Hindawi
Mobile Information Systems
Volume 2022, Article ID 1244352, 8 pages
https://doi.org/10.1155/2022/1244352

Research Article

Packaging Decorative Pattern Design Based on Genetic Algorithm

Yunming Zhu[1] and Tong Gao[2]

[1]*Tongmyong University, Department of Design, 428 Sinseon-ro, Nam-gu, Busan, Republic of Korea*
[2]*College of Media and Art, Nanjing University of Posts and Telecommunications, Nanjing, China*

Correspondence should be addressed to Tong Gao; gaotong@njupt.edu.cn

Received 27 May 2022; Revised 7 July 2022; Accepted 22 July 2022; Published 31 August 2022

Academic Editor: Mian Ahmad Jan

The science, art, and technology of covering or protecting things for distribution, storing, selling, and usage are known as packaging. Decorative design is the process of applying color, line, texture, or pattern to an object. Decorative design can be used to complement a structural design, or it can simply be a piece of art. Designers not only have the necessary design knowledge and expertise but they should also have certain observation, inventiveness, and other conditions. When it comes to packaging materials, most businesses want their products to be unique and attractive. The inventive design has entirely relied on people's thinking, and there is no technology or approach that can substitute human creative thinking. A genetic algorithm (GA) is a metaheuristic motivated by natural selection that relates to the wider family of evolutionary algorithms (EA) in computer science and operations research. Therefore, this paper first introduces the theoretical basis of genetic algorithms and then describes the generation process of packaging decorative patterns. Following the generation of the first population by coding the control points of the initial design line, the genetic algorithm is used to perform genetic operations on it in order to obtain multiple design solutions. The study found that this approach can alter the basic design scheme and provides an efficient way for the inventive creation of packaging decorative patterns, assisting designers in improving design efficiency and developing design ideas.

1. Introduction

The science, art, and technology of enclosing or protecting things for distribution, storage, sale, and usage are known as packaging. The process of creating, assessing, and making packages is sometimes referred to as packaging [1]. Packaging is a coordinated method of preparing items for transportation, warehousing, logistics, sale, and final use. It is thoroughly integrated into government, corporate, institutional, industrial, and personal use in many nations. Decorative design is the process of applying color, line, texture, or pattern to an object. Decorative design can be used to complement a structural design, or it can simply be a piece of art [2].

Design is a technology that requires a high level of comprehensive competence. Designers must not only have appropriate design knowledge and expertise but they must also possess particular observation, inventiveness, and other characteristics [3]. When it comes to packing materials, businesses often want their products to stand out and be attractive. As a result, there is an urgent need for a computer-aided calculation model and calculation tool to create an atmosphere in which designers can inspire creative inspiration and generate original and distinctive designs. The packaging decorative pattern works by adjusting parameters to meet the needs of various businesses. [4].

In this study, a genetic algorithm is used in the design process using analysis and research of the packaging decorative pattern design process, providing a platform for designers to generate ideas and enhance efficiency [5]. The genetic algorithm-based curve-fitting algorithm is a robust and high-precision fitting algorithm. The adaptive genetic algorithm is used, and the generation of the initial population is constrained by a basic curve tool based on existing data points, which can enhance the speed of curve fitting [6]. A package decoration pattern design model based on genetic algorithms is created during the packaging decoration pattern design process [7]. The pattern material is analyzed and expressed using the curve-fitting tool, and the designer through modifying generates different design schemes or directly genetic operation on the design curve, and the curve fitting is applied to the design process to achieve the goal of effective material use.

The rest of the paper is organized as follows: in section 2, the genetic algorithm is described, section 3 is composed of the packaging decorative pattern generation process, and finally, the paper is concluded in section 4.

2. Genetic Algorithm

Genetic algorithm is a heuristic global search algorithm. Genetic algorithms have produced substantial application successes and theoretical research advancements after more than 20 years of development [8]. Evolutionary intelligence has emerged as an important area of study in artificial intelligence [9]. The use of genetic algorithms in the field of design is made possible by research on innovative design based on genetic algorithms [10]. Although the calculating procedure and structure of the genetic algorithm are simple, the mechanism through which it operates is extremely complex [11]. Genetic algorithm contains the characteristics of a search process as well as an optimization mechanism, and its mathematical aspects can be described using analysis such as pattern theorem and construction hypothesis [12]. Markov chain is another excellent tool for evaluating genetic algorithm [13]. With the application of genetic algorithms in difficult optimization problems and practical engineering design, people are paying greater attention to genetic algorithm theory. Because the benefits of genetic algorithms in application research are primarily in their effectiveness of solution, reliability, extensibility, and ease of combination with other methods, it is not difficult to predict that with the continuous deepening of relevant theoretical research and the continuous expansion of application fields, genetic algorithms and evolutionary computing will make significant progress [14].

2.1. Basic Principle of Genetic Algorithm. Since the success of simulating biological evolution towards problem-solving, people have attempted to conceptually examine it in order to explain the usefulness of genetic algorithms. Holland presented the pattern theorem and implicit parallelism theory based on pattern analysis to explain the efficacy of the genetic algorithm, which is the underlying theorem of the genetic algorithm [15].

Schema: a template describes a set of strings. Some points in the collection of strings have similarities.

Scheme order: the order of the mode is the number of locations defined in mode (h) which is recorded as $O(H)$.

Distance of mode: the distance between the first determined position and the last determined position in mode h is recorded as $\delta(H)$.

Pattern theorem: patterns with low order, short defining moment, and average fitness greater than population fitness will increase exponentially in offspring as a result of selection, crossover, and mutation of genetic operators. The amount of a certain mode in the next generation can be generally expressed as (1) under the action of selection crossover and mutation operation:

$$m(H, t+1) \geq m(H, t)\frac{f(H)}{\overline{f}}\left[1 - \frac{P_c \cdot \delta(H)}{l-1} - O(H)\cdot P_m\right]. \tag{1}$$

Goldberg made a reasonable analysis of the parallelism of genetic algorithm and pointed out that the efficiency of genetic algorithm in pattern processing is $O(n^3)$, which Holland called "implicit parallelism" of the genetic algorithm [16]. This finding demonstrates that genetic algorithms may effectively parallelize operations.

2.2. Convergence of Genetic Algorithms. The analysis for the global convergence of genetic algorithms has important theoretical significance. As a result, more study on the temporal complexity for global convergence of genetic algorithms is equally important to the actual use of genetic algorithms [17]. Pattern-based efficiency analysis is one of the examples for the convergence of genetic algorithms. The running process of a genetic algorithm, from the perspective of a pattern, is the process of searching for the best pattern in the pattern space, and the search duration is determined by the size of the pattern space and the search efficiency of each genetic algorithm [18]. Because the former is dictated by specific challenges, the efficiency of each genetic algorithm is critical to the efficiency of the genetic algorithm. It is determined by two factors: the new pattern created by each genetic algorithm, and the pattern effectively kept by each genetic algorithm. An effective genetic algorithm should strike a balance between these two factors. To achieve a high search efficiency, it must be dynamically unloaded throughout the operation of the genetic algorithm.

2.3. Operation Process of Genetic Algorithm. The genetic operation of genetic algorithms is random throughout the evolution process, but it is not a completely random search. It can successfully utilize previous data to predict the merit-seeking set with a better-predicted performance of the next generation [19]. This constant evolution from generation to generation eventually converges on an entity, which is suited to the environment and achieves the best solution to the challenge. The genetic algorithm consists of five components [20]: parameter coding, setting of initial population, design of fitness function, design of genetic operation, and setting of control parameters. The fundamental steps are as follows:

Step 1. select the coding strategy and then define the fitness function and various parameters

Step 2. initialize the generated group P

Step 3. after decoding, compute the fitness of particular position strings in the group

Step 4. determine whether the optimization conditions are satisfied. If so, choose the best candidate, and the algorithm will be terminated.

Step 5. after selecting, crossing, and mutating the group, proceed to Step 3.

3. Packaging Decorative Pattern Generation Process

The generation process of packaging decorative pattern is consisting of the following sections.

3.1. Generation of Initial Population. Before designing, the range of solution space or the size of pattern segments must be determined. Because the geometry of a segment in a circular pattern is a sector, the angle and radius can uniquely identify a segment. Since the radius is a variable that can be altered at any moment, the angle value is usually employed to define the size of a segment. $\alpha = \pi/4$ and $r = 1$ are selected in the context of this work. Once the sector size is determined, the search space is limited to the plane of the sector. For simplicity, the quadratic parabolic curve and straight line with three type value points are selected as essential primitives. A conic's type value points are the fixed points and two endpoints of a parabola, while a straight line's type value points are its two endpoints. The initial design scheme is shown in Figure 1.

Figure 2 shows the control points of the segment, where a, b, c and d, e, f are two parabolic curves while h, g and g, i are two straight-line segments. The two edges can be used as a pattern segment or just as a reference boundary line. The two boundaries are shown in the clip, and curves and line segments are regarded together as lines. The segments are first replicated symmetrically and then rotated to make a complete segment. The general pattern in the real design is not confined to the fan-shaped region, as detailed in the following supplemental instructions. Except for points e and b, it is easy to observe that the remaining control points are on the sector's borders. Figure 2 is presented although the rationale is for the general curve closure, that is, using the curve to construct other forms. For the sake of clarity, the following definitions are provided.

Inner points: the inner point of a fragment is the type value point that falls within a fragment. Figure 2 shows points e and b.

Outer points: the type value points that lie on the segment border are referred to as the segment's outer points. Points other than e and b are depicted in Figure 2.

3.2. Binary Coding. There is no difference between the inner point and the outer point in terms of the control curve. The only difference is one position coordinate of the outer point, radius r. The coding is relatively simple, and the radius value is directly binary converted. The position coordinates of the inner point are composed of radius r and angle α, and the radius and angle variables are coded separately. According to research, there are two approaches to dealing with the coding of two-point initial populations.

(1) Limited interior points: if the number of interior points is less than 5, the algorithm is set to 2, the angle value, as well as radius coding, creates a population, and the genetic operation is performed concurrently.

(2) Many interior points: if there are more than 5 interior points, the angular values of the interior points are coded to establish an initial population, and the genetic operation is conducted individually.

As the angle and radius indicate two separate plane locations, they should be addressed differently in theoretical aspects. However, throughout the research process, it was discovered that when there are fewer interior locations, the effect of the genetic operation is equally optimal. When there are several interior sites, the genetic operation of a population must be performed independently. Because the decimal point is precise to three digits, it can be represented by 10 binary digits. Perform binary conversion on the starting population. The conversion results are shown in Table 1.

3.3. Genetic Manipulation. After the generation of the original population, some individuals are chosen for genetic operation, and changes in segments as well as the whole are monitored. Because there is no fitness function and the designer specifies the beginning curve and branch line, the constraint criteria must be satisfied. As a result, people are chosen at random for genetic procedures.

3.3.1. Crossover Operation. Here, point crossing or multipoint crossing can be selected for crossing operation. Both methods can bring changes to lines. Here, the method of multipoint intersection is used as an example. Suppose that the two selected points are r_a and r_g, the segment length is 4 bits. Randomly select the intersection, where the underline represents the crossed bit segment, and the results are as follows:

Father: r_a 1 1 1 1 0 0 1 1 0 0 .
r_g 0 0 1 1 1 0 1 0 1 0 .
Son: r_{a1} 1 0 1 1 1 0 1 1 0 0 .
r_{g1} 0 1 1 1 0 0 1 0 1 0 .

The values of r_g after crossing have changed, and the binary code of the population after crossing is shown in Table 2. After crossing, the segment and the whole are compared, as shown in Figure 3.

The forms of segments before and after crossing are quite comparable, as shown in Figures 1 and 3, but the changes between the whole before and after crossing are more visible than those between segments. The intersection in this case chooses the higher bit segment. If the intersection chooses the lower bit, the change will be minimal. For example, select the lower 4 bits to cross, and the results are as follows:

r_a 1 1 1 1 0 0 1 1 0 0 .
r_g 0 0 1 1 1 0 1 0 1 0 .

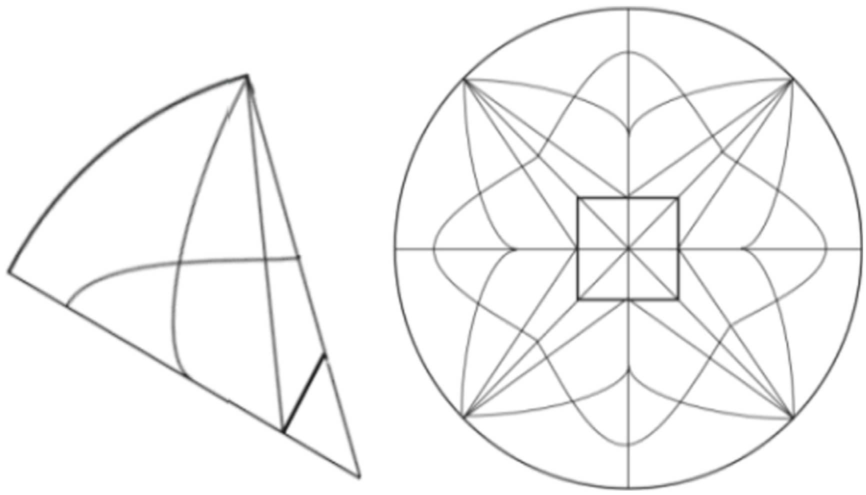

Figure 1: Fragment and overall schematic diagram of the initial design scheme.

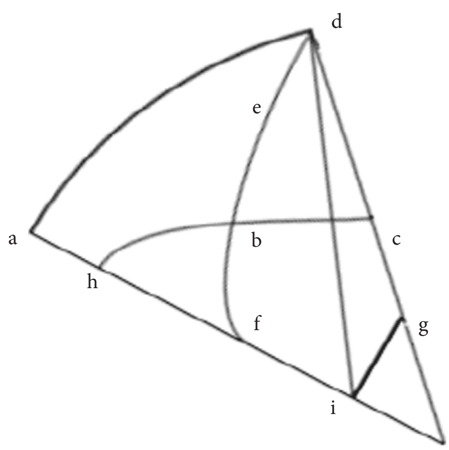

Figure 2: Schematic diagram of control points in clip.

After crossing:

r_{a2} 1 1 1 1 0 0 1 0 1 0 .

r_{g2} 0 0 1 1 2 0 1 1 0 0 .

At this time, the values of r_a and r_g have changed little. If they are displayed in a relatively simple pattern, there is almost no change. Therefore, the level of intersection points can be used to control the degree of pattern change during the crossing operation. To make the change significant, either specify a higher bit segment for genetic operation or increase the selection probability of the higher bit segment.

3.3.2. Mutation Operation. When compared to crossover operations, mutation operations make it easier to regulate an individual. Multiple individuals can be chosen to define greater variation digits, resulting in significant changes. Two individuals are used for illustration in this case. Mutations at single or multiple locations can be employed. A single-point mutation is employed in this case. Here, points d and i of the initial population are selected for mutation operation. In order to make the change significant, the mutation position is designated as the highest position. The results of the mutation operation are as follows:

Table 1: Binary code table of the initial population.

r_a	0.950	1	1	1	1	0	0	1	1	0	0
r_b	0.672	1	0	1	0	1	1	0	1	0	1
r_c	0.434	0	1	1	0	1	1	1	0	1	0
r_d	0.803	1	1	0	0	1	1	0	0	0	0
r_e	0.782	1	1	0	0	0	1	1	1	1	1
r_f	0.481	0	1	1	1	0	0	1	0	0	1
r_g	0.228	0	0	1	1	1	0	1	0	1	0
r_h	0.762	1	0	1	1	1	1	0	1	1	0
r_i	0.253	0	1	0	0	0	0	1	1	0	1
α_b	0.432	1	0	0	0	0	0	1	1	0	1
α_e	0.153	0	0	1	1	1	1	0	0	1	0

Table 2: Binary code of population after crossing.

r_{a1}	0.732	1	0	1	1	1	0	1	1	0	0
r_b	0.674	1	0	1	0	1	1	0	1	0	1
r_c	0.434	0	1	1	0	1	1	1	0	1	0
r_d	0.802	1	1	0	0	1	1	0	0	1	1
r_e	0.782	1	1	0	0	0	1	1	1	1	1
r_f	0.476	0	1	1	1	0	0	1	0	0	1
r_{g1}	0.476	0	1	1	1	0	0	1	0	1	0
r_h	0.764	1	0	1	1	1	1	0	1	1	0
r_i	0.256	0	1	0	0	0	0	0	0	1	1
α_b	0.402	1	0	0	0	0	0	1	1	0	1
α_e	0.183	0	0	1	1	1	1	0	0	1	0

Variation of point d.

Father: r_d 1 1 0 0 1 1 0 0 1 1 .

Son: r_{d1} 0 1 0 0 1 1 0 0 1 1 .

Variation of point e.

Father: r_i 1 1 0 0 0 0 0 0 0 1 .

Son: r_{i1} 0 1 0 0 0 0 0 0 0 1 .

The binary code of the mutated population and comparison between the segment and the whole after mutation are shown in Table 3 and Figure 4. It can be seen that the

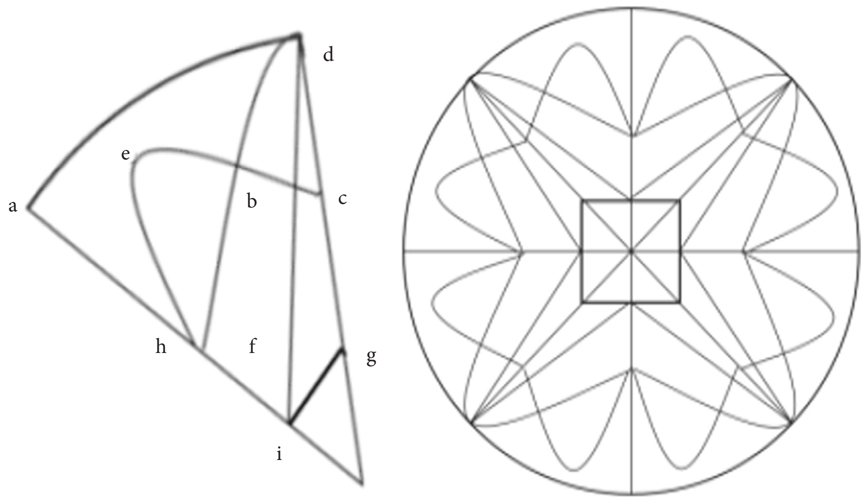

Figure 3: Schematic diagram of segments and whole after crossing.

Table 3: Binary codes of a mutated population.

r_{a1}	0.952	1	0	1	1	1	0	1	1	0	0
r_b	0.672	1	0	1	0	1	1	0	1	0	1
r_c	0.433	0	1	1	0	1	1	1	0	1	0
r_{d1}	0.301	1	1	0	0	1	1	0	0	1	1
r_e	0.783	1	1	0	0	0	1	1	1	1	1
r_f	0.477	0	1	1	1	0	0	1	0	0	1
r_{g1}	0.226	0	1	1	1	0	0	1	0	1	0
r_h	0.761	1	0	1	1	1	1	0	1	1	0
r_{i1}	0.782	0	1	0	0	0	0	0	0	1	1
α_b	0.431	1	0	0	0	0	0	1	1	0	1
α_e	0.182	0	0	1	1	1	1	0	0	1	0

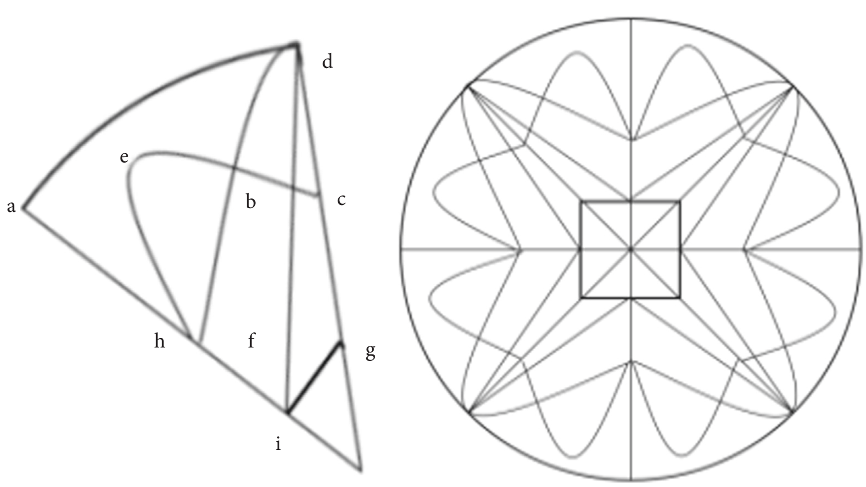

Figure 4: Schematic diagram of fragment and whole after variation.

variation operation is a separate operation on the control points, and the change of lines will be significant, which will help to produce richer pattern results.

3.3.3. Constraints. There are two functions of setting constraints. When there are many curves, it will not correspond to the features of decorative patterns in order to avoid undesired crossings. Constraints are imposed by the opposite party in order to preserve particular line features. For example, when designing, the d, e, and f curves in the first segment should be concave-convex, which means that the opening direction of the parabola should always point to the sector's center, to maintain the overall effect. If the d, e, f

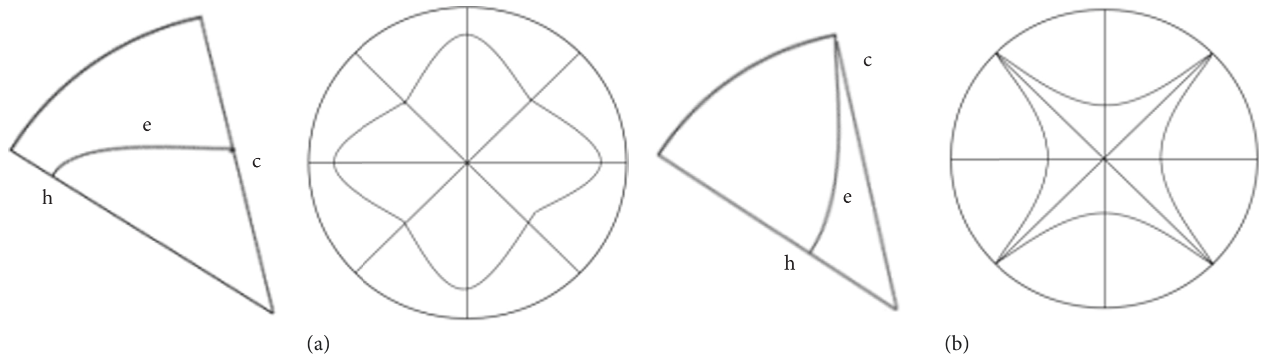

Figure 5: Change caused by curve concavity and convexity in the segment. (a) Protrude outward, (b) Concave outward.

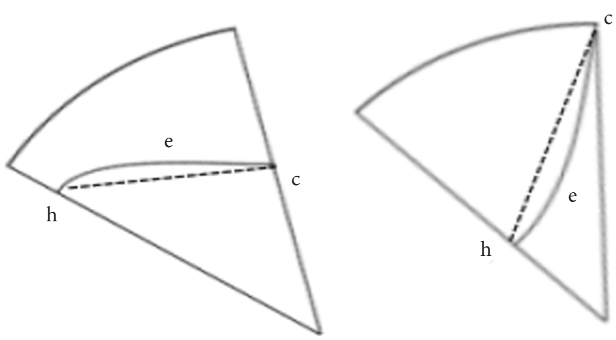

Figure 6: Positional relationship between the vertices of the quadratic curve and the midline connecting the two ends.

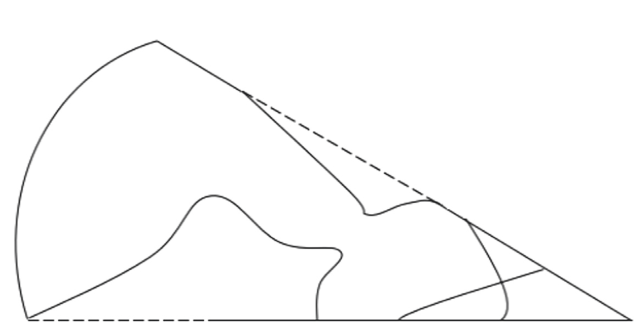

Figure 8: Schematic diagram of the line generated by genetic operation of the initial curve.

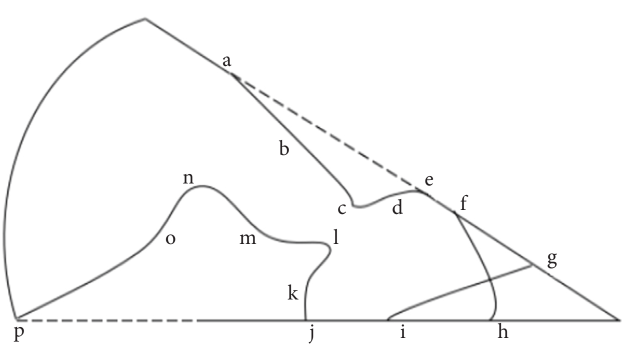

Figure 7: Initial curve and corresponding control points.

opening is outward, the overall shown curve is rather smooth, reflecting a wide circular form; if the opening is inward, a sharp shape is presented on the whole, changing the overall features, as shown in Figure 5.

Designers might not want the shift in segment concavity and convexity to occur while creating decorative patterns. One solution is to increase the number of iterations, change the lines as much as possible, and manually select the shape that meets the needs. Another method can constrain the convexity and convexity of the parabola, if this is not required but arbitrarily changed, then these constraints are not required. To determine the concavity and convexity of a curve, examine the entire functional equation, with the important information being the control point or type value point of the curve. Therefore, the method of comparing value points and related points can be used to speed up the judgement of concavity and convexity of the curve and increase the generality of the judgement method. For example, in this procedure, compare the radius of the parabola's vertex and the point at the two endpoints directly.

If the curve is to be convex, it must meet $f = (r_e + r_m) > 0$, and if there are several constraints, simply multiply them. Let one constraint be f_i, $1 = 1, 2, \ldots n$, where n represents the number of constraints, and the overall constraint is represented by F.

$$F = \prod_{i=1}^{n} f_i. \tag{2}$$

3.4. Design Generation. Select $\alpha = \pi/8$, and the segment is obtained by combining the elements of conic and straight line segments, as shown in Figure 7.

Assume that the intended outcome feature is to maintain the curves ac and ce distinct. The curves pn, nl, and lj do not intersect. Keep the bump on all bends. Point p is still in its original location. As illustrated in Figure 8, a new line can be obtained following the initial genetic procedure.

When a suitable result is stored, the user can select whether to halt the genetic procedure or continue in order to obtain additional optional outcomes. Figure 9 shows a pretty

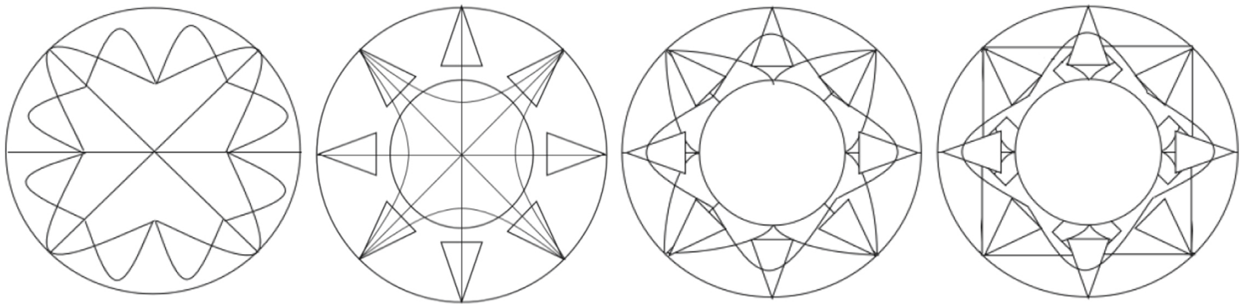

Figure 9: After 100 steps of genetic operation, a relatively satisfactory result graph is produced.

excellent outcome pattern obtained after 100 stages of genetic operation.

4. Conclusion

Pattern design plays a very important role in the field of the art design. Packaging decorative patterns have been used in a range of decorative industries due to their variety, numerous shapes, and great flexibility. Packaging decorative pattern design has always been a source of broad concern as one of the most essential and vast elements of pattern design. However, developing packaging decorative patterns is a highly complicated as well as free labor, and the algorithm has not yet reached the standard of producing packaging decorative patterns arbitrarily. Therefore, it is required to learn more from the design structure and features of certain commercial or open-source pattern design software and to improve the function and efficiency, such as by introducing more convenient curve tools and color systems. As a result, in this study, a genetic algorithm is used in the design process through analysis and research of the packaging decorative pattern design process, providing a platform for designers to create ideas and enhance efficiency. Curve fitting based on a genetic algorithm is a stable and high-precision fitting algorithm. The adaptive genetic algorithm is used in the method, and the initial population generation is constrained by a basic curve tool based on existing data points, which can enhance the speed of curve fitting. A package decoration pattern design model based on genetic algorithms is created during the packaging decoration pattern design process. The pattern material is analyzed and expressed using the curve fitting tool, and then, different design schemes are generated by the designer through modifying or directly performing genetic operations on the design curve, and the curve fitting is applied to the design process to achieve the goal of effective material use. Then, several algorithms and color matching techniques are employed for pattern color matching, and finally, the packaging decorative pattern works with visual impact are developed.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] W. Lu, Y. Chen, and X. Bian, "Contemporary packaging design on the elements of the Republic of China," *IOP Conference Series: Materials Science and Engineering*, vol. 573, no. 1, Article ID 012065, 2019.

[2] Z. Kazlacheva, "Pattern design of twisted draperies with decorative and constructive function," *Applied Researches in Technics, Technologies and Education*, vol. 7, no. 1, pp. 1–9, 2019.

[3] W. Wang, G. Zhang, L. Yang, and W. Wang, "Research on garment pattern design based on fractal graphics," *Eurasip journal on image and video processing*, vol. 2019, no. 1, p. 29, 2019.

[4] D. Reiser, A. von Keudell, and T. Urbanietz, "Determining Chemical Reaction systems in Plasma-assisted conversion of Methane using genetic algorithms," *Plasma Chemistry and Plasma Processing*, vol. 41, no. 3, pp. 793–813, 2021.

[5] P. B. Kumar and D. R. Parhi, "Intelligent hybridization of regression technique with genetic algorithm for navigation of humanoids in complex environments," *Robotica*, vol. 38, no. 4, pp. 565–581, 2020.

[6] T. Delaforge and S. Mariethoz, "Design automation of power electronic converters a grid elitist multiobjective genetic algorithm," in *Proceedings of the 2020 IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 2892–2899, IEEE, New Orleans, LA, USA, 2020, March.

[7] S. Jafar-Zanjani, S. Inampudi, and H. Mosallaei, "Adaptive genetic algorithm for optical metasurfaces design," *Scientific Reports*, vol. 8, no. 1, p. 11040, 2018.

[8] N. Thirer, "An FPGA implementation of a self-adaptive genetic algorithm," in *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications II*, vol. 11413, pp. 655–661, SPIE, 2020, May.

[9] W. Zhang, X. Zhang, H. Shi, and L. Zhou, "RDDSACCGA: a reliable data distribution solution assisted by cloud computing based on genetic algorithm," *Cluster Computing*, vol. 22, no. S5, pp. 11069–11077, 2019.

[10] M. Jäger, R. Schäfer, and R. L. Johnston, "GIGA: a versatile genetic algorithm for free and supported clusters and nanoparticles in the presence of ligands," *Nanoscale*, vol. 11, no. 18, pp. 9042–9052, 2019.

[11] W. Pensasitorn, "The use of images in graphic design on packaging of food and beverages," *Journal of Economics, Business and Management*, vol. 3, no. 12, pp. 1159–1163, 2015.

[12] J. H. Lee and M. L. Chang, "Stimulating designers' creativity based on a creative evolutionary system and collective intelligence in product design," *International Journal of Industrial Ergonomics*, vol. 40, no. 3, pp. 295–305, 2010.

[13] F. Petrie, *3,000 Decorative Patterns of the Ancient World*, Courier Corporation, 2013.

[14] R. Triwurjani, I. Fahriani, K. T. Muda, N. Ihsan, and P. E. Nugraheni, "Decorative Motifs as identity of Austronesian Speakers in South Minahasa: Preliminary analysis of decorative patterns of Pottery," in *Proceedings of the 9th Asbam International Conference (Archeology, History, & Culture In The Nature of Malay)(ASBAM 2021)*, pp. 190–195, Atlantis Press, Lansot Site, Tomohon City, North Sulawesi Province, 2022, April.

[15] D. E. Golberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, p. 36, Addion wesley, Boston, MA. United States, 1989.

[16] A. Kumar and V. C. Hand, "Feasibility of using neural networks and genetic algorithms to predict and optimize coated paper and board brightness," *Industrial & Engineering Chemistry Research*, vol. 39, no. 12, pp. 4956–4962, 2000.

[17] A. O. Bajeh and K. O. Abolarinwa, "Optimization: a comparative study of genetic and tabu search algorithms," *International Journal of Computer Applications*, vol. 31, no. 5, pp. 43–48, 2011.

[18] C. S. Chin and W. P. Lin, "Robust genetic algorithm and fuzzy inference mechanism embedded in a sliding-mode controller for an uncertain underwater robot," *IEEE*, vol. 23, no. 2, pp. 655–666, 2018.

[19] S. Chao, "Construction model of E-Commerce Agricultural product online Marketing system based on Blockchain and improved genetic algorithm," *Security and Communication Networks*, vol. 2022, Article ID 4055698, 2022.

[20] X. Y. Liu, Y. Liang, S. Wang, Z. Y. Yang, and H. S. Ye, "A hybrid genetic algorithm with wrapper-embedded approaches for feature selection," *IEEE Access*, vol. 6, pp. 22863–22874, 2018.