Journal of Network Intelligence ©2024 ISSN 2414-8105
Taiwan Ubiquitous Information Volume **9**, Number **4**, November **2024**

Bionic Optimization Algorithm Based Multi-style Styling Element Design for Social Networks

Bing Feng*
School of Arts and Design
Nanjing Vocational University of Industry Technology, Nanjing 210016, P. R. China
fengbingking@163.com

Chen Jiang
Dongseo University, Busan 47011, South Korea
kg3950@163.com

*Corresponding author: Bing Feng
Received January 5, 2024, revised April 21, 2024, accepted July 29, 2024.

Abstract. *The innovation of styling elements in modern social networks requires a great deal of mental effort on the part of designers and requires a combination of emotional and rational thinking. Therefore, high-quality styling element design takes a considerable amount of time. In order to effectively improve the design efficiency while ensuring the diversity and quality of the elements, a styling element design method based on bionic optimisation algorithm for social networks is proposed. Firstly, the geometric topological properties of the model are analysed based on the styling elements, and the triangular mesh sample points are generated. Then, the traditional genetic operator is improved and an adaptive strategy based on the fitness value is proposed. A novel bionic optimisation algorithm (GAWWO) is proposed by introducing the improved genetic operator and the chaotic optimisation strategy in the water wave optimisation algorithm in order to obtain a better search capability. Secondly, the distance and area characteristics between any two points on the surface of the model are used to calculate the eigenvalues of the modelling structure. The corresponding adjacency matrix is constructed through the analysis of face similarity calculation and the GAWWO algorithm is used to mine the target maximal clusters, thus simulating the design process of human innovative thinking. Finally, the test results of five generalised arithmetic cases show that the GAWWO algorithm possesses better performance and is able to better balance the local and global search capabilities. In addition, MALTAB simulation was conducted using cartoon expression modelling as an example, and the results show that the time consumption of the proposed method is only 0.02109 s, and the similarity between modelling element structures is only 0.608.*
Keywords: Modelling elements; bionic optimisation algorithm; water wave optimisation algorithm; genetic algorithm; shape distribution; similarity

---

1. **Introduction.** The dynamic development of modern social network platforms has brought new opportunities and challenges for styling design. In this changing environment, styling design needs to constantly innovate elements to meet the diverse and personalised needs of users [1, 2]. Designers need to provide customisable and personalised styling elements, such as avatar frames, personalised stickers, special effects and filters, etc., to meet users' needs for self-expression.

However, the innovation of styling elements in modern social networks requires a great deal of mental effort on the part of designers and a combination of emotional and rational thinking [3, 4]. Therefore, the design of high-quality styling elements takes a considerable amount of time. Designers need to fully understand the needs and preferences of users, which requires designers to conduct a large number of user studies and market research, and deeply understand the psychology and aesthetic tendencies of different user groups [5]. Each element requires designers to repeatedly think, draw sketches and adjust details. High-quality handmade products need time to accumulate. It is also necessary to consider the match and unity between the elements and the overall visual style of the platform, which requires global observation and coordination [6, 7]. Adopting new visual styles and creative elements also requires a certain amount of trial-and-error time, which requires continuous testing and iterative optimisation. Therefore, the intelligent design of diverse modelling elements has become a hot direction of current research.

Intelligent design can shorten the product development cycle, maximise product modularity and serialisation, improve the efficiency of design and product quality, reduce design costs and maximise profitability. The intelligence and parallelism of bionic optimization algorithms allow for fast convergence, fewer parameters to be adjusted and the ability to evaluate individuals based on neighbourhood structure and set metrics [8]. Bionic optimisation algorithms continuously obtain better species based on the evolutionary criterion of Darwin's theory of biological evolution and optimise the population with different updating strategies to ultimately achieve the optimal solution. Bionic optimisation algorithms are one of the most effective optimisation methods developed in recent years and have been widely used in the fields of transportation, communication and 3D animation model design [9, 10], however, they have not been promoted in the field of innovative design applications. Based on this, the purpose of this study is to apply bionic optimisation algorithms to the design of styling elements for social networks, and to explore new ideas for intelligent innovation design.

1.1. **Related work.** product appearance, cartoon design or element design design design can be a comprehensive use of cognitive science, design thinking, design psychology, machine learning and other theories [11] for intelligent design of product modelling, belonging to a kind of product modelling design.

The use of genetic algorithms, meta-heuristic algorithms, neural network algorithms, etc. is an effective means of design automation. These algorithms simulate the design process of designers, and can output design results that are novel and meet the expectations. Fung et al. [12] proposed a genetic algorithm to optimise the design of mobile phone products. By controlling the parameters of mobile phone shape through coding and using genetic algorithms for multi-generation evolutionary optimisation, a mobile phone shape that meets the predefined design requirements can be automatically generated. Example validation shows that the method can effectively explore the design space and automatically generate innovative mobile phone shapes that meet the target requirements. Compared with manual design, this method is more efficient and can produce novel design solutions that cannot be thought of by hand. Zhao and Ghasvari [13] systematically studied and compared the advantages of various meta-heuristic algorithms for industrial design problems. It is pointed out that heuristic algorithms can effectively handle continuous design variables and produce smooth shapes. Particle swarm algorithms can achieve global optimisation of formal features while ant colony algorithms can enhance the exploration of the design space. Application to footwear and eyewear design cases are given, and the results show that these algorithms can automatically generate novel and stylistically consistent designs.

Li et al. [14] used GAN network and style migration algorithm to automatically generate a 3D cartoon avatar that meets the requirements based on the text description input by the user. Intelligent automation of cartoon avatar design was realised. Not only the End-to-End generative network architecture is designed to achieve the automatic generation of avatars, but also the visual quality of the generated images is objectively evaluated to give a quantitative analysis. The results demonstrate that the method can generate feature-compliant anime avatars based on text descriptions, and the image quality is better than the benchmark model of GAN. The system releases the creative pressure of designers and can directly serve customers. Huang and Zheng [15] proposed a data-driven graphic design framework. The framework mainly uses deep learning and Human-Computer Interaction (HCI) to automate and optimise the design process. As a hybrid approach, this framework combines the two types of methods, deep learning and HCI, to complement each other. Through HCI, user feedback is fed back to the model, which in turn optimises the design. It takes into account not only innovation and design efficiency, but also human involvement, making this approach more flexible and interactive than traditional fully automated design.

However, the quality of images generated by deep learning methods based on GAN networks and other [16, 17] methods still falls short of real images, and there are often non-conforming design results. In addition, traditional meta-heuristic algorithms are prone to fall into local optimal solutions rather than global optimal solutions [18, 19, 20], which leads to the limited innovation of design works, and there is still a certain gap compared with manual design. The bionic optimisation algorithm simulates the evolution mechanism of natural organisms, which can improve the global search ability in the design space. The bionic optimisation algorithm adopts the diversity preservation strategy, which can avoid falling into the local optimal solution.

1.2. **Motivation and contribution.** In order to satisfy users' diverse styling element needs, intelligent design systems need to simulate designers' thinking characteristics. Bionic optimisation algorithms can produce more innovative designs by integrating multiple biological inspirations. Water Wave Optimization (WWO) algorithm is a novel intelligent algorithm for bionic optimisation [21, 22]. Therefore, this work proposes a method for designing styling elements for social networks based on the improved WWO algorithm. The main innovations and contributions of this work include:

(1) The set of product modelling elements is obtained using morphological analysis, and the set of 19 modelling element node parameters is obtained by quantitatively describing the design parameters.

(2) Improve the traditional genetic operator and propose an adaptive strategy based on the fitness value. By introducing the improved genetic operator and chaotic optimisation strategy into the WWO algorithm, a hybrid GAWWO algorithm is proposed.

(3) The distance and area characteristics between any two points on the model surface are used to calculate the modelling structure eigenvalues. The corresponding adjacency matrix is constructed by face similarity calculation and analysis, and the target maximal clusters are mined using the GAWWO algorithm.

2. **Social network modelling elements and their sample point selection.**

2.1. **Model geometry topology properties.** For social network modelling element design, the form of the product is mainly based on modelling elements, and different imagery is expressed through the combination of different sample elements.

Social network styling elements refer to the icons, colours, fonts, typography and other visual elements used within social network platforms. These elements are often designed to be highly recognisable and distinguishable to help users easily identify different features, pages and brands. For example, Facebook's blue and white colour combination, Twitter's bird logo, Instagram's camera logo and many other important elements of social networking platforms are part of their styling elements. These elements not only improve the recognition of the social networking brand, but also increase the user's trust and experience of using the social networking platform.

In order to avoid the difficulty of generating new ideas from the similar elements and causing the singularity of thinking, designers need to apply morphological analysis to decompose the reference element samples, whose key features include the face shape, eyes and mouth. In this paper, we take the cartoon emoji modelling as an example of social network modelling element design, and the key features (i.e. modelling elements) of the model's geometrical topological attributes include the mouth, eyes and nose, with a total of 19 elements, as shown in Figure 1. Cartoon emoticons, as commonly used emoticons on social network platforms, require a unique and easily recognisable icon design. Designers usually choose cute, vivid and amusing images as cartoon emoji charms, which can attract users' attention and keep in line with the theme of the platform. The typography and layout of cartoon emoji on social platforms are also important. They are usually required to be displayed in a relatively small space, and designers need to use typography and layout skills wisely to make the cartoon emoji meet the design requirements, be clearly visible and easy to use. This study analyses the imagery of 100 emoji styling elements

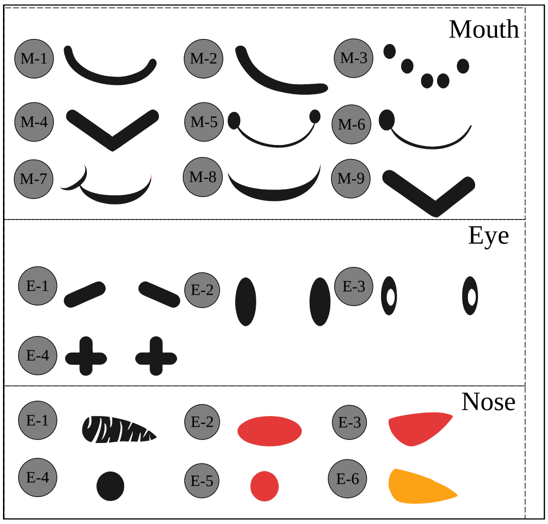

Figure 1. Shape element diagram

from a semiotic and aesthetic point of view to get the real user needs. Each emoticon corresponds to an image of "likeability". Based on the semantic differential analysis method, a five-order questionnaire was created to evaluate the target imagery of each of the 100 emoji styling elements. In order to facilitate the data processing, the five levels of 1, 2, 3, 4 and 5 were used in this study. 1 means very disliked, 2 means relatively disliked, 3 means between liked and disliked, 4 means relatively liked, and 5 means very liked, and finally the imagery evaluation value of each emoji modelling element about likeability is obtained.

In order to obtain users' perceptual imagery needs, 20 questionnaires were statistically analysed and the mean and variance of each emoji styling element regarding likability were calculated separately.

$$\overline{A_i} = \frac{1}{K} \sum_{k=1}^{K} A_i^k \tag{1}$$

$$\sigma = \sqrt{\frac{1}{K-1}\sum_{k=1}^{K}\left(\overline{A_i} - A_i^k\right)^2} \tag{2}$$

where σ denotes the variance of the i-th expression styling element against the likability rating value, and $\overline{A_i}$ denotes the mean of the i-th expression styling element against the likability rating value.

Based on the 3σ criterion, the remaining data after eliminating the data that meet the conditions is the mean value, and the evaluation value of the expression modelling elements is shown in Table 1.

Table 1. Evaluating the value of expression modelling elements

Elemental	**Likeability**	**Elemental**	**Likeability**	**Elemental**	**Likeability**
M-1	0.27	E-1	0.23	N-1	0.39
M-2	0.41	E-2	0.46	N-2	0.43
M-3	0.79	E-3	0.49	N-3	0.71
M-4	0.38	E-4	0.56	N-4	0.42
M-5	0.49			N-5	0.33
M-6	0.38			N-6	0.48
M-7	0.64				
M-8	0.73				
M-9	0.29				

In this study, based on the styling element diagram and combining with user needs and their own understanding of styling, the expressions were re-planned and two representative samples shown were designed and used as the basis for product innovation design.

2.2. **Sample point generation.** After analysing the geometric topological properties of the model based on the modelling elements, the triangulation mesh is processed using the meshing function *Triangulation*. In order to generate N random points on the surface of the triangular mesh model, the triangular mesh model is defined as $S = \{T_1, T_2, ..., T_k\}$, and T_i denotes the triangular face piece in the model.

According to the Monte Carlo method, the sample points are collected according to the principle of equal area by using the relationship between the distribution probability of points in an area and the area of that area. The main steps are shown below:

(1) Traverse the model triangular mesh face sheets, calculate and store the area cumulative sum;

(2) Generate random numbers based on area;

(3) Perform a search operation to find the index number of the triangular facet labelled by a random number;

(4) Random sample points were taken inside the face sheet following a mathematical method.

The detailed process of taking points inside the mesh face sheet is shown as follows:

Define the face piece in the triangular mesh as $T = \{P_1, P_2, P_3\}$, and its area AT is calculated as shown below:

$$AT = \frac{\|(P_2 - P_1) \times (P_3 - P_1)\|}{2} \tag{3}$$

The total area obtained by accumulating the triangular face sheets AT approximates the actual surface area of the face.

$$AS^2 = \sum_{i=1}^{k}(AT_i) \tag{4}$$

The probability of selecting any triangular face piece T_j is shown below:

$$p(T_j) = \frac{AT_j}{AS} \tag{5}$$

As shown in Figure 2, dividing the interval $[0, 1]$ into k segments, the j-th segment is equal to $p(T_j)$. At this point it is sufficient to generate a certain random number between $[0, 1]$ to find the facet index number j. As shown in Figure 3, for a triangle with vertices (A, B, C), the constructed sample points on its surface are shown below:

$$p = (1 - \sqrt{r_1})A + (\sqrt{r_1} - r_2)B + \sqrt{r_1 r_2}C \tag{6}$$

where r_1 denotes the ratio along an edge and $\sqrt{r_2}$ denotes the vertex-to-edge ratio.

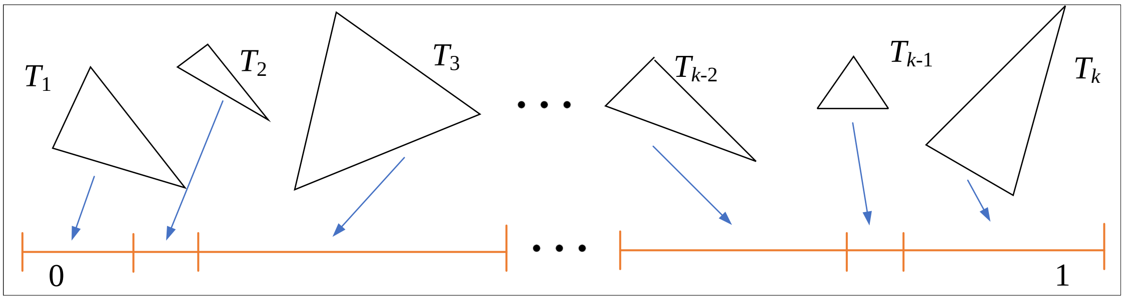

Figure 2. Selection of triangular patches

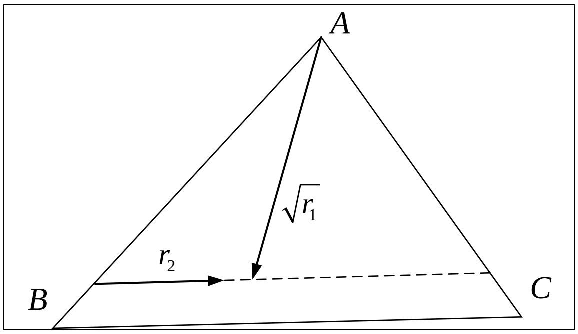

Figure 3. Selection point from triangular patches

The generation of random numbers is one of the important steps in the process of random point taking. In the true sense of the word, random numbers are generated randomly according to the experimental process after the performance of the distribution of probability, and its results are unpredictable and invisible. Random numbers in the computer are generated by simulation according to a certain algorithm, and the results are certain and visible. Traditional pseudo-random number generators generate random numbers with poor uniformity and high local clustering. For this reason, in this paper, Halton-Sequence in quasi-random number generator is used to generate random numbers. In case of taking the same number of points, the uniformity of the points generated by taking the Halton-Sequence method is better as compared to the pseudo-random number generator.

3. Hybrid water wave optimisation algorithm based on genetic operators.

3.1. Principle of water wave optimisation algorithm. The WWO algorithm is a novel intelligent algorithm for bionic optimisation [23, 24]. It is based on the shallow water wave theory and solves the optimisation problem by simulating the motion search of water waves in the seabed.

In the WWO algorithm, each water wave represents a solution to the problem, and each water wave has two attributes, λ for wavelength and h for wave height. In addition, each water wave has its own fitness value, the magnitude of which is inversely proportional to its vertical distance to the sea floor. The water waves are constantly moving and changing through three operations: propagation, wave breaking, and refraction, and finally the water wave with the highest fitness is sought, which in turn leads to the optimal solution of the problem.

(1) Dissemination

Propagation is the main operation of the algorithm and the water wave is propagated once for each iteration. If the initial water wave is assumed to be X, it is updated in each dimension according to Equation (7) to generate a new water wave X'.

$$X'_d = X_d + rand(-1, 1) \cdot \lambda L_d \tag{7}$$

where $rand()$ is a random generator function that serves to randomly generate a uniformly distributed random number in the range $[-1, 1]$, and L_d denotes the length of the d-th dimensional search space ($1 < d < D, D$ is the problem dimension).

(2) Broken Waves

In the shallow water wave theory, with the continuous propagation of the water wave, the energy contained in the water wave is also increasing, and its height is increasing, and in the end when the water wave energy reaches the peak, the water wave will be broken, and become a series of water waves, and this process is the process of the broken waves of the water wave. This process is simulated in the WWO algorithm.

$$X'_d = X_d + norm(0, 1) \cdot \beta L_d \tag{8}$$

where β is the broken wave coefficient and $norm(0, 1)$ denotes a Gaussian random number with mean 0 and variance 1.

(3) Refraction

During the movement of water waves, the refraction phenomenon will occur. In WWO algorithm, in order to simulate the energy consumption of water waves in the process of propagation, the wave height of water waves will be reduced by 1. When the wave height of water waves decays to 0, the water waves will be stagnant, and in order to avoid the appearance of this phenomenon, the refraction operation will be carried out on the water waves.

$$X' = norm\left(\frac{X^*_d + X_d}{2}, \frac{X^*_d - X_d}{2}\right) \tag{9}$$

where $norm()$ function denotes a Gaussian random number (with mean μ and variance σ) and X^* denotes the optimal water wave in the current population.

The refraction operation not only avoids the stagnation of the search, but also enables the water wave generated by the refraction to learn from the optimal water wave, which enables the algorithm to converge faster. After refraction, the wavelength of the new water wave is updated as following:

$$\lambda' = \lambda \frac{f(X)}{f(X')} \tag{10}$$

3.2. **Chaotic optimisation strategy.** As a bionic optimisation algorithm, the WWO algorithm usually starts with randomly generating an initialised population in the search space in the process of solving the optimal solution of the problem. Then, based on the motion of water waves, the population is searched in the search space through three operations, namely propagation, wave breaking and refraction, from generation to generation. When a solution better than the current optimal solution is found, it replaces the original optimal solution until the termination condition of the algorithm is satisfied, the algorithm ends and the optimal solution is output.

It can be understood through the optimality search process of the WWO algorithm that the solution search process of the bionic optimisation algorithm depends largely on the random initial solution selected at the very beginning. If the randomly generated initialisation population is within the vicinity of the optimal solution, the algorithm will be able to search for the optimal solution faster and converge relatively quickly. If the initialised population is far away from the optimal solution, the algorithm will be difficult to converge to the optimal solution, with slow convergence speed and low search accuracy. Thus, the WWO algorithm is generally very sensitive to the initial value.

In order to solve the problem that the random initialisation distribution is not ideal, this work adopts the chaos optimisation strategy to improve the WWO. Chaos in the chaos optimisation strategy refers to the ability to achieve traversal of all states within a specified range according to its own law without repetition [25]. The basic idea is to transform the variables to be optimised in the solution space into chaotic variables by means of chaos rules, so that they can be searched between the chaotic space ($[0, 1]$), and then mapped back to the original solution space when a solution is searched. The chaotic variables transformed by the chaos rules have the characteristics of randomness, ergodicity, regularity, etc., and thus can search the whole solution space. The use of chaotic strategy chaotic initialisation of the population ensures that the individuals in the population can traverse the search in the solution space, effectively reducing the impact caused by the uneven distribution of the initialisation.

Currently there are many bionic optimisation algorithms that use chaotic optimisation strategy, and the experimental results show that the performance of the algorithms can be greatly improved, and the effect of uneven initialisation distribution is effectively reduced. There are various chaotic optimisation strategies, and in this paper we decided to use Logistic mapping [26]. This is because Logistic mapping is easy to implement, simple and efficient, and can generate the initial population of chaotic sequences in a simple and effective way. The functional form of Logistic mapping is shown below:

$$X_{k+1} = \mu X_k (1 + X_k) \tag{11}$$

where μ denotes the chaos coefficient (it has been shown that its effective value range is $[0, 4]$, and in this paper, we decide to take $\mu = 3$), and X_k denotes the original initialisation sequence.

3.3. **Improvement of the genetic operator.** The crossover operator determines the frequency of crossover operations and the frequency of individual updates of the population, which is the key to the behaviour and performance of the genetic algorithm [27, 28]. An effective and reasonable crossover operator can greatly accelerate the search speed during the evolution of the population. However, in traditional genetic algorithms, the crossover operator is generally set as a fixed parameter, which causes that if the crossover probability is set too large, it will make the individuals with high adaptability be destroyed as a result.

The crossover operator is an important source for the generation of new individuals and is the key to achieving the global search of the algorithm. In terms of the overall evolutionary process of the population, the crossover probability should change with the change of fitness value, and gradually approach a certain stable value in the late evolutionary stage to avoid the algorithm failing to converge or taking too long to converge. From the point of view of new individual generation, all individuals in the population should have the same probability to perform the crossover operation during the crossover process to ensure the uniformity of the algorithm in all directions. Therefore this work proposes the crossover operator that self-adapts according to the fitness value.

$$p_c = \begin{cases} \frac{p_{c3}(f_{avg}-f)+p_{c2}(f-f_{min})}{f_{avg}-f_{min}} & f < f_{avg} \\ \frac{p_{c2}(f_{max}-f)+p_{c1}(f-f_{avg})}{f_{max}-f_{avg}} & f \geq f_{avg} \end{cases} \tag{12}$$

where f denotes the larger fitness value of the two crossing individuals, f_{min} denotes the lowest fitness value, f_{avg} denotes the average fitness value, and f_{max} denotes the maximum fitness value.

In addition to the crossover operator, the variation operator determines the probability of individual variation, which affects the variation of the whole population, and proper variation of individuals ensures the diversity of the population [29]. And in the basic genetic algorithm, the variation probability is kept as a constant. If it is set to a small value, it will cause the individuals in the population to have similar adaptations in the late stage of evolution, which is easy to fall into the local optimum, and seriously affects the operation efficiency of the algorithm; if it is set to a larger data, it will cause the individual variation to be too large, and the algorithm will be similar to a random algorithm, which loses the characteristics of genetic evolution. Therefore, the adaptive mutation operator can effectively improve this situation. From the above, it can be seen that the simple adaptive mutation operator has certain defects and shortcomings. Similarly, this paper is also based on the variation of adaptation value change.

$$p_m = \begin{cases} \frac{p_{m1}(f_{avg}-f)+p_{m2}(f-f_{min})}{f_{avg}-f_{min}} & f < f_{avg} \\ \frac{p_{m2}(f_{max}-f)+p_{m3}(f-f_{avg})}{f_{max}-f_{avg}} & f \geq f_{avg} \end{cases} \tag{13}$$

where f denotes the fitness of the mutant individual.

3.4. **GAWWO design.** In this paper, the purpose of improving the GAWWO algorithm is to speed up the convergence rate and improve its convergence accuracy.

To address the shortcomings of WWO algorithm, we refer to the combination of PSO algorithm and GA algorithm. In this paper, we plan to introduce the crossover operation and mutation operator of GA algorithm in WWO algorithm. The introduction of crossover and mutation operators enables the WWO algorithm to achieve better search performance while making the population diversity of the WWO algorithm guaranteed. Therefore, the hybrid algorithm will theoretically perform better than the standard WWO algorithm.

Firstly, water wave optimisation is performed on the whole population to obtain the current optimal water wave set. Then, a cross-mutation operation is performed on the selected water wave cluster to obtain a new generation of population, evaluate its fitness value, and output the optimal solution if it satisfies the output condition, otherwise the new population obtained will repeat the previous action again. This approach accelerates the convergence speed and strengthens the local search, and at the same time, the accelerated convergence speed will not cause the reduction of population diversity. In this way, it can also avoid the weakening of the global capacity, and better balance between the global search capacity and convergence speed. The pseudo-code of the proposed GAWWO is shown in Algorithm 1.

Algorithm 1 GAWWO

Inputs: n - population size, h - max wave height, a - attenuation coefficient, b - wave breaking coefficient, k - max wave breaking dimension, p - chaos coefficient, p_c - crossover probability, and p_m - mutation probability

Outputs: P - the population of water wave solutions, where the best solution is the optimized result.

```
Initialize population P of size n
while stopping criteria not met do
    P′ = WWO(P)                  ▷ perform WWO propagation, breaking, refraction
    P″ = Selection(P′)           ▷ select fittest individuals
    for i in range(n) do
        if rand() < p_c then     ▷ perform crossover on P″
            parent1, parent2 = SelectParents(P″)
            child = Crossover(parent1, parent2)
            if Fitness(child) > Fitness(parent1) then
                Replace parent1 with child in P″
            end if
        end if
    end for
    for i in range(n) do
        if rand() < p_m then     ▷ perform mutation on P″
            parent = SelectParent(P″)
            child = Mutate(parent)
            if Fitness(child) > Fitness(parent) then
                Replace parent with child in P″
            end if
        end if
    end for
    P = P″
end while
Return best solution in P
```

4. GAWWO-based modelling element mining.

4.1. Eigenvalue calculation and comparison.

In this paper, the distance and area characteristics between any two points on the surface of the modelling element are used to calculate the modelling structure eigenvalues. Assuming the existence of two sampling points $P_1(x_1, y_1, z_1)$ and $P_2(x_2, y_2, z_2)$, the corresponding point-to-point distance calculation solution is shown below:

$$d^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2 \tag{14}$$

The algorithmic matrix is evaluated using the Manhattan distance, then the similarity measure function for two different modelling element models is calculated below:

$$dis(M_1, M_2) = \sum_{i=1}^{500} \sum_{j=1}^{100} |m_{ij}(1) - m_{ij}(2)| \tag{15}$$

where M_1 and M_2 denote the distance pinch distribution matrices of the two modelling element models, respectively, with matrix size 500×100.

The difference between the modelling element models represented by the matrices can be inferred through Equation (15). The larger the value of $dis(M_1, M_2)$, the greater the difference between the two matrices, i.e., the greater the difference between the two modelling element models is indicated.

4.2. **Analysis of Face Similarity Calculation.** The face similarity evaluation stage is the key to the construction of attribute maps, and the way of handling its various factors is analysed below. Considering the fact that auxiliary structures such as chamfered corners can affect the normal similarity evaluation, the relative area approach is used here to filter the faces of modelling elements. In other words, the area ratio between a particular face and its neighbours is calculated, and if it is less than a specified threshold, the corresponding node of the graph is deleted.

After the distribution matrix of distance angle between different modelling element models is calculated, the corresponding adjacency matrix needs to be constructed through the analysis of face similarity calculation. Firstly, according to the number of sides and area of the faces are analysed, and the calculation of the similarity between the two is shown as follows:

$$S_{e_{ij}} = 1 - \frac{|n_i - n_j|}{\max(n_i, n_j)} \tag{16}$$

$$S_{a_{ij}} = 1 - \frac{|a_i - a_j|}{\max(a_i, a_j)} \tag{17}$$

where n_i and a_i denote the number of sides and area of a face f_i, respectively.

The similarity coefficient of two different faces is ϕ, which is calculated as shown below:

$$\phi = \begin{cases} 0, & T_s = 0 \\ \alpha_1 S_{e_{ij}} + \alpha_2 S_{a_{ij}}, & T_s = 1 \end{cases} \tag{18}$$

where α_1 and α_2 denote the weight values and $\alpha_1 + \alpha_2 = 1$.

If the number of components of the modelling element is not equal to 3 then $T_s = 0$, otherwise $T_s = 1$.

4.3. **Mining design process.** In this paper, the distance and area characteristics between any two points on the model surface are used to calculate the modelling structure eigenvalues. The corresponding adjacency matrix is constructed through the analysis of face similarity calculation, and the target maximal group is mined using the GAWWO algorithm, thus simulating the design process of human innovative thinking.

After performing the similarity calculation, the different models are to be combined with certain weights. It is assumed that a complete combined modelling model case contains the property map of a substructure and its adjacency matrix. Then the corresponding adjacency matrix of the maximal clusters of the substructure property graph is shown below:

$$R = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 3 & 0 \\ 0 & 3 & 0 & 5 \\ 1 & 0 & 5 & 0 \end{pmatrix} \tag{19}$$

In this paper, the GAWWO algorithm is used to implement the problem of optimally solving the largest clusters in the graph. The inputs to the GAWWO algorithm are the association graph G and the substructure to be mined S. The association matrix M and the matrix size n are calculated. The substructure size k is set to 3.

The selected GAWWO adaptation value function is defined as follows:

$$fit(S) = k \cdot (k-1) - M_{ij} \tag{20}$$

where M_{ij} is calculated by Equation (15), specifically by calculating $dis(M_i, M_j)$.

Enter the loop and calculate the adaptation value of the new structure f_{new}. For the k water waves obtained after updating, they are evaluated for adaptation. If any of them is superior to the original water wave, the water wave with the highest adaptation value is used to replace the original water wave. If all the resulting water waves are inferior to the original water wave, the original water wave is kept unchanged. Assuming that the adaptation value of the initial subgraph is f_{int}, then if $f_{new} \leq f_{int}$, then the structure is retained, otherwise it is accepted with some probability.

5. **Experimental results and analyses.**

5.1. **Experimental environment.**

The PC processor is Intel(R) Xeon(R) CPU E5-16200@3.60GHz, 8GB of RAM and Windows 10 operating system. The application software is MATLAB 2014b.

5.2. **Analysis of algorithm experimental results.**

In order to verify the effectiveness of the proposed novel bionic optimisation algorithm, this paper compares the performance of the GAWWO algorithm with four algorithms, namely GA, PSO, WWO and GAPSO. A total of five test functions, two single-peak functions and three multi-peak functions [30], are used for testing. The single-peak test functions are suitable for the evaluation of the mining process and the multi-peak test functions are suitable for the evaluation of the detection process.

Among the five test functions, F1 and F2 are single-peak functions with one global extreme point, and F3, F4 and F5 are multi-peak functions with multiple local extreme points. The test function information is shown in Table 2 below.

Table 2. Test Functions

No.	Function (math.)	An equation
F1	Sphere	$f_1(x) = \sum_{i=1}^{n} x_i^2$
F2	Rosenbrock	$f_2(x) = \sum_{i=1}^{n} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$
F3	Rastrigrin	$f_3(x) = \sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi x_i) + 10)$
F4	Ackley	$f_4(x) = -20\left(\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right)\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$
F5	Griewank	$f_5(x) = -\frac{1}{4000}\sum_{i=1}^{n} x_i^2 + \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$

The parameters of the GAWWO algorithms are set as follows: wavelength $\lambda = 0.5$; wave height $h = 12$; wavelength attenuation coefficient $\alpha = 1.003$; wave breaking coefficient $\beta \in [0.001, 0.25]$; chaos coefficient $\mu = 3$; adaptive crossover probabilities $P_{c1} = 0.4$,

$P_{c2} = 0.6$, and $P_{c3} = 0.9$; and adaptive variance probabilities $P_{m1} = 0.001$, $P_{m2} = 0.01$, and $P_{m3} = 0.1$.

In the experiment, each algorithm on each of the five test functions independently and repeatedly run 30 times to ensure the reliability of the results. Table 3 summarises the results of the optimal value mean and standard deviation of the 5 algorithms on the five test functions.

The GAWWO algorithm achieves optimal mean values on F1, F2, F3, and F5, and better values on F4, and the corresponding standard deviations are also the smallest or second smallest. These experimental results show that GAWWO has strong robustness and adaptability.

Table 3. Statistics of optimisation results of different algorithms on F1-F5

Function	Mean/variance	GA	PSO	WWO	GAPSO	GAWWO
F1	average value	6.98E+04	5.50E-01	1.38E+02	2.13E-28	1.56E-45
	(statistics) standard deviation	3.63E+04	3.36E-01	1.65E+02	1.76E-28	5.07E-46
F2	average value	2.08E+08	8.70E+02	1.22E+05	2.35E+03	7.35E-02
	(statistics) standard deviation	1.09E+08	1.02E+03	1.69E+05	3.62E+03	2.14E-01
F3	average value	5.07E+02	1.86E+02	1.73E+02	1.34E+02	1.32E+01
	(statistics) standard deviation	2.68E+02	9.69E+01	1.33E+02	7.54E+01	6.23E+00
F4	average value	1.32E+00	9.15E-04	1.17E-12	1.57E-14	2.01E-13
	(statistics) standard deviation	1.07E+00	8.14E-04	2.04E-12	1.61E-14	1.50E-15
F5	average value	1.82E+01	6.51E-02	6.85E-01	2.12E-02	1.12E-16
	(statistics) standard deviation	9.79E+00	4.26E-02	3.77E-01	3.24E-02	1.54E-16

5.3. **Analysis of results of modelling examples.** After reaching the subgraph number with the minimum target fitness value, the combined result of snowman expression modeling elements is obtained, and the combined modeling result is shown in Figure 6. After the user satisfaction survey, it is concluded that all 15 cartoon expressions get more than 70 (out of 100).

Furthermore, the similarity value metrics were employed for evaluating the maximum clusters mining outcomes of the GAPSO and GAWWO algorithms in order to precisely and quantitatively evaluate the mining performance of the methods.

The average mining metrics and time comparisons of the results of the 15 cartoon expression modelling elements are shown in Table 4. It can be seen that compared with the GAPSO algorithm, the GAWWO algorithm consumes less time, only 0.02109 s. At the same time, the similarity between the structures produced by the GAWWO algorithm is lower, only 0.608, which is 23.1

Table 4. Mining indicators and time comparison

Arithmetic	Similarity	Time/s
GAPSO	0.791	0.2510
GAWWO	0.608	0.2109

6. **Conclusion.** This work proposes a method for designing diverse styling elements for social networks based on the GAWWO algorithm. Based on the styling elements, triangular mesh sample points conforming to geometric topological properties are generated. A novel bionic optimisation algorithm (GAWWO) is proposed by introducing improved genetic operators and chaotic optimisation strategies into the water wave optimisation algorithm in order to obtain better search capabilities.The test results of five generalised

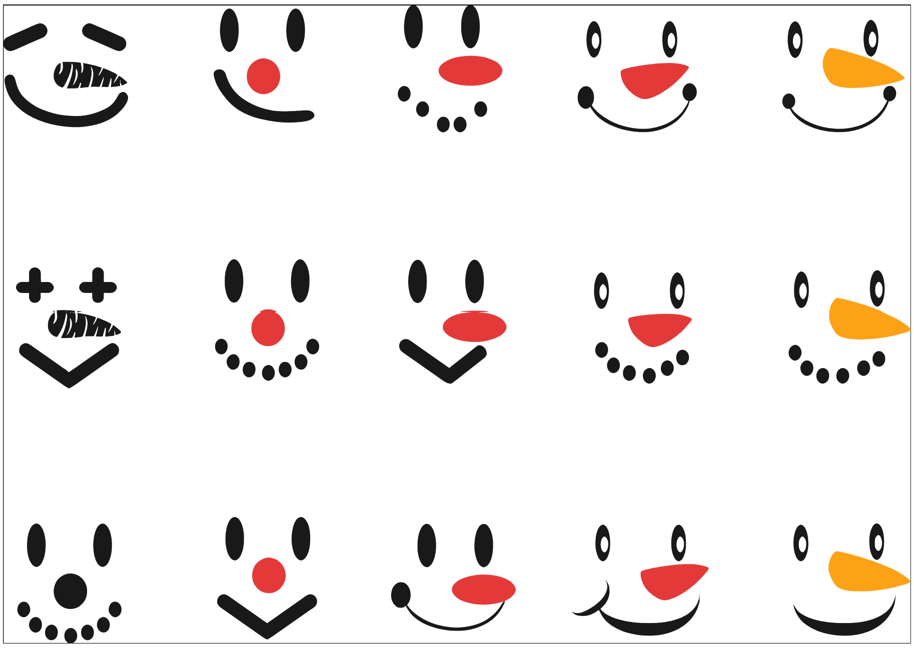

Figure 4. Cartoon expression results

arithmetic cases show that the GAWWO algorithm possesses a better performance and is able to better balance the local and global search capabilities. Finally, the effectiveness and practicality of the proposed method is verified by using the GAWWO algorithm to mine the optimisation problem of the largest group to solve the optimisation problem with a specific application of cartoon expression modelling. However, only the similarity value as a quantitative assessment index cannot objectively evaluate the mining performance of the algorithm in a more comprehensive way, which means that the practical application still needs to find indicators with wider applicability. Therefore, further research will be carried out on the evaluation indexes of product styling design based on intelligent evolutionary algorithm.

Acknowledgement. This work was supported by Jiangsu Provincial Social Science Foundation Project (No. 20YSB010), Research topic on philosophy and social sciences in universities in Jiangsu Province (No. 2022SJYB0556), and Philosophy and Social Research Science Institute of Nanjing Vocational University of Industry Technology (No. 2021SKYJ21).

REFERENCES

[1] S. L. Han, P. Y. Chan, P. Venkatraman, P. Apeagyei, T. Cassidy, and D. J. Tyler, "Standard vs. upcycled fashion design and production," *Fashion Practice*, vol. 9, no. 1, pp. 69-94, 2017.

[2] J. He, "Application of cartoon modeling in VI design: Take 'Solar Eclipse' as an example," *Highlights in Art and Design*, vol. 1, no. 1, pp. 1-6, 2022.

[3] L. Li, F. Liu, C.-h. Peng, and B. Gao, "Computer-aided 3D human modeling for portrait-based product development using point-and curve-based deformation," *Computer-Aided Design*, vol. 45, no. 2, pp. 134-143, 2013.

[4] M. R. González-Rodríguez, M. C. Díaz-Fernández, and C. P. Gómez, "Facial-expression recognition: An emergent approach to the measurement of tourist satisfaction through emotions," *Telematics and Informatics*, vol. 51, 101404, 2020.

[5] Y. Jing and Y. Song, "Application of 3D reality technology combined with CAD in animation modeling design," *Computer-Aided Design and Applications*, vol. 18, no. S3, pp. 164-175, 2020.

[6] P. Hong, Z. Wen, and T. S. Huang, "Real-time speech-driven face animation with expressions using neural networks," *IEEE Transactions on Neural Networks*, vol. 13, no. 4, pp. 916-927, 2002.
[7] M. Fang and F. Zhou, "Research on derivative products design of pottery and cartoon," *Applied Mechanics and Materials*, vol. 312, pp. 972-975, 2013.
[8] T. Weise, S. Bouaziz, H. Li, and M. Pauly, "Realtime performance-based facial animation," *ACM Transactions on Graphics (TOG)*, vol. 30, no. 4, pp. 1-10, 2011.
[9] T.-Y. Wu, A. Shao, and J.-S. Pan, "CTOA: Toward a chaotic-based tumbleweed optimization algorithm," *Mathematics*, vol. 11, no. 10, 2339, 2023.
[10] T.-Y. Wu, H. Li, and S.-C. Chu, "CPPE: An improved phasmatodea population evolution algorithm with chaotic maps," *Mathematics*, vol. 11, no. 9, 1977, 2023.
[11] 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.
[12] C. K. Fung, C. K. Kwong, K. Y. Chan, and H. Jiang, "A guided search genetic algorithm using mined rules for optimal affective product design," *Engineering Optimization*, vol. 46, no. 8, pp. 1094-1108, 2014.
[13] M. Zhao and M. Ghasvari, "Product design-time optimization using a hybrid meta-heuristic algorithm," *Computers & Industrial Engineering*, vol. 155, 107177, 2021.
[14] Z. Li, L. Chen, C. Liu, F. Zhang, Z. Li, Y. Gao, Y. Ha, C. Xu, S. Quan, and Y. Xu, "Animated 3D human avatars from a single image with GAN-based texture inference," *Computers & Graphics*, vol. 95, pp. 81-91, 2021.
[15] L. Huang and P. Zheng, "Human-computer collaborative visual design creation assisted by artificial intelligence," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 9, pp. 1-21, 2023.
[16] N. U. Din, K. Javed, S. Bae, and J. Yi, "A novel GAN-based network for unmasking of masked face," *IEEE Access*, vol. 8, pp. 44276-44287, 2020.
[17] S. Shahriar, "GAN computers generate arts? A survey on visual arts, music, and literary text generation using generative adversarial network," *Displays*, vol. 73, 102237, 2022.
[18] A. Kaveh and M. Khayatazad, "A new meta-heuristic method: ray optimization," *Computers & Structures*, vol. 112, pp. 283-294, 2012.
[19] K. S. Lee and Z. W. Geem, "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 36-38, pp. 3902-3933, 2005.
[20] H. Yapici and N. Cetinkaya, "A new meta-heuristic optimizer: Pathfinder algorithm," *Applied Soft Computing*, vol. 78, pp. 545-568, 2019.
[21] Y.-J. Zheng, "Water wave optimization: a new nature-inspired metaheuristic," *Computers & Operations Research*, vol. 55, pp. 1-11, 2015.
[22] Y.-J. Zheng, X.-Q. Lu, Y.-C. Du, Y. Xue, and W.-G. Sheng, "Water wave optimization for combinatorial optimization: Design strategies and applications," *Applied Soft Computing*, vol. 83, 105611, 2019.
[23] J. Zhang, Y. Zhou, and Q. Luo, "Nature-inspired approach: a wind-driven water wave optimization algorithm," *Applied Intelligence*, vol. 49, pp. 233-252, 2019.
[24] A. Kaur and Y. Kumar, "A new metaheuristic algorithm based on water wave optimization for data clustering," *Evolutionary Intelligence*, vol. 15, no. 1, pp. 759-783, 2022.
[25] S. B. Munch, T. L. Rogers, B. J. Johnson, U. Bhat, and C.-H. Tsai, "Rethinking the prevalence and relevance of chaos in ecology," *Annual Review of Ecology, Evolution, and Systematics*, vol. 53, pp. 227-249, 2022.
[26] Z. Man, J. Li, X. Di, Y. Sheng, and Z. Liu, "Double image encryption algorithm based on neural network and chaos," *Chaos, Solitons & Fractals*, vol. 152, 111318, 2021.
[27] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimedia Tools and Applications*, vol. 80, pp. 8091-8126, 2021.
[28] A. Sohail, "Genetic algorithms in the fields of artificial intelligence and data sciences," *Annals of Data Science*, vol. 10, no. 4, pp. 1007-1018, 2023.
[29] M. A. Albadr, S. Tiun, M. Ayob, and F. Al-Dhief, "Genetic algorithm based on natural selection theory for optimization problems," *Symmetry*, vol. 12, no. 11, pp. 1758, 2020.
[30] C. Kim, R. Batra, L. Chen, H. Tran, and R. Ramprasad, "Polymer design using genetic algorithm and machine learning," *Computational Materials Science*, vol. 186, 110067, 2021.