive poor performance of small files problem, there are many approaches [30-32]. In this paper, we reduce the number of files by merging files, then the files are merged into data nodes, and by using MapReduce to establish the mapping from the small files to the large files, thereby improving the efficiency of document retrieval and enhancing the performance of the storage system.

This paper is organized as follows. Section 2 gives the proposed algorithm, including Multi-resolution Model Family (MF), evaluation method of visualization effect, formal description of three-dimensional visualization Meta-mode and the design method of the Hadoop system. The implementation of the algorithm is provided in Section 3. Simulation

results and comparison studies to verify the capability of the proposed method are shown in Section 4.

2. The Three-dimensional visualization of intelligent optimization algorithm.

The combination optimization of model can be translated as the multi-objective optimization with resource constraints, which include the constraint of hardware resource and the resolution of the model. It starts with the initial sample set of multi-resolution models. The hardware resource directly affects the draw rate. The resolution of the model directly affects the visualization effect. Through evaluating the drawn image, the fitness value of the corresponding model combination can be obtained. The fitness is the image information entropy. Then, to modify the reference value, satisfactory results can be shown by gradual iteration. In each iteration solution process, the part of satisfactory intermediate results will be passed to the next step; it has a positive impact on the final image. And intuitive expression of the relationship between the model combination and the drawn image is provided. Figure 1 shows the proposed algorithm.

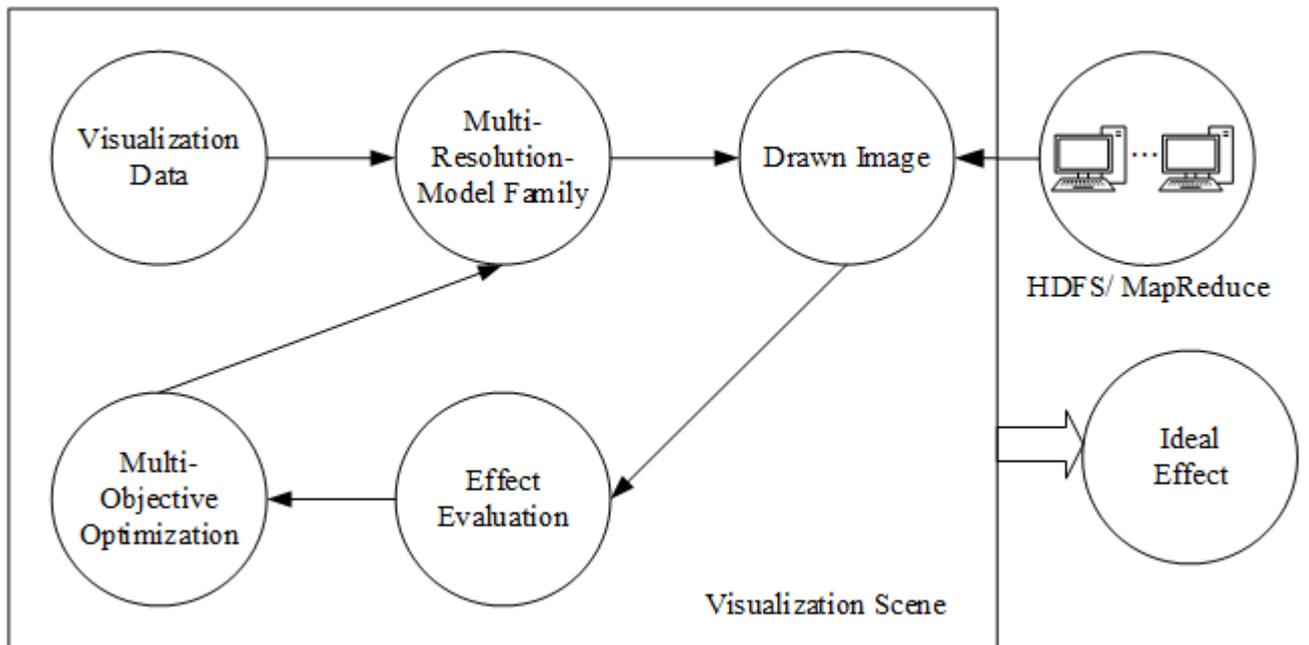


Figure 1. The proposed algorithm

2.1. Multi-resolution Model Family. To establish the different granularity models for the same system, object, phenomenon or process is called the multi-resolution model. The meaning of resolution is the accuracy and the level of detail of the model describing the real world in the modelling and simulation. It is associated with the model fidelity.

Establishing a multi-resolution model is needed to keep the system described in these models or process consistency. The external characteristics of the multi-resolution model should be relatively independent and stable, the granularity of the internal model changes during the dynamic simulation process, and the use of resources reallocated during the process of scheduling, but the external characteristics need to be consistent, it is said that outside express of model and simulation process has seamlessness.

[14] gave the concept of Multi-Resolution-Model Family (**MF**). The set of different resolution models of the same entity is called the multi-resolution-model family. We further proposed the rigorous mathematical definition of MF based on the definition.

Definition 2.1. If we can use $N(N \geq 1)$ models M_1, \dots, M_N to express an object E , and the resolution of the N models are different, so the set $M = \{M_1, \dots, M_N\}$ composed of the models M_1, \dots, M_N is called a multi-resolution MF of object E .

We use an example in two resolution levels to discuss the principles of multi-resolution modelling. Let multi-resolution object E keep properties at high-resolution level HRL and low-resolution level LRL at any time, and the state of HRL and LRL state is always consistent, which is shown in Figure 2.

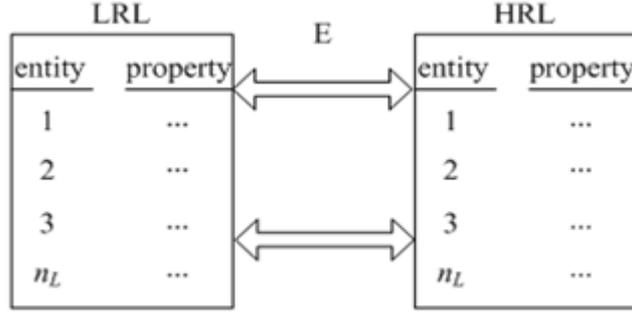


Figure 2. Two levels resolution modeling

Definition 2.2. We set a high resolution entity for E_H , low resolution entity for E_L , the high and low resolution entities can be respectively as follows:

$$\begin{aligned} E_H &= [P_1^H, P_2^H, \dots, P_s^H]^T \\ E_L &= [P_1^L, P_2^L, \dots, P_t^L]^T \end{aligned} \quad (1)$$

Generally, high-resolution entity contains more information than low-resolution entity, so the simulation relation between them can be expressed as:

$$E_H = K \cdot E_L \quad (2)$$

Where, s, t respectively represent the number of multi-resolution entity model, and content $s \leq t$; K is for the $s \times t$ order transformation matrix between the high/low resolution entity model, P for the attribute values of entity.

2.2. Evaluation Method of Visualization Effect. The three-dimensional visualization model is encoded as particles, thus particle evaluation is equivalent to the combination evaluation of a hierarchical resolution model for each entity. Generally, a good combination will produce high-quality images, so the evaluation of the model combination can be transformed into image evaluation.

Drawing and evaluation are necessary conditions to produce a new iteration. Once a new solution is produced, it must draw its corresponding images one by one. Based on the rendered image, we can determine the corresponding fitness values and carry out an evaluation of the results. The evaluation of the model combination is to use some parameters of image quality evaluation. The linear combination of these parameters builds the target evaluation function.

The parameters of Entropy and the image difference, etc., are used to evaluate the image quality. Other parameters can also be defined to evaluate depending on the application. The above two methods are for the evaluation of the dominant rendering image rather than the evaluation of the abstract and implicit MF. The fitness value $Fit(p_i(t))$ must follow the rules:

(1) If the corresponding rendering image of the particle $p_i(t')$ is better than $p_i(t)$ in visualization effect, then $Fit(p_i(t')) > Fit(p_i(t))$;

(2) If the corresponding rendering image of the particle $p_i(t)$ is better than $p_j(t)$ in visualization effect, then $Fit(p_i(t)) > Fit(p_j(t))$.

Entropy is a thermodynamic function, after introduced information theory, which is used to measure the amount of information and is known as information entropy. In information theory, one symbol coding system is more orderly, the lower entropy value is; conversely, the uncertainty of variables is greater, the higher entropy value is. Therefore, it can be said a measure for the degree of ordering.

In order to characterize it, we introduce the image two-dimensional entropy, which can be defined as follows:

$$H(IMG_2) = - \sum_{i=1}^{\text{width}} \sum_{j=1}^{\text{height}} p(i, j) \log_2 p(i, j) \quad (3)$$

Where, $p(ij) = x(ij) / \sum_{i=1}^{\text{width}} \sum_{j=1}^{\text{height}} x(i, j)$, *width*, *height* are respectively the width and height of the image, which means the size of the image. $x(ij)$ is the gray value at the point (i, j) .

When the image information entropy is calculated, the width and height are 1024 and 768. According to (3), we can obtain that the value of image information entropy is 4.559676. The test image and the corresponding gray scale image are shown in Figure 3.



Figure 3. Test image and the corresponding gray scale image

2.3. Formal Description of Three-dimensional Visualization Meta-mode. By making a formal description of the three-dimensional visualization meta-model, we can achieve a more accurate multi-resolution model by grammar, providing adequate grammar assurance for the next implementation of model reusability, but also providing an exchange platform for simulation users and developers.

To make a formal description of the meta-model, we chose BNF (Backus Naur Form). BNF is a typical meta-language, it can express grammatical rules strictly, and the description grammar is context-free grammar. We carry out a specific formal description of interactions among ship, copter, and sea for example (the test scene is given in Figure 4 and Table 1).

In Figure 4, the point “A” is meta-class. The entities are ship, copter and sea. The point “B” is the meta-attribute. The sea surface owns the attribute of reflection. The point “C” is the meta-action. The spray is generated by wind, the sea surface is changed. The point “D” is the meta-interaction. When the ship passes, the trail will be left. The interaction is between ship and sea. The point “E” is meta-constraint. The copter cannot be below the sea surface.

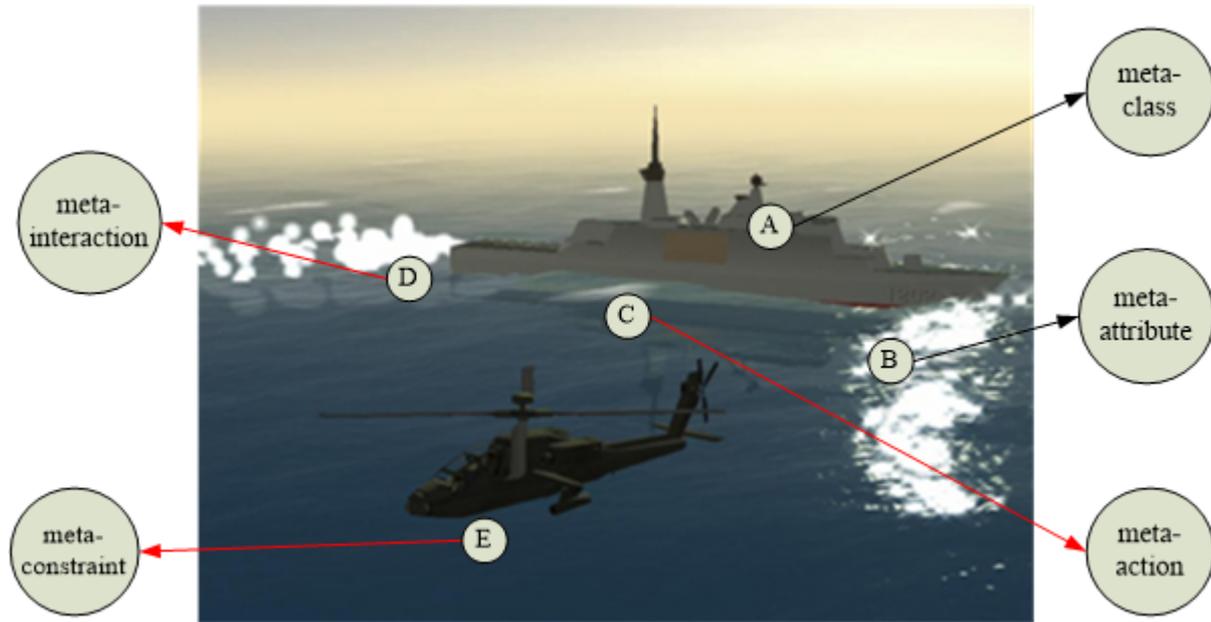


Figure 4. A test scene for a specific formal description of interactions

Table 1. A specific formal description for example

```

<Scene> ::= [ <Sea> <ship> <copter> ]
<Sea> ::= [ <Illumination> <Reflection effect> <Wind> <Spray> <Trail> ... ]
<ship> ::= ...
<copter> ::= ...
...

```

2.4. The design method of Hadoop system. The optimization structure of the overall system design by Hadoop is shown in Figure 5. System optimization includes two parts: optimized file storage and file retrieval. The file storage consists of file merging and a merge algorithm to combine individual small files into larger files, where each file is written to a separate sequence file.

In this paper, “SequenceFile” technology is employed to merge and create a new file format, reducing the number of system files. The merged files are stored in the “DataNode” file blocks. The merge algorithm primarily operates based on time, merging files created within a certain timeframe. File retrieval aims to enhance the read efficiency of “SequenceFile” through indexing design and index creation. Additionally, secondary indexes are constructed using a “Trie”-based approach to improve random access efficiency.

The optimization structures of the overall system design by Hadoop are shown in Figure 5. System optimization includes two parts, one is optimized file storage, and the other is the file retrieval. There are two mains in file storage: file merge and merge algorithm, to merge individual small files into large files, each file will be written to a separate sequence file.

3. Implementation of the algorithm based on IPSO.

3.1. Intelligent Combination of Multi-resolution Model. We achieve the intelligent combination and matching of multi-resolution models using IPSO (Improved Particle Swarm Optimization). The fundamental approach involves translating model matching

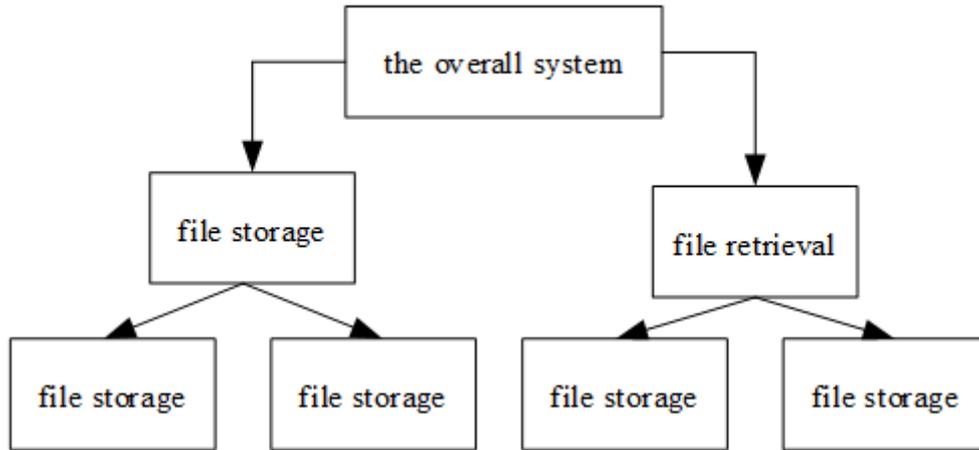


Figure 5. The optimization structure of the Hadoop system

into a global optimization problem. The process begins with an initial sample set of multi-resolution models, where the fitness value of each model combination is evaluated through its drawn image. Subsequently, modifications to the reference value are made iteratively to refine the results. Each iteration passes satisfactory intermediate results to the next step, contributing to the enhancement of the final image. The relationship between the model combination and the drawn image is illustrated in Figure 6.

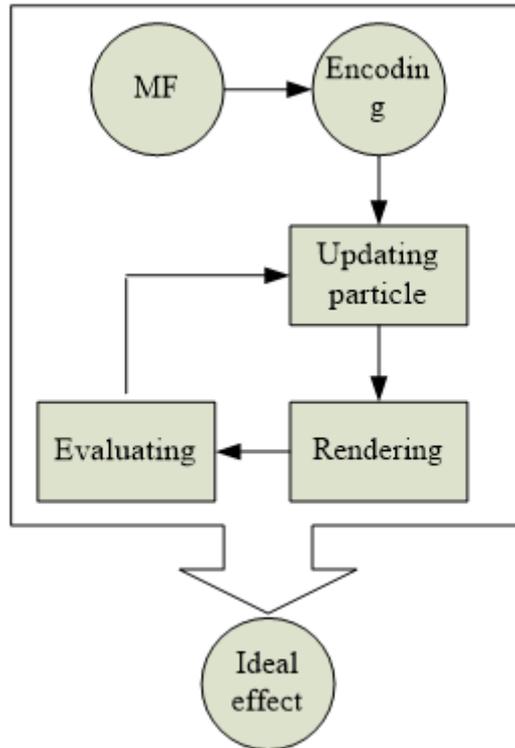


Figure 6. Intelligent combination algorithm of multi-resolution model

By using the IPSO for solving, its basic particle is the solution. Multiple particles can determine the global optimum flight path according to the current individual and global optimal solution, update in accordance with their fitness value, and gradually move closer to the optimal solution.

In the process of searching for the optimal combination of models, the first problem is to encode the original MF for the IPSO particle. Then, on the basis of the corresponding evaluation of the visualization effect, the fitness value of all particles can be calculated. The automatic design process of the proposed algorithm is shown in Figure 7, including the encoding, initialization, updating, decoding, rendering, and evaluation.

The automatic design process is as follows:

- (1) To encode the **MF** and determine the attributes of particles, such as the number of particles, the dimension of the particle, etc.
- (2) To initialize all particles.
- (3) To calculate the fitness value of each particle in the initial sample set and assign it to individual optimal solution $pBest$.
- (4) To determine the initial sample set of global optimal solution $gBest$.
- (5) If the satisfactory solution is not searched:
 - a. To calculate the velocity of each particle and update its location;
 - b. To calculate the fitness value of each particle and compare it with $pBest$, if the value is larger, update $pBest$ by using it;
 - c. To determine the global optimal solution.

In Figure 7, the work of the encoder/decoder has two parts: one part encodes the MF and sends the resulting set. The other part encodes the intermediate particle set generated by the solver and sends the candidate model combination to the rendering module.

The rendering module draws based on the candidate model combination and sends the image to the image evaluation module.

The image evaluation module evaluates the input image. We use the above objective evaluation, namely, the image information entropy.

The usefulness of the solver is generating an intermediate particle set based on the output of the encoder/decoder and image evaluation.

Then, there will be two sections to describe the specific implementation method.

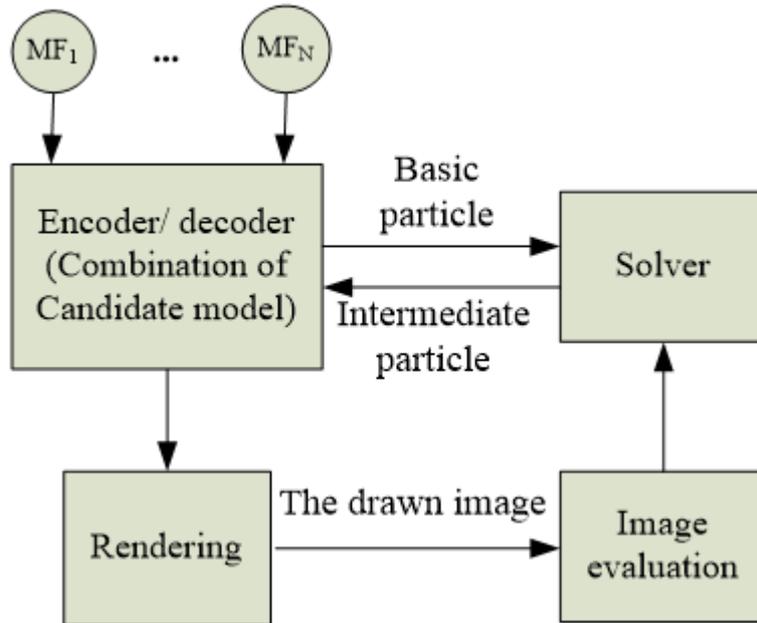


Figure 7. The automatic design process of model combination

3.2. Encoding and Initialization. Definition 3.1. Assuming that the particle sample set $P(t) = \{p_1(t), p_2(t), \dots, p_N(t)\}$ at time t , which is an N -dimensional vector; N is the size of the sample set. The initial sample set $P(0)$ can be pre-defined or randomly selected.

In the process of multi-resolution model combination of multi-resolution model, the encoding must be according to the practical problem, so that the particle can denote the solution of the problem. Therefore, we encode the multi-resolution model of visualization entities as the particles in IPSO. The number of attributes, namely, the dimension of the particle is equal to the number of entities in three-dimensional visualization.

$$N(p_i(t)) = N_E \quad (4)$$

Where $N(p_i(t))$ is the dimension of the particles, N_E is the number of entities in three-dimensional visualization.

Assuming that $N_E = m$, the m -dimensional encoding of the particle i at time t can be described as follows:

$$p_i(t) = \begin{cases} \text{Level}(s_1(x_1, y_1, z_1), s_2(x_1, y_1, z_1), \dots, s_m(x_1, y_1, z_1)) \\ \text{Level}(s_1(x_2, y_2, z_2), s_2(x_2, y_2, z_2), \dots, s_m(x_2, y_2, z_2)) \\ \vdots \\ \text{Level}(s_1(x_n, y_n, z_n), s_2(x_n, y_n, z_n), \dots, s_m(x_n, y_n, z_n)) \end{cases}$$

Where, $s_j(x_i, y_i, z_i)$ is the scalar value of the sample point i , $\text{Level}(s_1(x_n, y_n, z_n), s_2(x_n, y_n, z_n), \dots, s_m(x_n, y_n, z_n))$ is the value of multi-resolution level. Their range are $[0,1]$, which can be defined in setting PSO parameters.

The number of particles needed to be adjusted according to the actual problem for the solver. The number of particles may be too small so that the time of a single iteration will be shortened, but the average size of the searching area of each particle becomes larger. Thereby, the average number of required iterations will greatly increase for optimization, and the probability that the best results are not obtained is also higher. Conversely, too much of the particle will increase the time of a single iteration, and the positions overlap after updating the particle, which results in a great waste of resources. Generally speaking, about 10 particles meet the need for general questions, but the required number of particles may reach 100-200 for some more complex questions.

In the iteration process of IPSO, assuming that the used number of particles is N , then at the time t , the particle sample set is defined as follows:

$$P(t) = \{p_1(t), p_2(t), \dots, p_N(t)\} \quad (5)$$

It is an N -dimensional vector; N is the size of the sample set. Assumed that at time $t = T$ the best result is obtained by iteration; the sample set can be expressed as follows:

$$P(T) = \{p_1(T), p_2(T), \dots, p_N(T)\} \quad (6)$$

The particle $p_i(T)$ is the best particle, if and only if the index value $i = gBest$.

We set that the $P(0)$ is pre-defined. Randomly selected can ensure randomness of particle distribution, but without prior knowledge as an auxiliary, which increases search time and costs. The pre-defined can customize the user's experience, knowledge, and understanding of the original data to set the initial particles purposefully, which can avoid many unnecessary searching processes. Compared with the random selection method, it is faster to obtain the optimal result.

3.3. Optimization Process. From the initial set of **MF** to obtain the satisfactory model combination, it will need to go through several iterations. That's a gradual process of optimization. The entire process starts from the initial particle sample set, where the calculations of fitness values and updates are needed to generate new particles. Assuming that a new sample set is $P(t')$, including the $p_i(t')$ ($i = 1, 2, \dots, N$), we evaluate each particle in turn, obtain the fitness value $Fit(p_i(t'))$. Then to compare the fitness values of $p_i(t')$ and $pBest(i)$.

If $Fit(p_i(t')) > Fit(pBest(i))$, then to update the value of $pBest(i)$, $pBest(i) = p_i(t')$, and to compare the fitness values of $p_i(t')$ and $gBest$.

If $Fit(p_i(t')) > Fit(gBest)$, then to update $gBest = p_i(t')$.

Else if $Fit(p_i(t')) < Fit(pBest(i))$, then no action.

And according to the obtained $gBest$ and $pBest$, to update the velocity and position of each particle, thereby a new set of samples $P(t'')$ can be obtained. If this set is not the ideal model combination and does not reach the maximum limited number of iterations, the process will continue.

4. Experimental results and analyses.

4.1. The Structure of Solution Model Based on Gray Relational Degree. The above multi-objective optimization for the model combination can be described as follows:

$$\begin{cases} \max f_1 = -\sum_{i=1}^{\text{width}} \sum_{j=1}^{\text{height}} p(i, j) \log_2 p(i, j) \\ \max f_2 = FPS \end{cases} \quad (7)$$

Where the FPS represents the draw rate.

This paper set up a Hadoop cluster, which includes four hosts, one host is the server node, and the other is the node station.

The development environment of the proposed algorithm: the processor is Intel(R) Core (TM) i7CPU/870 @ 2.93GHz, the memory is DRRII800/ 4GB, the graphics card is NVIDIA GeForce GTX 260/1024MB, the hard is 500G/7200 16M.

The test scene contains five or six resolution levels. The corresponding vertex number and polygon number are given in TABLE 2 and TABLE 3. The TABLE 4 is the **MF** of the sea.

To transform the multi-objective optimization functions into three single objective optimization functions in (8), the solution value and objective value of single objective functions satisfying condition in (8) are shown in Table 5. l_1, l_2 and l_3 respectively represent the resolution level of ship, helicopter and sea.

Table 2. The resolution levels of ship and the corresponding vertex number and polygon number of Ship

Resolution level	Vertex number	Polygon number
5	9317	4610
4	7957	3934
3	4697	2286
2	3668	1824
1	2084	1016

Therefore, we can obtain an ideal point in the solution domain of the model: $f^* = (f_1^*, f_2^*) = (4.737679, 47.2)$. Assuming that the weight of three objective functions is the same, and then, the solution model according to (4) can be computed as follows:

Table 3. The resolution levels of helicopter and the corresponding vertex number and polygon number of helicopter

Resolution level	Vertex number	Polygon number
5	3458	1497
4	2202	938
3	1280	524
2	756	288
1	648	262

Table 4. The MF of the sea. The mean of "Y" is included, and the mean of "N" is not included. There are some numbers which represent their respective properties

Resolution level	Reflection effect	Illumination	Wind velocity	Spray	Fog density	Trail
6	Y	Y	12	Y	0.0015	Y
5	Y	Y	12	Y	0.0015	N
4	Y	Y	12	Y	0.001	N
3	Y	Y	12	Y	0	N
2	Y	N	8	Y	0	N
1	N	N	4	N	0	N

Table 5. Solution value and objective value of single objective functions

	l_1	l_2	l_3	objective value
$\max f_1$	3	5	5	4.737679
$\max f_2$	1	1	1	47.2

$$F = \sum_{k=1}^p \omega_k \frac{\min_k |f_k - f_k^*| + \zeta \max_k |f_k - f_k^*|}{|f_k - f_k^*| + \zeta \max_k |f_k - f_k^*|} \quad (8)$$

4.2. IPSO Algorithm for Solving the Model. According to the constraint condition of the above equations, we only need to determine two values and the other four unknowns can be determined. In the algorithm, we determine the particles x_1 and x_2 , the dimension of the cluster is 2. We set that population size is 3, the maximum evolution generation is 20, the maximum inertia weight is 1.0, the minimum inertia weight is 0.3, $c_1 = c_2 = 2.0$, and the maximum speed of the particles move to 0.25. The algorithm runs for 20, and the fitness function F is defined as follows:

$$\max \sum_{k=1}^3 \frac{1}{3} \cdot \frac{\min_k |f_k - f_k^*| + 0.5 \max_k |f_k - f_k^*|}{|f_k - f_k^*| + 0.5 \max_k |f_k - f_k^*|} \quad (9)$$

Using the steps in section 3 programming to realize, fitness value curves of evolutionary processes are shown in Figure 8. The optimal individual is at point $l_1 = 5, l_2 = 1, l_3 = 2$, where the optimal value of the gray relational degree is 0.969 263, the optimal value of (6) is $f^{**} = (f_1^{**}, f_2^{**}) = (4.627772, 35.6)$.

The Figure 8 is corresponding to the iterative process of the above sample set. From these image sequences, we can see that the entire process is toward a better solution program. The changes in the fitness value of each particle are shown in Figure 9. From the figures, we can observe that each particle is gradually to the optimal solution. Under the same conditions of MF, the optimal solution is the second particle of the 4th generation;

the corresponding value of image information entropy is 4.627772. The corresponding resolution levels are 5, 1 and 2; the draw rate is 38.5 FPS/s.

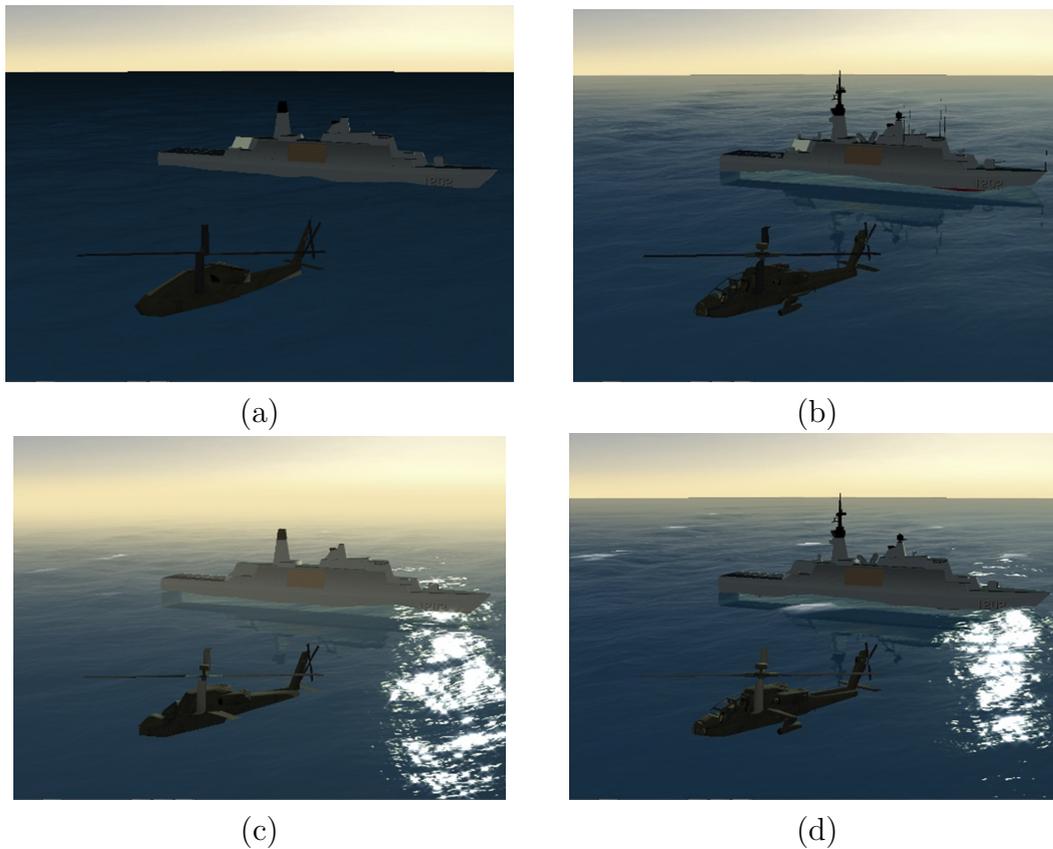


Figure 8. The evolution process of multi-objective optimization

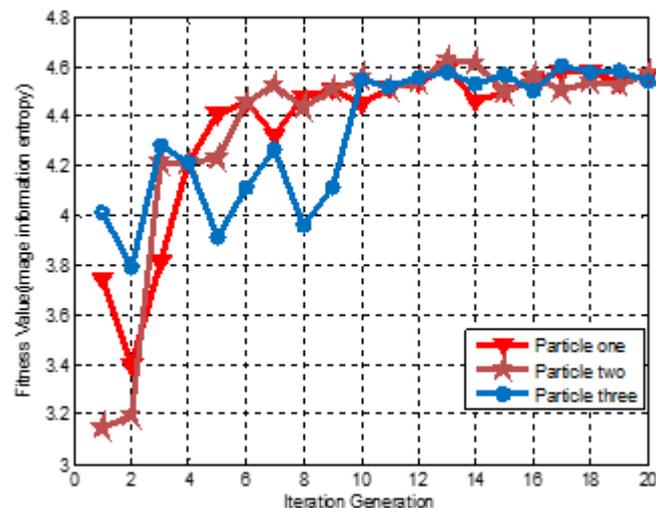


Figure 9. The fitness curve of each particle

4.3. Comparison Studies. Figure 10 shows the comparison of the rendering effect between the algorithm in [30] and the proposed approach. In Figure 10, by using the proposed algorithm, the corresponding resolution levels are 5, 1 and 2; the draw rate is

38.5 FPS/s; the corresponding value of image information entropy is 4.627772. But the corresponding value of image information entropy is 4.211484 and the draw rate is 30.5 FPS/s by using the algorithm in [30], the corresponding resolution levels are 2, 3, and 3. The comparison results are evaluated on image information entropy and draw rate, showing that our algorithm outperforms the algorithm in [30] and the algorithm can achieve better results.

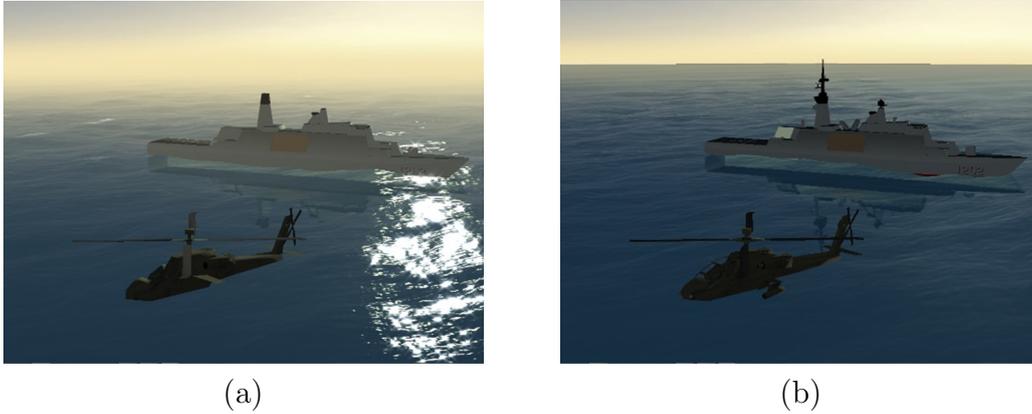


Figure 10. (a) The rendering effect with the proposed algorithm. (b) The rendering effect with the algorithm in [30].

4.4. Performance Analysis. We choose the number of the different resolution three-dimensional models for experimental design: 1000, 2000, 3000, and 4000, then they are directly written into systems or read out from the optimized system. Where the size distribution of these files is the same, the experimental results are shown in Figure 11 and Figure 12. In Figure 11, the Experimental Analysis of test results: The figure shows that read efficiency is higher. Because of by use of “SequenceFile” technology, the search work is only needed once. In Figure 12, the comparison shows that the efficiency of batch reading is still higher than the original system.

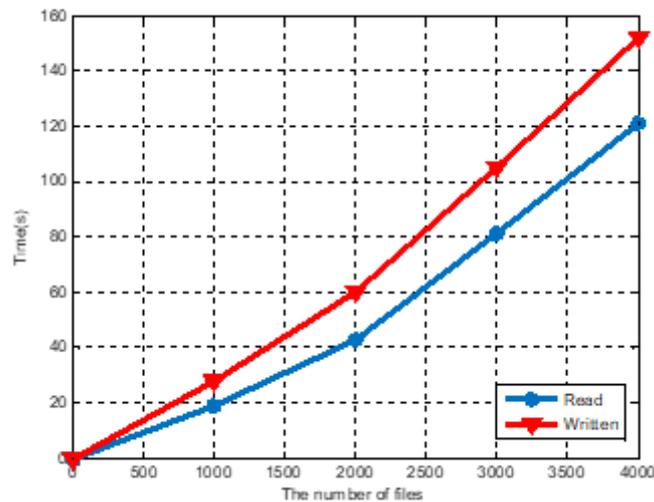


Figure 11. Written or read time of before and after optimization

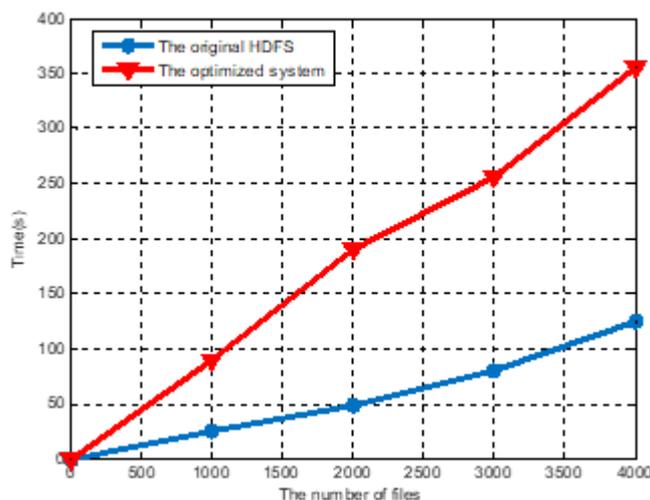


Figure 12. Written or read time of before and after optimization

5. **Conclusions.** In this work, a novel solution algorithm for a multi-objective optimization problem based on grey correlation was introduced. The approach improves the rendering efficiency and visual effect, and the automated combination of models will be realized.

In the process of three-dimensional visualization, the human factor is growing emphasis. The requirements are the increasing influence for visualization results. How to reflect the requirements will be our future work, which needs to use more artificial intelligence techniques for application in three-dimensional visualization. Or, we will further improve the proposed algorithm for the visualization results.

Acknowledgment. This work is partially supported by the National Natural Science Foundation of China (61503124 and 61304144), and the Key Scientific Research Projects of Henan Higher, China (15A520018).

REFERENCES

- [1] G.-H. Chen, C.-N. Luo, L.-X. Zhou, and X.-H. Rao, "Research on three-dimensional visualization system of Natech events triggered domino accidents in oil-gas depots," *Journal of Loss Prevention in the Process Industries*, vol. 81, p. 104953, 2023.
- [2] Y. Arai, M. Ogozama, H. Yamamoto, H. Takahashi, and H. Fujiwara, "Three-dimensional visualization using a rubber balloon as postpartum feedback for vessels traversing vasa previa," *European Journal of Obstetrics & Gynecology and Reproductive Biology*, vol. 279, pp. 191-192, 2022.
- [3] A. Kharaghani, H. Mahmood, Y. Wang, and E. Tsotsas, "Three-dimensional visualization and modeling of capillary liquid rings observed during drying of dense particle packings," *International Journal of Heat and Mass Transfer*, vol. 177, p. 121505, 2021.
- [4] J. Alonso, A. Fernández, and B. Javidi, "Augmented reality three-dimensional visualization with multifocus sensing," *OSA Continuum*, vol. 1, no. 2, pp. 55-365, 2023.
- [5] M. Lopatin, A. Briushinin, D. Savin, and I. Gureeva, "Development of a Universal System for Three-Dimensional Data Visualization," *Springer Proceedings in Physics*, vol. 268, pp. 173-181, 2022.
- [6] Q. Liu, and Y.-H. Ye, "Road detection and three-dimensional visualization management using deep learning and BIM integration," *Third International Conference on Computer Vision and Pattern Analysis*, vol. 12754, pp. 173-181, 2023.
- [7] Q. Guo, J.-M. Chen, T. Pu, Y.-J. Zhao, K. Xie, X.-P. Geng, and F.-B. Liu, "The value of three-dimensional visualization techniques in hepatectomy for complicated hepatolithiasis: A propensity score matching study," *Asian Journal of Surgery*, vol. 46, no. 2, pp. 767-773, 2023.

- [8] X. Wu, J.-Y. Yan, K.-A. Wu, and Y.-A. Huang, "Integral lucky imaging technique for three-dimensional visualization of objects through turbulence," *Optics & Laser Technology*, vol. 125, p. 105955, 2020.
- [9] L. Yang, P.-P. Wu, M.-F. Liao, Z.-Z. Tang, H.-B. Long, H.-H. Zhang, and X.-Y. Yu, "Three-dimensional modeling and visualization of rice root system based on the improved dual-scale automaton and L-system," *Computers and Electronics in Agriculture*, vol. 195, p. 106823, 2022.
- [10] Z.-W. Ye, J.-G. Wang, and J.-J. Yang, "A multi-objective optimization approach for a fault geothermal system based on response surface method," *Geothermics*, vol. 117, p. 102887, 2024.
- [11] L.-L. Li, W.-H. Zhang, Y. Li, C.-J. Jiang, and Y.-F. Wang, "Multi-objective optimization of turbine blade profiles based on multi-agent reinforcement learning," *Energy Conversion and Management*, vol. 297, p. 117637, 2023.
- [12] M. Dörterler, Ü. Atila, N. Top, and İ. Şahin, "A nested optimization approach for robot gripper multi-objective optimization problem," *Expert Systems with Applications*, vol. 239, p. 122163, 2024.
- [13] Y. Wang, M.-C. Zuo, and D.-W. Gong, "Migration-based algorithm library enrichment for constrained multi-objective optimization and applications in algorithm selection," *Information Sciences*, vol. 649, p. 119593, 2023.
- [14] X. Feng, Z.-Y. Ren, A.-Q. Pan, J.-C. Hong, and Y.-H. Tong, "A multi-preference-based constrained multi-objective optimization algorithm," *Swarm and Evolutionary Computation*, vol. 83, p. 101389, 2023.
- [15] L.-L. Kang, R.-S. Chen, N.-X. 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.
- [16] T.-T. Nguyen, T.-D. Nguyen, T.-G. Ngo, and V.-T. Nguyen, "An Optimal Thresholds for Segmenting Medical Images Using Improved Swarm Algorithm," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 13, no. 1, pp. 12–21, 2022.
- [17] T.-Y. Wu, A. Shao, and J.-S. Pan, "CTOA: Toward a Chaotic-Based Tumbleweed Optimization Algorithm," *Mathematics*, vol. 11, no. 10, p. 2339, 2023.
- [18] A. Afzal, and M.-K. Ramis, "Multi-objective optimization of thermal performance in battery system using genetic and particle swarm algorithm combined with fuzzy logics," *Journal of Energy Storage*, vol. 32, p. 101815, 2020.
- [19] A. Mokri, A. Noorpoor, and F.-A. Boyaghchi, "Evaluation and multi-objective salp swarm optimization of a new solid oxide fuel cell hybrid system integrated with an alkali metal thermal electric converter/absorption power cycle," *Process Safety and Environmental Protection*, vol. 176, pp. 797-816, 2023.
- [20] J. Kennedy, and R. Eberhart, "Particle Swarm Optimization," *Proceedings of ICNN'95-International Conference on Neural Networks*, pp. 1942-1948, 1995.
- [21] R. Eberhart, J. Kennedy, "A New Optimizer Using Particle Swarm Theory," *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39-43, 1995.
- [22] F.-Q. Meng, S. Wei, J.-D. Wang, P.-F. Wang, and B. Li, "An Information Feedback-based Particle Swarm Optimization Algorithm for Multi-regional Image Segmentation," *Journal of Network Intelligence*, vol. 8, no. 1, pp. 194-210, 2023.
- [23] W.-B. Liu, Z.-D. Wang, N.-Y. Zeng, F.-E. Alsaadi, and X.-H. Liu, "A PSO-based deep learning approach to classifying patients from emergency departments," *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 7, pp. 1939-1948, 2021.
- [24] X. Wang, D.-S. Yang, and S. Chen, "Particle swarm optimization based leader-follower cooperative control in multi-agent systems," *Applied Soft Computing*, vol. 151, p. 111130, 2024.
- [25] Y.-L. Yuan, C.-M. Hu, L. Li, and X.-Y. Wang, "Regional-modal optimization problems and corresponding normal search particle swarm optimization algorithm," *Swarm and Evolutionary Computation*, vol. 78, p. 101257, 2023.
- [26] H. Li, J. Li, P.-S. Wu, Y.-C. You, and N.-Y. Zeng, "A ranking-system-based switching particle swarm optimizer with dynamic learning strategies," *Neurocomputing*, vol. 494, pp. 356-367, 2022.
- [27] X.-L. Li, Y.-Q. Wang, G.-B. Ma, X. Chen, and B. Yang, "Prediction of electricity consumption during epidemic period based on improved particle swarm optimization algorithm," *Energy Reports*, vol. 8, pp. 437-446, 2022.
- [28] Z. He, T.-H. Liu, and H. Liu, "Improved particle swarm optimization algorithms for aerodynamic shape optimization of high-speed train," *Advances in Engineering Software*, vol. 173, p. 103242, 2022.

- [29] G. Zhang, J.-J. He, W.-S. Li, X.-L. Lan, and Y.-P. Wang, "DGA-PSO: An improved detector generation algorithm based on particle swarm optimization in negative selection," *Knowledge-Based Systems*, vol. 278, p. 110892, 2023.
- [30] C. E. D. Martin, "Adaptive Decision-Making Frameworks for Multi-Agent Systems," *University of Texas: Doctor Thesis*, 2001.
- [31] V. Mothukuri, S.-S. Cheerla, R.-M. Parizi, Q. Zhang, and K.-R. Choo, "BlockHDFS: Blockchain-integrated Hadoop distributed file system for secure provenance traceability," *Blockchain: Research and Applications*, vol. 2, no. 4, p. 100032, 2021.
- [32] J.-N. Bi, X.-J. Chen, and S. Wang, "Development of Network Public Opinion Analysis System in Big Data Environment Based on Hadoop Architecture," *Procedia Computer Science*, vol. 228, pp. 291-299, 2023.