

HYBRID OPTIMIZATION AND DEEP LEARNING BASED
COLOR CLASSIFICATION FOR FRACTAL PATTERN
GENERATION IN DIGITAL DESIGN

Tastanova S. ¹, Tastanov M. , Aralov J. ,
Mosavi A. , Csoka M. , Gocs L. ,

Abstract Fractal-based methods are used to create geometric patterns for textile and carpet design. Color selection is constrained by production cost, composition, and visual quality. A computational method is proposed for color generation, classification, and optimization. Global search is performed by a genetic algorithm. Local refinement is conducted by the simplex method. Color combinations are classified by a convolutional neural network. Production cost and color quality are balanced through a weighted objective function. Optimization weights are adjusted through user feedback. Design priorities are therefore modified during each iteration. The method is evaluated with fractals derived from Pascal's triangle. Multiple candidate solutions are produced and locally refined. Patterns are assigned to warm, cool, or high-contrast categories. An AUC of 1.00 is obtained under the experimental conditions. A labor cost reduction of 10–15% is reported for the evaluated setting. Automated color selection for digital carpet design is supported by the method.

Key words: Carpet design; Color optimization; Convolutional neural network; Fractal geometry; Data science; Simplex method; Information systems; Machine learning; Modeling and simulation; Applied mathematics; Artificial intelligence.

AMS Mathematics Subject Classification: 68W50, 68T07, 68U10, 90C59, 28A80.

DOI: 10.32523/2306-6172-2026-14-3-127-140

1 Introduction

Fractals are defined by self-similarity, scale-dependent complexity, and recurrent geometric forms. Irregular structures in mathematics and nature are represented by fractal models [1, 2]. Fractal geometry is applied in mathematics, graphics, image analysis, and generative design. Complex visual structures are produced from simple mathematical rules. Such methods are examined in digital art and product design [3, 4, 5, 6]. These methods are valued in textile and carpet design. Pattern diversity, visual complexity, and color harmony are required in these fields. Diverse patterns are produced through recursive mathematical procedures. Geometry, spatial order, and visual composition are controlled by model parameters. Many design alternatives are therefore produced from consistent mathematical rules. Fractal methods are also applied in architecture and visual design. Relations between geometric complexity and visual appearance are examined [7, 8, 11, 12]. Pascal-type constructions are used to produce computational fractal patterns [4, 5]. Complex visual structures are thus obtained from few parameters. These methods are applied in textile and carpet production. Multiple geometric forms and color combinations are produced efficiently. Consistent mathematical rules are preserved across design variants. Fractal colors for carpet design are optimized with genetic algorithms [9, 23]. Alternative patterns are adapted to aesthetic and production requirements.

¹Corresponding Author.

However, production suitability is not ensured by visual complexity alone. Color parameters must also be selected and optimized.

Color selection is treated as a major textile design task. Visual harmony, contrast, perceived quality, and product appearance are affected. Many regions in complex fractals are assigned coordinated colors. Manual adjustment is slowed by numerous design alternatives. Dye costs and production requirements must also be considered [14, 15]. Color relationships are organized through established color theories. Itten's theory is used to assess visual color combinations [24]. The Ostwald system is used for systematic color analysis [25]. Perceptual color differences are represented through the CIE L*a*b* space [26].

Several opposing objectives are addressed during color selection. Material costs are reduced through lower-cost dyes. Additional resources may be required for preferred color combinations. Economic and aesthetic factors are therefore considered together. Parts of this process are automated through optimization and artificial intelligence. Suitable color parameters are identified through optimization methods. Generated patterns are classified through machine learning models [10, 16, 17, 21]. Complex parameter spaces are explored through genetic algorithms [9, 23, 27]. Separate objectives are treated through methods such as NSGA-II [30]. Color combinations are organized through broad theoretical categories. Warm, cool, and high-contrast groups are predefined in this study. Generated fractals are assigned to these groups during classification. However, aesthetic quality is not fully measured by these categories. Fractal generation and color optimization are addressed in earlier studies [9, 13, 23]. Economic optimization, user feedback, and automated classification are not fully integrated. Therefore, a combined computational method is proposed. Color parameters are generated, classified, and optimized for digital carpet design. Global exploration is performed by a Genetic Algorithm (GA). Local refinement is performed by the Simplex Method. Generated combinations are classified by a Convolutional Neural Network (CNN). Production cost and pattern quality are treated as optimization criteria. Criterion weights are adjusted through user feedback. Simplex refinement is supported by established function minimization methods [18]. Genetic and multi-objective methods are also considered [27, 30]. Image classification is performed through established CNN methods [28, 29]. Fractal designs and production parameters are therefore evaluated together.

2 Materials and Methods

The mathematical model, optimization procedure, feedback mechanism, and CNN classifier are described. Global and local methods are combined for cost and quality optimization.

2.1 Problem Formulation and Objective Function

The proposed system is used to optimize fractal color composition. Carpet and textile applications are considered. The color palette is represented by $\mathbf{x} = (x_1, x_2, \dots, x_n)$. The proportion of color i is denoted by x_i . The total color count is denoted by n . Color parameters can be represented in the HSV color space. The representation is selected according to generation and classification requirements. Two primary objectives are considered. Dye cost is minimized, and visual quality is maximized. Complex spaces are explored through evolutionary operators [27]. Genetic algorithms are applied to fractal carpet colors in earlier studies [9, 23]. Conflicts may be caused by costly preferred color combinations. A normalized weighted objective is defined for both criteria:

$$J(\mathbf{x}) = w_c C(\mathbf{x}) + w_q [1 - Q(\mathbf{x})], \quad (1)$$

$J(\mathbf{x})$ denotes the total optimization objective. Normalized cost is denoted by $C(\mathbf{x})$. Normalized quality is denoted by $Q(\mathbf{x})$. Cost and quality weights are denoted by w_c and w_q . These conditions are imposed:

$$w_c + w_q = 1, \quad w_c, w_q \geq 0. \quad (1)$$

The quality term is expressed as $1 - Q(\mathbf{x})$. The problem is defined as a minimization problem. The objective value is reduced by a higher quality score. Relative priorities are controlled by w_c and w_q . Cost reduction is emphasized by larger w_c values. Visual quality is emphasized by larger w_q values.

2.2 Cost and Quality Modeling

Material expenditure is estimated by the cost function. Unit dye prices and usage proportions are included:

$$C(\mathbf{x}) = \frac{\sum_{i=1}^n p_i x_i}{C_{\max}}, \quad (2)$$

The unit price of color i is denoted by p_i . Its usage proportion is denoted by x_i . Reference cost is denoted by $C_{\max}$. Cost normalization is provided by $C_{\max}$. The normalized cost is combined with the quality score.

Visual properties are evaluated by the quality function. Individual color effects and pairwise interactions are included:

$$Q(\mathbf{x}) = \sum_{i=1}^n f_i(x_i) + \sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq i}}^n g_{ij}(x_i, x_j), \quad (3)$$

The visual contribution of color i is represented by f_i . Pairwise color interactions are represented by g_{ij} . Color harmony, contrast, and compatibility can be represented by these interactions.

The quality score should be normalized to $[0, 1]$ before Equation (1). Functions f_i and g_{ij} are defined by selected evaluation criteria. User data or classification results may also be included. Established color theories can be used for evaluation [24, 25]. Numerical differences can be measured in CIE $L^*a^*b^*$ space [26].

2.3 Constraints and Penalty Function

Two main constraints are imposed on color proportions. First, negative proportions are prohibited:

$$x_i \geq 0, \quad i = 1, \dots, n. \quad (4)$$

Second, the proportions are required to sum to unity:

$$\sum_{i=1}^n x_i = 1. \quad (5)$$

Invalid negative quantities are prevented by the first constraint. A complete composition is ensured by the second constraint.

Constraint violations are penalized through the function below:

$$P(\mathbf{x}) = k_1 \sum_{i=1}^n \max(0, -x_i)^2 + k_2 \left(\sum_{i=1}^n x_i - 1 \right)^2, \quad (6)$$

Penalty coefficients are denoted by k_1 and k_2 . The penalized objective is defined as:

$$J_p(\mathbf{x}) = J(\mathbf{x}) + P(\mathbf{x}). \quad (7)$$

Penalty coefficients should be selected experimentally. Infeasible solutions should receive larger objective values. The same penalty is applied during global and local optimization.

2.4 User Feedback and Dynamic Optimization

User feedback is incorporated to adjust cost and quality priorities. A candidate fractal pattern is first generated and classified. Design satisfaction is then assessed by the user. Criterion weights are updated for later iterations. This process is illustrated in Figure 1.

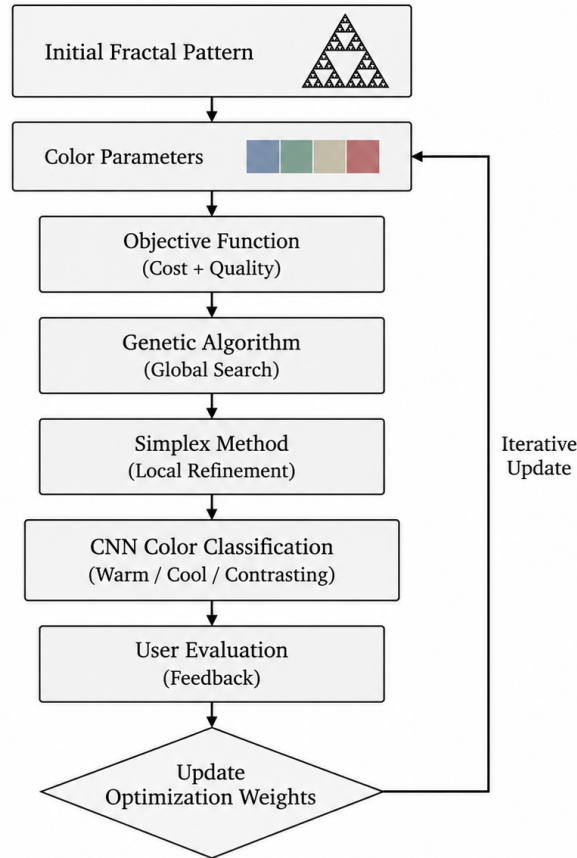


Figure 1: Iterative optimization with user feedback for fractal color selection.

The objective at iteration t is expressed as:

$$J_t(\mathbf{x}) = w_{c,t} C(\mathbf{x}) + w_{q,t} [1 - Q(\mathbf{x})], \quad (8)$$

Weights at iteration t are denoted by $w_{c,t}$ and $w_{q,t}$. The following condition is imposed:

$$w_{c,t} + w_{q,t} = 1. \quad (2)$$

User satisfaction is denoted by $s_t \in [0, 1]$. Greater satisfaction is represented by larger values. The update rule below is applied:

$$w_{c,t+1} = \text{clip} [w_{c,t} + r(s^* - s_t), 0, 1], \quad (9)$$

The update rate is denoted by r . Target satisfaction is denoted by s^* . The updated value is restricted to $[0, 1]$ by $\text{clip}(\cdot)$. The quality weight is then calculated as:

$$w_{q,t+1} = 1 - w_{c,t+1}. \quad (10)$$

Below target satisfaction, the cost weight is reduced. The quality weight is increased. Above target satisfaction, greater emphasis can be placed on cost reduction. Optimization priorities are therefore adapted to user evaluations.

2.5 Hybrid Optimization Procedure

The Genetic Algorithm is selected for its population-based global search [27]. The Simplex Method is selected for derivative-free local refinement [18]. Both methods are combined within the proposed optimization procedure. An initial population of color vectors is generated by the Genetic Algorithm. Each candidate is assessed with the penalized objective in Equation (7). New solutions are produced through selection, crossover, and mutation. Constraint violations are penalized by the penalty function. The parameter space is then explored by the Genetic Algorithm. A suitable candidate is selected and refined by the Simplex Method. The optimized fractal pattern is then generated and classified. User feedback is incorporated, and the objective weights are updated. The process is repeated until the termination criteria are satisfied. Cost and quality are combined through a weighted objective. Alternatively, separate objectives can be defined through NSGA-II [30].

2.6 CNN-Based Color Classification

A Convolutional Neural Network (CNN) is used to classify generated fractal images. Such networks are widely used for image classification and visual recognition [28, 29]. Color composition data are supplied to the optimization process. The classification scheme is based on Itten and Ostwald color theories [24, 25]. Patterns are assigned to warm, cool, or high-contrast groups. Basic color properties are represented by these groups.

2.6.1 Dataset Preparation

Fractal image fragments are labeled by their dominant color category. Representative regions are selected, labeled, and organized for CNN evaluation. Itten's color wheel is used as a reference in Figure 2.

The dataset should be divided into training, validation, and test subsets. Sample counts, class distributions, and split procedures should be reported. These details should be included when experimental records are available.



Figure 2: Itten's color wheel as a reference for color classification.

2.6.2 CNN Architecture and Training

Several CNN depths are evaluated for color classification. Initial depth increases are associated with improved classification performance. Further layers are associated with lower performance. A two-layer CNN is therefore selected for the final implementation.

The model is trained for a maximum of 200 cycles. The relation between cycle count and classification accuracy is examined. Limited gains are reported after additional cycles. The 200-cycle limit is selected to balance accuracy and computational cost. Generated fractal patterns are assigned to predefined color categories. For reproducibility, all model and evaluation settings should be specified. Filters, activations, optimizer, learning rate, and dataset size should be specified. The evaluation protocol should also be reported.

3 Results

The proposed software system is evaluated through computational experiments. Fractal generation, color optimization, hybrid search, and CNN classification are examined. Fractals from Pascal's triangle are used as primary test structures. Optimization performance and CNN classification accuracy are assessed. Fractal patterns are obtained through a modular transformation of Pascal's triangle. A modulus value of 5 is used in the experiments. Distinct and complex patterns are produced for color assignment and optimization.

The software interface is used for fractal generation and color optimization. Generation and color parameters are modified through the interface. The produced patterns are supplied for subsequent analysis.

Recursive spatial organization is obtained from Pascal's triangle with $\text{Mod} = 5$. This structure is used to produce complex patterns with different colors.

The same mathematical construction is used to produce several fractal configurations.

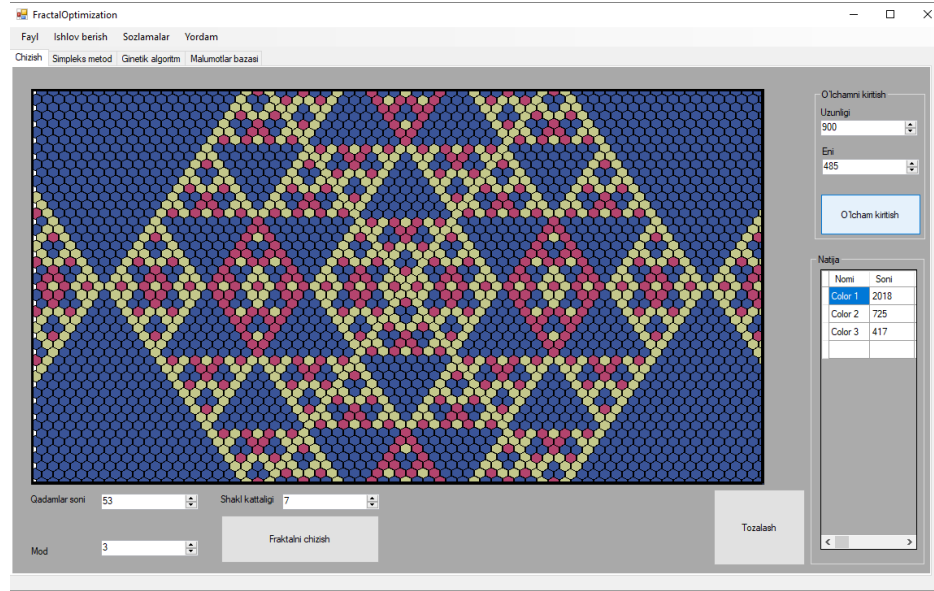


Figure 3: Software interface with a generated fractal and its optimization results.

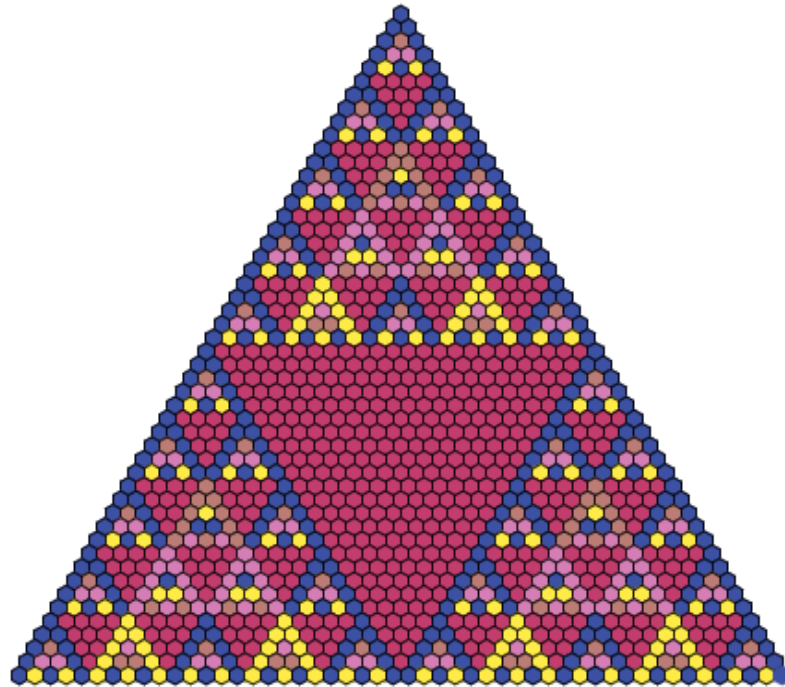


Figure 4: Fractal pattern from Pascal's triangle with $\text{Mod} = 5$.

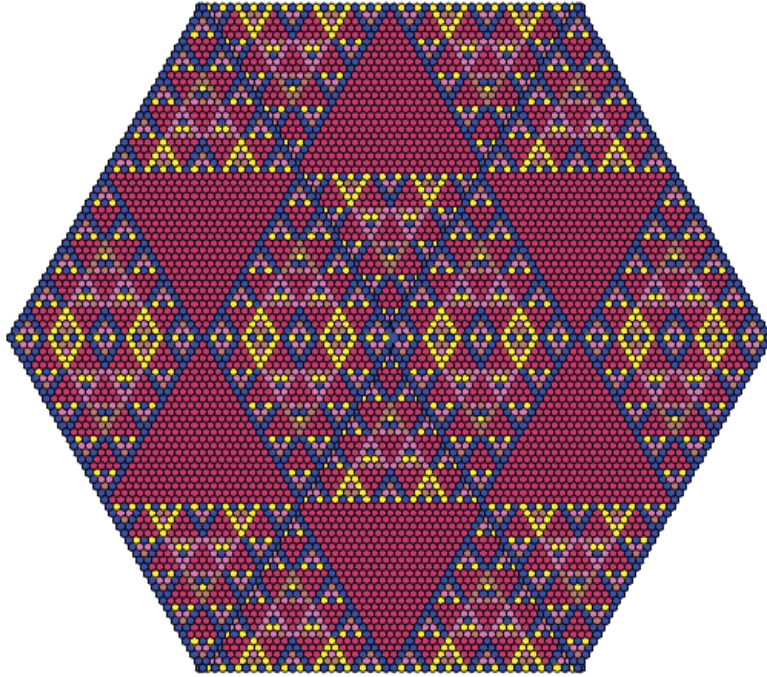


Figure 5: Complex fractal structures from Pascal's triangle with $\text{Mod} = 5$.

These patterns are used as inputs for color generation and optimization. The optimization procedure is applied to each fractal's color parameters. Dye costs and selected color proportions are included in the experiments. Global exploration is performed by the Genetic Algorithm. Local refinement is performed by the Simplex Method. Multiple candidates are produced by the Genetic Algorithm. A selected solution is refined toward a local optimum.

Table 1: Comparative performance of the Simplex Method and Genetic Algorithm

Parameter	Simplex Method	Genetic Algorithm
Execution time (s)	1.5	3.2
Number of iterations	30	50
Reported accuracy	99%	95%
Number of generated results	1	100

Different computational characteristics are reported for both methods. For the Simplex Method, 1.5 s and 30 iterations are reported. For the Genetic Algorithm, 3.2 s and 50 iterations are reported. One refined solution is produced by the Simplex Method. A population of 100 candidates is generated by the Genetic Algorithm. Broader exploration is provided by the Genetic Algorithm. Local refinement is provided by the Simplex Method. Optimized design parameters are presented in Table 2.

Generation steps are reduced from 100 to 80. Figure size is reduced from 500 to 450. Color count is reduced from five to four. A more compact configuration is therefore identified. Target fractal structures are still retained. Production indicators are compared before and after software implementation.

Labor costs are reduced by approximately 10–15%. This reduction is reported after software implementation. Manual intervention may therefore be reduced during design. The

Table 2: Comparison of initial and optimized design parameters

Parameter	Initial value	Optimized value
Number of steps	100	80
Figure size	500	450
Number of colors	5	4

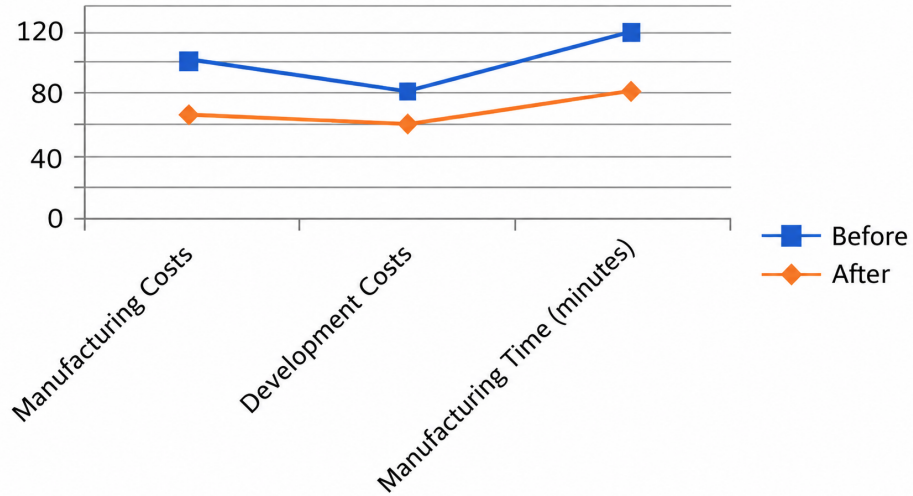


Figure 6: Production indicators before and after software implementation.

result should be interpreted under the reported experimental conditions. The comparison baseline should also be considered. CNN classification is evaluated in the second experiment. Warm, cool, and high-contrast classes are distinguished. Loss and accuracy are measured across training epochs.

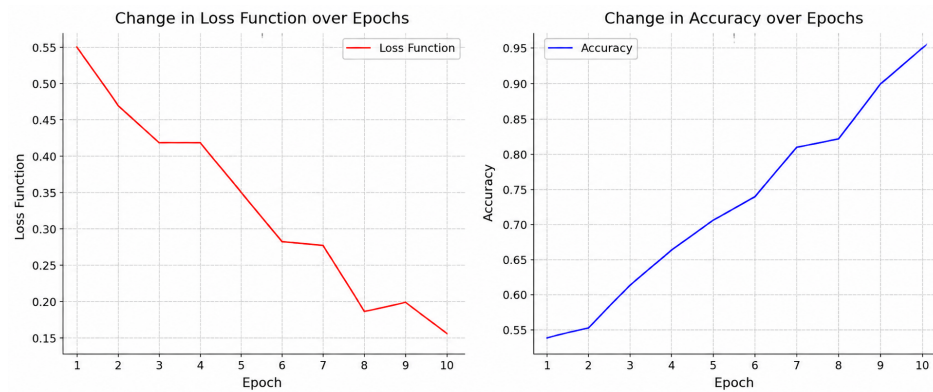


Figure 7: CNN loss and accuracy across training epochs.

Classification loss is reduced during training. Classification accuracy is increased. Visual class features are therefore learned by the CNN. A limit of 200 epochs is applied. The stated experimental configuration is followed. Classification performance is also evaluated through ROC analysis.

An area under the curve (AUC) of 1.00 is reported. Perfect separation is indicated for the evaluated dataset and procedure. However, this value can be obtained from training-data

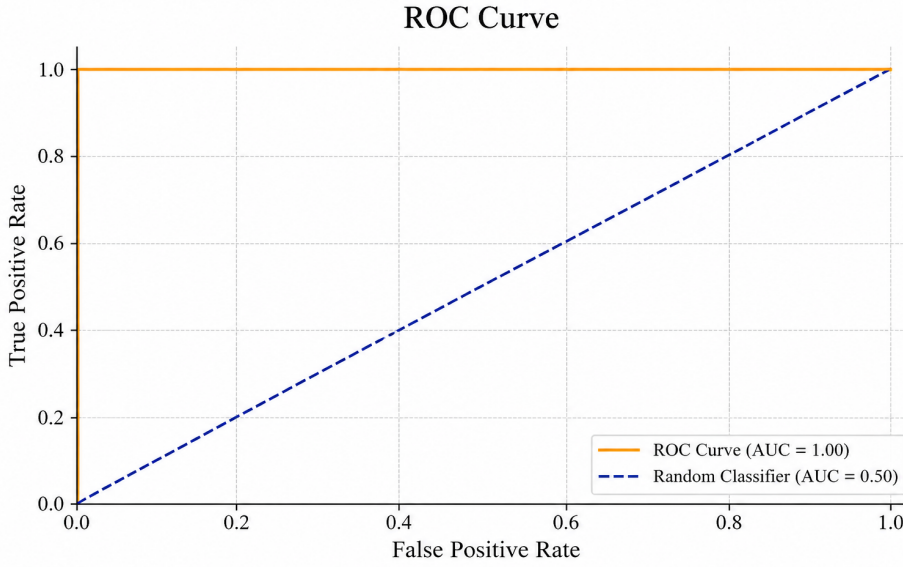


Figure 8: ROC curve for the CNN-based color classification model.

evaluation. It can also be obtained from datasets with limited complexity. Dataset splits and independent test procedures should therefore be reported. Class-level results are also examined through a confusion matrix.

Correct and incorrect predictions are presented for each color category. Correct predictions are represented along the diagonal. Errors are represented by off-diagonal elements. High agreement is reported between predicted and assigned categories. The trained CNN is applied to representative generated fractals. Two classification examples are presented below.

Table 3: CNN classification results for representative fractal patterns

Input	Predicted class	Reported contrast	Example
Fractal color combination	Cool colors	28%	Pascal's-triangle-based fractal pattern
Fractal color combination	Warm colors	31%	Pascal's-triangle-based fractal pattern

Classifier application is demonstrated by these examples. Patterns are assigned to predefined color categories. Associated classification outputs are provided. These outputs are integrated with color optimization. Three principal outcomes are obtained. First, diverse Pascal-based patterns are produced as design candidates. Second, global exploration and local refinement are combined for color optimization. Third, generated combinations are classified into predefined color groups. These results are obtained under the reported experimental conditions. Automated fractal design and color selection are therefore supported.

4 Discussion

Fractal generation, hybrid optimization, feedback, and CNN classification are combined. Automated color selection for carpet design is supported. Cost and visual properties are considered during color exploration. The method is based on earlier fractal color studies [9, 13, 23]. Global search is performed by the Genetic Algorithm. Local refinement is performed by the

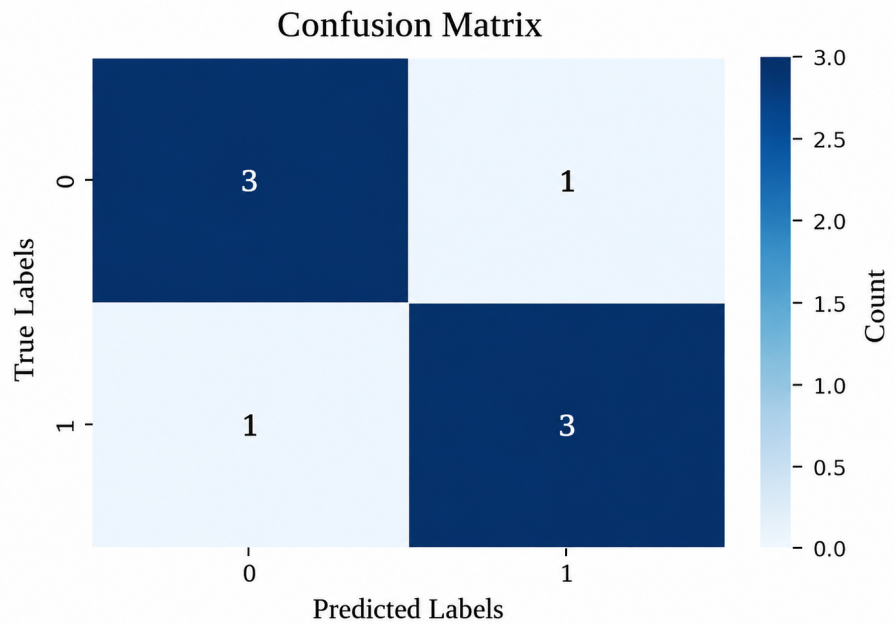


Figure 9: Confusion matrix of the CNN-based color classification model.

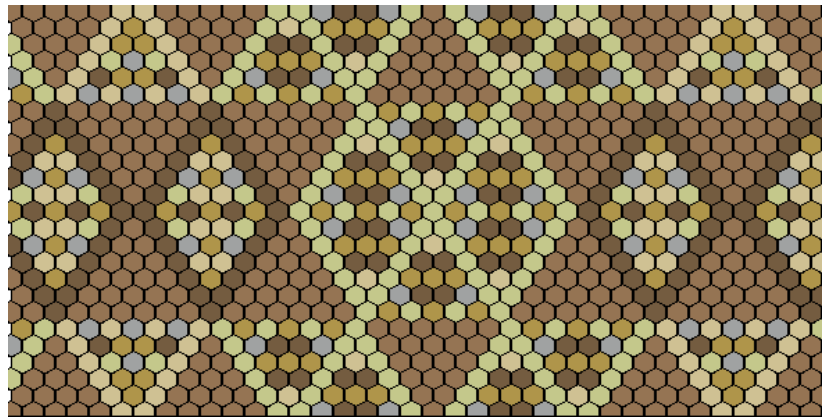


Figure 10: Pascal's-triangle fractal classified as a cool-color combination.

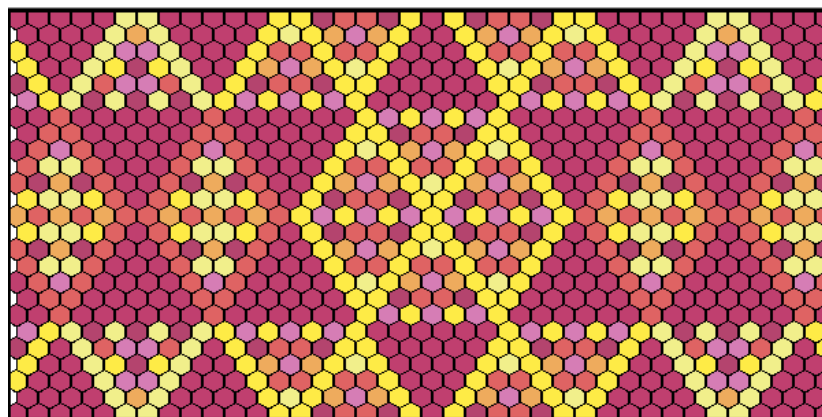


Figure 11: Pascal's-triangle fractal classified as a warm-color combination.

Simplex Method. Multiple candidates are generated by the Genetic Algorithm. A selected solution is refined by the Simplex Method. This distinction is established between evolutionary and derivative-free methods [18, 27]. Adaptation is provided through user feedback. Cost and quality weights are modified after user evaluation. Revised design preferences are therefore represented during optimization. User involvement is retained during final selection. Generated patterns are assigned to warm, cool, and high-contrast groups. CNNs are widely used for image classification and visual recognition [28, 29]. Strong classification performance is reported under the experimental conditions. However, the AUC of 1.00 should be interpreted cautiously. Dataset size, class distribution, and test procedures should be reported. Classifier generalization is determined by these details. Labor costs are reduced by approximately 10–15%. Potential practical benefits are indicated by this reduction. However, the result should be interpreted within the reported conditions. Broader industrial validation is required for other production environments. More complete quality measures can be included. Texture, fractal complexity, color harmony, and perception can be evaluated. These extensions can be supported by established color systems [24, 25, 26]. Cost and quality can also be treated as separate objectives. NSGA-II can be used for this purpose [30].

5 Conclusions

A computational method is presented for fractal color generation and optimization. Digital carpet and textile applications are considered. Production cost and visual quality are balanced. Global exploration is performed by the Genetic Algorithm. Local refinement is performed by the Simplex Method. Multiple candidates are generated and locally refined. CNN classification is applied to predefined color groups. Generated color combinations are therefore analyzed automatically. User feedback is integrated with optimization and classification. Practical value is indicated by the reported labor cost reduction. However, larger datasets and independent tests are required. Real production conditions should also be examined. Improved quality functions and perceptual features should be evaluated. Multi-objective methods should also be tested for cost and aesthetic optimization.

Aknowledgements

The authors acknowledge that AI tools were used for language editing and proofreading. PRISM OpenAI was used for this purpose. In addition, one author is a member of the journal editorial board. The authors acknowledge the support from project 2020-1.1.2-PIACI-KFI-2020-00129. This work was also supported by the National Key Research and Development Program of China (2013YFB3406801).

References

- [1] B. B. Mandelbrot, *The Fractal Geometry of Nature*. San Francisco, CA, USA: W. H. Freeman, 1982.
- [2] A. Husain, M. N. Nanda, M. S. Chowdary, and M. Sajid, “Fractals: An eclectic survey, Part-I,” *Fractal Fract.*, vol. 6, no. 2, Art. no. 89, 2022, doi: 10.3390/fractalfract6020089.
- [3] R. Trombin, *Working with Fractals: A Resource for Practitioners of Biophilic Design*. New York, NY, USA: Terrapin Bright Green, 2020.
- [4] B. A. Bondarenko, “Generalized triangles and Pascal pyramids, their fractals, graphs and applications,” *Discrete Math. Appl.*, vol. 3, no. 3, pp. 159–160, 1991.
- [5] F. M. Nuraliev, Sh. A. Anarova, and O. M. Narzulloev, “Mathematical and software of fractal structures from combinatorial numbers,” in *Proc. Int. Conf. Inf. Sci. Commun. Technol. (ICISCT)*, Tashkent, Uzbekistan, 2019, pp. 10–14.

- [6] Y. Wu, "Fractal-based re-design," Ph.D. dissertation, Cardiff Univ., Cardiff, U.K., 2006.
- [7] R. Zhang, P. Ji, J. Wang, and B. Zhang, "Creative design and visual effect simulation of IoT dynamic interfaces by fractal algorithms and artificial intelligence," *IEEE Trans. Consum. Electron.*, early access, 2025.
- [8] I. M. Rian, "Fractal-based porous concrete components design and 3D printing," in *The Routledge Companion to Smart Design Thinking in Architecture & Urbanism for a Sustainable, Living Planet*. London, U.K.: Routledge, 2024, pp. 376–387.
- [9] F. Nuraliev, O. Narzulloev, and S. Tastanova, "Optimization of fractal structure pattern colors in carpet design using genetic algorithm," in *Intelligent Human Computer Interaction (IHCI)*. Cham, Switzerland: Springer, 2023, pp. 575–585.
- [10] P. Wang, E. Fan, and P. Wang, "Comparative analysis of image classification algorithms based on traditional machine learning and deep learning," *Pattern Recognit. Lett.*, vol. 141, pp. 61–67, 2021, doi: 10.1016/j.patrec.2020.07.042.
- [11] M. Ivanovici, "Fractal dimension of color fractal images with correlated color components," *IEEE Trans. Image Process.*, vol. 29, pp. 8069–8082, 2020.
- [12] J. Song, B. Wang, Q. Jiang, and X. Hao, "Exploring the role of fractal geometry in engineering image processing based on similarity and symmetry: A review," *Symmetry*, vol. 16, no. 12, Art. no. 1658, 2024, doi: 10.3390/sym16121658.
- [13] F. Nuraliev, I. S. Nabiyev, and S. Tastanova, "Automated system for visualization and optimization of color parameters of fractal images using artificial intelligence," in *Proc. Int. Conf. Inf. Sci. Commun. Technol. (ICISCT)*, Seoul, South Korea, 2024, pp. 93–97.
- [14] M. Abdel Halim, G. Ibrahim, and F. Abdel Tawab, "Utilizing artificial intelligence technical to develop some textile craft industries," *Int. Des. J.*, vol. 14, no. 5, pp. 43–63, 2024.
- [15] N. A. Salingeros, "Environments that boost creativity: AI-generated living geometry," *Multimodal Technol. Interact.*, vol. 9, no. 5, Art. no. 38, 2025, doi: 10.3390/mti9050038.
- [16] J. Li, "Innovative use of visual communication style in product design based on fractal geometry," *J. Combin. Math. Combin. Comput.*, vol. 127, pp. 2465–2490, 2025.
- [17] A. S. Buriboev, D. Sultanov, Z. Ibrohimova, and H. S. Jeon, "Mathematical modeling and recursive algorithms for constructing complex fractal patterns," *Mathematics*, vol. 13, no. 4, Art. no. 646, 2025, doi: 10.3390/math13040646.
- [18] J. A. Nelder and R. Mead, "A simplex method for function minimization," *Comput. J.*, vol. 7, no. 4, pp. 308–313, 1965, doi: 10.1093/comjnl/7.4.308.
- [19] I. L. Akulich, *Mathematical Programming in Examples and Problems*, 3rd ed. Saint Petersburg, Russia: Lan', 2011, pp. 277–312 (in Russian).
- [20] Yu. A. Zak, *Applied Problems of Multiobjective Optimization*. Moscow, Russia: Ekonomika, 2014 (in Russian).
- [21] S. Tastanova and F. Nuraliev, "Automated system for optimization of color parameters of fractal images using artificial intelligence," in *Actual Problems of Mathematical Modeling, Algorithmization, and Programming*, Tashkent, Uzbekistan, 2023, pp. 124–130.
- [22] J. W. Cannon, "Review of *The Fractal Geometry of Nature*, by Benoit B. Mandelbrot," *Amer. Math. Monthly*, vol. 91, no. 9, pp. 594–598, 1984.
- [23] F. Nuraliev, S. Tastanova, and T. Bulanova, "The use of methods for optimizing the genetic algorithm for the colors of a fractal structure pattern in carpet design," *Farabi J. Soc. Sci.*, vol. 8, no. 2, 2022.
- [24] J. Itten, *The Elements of Color*. New York, NY, USA: Van Nostrand Reinhold, 1970.
- [25] C. E. Foss, D. Nickerson, and W. C. Granville, "Analysis of the Ostwald color system," *J. Opt. Soc. Amer.*, vol. 34, no. 7, pp. 361–381, 1944, doi: 10.1364/JOSA.34.000361.
- [26] *Colorimetry — Part 4: CIE 1976 L*a*b* Colour Space*, ISO/CIE 11664-4:2019, CIE, Vienna, Austria, 2019.
- [27] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI, USA: Univ. Michigan Press, 1975.
- [28] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 25, 2012, pp. 1097–1105.
- [29] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.

- [30] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002, doi: 10.1109/4235.996017.

Saida Tastanova*
Television-Media Technologies, TUIT,
100084, Tashkent, Uzbekistan.
Email: a.tastanova@tuit.uz

Amir Mosavi
Obuda University, Hungary.
J. Selye University, Slovakia.
Karaganda Technical University, Kazakhstan.
Ludovika University of Public Service, Hungary.

Marat Tastanov
IT Center under NASP,
Tashkent, Uzbekistan.

Mark Csoka
J. Selye University, Slovakia.

Jumamurod Aralov
Television-Media Technologies, TUIT,
100084, Tashkent, Uzbekistan.

Laszlo Gocs
John von Neumann University,
Kecskemet, Hungary.

Received 02.01.2026, Accepted 11.04.2026, Available online 30.09.2026.