Computational Visual Media
https://doi.org/10.1007/s41095-023-0359-z Vol. 10, No. 3, June 2024, 559–576

Research Article

Symmetrization of quasi-regular patterns with periodic tilting of regular polygons

Zhengzheng Yin[1], Yao Jin[1,2] (✉), Zhijian Fang[1], Yun Zhang[3], Huaxiong Zhang[1,2] (✉), Jiu Zhou[1] and Lili He[1,2]

Abstract Computer-generated aesthetic patterns are widely used as design materials in various fields. The most common methods use fractals or dynamical systems as basic tools to create various patterns. To enhance aesthetics and controllability, some researchers have introduced symmetric layouts along with these tools. One popular strategy employs dynamical systems compatible with symmetries that construct functions with the desired symmetries. However, these are typically confined to simple planar symmetries. The other generates symmetrical patterns under the constraints of tilings. Although it is slightly more flexible, it is restricted to small ranges of tilings and lacks textural variations. Thus, we proposed a new approach for generating aesthetic patterns by symmetrizing quasi-regular patterns using general k-uniform tilings. We adopted a unified strategy to construct invariant mappings for k-uniform tilings that can eliminate texture seams across the tiling edges. Furthermore, we constructed three types of symmetries associated with the patterns: dihedral, rotational, and reflection symmetries. The proposed method can be easily implemented using GPU shaders and is highly efficient and suitable for complicated tiling with regular polygons. Experiments demonstrated the advantages of our method over state-of-the-art methods in terms of flexibility in controlling the generation of patterns with various parameters as well as the diversity of textures and styles.

Keywords quasi-regular patterns (QRP); k-uniform tilings; invariant mappings; symmetry; aesthetic patterns

1 Introduction

A pattern is an arrangement of motifs that is repeated in a consistent manner. It is used as a basic material with wide applications in the design of various products, such as fabrics, neckties, jewelry, carpets, and wallpapers. The traditional pattern creation method involves manual drawings, which generally require a long time to shape a single pattern, even with the assistance of software such as Adobe Photoshop and Illustrator, etc. These patterns can be generated automatically and efficiently using various mathematical methods [1, 2]. Among these methods, shape grammar [3], fractals [4, 5], dynamical systems [6, 7], and their variants are the most commonly used. Although they can help create splendidly aesthetic patterns, they always suffer from uncontrollable problems; that is, it is impossible to establish a relationship between the parameters of the mathematical models and their generated patterns, which prevents artists from wide applications. Quasi-regular pattern (QRP) is another type of digital pattern generated by visualizing the smoothed form of the Hamiltonian function [8]. These patterns are typically characterized by translational and rotational symmetries with near-regular textures [9–11]. Although the parameters of the mathematical model have a specific geometrical significance, it is

1 School of Computer Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China. E-mail: Z. Yin, 202020503044@mails.zstu.edu.cn; Y. Jin, jinyao@zstu.edu.cn (✉); Z. Fang, hptnthptnt@zstu.edu.cn; H. Zhang, zhxhz@zstu.edu.cn (✉); J. Zhou, zhoujiu34@126.com; L. He, llhellhe@zju.edu.cn.

2 Zhejiang Provincial Innovation Center of Advanced Textile Technology, Shaoxing 312000, China.

3 School of Media Engineering, Communication University of Zhejiang, Hangzhou 310018, China, E-mail: zhangyun@cuz.edu.cn.

Manuscript received: 2023-02-28; accepted: 2023-06-04

still difficult to predict the shape and texture of a pattern according to any given QRP model.

By contrast, it is much easier to control the spatial layout of the pattern than the texture details. Therefore, to enhance aesthetics and controllability, patterns are typically created with symmetric layouts. One method to generate such patterns is to construct invariant functions with symmetries from the dynamics [2, 12, 13]. The designed function determines the layout structure, which is typically restricted to certain simple planar symmetry types. Another method to generate patterns is to use dynamical systems or fractals under the constraint of predefined tilings [14, 15]. Although it is more flexible than those in previous works, it is usually confined to a small range of tiling structures, for example, Penrose tilings [14] or Archimedes tilings [15]. In addition, constructing continuous conditions across the tiling edges reduces texture richness, making the pattern visually monotonous.

In this paper, we proposed a new pattern generation method that symmetrizes quasi-regular patterns with the constraints of general k-uniform tilings. First, we reconstructed the k-uniform tiling using the integer representation method [16]. We then rearranged the basic elements of the tilings to ensure texture continuity across the tiling edges. Next, we constructed three types of invariant mappings with continuity conditions at the boundaries of the fundamental regions. Finally, we colored the image space using QRP models with the help of invariant mapping. Our method can flexibly control the pattern from various aspects, and the generated patterns are rich in texture details and visually pleasing. Overall, the contributions of our method can be summarized as follows:

- We proposed a new method for generating aesthetic patterns by integrating the tiling structures with the quasi-regular pattern models, which can flexibly control both the textures and layout of the pattern.
- We presented a unified scheme of constructing fundamental regions and invariant mappings for any k-uniform tiling, which is capable of batch generation of numerous patterns.
- We constructed three kinds of invariant mappings, which ensures natural transition both on the boundaries of regular polygons and fundamental regions.

2 Related work

Extensive research has been conducted on the generation of aesthetic patterns. Gieseke et al. [17] recently surveyed the control mechanisms of pattern generation. Herein, we focused only on the most closely related ones.

2.1 Pattern generation models

Popular mathematical models for generating aesthetic patterns include shape grammar, fractal geometry, dynamical systems, and quasi-regular patterns. Shape grammar applies shape rules, such as the addition and subtraction of shapes, and various transformations to construct geometric patterns [18], such as Islamic geometric patterns [3]. Pattern shapes can be controlled by grammar, which requires creative thinking and ideas. Research on fractal geometry [4, 19] and dynamical systems [6] has a long history, remains active, and produces several commonly used methods, such as the escape time and orbit trap methods. They can easily and efficiently create various styles with beautiful patterns. However, it is not intuitive for the user to control the pattern simply by adjusting the parameters of the mathematical models.

The quasi-regular pattern model generates colorful patterns by visualizing a special type of smooth function [9–11]. It has several geometrically significant parameters, and the generated patterns typically have local symmetries and near-regular structures. These patterns are widely used in textile and garment designs [20, 21]. It has better controllability than the other models but is still difficult for arbitrary QRP models.

2.2 Patterns based on symmetric functions

To control the spatial layout of the pattern and enhance aesthetics, researchers have resorted to constructing functions with symmetrical properties that are used in dynamical systems. Chung and Chan [12] constructed functions with all the wallpaper symmetries using a Fourier series. Carter et al. [22] extended this method to construct equivariant functions with respect to more symmetries, such as frieze and crystallographic groups. Zou et al. [23] combined chaotic functions with cyclic and dihedral symmetries by using the orbit trap method to render aesthetic patterns. Lu et al. [13] generated patterns by constructing trigonometric

and polynomial functions with specific wallpaper symmetry. Gdawiec [24] applied fixed-point theory to create symmetric functions. Recently, they proposed a modified orbit trap method [2], which can obtain various interesting patterns. In addition, non-Euclidean geometries are widely used to generate patterns with distinctive styles, such as spherical [25] and hyperbolic geometries [26]. However, all these methods require the analysis and design of specific functions with certain symmetries and are generally confined to a small range of symmetry types.

2.3 Patterns constrained by tilings

Another method to control the spatial layout of a pattern is using predefined tilings. The key to this method is to construct invariant mappings for the tiling to maintain texture continuity on the boundaries of polygons. One class of methods uses dynamical systems based on several types of tiling, such as chair tiling [27], Penros tiling [14], spiral tiling [28], and Archimedean tiling [15]. Another class uses complicated fractal models with a tiling structure, which possesses a self-similarity property [29–31]. These methods can flexibly control the pattern layout. However, all of them suffer from at least two problems: (1) invariant mappings should be designed separately for each tiling, and (2) the texture of the pattern is monotonous near the boundaries of the fundamental regions because of simple continuous conditions. Our method integrates general k-uniform tilings with quasi-regular patterns and devises a simple and effective scheme to eliminate texture seams across the boundaries, which could overcome the abovementioned problems.

3 Preliminaries

To make the paper self-consistent, we introduced some preliminaries on generating quasi-regular patterns and k-uniform tilings in this section.

3.1 Generation of quasi-regular patterns

A quasi-regular pattern is obtained by visualizing a specific type of function whose geometry is a smooth 2-manifold surface, specifically a smooth height field defined in R^2. Its basic mathematical model is derived from the weak chaos theory [8] and is expressed by Eq. (1):

$$H_q(x,y)=\sum_{i=1}^{\lfloor q\rfloor}\cos\left(x\cos\left(\frac{2\pi i}{q}\right)+y\sin\left(\frac{2\pi i}{q}\right)\right) \tag{1}$$

where q denotes the number of resonances, $(x,y)\in R^2$ are the coordinates, and H_q is named as "QRP model" here. Please refer to Ref. [8] for details on this deduction. A visualization of the surface defined by Eq. (1) is presented in Fig. 1(a).

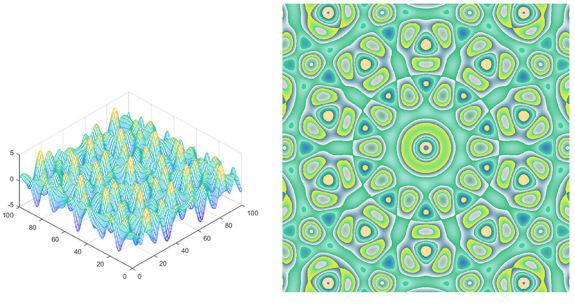
Noting that the contours of the QRP model H_q constitute a series of closed curves of various shapes, Zhang and Li [9] proposed a method for visualizing the contours of surface H_q. They partitioned the height field of QRP model H_q into several disjoint intervals $h_1,h_2,...,h_n(\cup_{i=1}^n h_i=H_q)$, and the connected region corresponding to each interval was assigned the same predefined color value. The generated colorful image is called a quasi-regular pattern (see the example in Fig. 1(b)). To elaborate on the details, each pixel (n_x,n_y) in the image space was transformed into a normalized space $[x_t,y_t]\times[x_t+s\pi,y_t+s\pi]$.

$$(x,y)=\left(x_t+n_x\frac{s\pi}{W_x},y_t+n_y\frac{s\pi}{W_y}\right) \tag{2}$$

where W_x and W_y denote the width and height of the canvas, respectively, (x_t,y_t) is the vector for the translation, and s is used to control the scale of the normalized space. Subsequently, the color value for the pixel (n_x,n_y) in the connected region corresponds to its partitioned interval. The details of the algorithm are presented in Algorithm 1.

3.2 Reconstruction of k-uniform tilings

A k-uniform tiling composed of regular polygons is an edge-to-edge tiling whose basic elements (tiles) consist of five types of regular polygons: triangular, square, hexagonal, octagonal, and dodecagon. It has



(a) Surface of the QRP model (b) Colored pattern of the QRP model

Fig. 1 Quasi-regular pattern of basic model with parameters $q=5, s=18, x_t=y_t=0$.

Algorithm 1 Algorithm for generating quasi-regular patterns

Input: Canvas width: W_x, height: W_y, parameters q, s, x_t, y_t
Output: Quasi-regular pattern
for all pixels in canvas (n_x, n_y) **do**
  Calculate (x, y) with Eq. (2) ;
  Calculate H_q with Eq. (1);
  Set the color for the pixel (n_x, n_y) according the partition of the height field H_q;
end for

k distinct transitivity classes of vertices and, therefore, has k equivalence classes of vertices with respect to their symmetries. Figure 2 shows 15 vertex types that appear in the k-uniform tilings. In this study, a tiling is named according to the number of vertex types it contains, where each type corresponds to a capital letter, as shown in Fig. 2. A k-uniform tiling may contain several different vertex types and thus can be named as a combination of those letters. Specifically, when $k = 1$, it degenerates into a special class of tiling known as Archimedean tiling [15].

Although Archimedean tilings can be represented simply by vertex types, the representation of a general k-uniform tiling is relatively complicated. Medeiros e Sá et al. [32] presented a simple representation for periodic tilings of the plane by regular polygons. This approach explicitly represents a minimal subset of vertices from which all vertices of the tiling are systematically generated by translations. After that, Soto Sánchez et al. [16] improved the method and

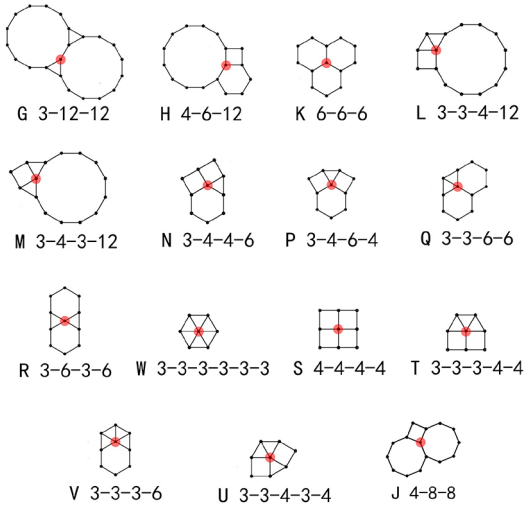


Fig. 2 15 different vertex types that appear in k-uniform tilings with regular polygons.

proposed an integer representation for any k-uniform tiling.

According to their definition, the coordinates of the tiling vertices are represented as complex numbers under the basis $\{1, \omega, \omega^2, \omega^3\}$: $a_0+a_1\omega+a_2\omega^2+a_3\omega^3$, where $[a_0, a_1, a_2, a_3]$ are four integers representing the lattice coordinates, and ω^i is one of the principal 12th roots of unity. Thus, k-uniform tiling can be represented concretely by a $(2 + n) \times 4$ integer matrix containing the lattice coordinates for the two translation vectors (the first two rows) and n seed vertices (remaining rows). See Ref. [16] for further details.

Given an integer representation of k-uniform tilings, the calculation of k-uniform tilings composed of regular polygons is straightforward and can be summarized in the following steps:

- *Reconstruct the translation grid*: reconstruct lattice coordinates for all the vertices of a representative translation cell by using the seed coordinates.
- *Convert lattice coordinates to Cartesian coordinates*: The Cartesian coordinates of the tiling's vertices for rendering can be obtained from the corresponding lattice coordinates, which are calculated as Eq. (3):

$$(x, y) = \begin{pmatrix} a_0 & a_1 & a_2 & a_3 \end{pmatrix} \begin{bmatrix} 1 & 0 \\ \frac{\sqrt{3}}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{\sqrt{3}}{2} \\ 0 & 1 \end{bmatrix} \tag{3}$$

- *Translate the grid to construct the tiling*: Translate the grid to cover the whole plane, where the translation vector can be also calculated by Eq. (3).

4 Symmetrization of quasi-regular patterns

4.1 Overview

We aimed to generate aesthetic patterns from quasi-regular patterns by controlling their spatial layout and, more precisely, symmetrize the patterns under the constraint of any given k-uniform tiling. The key problem is constructing invariant mappings under specific symmetry groups that can maintain the continuity of the rendered pattern across the edges of the tiling and the axes of the local symmetrical

transformations.

In contrast to the existing methods [14, 15], which must tediously design invariant mapping for each tiling, we proposed a unified method that can be applied to all types of k-uniform tilings. We first reconstructed the selected k-uniform tiling using the integer representation method [16]. We then rearranged the basic elements of the tiling for further construction of the fundamental region, which helped transit the texture continuously across the edges of the tiling. Here, the fundamental region is defined as a connected set that covers all basic elements under the action of a certain symmetry group S without overlapping except at their boundaries [15]. Next, we constructed the fundamental region and corresponding invariant mappings under the three types of symmetric groups. Finally, we set the colors in the image space using QRP models with invariant mapping. The details of this algorithm are described in Algorithm 2.

4.2 Rearrangement of basic elements

Suppose that a k-uniform tiling is composed of r ($1 \leqslant r \leqslant 5$) types of basic elements, that is, tiles, each of which is a regular m polygon P_m. We would use the fundamental region U as the element to cover the entire canvas via symmetry transformations.

Therefore, instead of directly constructing a fundamental tiling region [14, 15], we proposed a scheme that rearranges all basic elements to form the target domain, where the fundamental region can be defined. The scheme aims to transform each basic element such that one edge of each basic element overlaps, which helps eliminate texture seams between neighboring polygons.

For the convenience of the later construction of the fundamental region, we first built a mapping to transform each polygon in the tiling to a fixed location. The mapping translates the polygon to the origin of the coordinate system such that the centroid of the polygon coincides with the origin and then rotates it by one of its edges to ensure that it is perpendicular to the x-axis (an illustration is shown in Fig. 3). Therefore, the mapping can be expressed as

$$\begin{pmatrix} p'_x \\ p'_y \end{pmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{pmatrix} p_x - c_x \\ p_y - c_y \end{pmatrix} \tag{4}$$

where (c_x, c_y) is the centroid of the polygon and (p_x, p_y) and (p'_x, p'_y) are the point pairs for the polygon before and after the mapping. The rotation angle θ can be calculated as Eq. (5):

$$\theta = \alpha - \beta, \quad \alpha = \begin{cases} 0, & n = 3 \\ \dfrac{\pi}{n}, & n \neq 3 \end{cases}, \quad \beta = \arccos\left(\frac{V_x}{\sqrt{V_x^2 + V_y^2}}\right) \tag{5}$$

where (V_x, V_y) is the position of any vertex of the polygon that falls into the first and second quadrants after the transformation.

We then rearranged all the basic elements of a tiling to a state in which one side of each element overlapped. By default, the overlapping edges and their perpendicular bisector lines coincide with y- and x-axes, respectively. Note that there are 2^n cases for such an arrangement, where n is the number of basic elements. Each case leads to a different fundamental region U and, thus, produces very different results, which gives us another DOF to control the pattern. To explain the details, we employed a tiling composed of three distinct types of regular polygons as an illustrative example, where the basic elements include

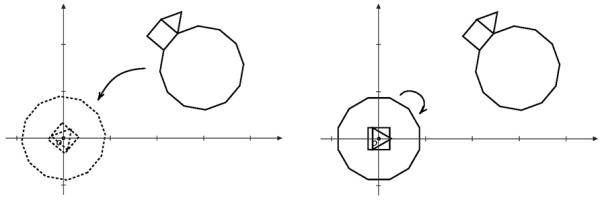
Algorithm 2 Symmetrize QRP with k-uniform tilings

Input:
1. Integer matrix representing a given k-uniform tilt T.
2. The QRP model H and its parameters q, s, x_t, y_t.
3. Canvas width: W_x, height: W_y,

Output: Tiling-constrained quasi-regular pattern

Initialize: Reconstruct the tiling with T by calculating Cartesian coordinates (p_x, p_y) of all polygons P_m with Eq. (3).

for all P_m **do**
  for all $(p_x, p_y) \in P_m$ **do**
    Calculate (p'_x, p'_y) with Eq. (4);
    Calculate $d_{OO'}$ with Eq. (6);
    $p'_x = p'_x + d_{OO'}$;
    Calculate (x, y) by the invariant mapping M which will be introduced in Section 4.3;
    Calculate the corresponding pixel color of (x, y) with the QRP algorithm (Algorithm 1);
  end for
end for



(a) Translation (b) Rotation

Fig. 3 Initialize the location of the basic elements.

a regular triangle, quadrilateral, and dodecagon. It can be verified that 8 different arrangements can be made in total, which can be represented as $3^l4^l12^l$, $3^l4^l12^r$, $3^l4^r12^l$, $3^r4^l12^l$, $3^l4^r12^r$, $3^r4^l12^r$, $3^r4^r12^l$, and $3^r4^r12^r$, where the digits denote the number of sides of the basic elements and the superscript letters l and r indicate whether the corresponding element is placed on the left (l) or right (r). Figure 4 displays two of the 8 cases, that is, $3^l4^r12^r$ and $3^r4^r12^r$. In all cases, each polygon is initially located in the the center of the coordinate system(Fig. 3) along the x axis to obtain a new arrangement, and the displacement is calculated as

$$d_{OO'_m} = (-1)^{l_m} \frac{|e|}{2\tan\left(\frac{\pi}{m}\right)} \quad (m = 3, 4, 6, 8, 12) \quad (6)$$

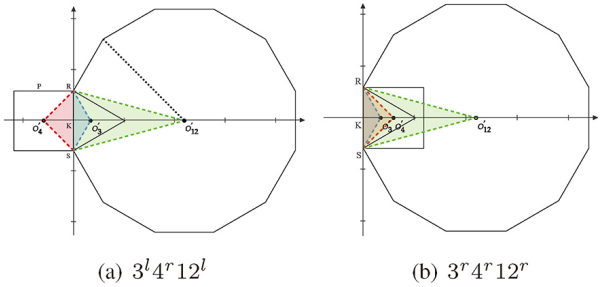
where $|e|$ is the edge length of the polygons and l_m is a Boolean that denotes whether the polygon is on the left ($l_m = 1$) or right ($l_m = 0$).

Finally, we constructed the target domain being used to build the fundamental region by connecting the centroid of each polygon to the two endpoints of the overlapping edge. The defined fundamental region is marked as the colored area in Fig. 4.

4.3 Construction of invariant mappings

We constructed three types of invariant mappings associated with general k-uniform tilings without strange seams to enrich the spatial structure of quasi-regular patterns. The generated patterns have the expected local symmetries whose textures are continuous across the edges of the polygons.

The symmetry of each tiling was isometric. All symmetries of a tiling constitute a symmetry group that can be represented as a set of generators $\{g_1, g_2, ..., g_p\}$ [33]. Let G_m be the symmetry group of P_m; then, the fundamental region U for a k-uniform tiling is the union of the fundamental regions of all its basic elements, that is, $U = \cup_m U_m$. We built invariant mappings M based on fundamental regions U. In general, invariant mapping is a transformation that maintains a specific property of an object after its application. The goal of invariant mapping is to make the mapped and fundamental regions congruent. Formally, let T be a symmetry transformation. Then, for any point (x, y) in the fundamental region, it holds $M(x, y) = M(T(x, y))$.



(a) $3^l4^r12^l$ (b) $3^r4^r12^r$

Fig. 4 Two of the 8 cases for rearranging the basic elements, where O'_m is the centroid of the m polygon P_m, R and S represent the endpoints of their coincident edges, and K denotes the intersection point of this edge with the x-axis.

We took the fundamental region constructed in Fig. 5 as an example to describe how to construct three types of invariant mappings, denoted as M_1, M_2, and M_3. Let R_{AB} be the reflection transformation of the line $\overline{AB}$. To construct a symmetrical group, we used different composite transformations of R_{AB} as the generator.

4.3.1 Invariant mappings with dihedral symmetries

The dihedral symmetry corresponds to the dihedral group D, and each regular polygon P_m contains a dihedral group consisting of m-fold rotations and m-fold reflections. Thus, any k-uniform tiling is naturally equipped with this symmetry.

Without loss of generality, we used the GLM-type tiling illustrated in Fig. 5(a) as an example to construct an invariant mapping M_1 with dihedral symmetry. Let $D_3 = \{\boldsymbol{R}_{O'_3K}, \boldsymbol{R}_{O'_3R}\}$, $D_4 = \{\boldsymbol{R}_{O'_4K}, \boldsymbol{R}_{O'_4R}\}$, and $D_{12} = \{\boldsymbol{R}_{O'_{12}K}, \boldsymbol{R}_{O'_{12}R}\}$ be the dihedral symmetry group of basic elements (polygons) P_3, P_4, and P_{12}, respectively. The fundamental regions corresponding to each polygon were $\triangle O'_3RK, \triangle O'_4RK$, and $\triangle O'_{12}RK$. By definition, for any point in the fundamental region U of the tiling, for example, $A_1 \in \overline{RK}$ marked in Fig. 5(a), we obtained rotationally symmetrical points

$$\begin{cases} A_2 = \boldsymbol{R}_{O'_4R}(A_1) \in \overline{RP} \\ A_3 = \boldsymbol{R}_{O'_3R}(A_1) \in \overline{RM} \\ A_4 = \boldsymbol{R}_{O'_{12}R}(A_1) \in \overline{RQ} \end{cases} \quad (7)$$

as well as its reflectional symmetrical points.

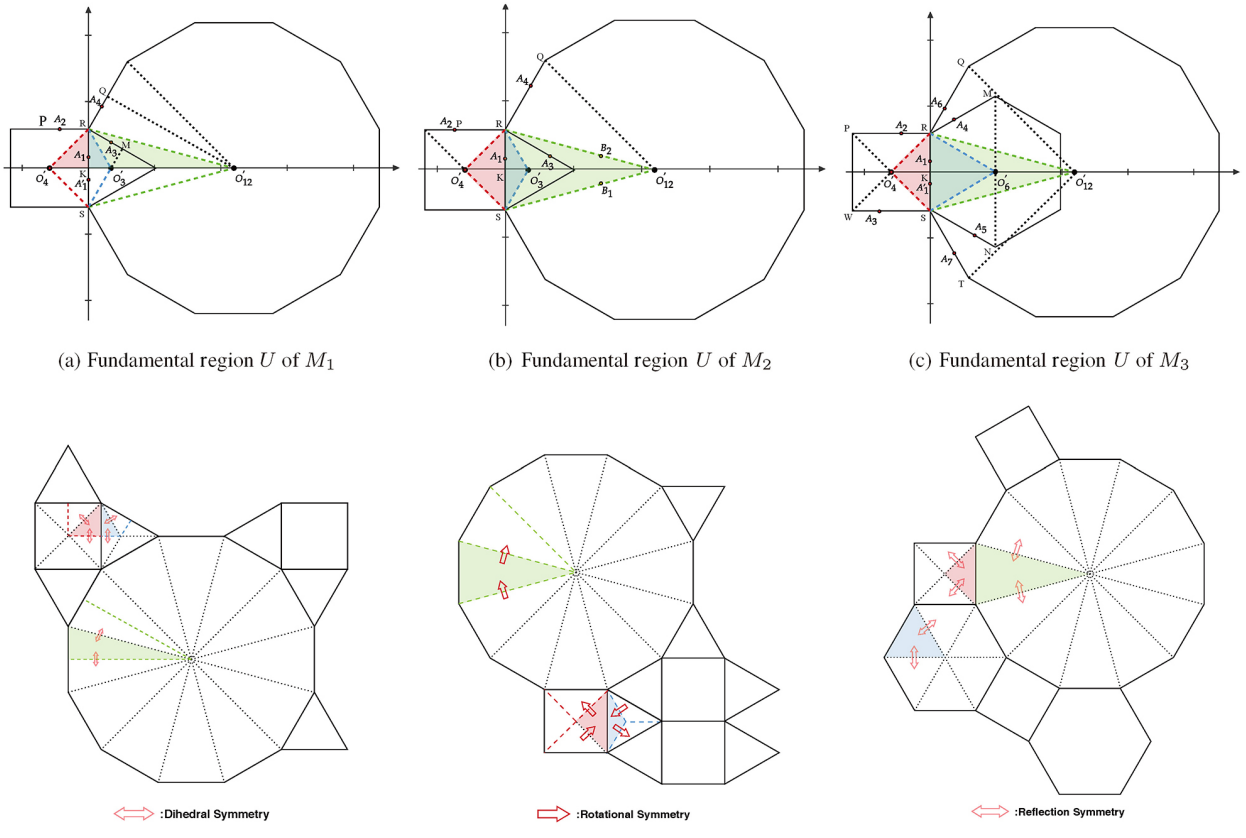
$$A'_1 = \boldsymbol{R}_{O'_4K}(A_1) = \boldsymbol{R}_{O'_3K}(A_1) = \boldsymbol{R}_{O'_{12}K}(A_1) \in \overline{RS} \quad (8)$$

where A_1, A_4, and A'_1 lie on the boundary of fundamental region U.

For any point $(x, y) \in P_m$, we can map it to its fundamental region U_m using the dihedral symmetry group D_m as Eq. (9):

$$(x', y') = \gamma_m(x, y) \in U_m, \quad \text{s.t. } \gamma_m \in D_m \quad (9)$$

Note that all symmetrical points lead to the same coordinates (x', y'). Thus, mapping M_1 defined as



(a) Fundamental region U of M_1 (b) Fundamental region U of M_2 (c) Fundamental region U of M_3

(d) Symmetry for the GLM-type tiling (e) Symmetry for the GLMT_1-type tiling (f) Symmetry for the H-type tiling

Fig. 5 Examples of three kinds of invariant mappings. The colored areas marked in (a), (b), and (c) are the fundamental regions of M_1, M_2, and M_3, respectively. (d), (e), and (f) illustrate the symmetries corresponding to (a), (b), and (c) , respectively.

$M_1(x, y) = H_q(x', y')$ is an invariant mapping, where H_q is the QRP model defined in Eq. (1).

An illustration of the mapping under dihedral symmetry is shown in Fig. 5(d), where the double red arrows represent the symmetric relationship between the fundamental region U and the others. Examples of the generated pattern and its corresponding surface are shown in Fig. 6(a) and Fig. 6(b).

4.3.2 Invariant mappings with rotational symmetries

An n-fold rotational symmetry is a cyclic group C_n, which makes it somewhat complicated to construct invariant mapping compared with the previous case. To further clarify the mapping M_2, we utilize Fig. 5(b) as an illustrative example. To elaborate on mapping M_2, we consider Fig. 5(b) as an example. In this mapping, we must consider the continuity of the seams across both the boundary edges of the polygons and the fundamental region U. In Fig. 5(b), let C_m $(m = 3, 4, 12)$ be the cyclic symmetry of P_m. Then $(\boldsymbol{R}_{O'_3K} \cdot \boldsymbol{R}_{O'_3R})$, $(\boldsymbol{R}_{O'_4K} \cdot \boldsymbol{R}_{O'_4R})$, and $(\boldsymbol{R}_{O'_{12}K} \cdot \boldsymbol{R}_{O'_{12}R})$ are the generators of C_3, C_4, and C_{12}, respectively. All of these are counterclockwise rotations about O_m at an angle $2\pi/m$. To construct M_2, the fundamental regions corresponding to each polygon become $\triangle O'_3RS$, $\triangle O'_4RS$, and $\triangle O'_{12}RS$. Then, we obtained the symmetrical points of $\forall A_1 \in \overline{RS}$ as

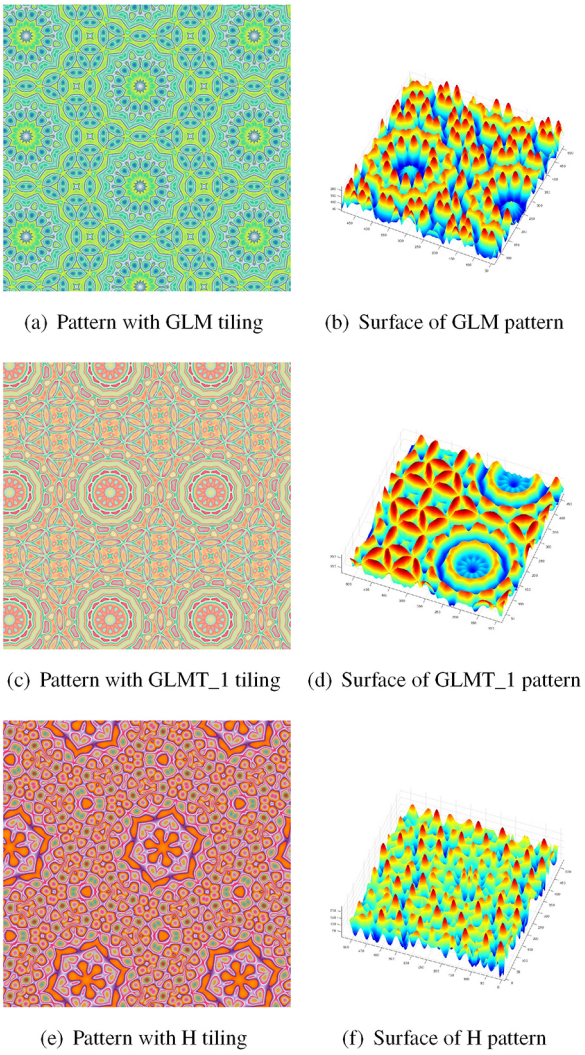
$$\begin{cases} A_2 = (\boldsymbol{R}_{O'_4K} \cdot \boldsymbol{R}_{O'_4R})^{-1} (A_1) \in \overline{RM} \\ A_3 = (\boldsymbol{R}_{O'_3K} \cdot \boldsymbol{R}_{O'_3R})^{-1} (A_1) \in \overline{RP} \\ A_4 = (\boldsymbol{R}_{O'_{12}K} \cdot \boldsymbol{R}_{O'_{12}R})^{-1} (A_1) \in \overline{RQ} \end{cases} \tag{10}$$

and $\forall B_2 \in \overline{O'_{12}R}$ as

$$B_1 = (\boldsymbol{R}_{O'_{12}K} \cdot \boldsymbol{R}_{O'_{12}R}) (B_2) \in \overline{O'_{12}Q} \tag{11}$$

The symmetry of the fundamental regions for other polygons can also be deduced similarly from Eq. (10) and Eq. (11) using their corresponding generators.

In contrast to dihedral symmetry, the definition of rotational symmetry cannot ensure continuity across the boundaries of fundamental regions. For the symmetry transformation $\gamma_m \in C_m$ defined on P_m, we can map any point $(x, y) \in P_m$ to the fundamental region U_m under γ_m and obtain a new point $\boldsymbol{p} = (x_\gamma, y_\gamma) = \gamma_m (x, y) \in U_m$. We then



(a) Pattern with GLM tiling (b) Surface of GLM pattern

(c) Pattern with GLMT_1 tiling (d) Surface of GLMT_1 pattern

(e) Pattern with H tiling (f) Surface of H pattern

Fig. 6 Aesthetic patterns of three kinds of invariant mapping: (a), (c), and (e) are the patterns generated by the model Eq. (1) ($q = 4.4$, $s = 8$, $x_t = y_t = 0$) corresponding to M_1, M_2, and M_3, respectively. (b), (d), and (f) are the surface of pattern corresponding to (a), (b), and (c), respectively.

constructed a composite mapping h applied to γ_m to satisfy the continuity condition:

$$\begin{aligned}(x', y') &= h\left(\gamma_m(x, y)\right) \\ &= (x_\gamma, y_\gamma)\,\sigma\left(\frac{d_{\min}(\boldsymbol{p})}{d_{\text{sum}}(\boldsymbol{p})}\right) \\ &\quad + (x_\gamma, \eta\,|y_\gamma|)\left[1 - \sigma\left(\frac{d_{\min}(\boldsymbol{p})}{d_{\text{sum}}(\boldsymbol{p})}\right)\right] \end{aligned} \tag{12}$$

where $\eta \in (0, 1]$ is a parameter used to control the texture details and $d_{\min}(\boldsymbol{p})$, $d_{\text{sum}}(\boldsymbol{p})$, and $\sigma(x)$ are the three functions. $d_{\min}$ denotes the minimum distance from a point in the fundamental region $\boldsymbol{p}$ to its boundary, ∂U_m.

$$d_{\min}(\boldsymbol{p}) = \min_{e \in \partial U_m} d(\boldsymbol{p}, e), \quad \boldsymbol{p} \in U_m \tag{13}$$

d_{sum} is the summation of all distances defined in Eq. (13):

$$d_{\text{sum}}(\boldsymbol{p}) = \sum_{e \in \partial U_m} d(\boldsymbol{p}, e), \quad \boldsymbol{p} \in U_m \tag{14}$$

σ is a smooth function used to encourage each edge of the fundamental region U_m to have the same color after rotation γ_m is applied, which can be defined as a Sigmoid function:

$$\sigma(x) = \frac{1}{1 + \mathrm{e}^{-(b+\omega x)}} \tag{15}$$

where b and ω are parameters of the linear function of x. We can easily verify that $r = d_{\min}(\boldsymbol{p})/d_{\text{sum}(\boldsymbol{p})} \in [0, 1/3]$. When point $\boldsymbol{p} \in U$ approaches ∂U, $r \to 0$ and its mapped point is expected to approach its symmetrical point about the $x$axis, that is, $\sigma(r) \to 0$. In contrast, when $\boldsymbol{p}$ is away from ∂U, $r \to 1/3$, we would like to retain the original color calculated by QRP model as much as possible, that is, $\sigma(r) \to 1$. Thereafter, we set $b = -5$ and $\omega = 25$ by default, which satisfy the properties.

Because all symmetrical points lead to the same coordinate (x', y') as in Eq. (12), the mapping M_2 defined as $M_2(x, y) = H_q(x', y')$ is an invariant mapping. An illustration of the mapping is shown in Fig. 5(e), and the generated patterns and surfaces of the QRP model are shown in Fig. 6(c) and Fig. 6(d).

4.3.3 Invariant mappings with reflection symmetries

The patterns generated using M_3 have local reflection symmetries, the axes of which are lines that end at the polygon's centroid and one of its vertices. The constructions of M_3 and M_1 have certain similarities. However, unlike M_1, M_3 has a larger fundamental region area and must consider the boundary continuity between the polygons.

We used Fig. 5(c) as an example to demonstrate the construction details. In Fig. 5(c), let D'_m be dihedral symmetry group of P_m ($m = 4, 6, 12$). $\boldsymbol{R}_{O'_4 R}$ and $\boldsymbol{R}_{O'_4 S}$, $\boldsymbol{R}_{O'_6 R}$ and $\boldsymbol{R}_{O'_6 S}$, and $\boldsymbol{R}_{O'_{12} R}$ and $\boldsymbol{R}_{O'_{12} S}$ are the generators of D'_4, D'_6, and D'_{12}, respectively. The fundamental regions U corresponding to each polygon are $\triangle O'_4 RS$, $\triangle O'_6 RS$, and $\triangle O'_{12} RS$. Then, for any point $A_1 \in \overline{RK}$, the symmetrical points under invariant mapping M_3 lying at P_4, P_6, and P_{12} are as Eq. (16):

$$\begin{cases} A_2 = \boldsymbol{R}_{O'_4 R}(A_1) \in \overline{RP} \\ A_3 = \boldsymbol{R}_{O'_4 S}(A_1) \in \overline{SW} \\ A_4 = \boldsymbol{R}_{O'_6 R}(A_1) \in \overline{RM} \\ A_5 = \boldsymbol{R}_{O'_6 S}(A_1) \in \overline{SN} \\ A_6 = \boldsymbol{R}_{O'_{12} R}(A_1) \in \overline{RQ} \\ A_7 = \boldsymbol{R}_{O'_{12} S}(A_1) \in \overline{ST} \end{cases} \tag{16}$$

Unlike M_1, the construction of M_3 must eliminate the texture seams at the boundaries of the tilings. To this end, we constructed a mapping that can make each edge of polygon P_m symmetrical about its perpendicular bisector. Therefore, for $\forall (x,y) \in P_m$, we first mapped it to its fundamental region U_m by symmetry transformation $\gamma_n \in D'_m$, that is, $\boldsymbol{p} = (x_\gamma, y_\gamma) = \gamma_m (x,y) \in U_m$, and then apply a composite function to make it symmetrical.

$$(x', y') = (x_\gamma, y_\gamma)\, \sigma(d) + (x_\gamma, \eta\, |y_\gamma|)\, [1 - \sigma(d(\boldsymbol{p}))] \tag{17}$$

where σ is defined in Eq. (15) and $d(\boldsymbol{p})$ is the distance from $\boldsymbol{p}$ to $\overline{RS}$, that is, $d(\boldsymbol{p}) = d(\boldsymbol{p}, \overline{RS}) = \min_{\boldsymbol{q} \in \overline{RS}} ||\boldsymbol{p} - \boldsymbol{q}||$.

M_3 is an invariant mapping under the definition in Eq. (17). An illustration is shown in Fig. 5(f), and the generated pattern and its QRP surface are shown in Fig. 6(e) and Fig. 6(f).

4.4 GPU implementation

Because of the enormous number of pixels in a pattern to be calculated, serial implementation on the CPU is inefficient. Thus, we took advantage of the multi-pipeline feature of the GPU, leveraging it to render all pixels of a pattern in parallel, and applied OpenGL Shading Language (GLSL) for GPU computing.

In the vertex shader, the projection matrix was received from the CPU memory. The grids were translated by the matrix to make the tile cover the entire window, as shown in Fig. 7.

The fragment shader obtains the vertex information passed in from the vertex shader as well as other parameters. Substitute these parameters into Eqs. (4)–(6) in Section 4.2 to construct the fundamental region. Simultaneously, three different types of invariant mappings, M_1, M_2, and M_3 as well as predefined QRP models, are all implemented in the fragment shader in Section 4.3.

We then colored all pixels of the pattern using texture mapping. Each pixel corresponds to a texture coordinate, and its color is determined by the value H, which can be efficiently calculated using the QRP model.

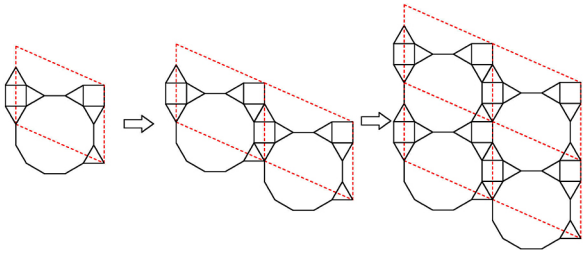


Fig. 7 Illustration of tiling construction by translating grid.

5 Experimental results

The proposed pattern generation algorithm is implemented using GPU shaders and ran on a PC with Intel Xeon (2.40 GHz) processor, NVDIA Quadro P620 (2 GB), Windows 10 OS, and all patterns are rendered to image with a resolution of 1024 pixels × 1024 pixels. The tiling dataset was collected from Ref. [34] and contained 212 tilings. By default, we set the basic model (Eq. (1)) as the QRP model with parameters $q = 5, s = 18, x_t = y_t = 0$. The parameter q was originally defined as a positive integer but could be set as any positive real number $q \geqslant 1$ in practice, which also works well.

In the following sections, we experimentally demonstrated the effectiveness of the algorithm using several examples and comparisons from diverse perspectives.

5.1 Influence factors of the algorithm

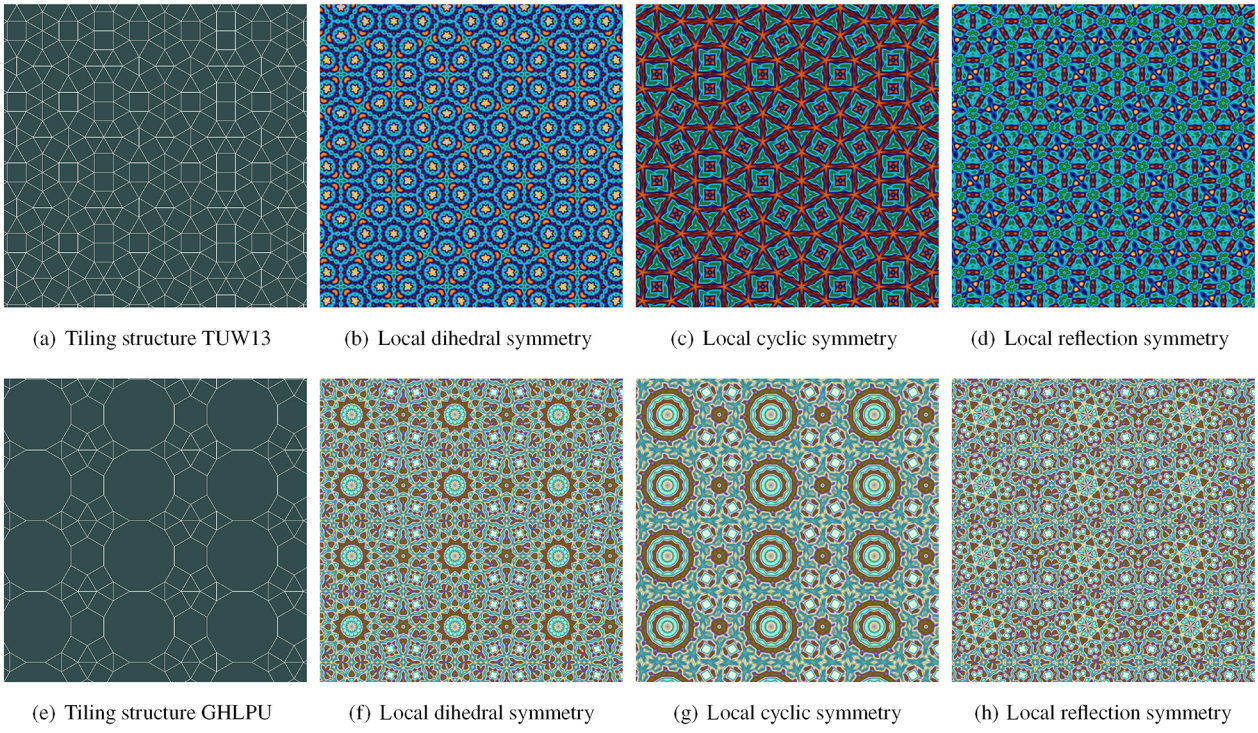
There are various ways to control the generated patterns for our algorithm, including η in Fig. 8, the type of local symmetry, the method of overlaying basic polygons, models, and parameters of quasi-regular patterns, and the type of tiling structures, etc. All of these factors significantly affected the results.

The parameter η in Eq. (12) is used to rescale the local detail of the texture. This is an additional degree for controlling the variation in the patterns. When $\eta = 1$, the position of the mapped point was maintained, and the pattern displayed its original texture detail. When $\eta < 1$, the position of the mapped point about the y axis was rescaled; thus, the details of the pattern were magnified. The smaller the value of η, the larger the magnification and the clearer the texture. The results generated by different values of η are shown in Fig. 8. We empirically set $\eta = 0.3$ by default.

The next example shows the patterns generated with three different types of local symmetries with fixed tiling structures (Fig. 9(a) and Fig. 9(e)) and the same QRP models. The results show that local dihedral symmetry (Fig. 9(b) and Fig. 9(f)), rotational symmetry (Fig. 9(c) and Fig. 9(g)), and reflection symmetry (Fig. 9(d) and Fig. 9(h)) have

(a) $\eta = 0.2$ (b) $\eta = 0.3$ (c) $\eta = 0.5$ (d) $\eta = 0.7$ (e) $\eta = 1.0$

Fig. 8 Results with different values of η in Eq. (12), where the QRP model is defined in Eq. (1) ($q = 3.6$, $s = 3$, $x_t = 2$, $y_t = 0$) and the invariant mapping is M_2.



(a) Tiling structure TUW13 (b) Local dihedral symmetry (c) Local cyclic symmetry (d) Local reflection symmetry

(e) Tiling structure GHLPU (f) Local dihedral symmetry (g) Local cyclic symmetry (h) Local reflection symmetry

Fig. 9 Patterns generated by M_1, M_2, and M_3. (a) and (e) are the tiling structure of the generated patterns (b)–(d) and (f)–(h), respectively.

their own art styles and differ significantly from each other in appearance. The reason behind this is that the constructed fundamental region of each type is different, which leads to different shapes of QRP defined in that region.

The third example in Fig. 10 shows the effectiveness of different methods for rearranging the basic tiling elements. We used the same tiling structure (HLMP), QRP models, and local symmetry type (dihedral) but with different rearrangements of the basic elements. The results show that they share some similarities in global structure but differ in local texture details. This is because the tiling structure and local symmetry type reflect global features, whereas the QRP model determines the local texture features.

Then, we showed examples to demonstrate the colorful texture patterns of QRP models and their parameters, with all remaining factors fixed. Figure 11 displays tiling-constrained patterns generated with different QRP models and the same parameters ($q = 4.8, s = 8, x_t = 0, y_t = 0$), and their function expressions are varied from the basic models, which can be seen in Table 1. Owing to our scheme of building the fundamental region, any QRP model defined is continuous across the edges of the tiling, which naturally eliminates the texture seams on those edges.The patterns generated with different QRP models (Fig. 11) differ considerably for both the local and global textures. We also investigated the parameters (q, s, x_t, y_t) of the QRP model, and the corresponding results are shown in Fig. 12. Because

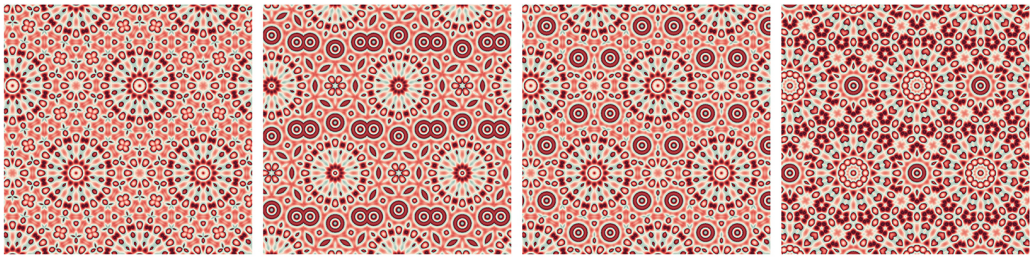
Table 1 QRP models used in the experiments

QRP model	Example
$H=\sum_{i=1}^{\lfloor q\rfloor}\cos\left(x\cos^3\left(\frac{2\pi i}{q}\right)+y\sin^3\left(\frac{2\pi i}{q}\right)\right)$	11(b)
$H=\sum_{i=1}^{\lfloor q\rfloor}\cos\left(\lvert x\rvert^{\frac{3}{4}}\cos\left(\frac{2\pi i}{q}\right)+\lvert y\rvert^{\frac{3}{4}}\sin\left(\frac{2\pi i}{q}\right)\right)$	11(c)
$H=\sum_{i=1}^{\lfloor q\rfloor}\cos\left[\Omega\right]^2+\frac{\sin(y)+\cos(x)}{5}$	11(d)
$H=\sum_{i=1}^{\lfloor q\rfloor}\cos\left[\Omega\right]^2+\frac{\cos(xy)}{5}-\frac{(x+y)}{100}$	11(e)
$H=\sum_{i=1}^{\lfloor q\rfloor}\sin\left\{\cos\left[\Omega\right]\right\}+\lvert\sin\left[\Omega\right]\rvert$	11(f)
$H=\sum_{i=1}^{\lfloor q\rfloor}\tan\left\{\sin\left[\Omega\right]\right\}+\cos\left[\Omega\right]^3$	11(g)
$H=\sum_{i=1}^{\lfloor q\rfloor}\cos\left[\Omega\right]^3+\cos\left(y\sin\left(x\right)\right)$	11(h)

Note: $\Omega=x\cos\left(\frac{2\pi i}{q}\right)+y\sin\left(\frac{2\pi i}{q}\right)$

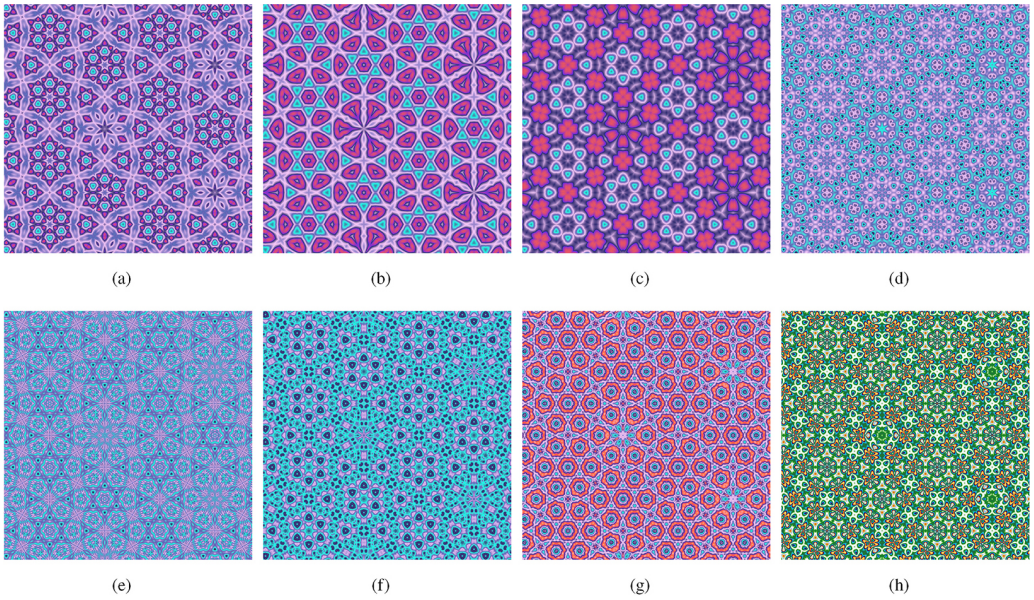
the parameter q reflects the complexity of the model, the local texture detail becomes more complicated as q increases (Figs. 12(a)–12(d)). The parameters x_t, y_t confine the domain of definition and, thus, affect the contour shape of the QPR models. The textural details of the pattern changed as these two parameters varied (Figs. 12(e)–12(l)). s was used to control the scale of the texture pattern, and the texture size decreased as the value of s increased. The translation period gradually increases for rendered images with the same resolution (Figs. 12(m)–12(p)).

The tiling structure determines the pattern layout. Moreover, the fundamental regions differ for different tiling structures; therefore, the QRP models defined therein are distinct. Figure 13 displays pattern generation results with different tiling structures of a different number of vertex types while keeping the remaining factors unchanged. It can be observed from the figure that the tiling structures significantly



(a) $3^r4^r12^r$ (b) $3^l4^r12^r$ (c) $3^r4^l12^r$ (d) $3^r4^r12^l$

Fig. 10 Patterns generated from different ways of rearrangement of basic elements, where the QRP model is defined in Eq. (1) ($q=5.0$, $s=8$, $x_t=0$, $y_t=0$) and the invariant mapping is M_1.



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 11 Patterns generated with different QRP models: (a) basic model (Eq. (1)). (b)–(h) correspond to seven different models listed in Table 1. All of them are generated by the parameters ($q=4.8, s=8, x_t=0, y_t=0$) and invariant mapping M_1.

Fig. 12 Patterns generated with different parameters of the QRP model.(a)–(d) are results tuned by the parameter q with HLW tiling structure and the mapping M_1; (e)–(h) are results tuned by the parameter s with the mapping M_3. (i)–(l) are results tuned by the parameter x_t with the mapping M_1. (m)–(p) are results tuned by the parameter y_t with the mapping M_1.

affect the global features as well as the local details of the generated patterns.

The last example shows the influence of the color palette selection. Keeping all other factors frozen, we selected different color palettes and numbers of colors to render the image. The results are shown in Fig. 14. Note that the texture detail of the pattern remains unchanged for palettes of different colors (Figs. 14(a)–14(d)). However, the texture detail decreases as the number of colors decreases (Figs. 14(e)–14(h)).

5.2 Comparisons

In this section, we compared the advantages of our algorithm against the state-of-the-art methods to

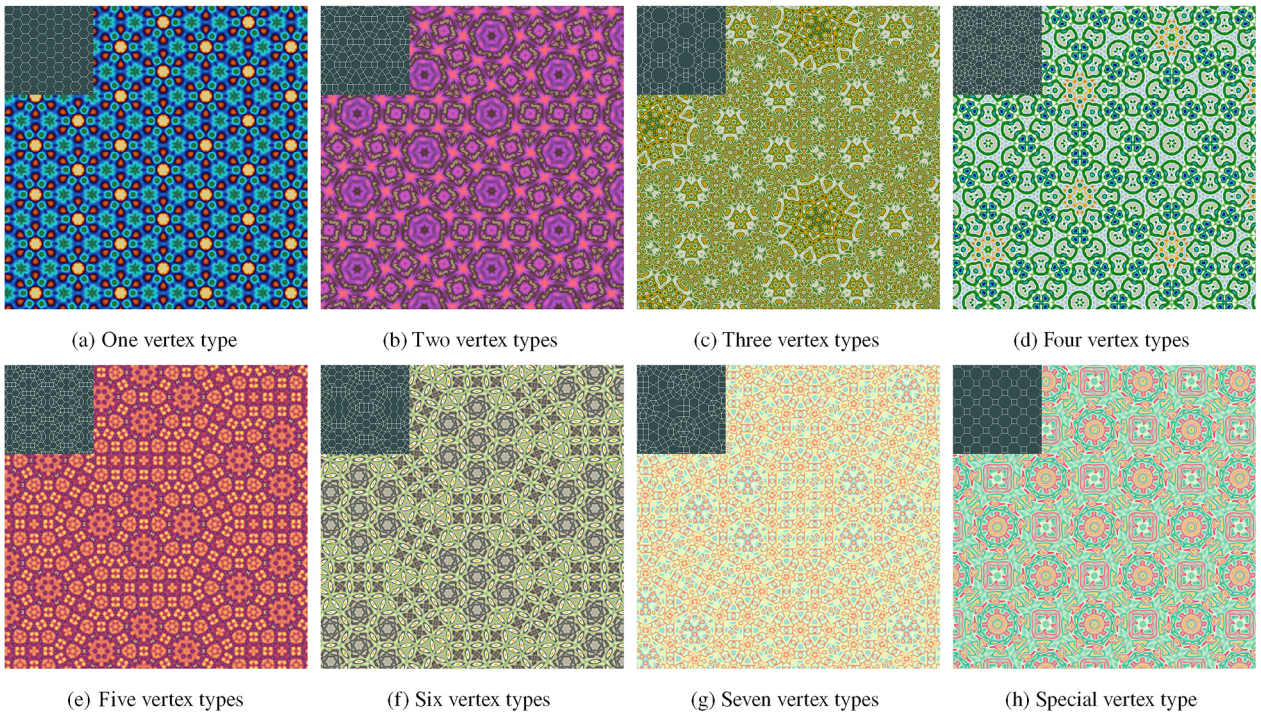


Fig. 13 Patterns generated with tilings of different vertex types.

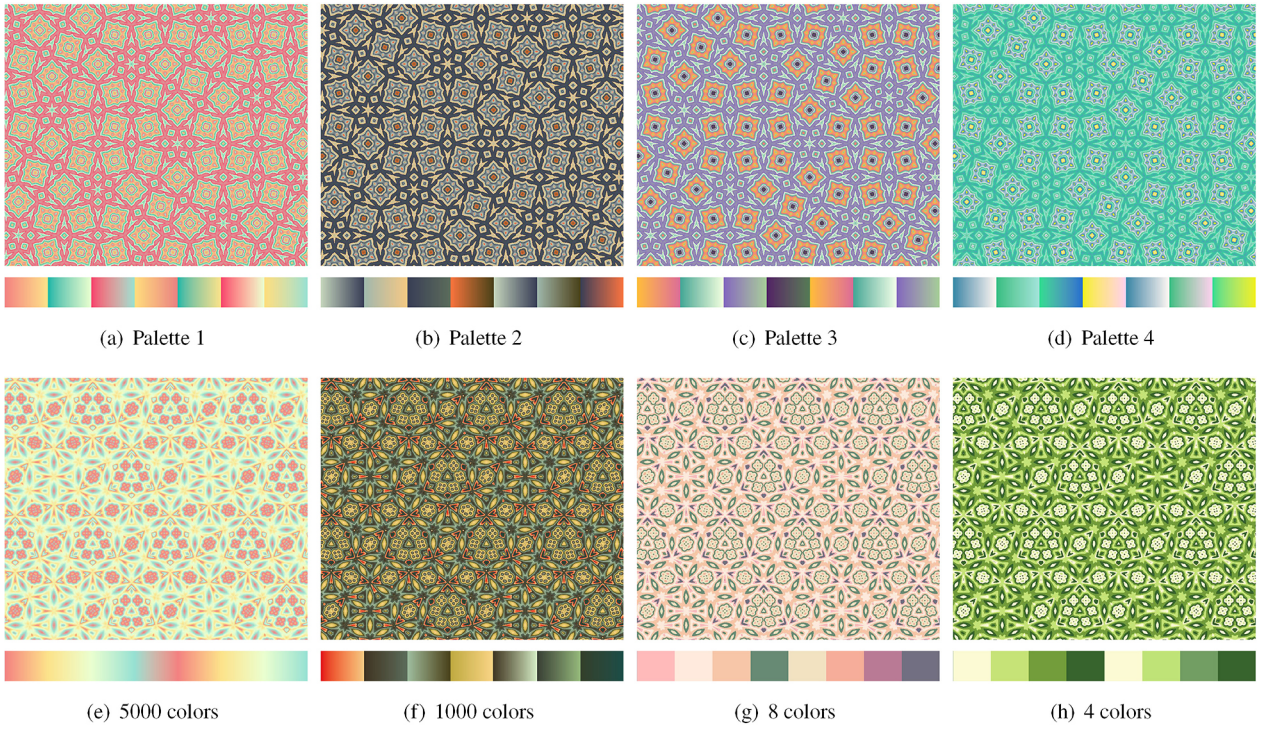


Fig. 14 Patterns generated with different color palettes. Patterns from (a) to (d) are generated with different color palettes under the same tiling structure. Patterns from (e) to (h) are generated with color palettes with different numbers of colors.

demonstrate its superiority.

A dynamical system is one of the most popular tools for generating aesthetic patterns with symmetrical properties. Thus, we first compared it with the generating patterns with simple planar symmetries from dynamical systems [2, 24]. The results are shown in Figs. 15(a)–15(h). In general, the patterns are regular and monotonous because of their simple,

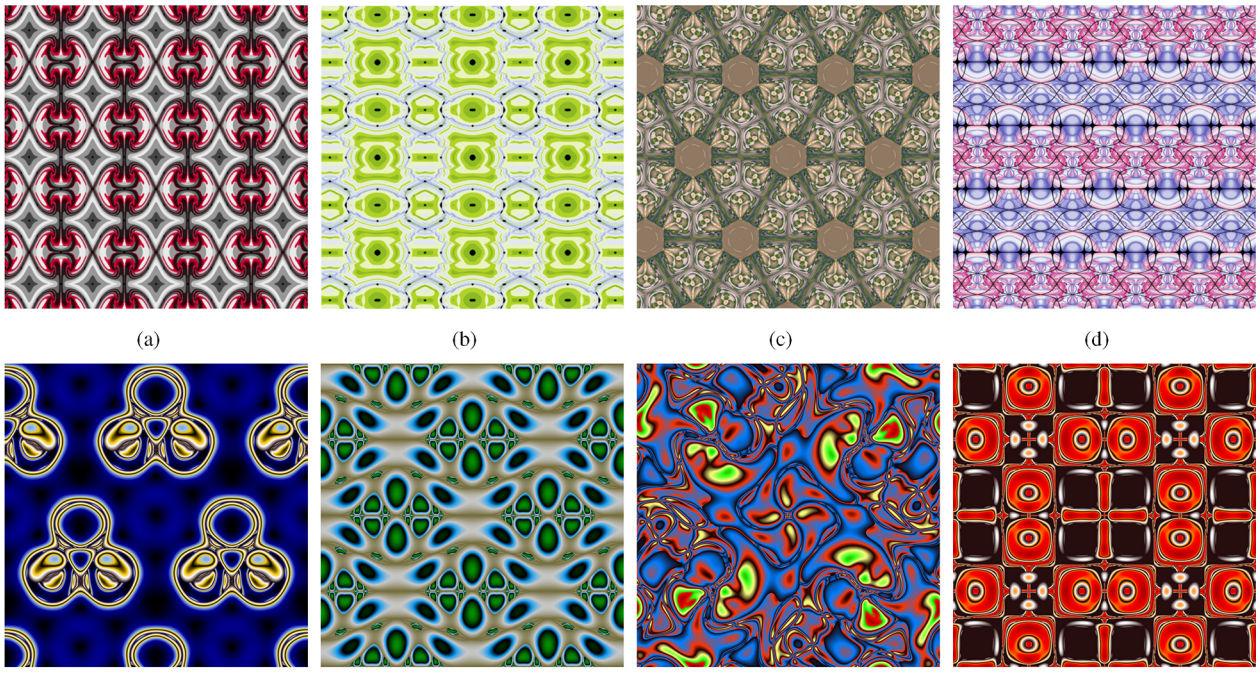


Fig. 15 Symmetrical patterns generated from dynamical systems. (a)–(d) Results of the method [24]. (e)–(h) Results of the method [2]. All the figures are extracted from Refs. [24] and [2].

symmetric wallpaper layout. Additionally, these methods [2, 24] require the design of specific functions with symmetric properties for dynamical system models, which is difficult and tedious. In comparison, users can flexibly control the pattern from various aspects using our method, and the generated patterns vary in both space structures and textures.

We also compared tiling-constrained pattern generation methods based on dynamical systems [14, 15]. We implemented both algorithms and present their results in Fig. 16 and Fig. 17. Comparisons show that our algorithm outperforms the others [14, 15] in several respects. First, our method can generate seamless patterns for any given k-uniform tilings. In contrast, the methods [14, 15] are limited to simple types of tilings (Penrose and Archimedean) and should design invariant mappings for each tiling case by case. Second, our method naturally eliminates seams across the tiling edges; thus, the generated patterns are rich in variation near these edges. However, both methods [14, 15] require the construction of explicit continuous conditions across the tiling edges, and visibility is usually observed in their patterns. Third, the QRP model used in our algorithm has several fixed and meaningful parameters that can conveniently control pattern variation. However, the parameters of the dynamic models are integrated into the functions, which have different geometrical significances for the

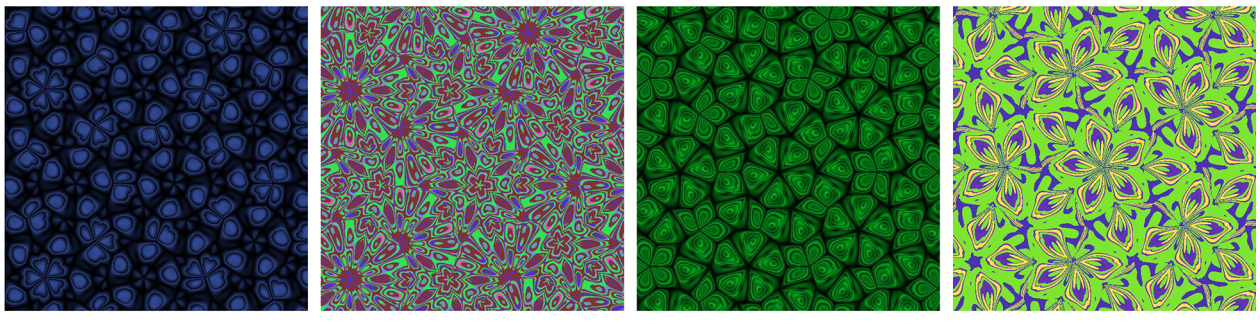
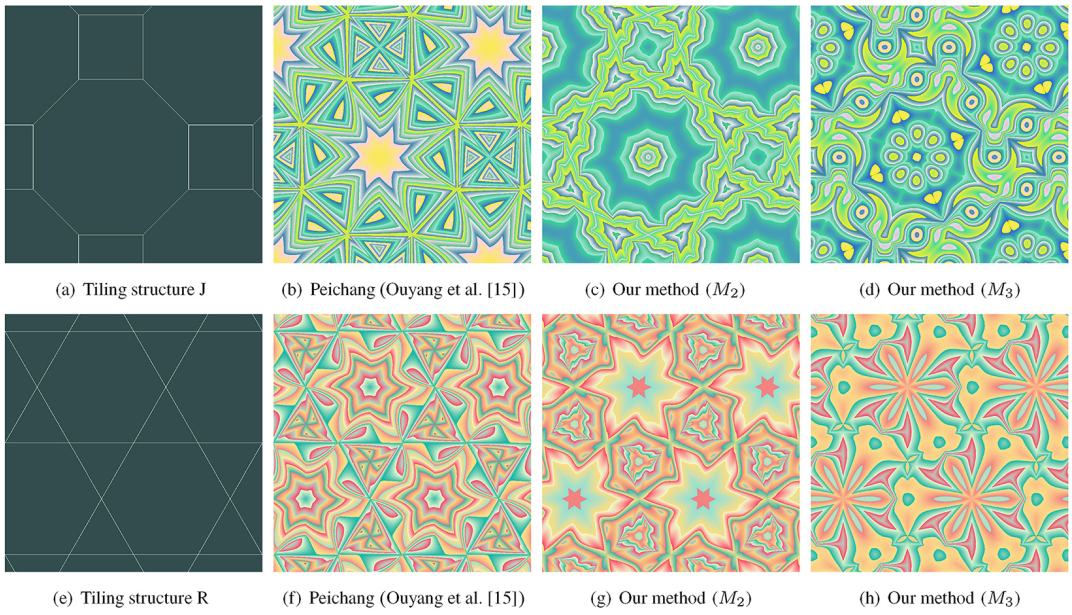


Fig. 16 Patterns generated with Penrose tilings [14].



(a) Tiling structure J (b) Peichang (Ouyang et al. [15]) (c) Our method (M_2) (d) Our method (M_3)

(e) Tiling structure R (f) Peichang (Ouyang et al. [15]) (g) Our method (M_2) (h) Our method (M_3)

Fig. 17 Comparison results with tiling-constrained pattern generation methods.(a) and (e) are the tiling structures of (b)–(d) and (f)–(h), respectively; (b) and (f) are generated by Ouyang et al. [15]; (c), (g), (d), and (h) are generated by our method.

different functions. Thus, one has to frequently change the function models to obtain patterns with different styles.

5.3 Performance

Owing to the parallel implementation on GPU, our algorithm achieves high performance, even for rendering images of large resolutions. Table 2 shows the time statistics for rendering images of different resolutions. Note that the time cost remains nearly the same as the resolution increases.

Table 2 Graphics generation speed at different resolutions

Pattern resolution	Cost time per pattern (s)
512×512	0.0403
1024×1024	0.0407
2048×2048	0.0411
4096×4096	0.0413
8192×8192	0.0420

5.4 Application

The patterns generated by our method can be used as materials in certain design fields, such as fabrics and ties. Here, we demonstrated the application of such patterns in the tie design (Fig. 18). Because the style and features of our patterns are similar to those of ties, slight alterations are needed for the final applications, which could greatly assist artists in improving the efficiency of design flows.

(a) (b) (c) (d)

Fig. 18 An application of our aesthetic patterns used in ties design.

6 Conclusions

We proposed a novel method for generating colorful patterns by symmetrizing quasi-regular patterns with general k-uniform tilings composed of regular polygons. Our method, which unifies the construction of invariant mappings for all k-uniform tilings, is flexible for controlling the generation of patterns with various parameters in both textures and symmetric layout structures. Compared with existing methods, our method can generate patterns with more variations in spatial layout structures and texture details. In addition, our method is fully automatic, which significantly reduces the effort required to design specific invariant mappings for any given k-uniform tiling.

Although the overall performance of our method is promising, it has a few limitations. We did not consider the aesthetic evaluation of specific domains of the generated patterns, which may hinder practical design applications. Two main factors affect aesthetics. The first is the strategy for partitioning the height field of QRP models, which significantly impacts the shape of the generated pattern. The other is color compatibility and spatial arrangements, which require artists to create secondary designs for practical applications. In the future, we plan to integrate the aesthetic model learned from a specific art design field into the method and study a method for generating quasi-regular patterns with the constraints of aperiodic tiling's.

Acknowledgements

This work was supported by the Key R&D Programs of Zhejiang Province (Nos. 2023C01224 and 2022C01220) and the National Natural Science Foundation of China (No. 61702458). Yun Zhang was partially supported by Zhejiang Province Public Welfare Technology Application Research (No. LGG22F020009), and Key Lab of Film and TV Media Technology of Zhejiang Province (No. 2020E10015).

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] Ouyang, P.; Chung, K. W.; Nicolas, A.; Gdawiec, K. Self-similar fractal drawings inspired by M. C. Escher's print *Square limit*. *ACM Transactions on Graphics* Vol. 40, No. 3, Article No. 1, 2021.

[2] Gdawiec, K.; Adewinbi, H. Procedural generation of artistic patterns using a modified orbit trap method. *Applied Sciences* Vol. 12, No. 6, Article No. 2923, 2022.

[3] Sayed, Z.; Ugail, H.; Palmer, I.; Purdy, J.; Reeve, C. Auto-parameterized shape grammar for constructing Islamic geometric motif-based structures. In: *Transactions on Computational Science XXVIII. Lecture Notes in Computer Science, Vol. 9590*. Gavrilova, M.; Tan, C.; Sourin, A. Eds. Springer Berlin Heidelberg, 146–162, 2016.

[4] Mandelbrot, B. B.; Aizenman, M. Fractals: Form, chance, and dimension. *Physics Today* Vol. 32, No. 5, 65–66, 1979.

[5] Helt, G. Extending mandelbox fractals with shape inversions. In: Proceedings of the Bridges 2018: Mathematics, Art, Music, Architecture, Education, Culture, 547–550, 2018.

[6] Szyszkowicz, M. Computer art from numerical methods. *Computer Graphics Forum* Vol. 10, No. 3, 255–259, 1991.

[7] Gdawiec, K.; Kotarski, W.; Lisowska, A. Automatic generation of aesthetic patterns with the use of dynamical systems. In: *Advances in Visual Computing. Lecture Notes in Computer Science, Vol. 6939*. Bebis, G. et al. Eds. Springer Berlin Heidelberg, 691–700, 2011.

[8] Zaslavsky, G. M.; Sagdeev, R. Z.; Usikov, D. A.; Chernikov, A. A.; Eugene Wayne, C. Weak chaos and quasi-regular patterns. *Physics Today* Vol. 45, No. 8, 70–71, 1992.

[9] Zhang, Y.; Li, D. Reseaches on fabric patterns designing with visual informations in non-linear dynamic systems. *Journal of Textile Research* Vol. 23, No. 5, 27–28, 2002. (in Chinese)

[10] Liu, S. The generation system of textile pattern draft based on quasi-regular pattern theory. *International Journal of Information Engineering and Electronic Business* Vol. 1, No. 1, 50–57, 2009.

[11] Zhang, Y.; Jin Y.; Chen K. *Quasi-Regular Pattern Art*. China Textile Press, 2017. (in Chinese)

[12] Chung, K. W.; Chan, H. S. Y. Symmetrical patterns from dynamics. *Computer Graphics Forum* Vol. 12, No. 1, 33–40, 1993.

[13] Lu, J.; Ye, Z.; Zou, Y. Automatic generation of colorful patterns with wallpaper symmetries from dynamics. *The Visual Computer* Vol. 23, No. 6, 445–449, 2007.

[14] Chung, K. W.; Wang, B. N. Automatic generation of aesthetic patterns on aperiodic tilings by means of dynamical systems. *International Journal of Bifurcation and Chaos* Vol. 14, No. 9, 3249–3267, 2004.

[15] Ouyang, P.; Zhao, W.; Huang, X. Beautiful math, part 5: Colorful Archimedean tilings from dynamical systems. *IEEE Computer Graphics and Applications* Vol. 35, No. 6, 90–96, 2015.

[16] Soto Sánchez, J. E.; Weyrich, T.; Medeiros e Sá, A.; de Figueiredo, L. H. An integer representation for periodic tilings of the plane by regular polygons. *Computers & Graphics* Vol. 95, 69–80, 2021.

[17] Gieseke, L.; Asente, P.; Mï£ · ch, R.; Benes, B.; Fuchs, M. A survey of control mechanisms for creative pattern generation. *Computer Graphics Forum* Vol. 40, No. 2, 585–609, 2021.

[18] Speller, T.; Whitney, D.; Crawley, E. Using shape grammar to derive cellular automata rule patterns. *Complex Systems* Vol. 17, 79–102, 2007.

[19] Gdawiec, K. Fractal patterns from the dynamics of combined polynomial root finding methods. *Nonlinear Dynamics* Vol. 90, No. 4, 2457–2479, 2017.

[20] Zhang, Y.; Fu, Y. Fabric pattern design based on quasi-regular pattern theory. *Journal of Textile Research* Vol. 26, No. 5, 58–62, 2005. (in Chinese)

[21] Jia, F. X.; Zhang, Y. Methods of clothing pattern designing based on quasi-regular patterns with shadow effect. *Journal of Textile Research* Vol. 38, No. 7, 124–129, 2017. (in Chinese)

[22] Carter, N. C.; Eagles, R. L.; Grimes, S. M.; Hahn, A. C.; Reiter, C. A. Chaotic attractors with discrete planar symmetries. *Chaos, Solitons & Fractals* Vol. 9, No. 12, 2031–2054, 1998.

[23] Zou, Y.; Li, W.; Lu, J.; Ye, R. Orbit trap rendering method for generating artistic images with cyclic or dihedral symmetry. *Computers & Graphics* Vol. 30, No. 3, 470–473, 2006.

[24] Gdawiec, K. Procedural generation of aesthetic patterns from dynamics and iteration processes. *International Journal of Applied Mathematics and Computer Science* Vol. 27, No. 4, 827–837, 2017.

[25] Liu, S.; Leng, M.; Ouyang, P. The visualization of spherical patterns with symmetries of the wallpaper group. *Complexity* Vol. 2018, Article No. 7315695, 2018.

[26] Ouyang, P.; Fathauer, R. W.; Chung, K. W.; Wang, X. Automatic generation of hyperbolic drawings. *Applied Mathematics and Computation* Vol. 347, 653–663, 2019.

[27] Chung, K. W.; Chan, H. S. Y.; Wang, B. N. Automatic generation of nonperiodic patterns from dynamical systems. *Chaos, Solitons & Fractals* Vol. 19, No. 5, 1177–1187, 2004.

[28] Ouyang, P.; Tang, X.; Chung, K.; Yu, T. Spiral patterns of color symmetry from dynamics. *Nonlinear Dynamics* Vol. 94, No. 1, 261–272, 2018.

[29] Ouyang, P.; Fathauer, R. W. Beautiful math, part 2: Aesthetic patterns based on fractal tilings. *IEEE Computer Graphics and Applications* Vol. 34, No. 1, 68–76, 2014.

[30] Ouyang, P.; Yi, H.; Deng, Z.; Huang, X.; Yu, T. Boundary dimensions of fractal tilings. *Fractals* Vol. 23, No. 4, 1550035, 2015.

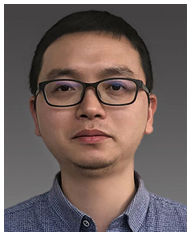
[31] Wang, X.; Ouyang, P.; Chung, K.; Zhan, X.; Yi, H.; Tang, X. Fractal tilings from substitution tilings. *Fractals* Vol. 27, No. 2, 1950009, 2019.

[32] Medeiros e Sá, A.; de Figueiredo, L. H.; Soto Sanchez, J. E. Synthesizing periodic tilings of regular polygons. In: Proceedings of the 31st SIBGRAPI Conference on Graphics, Patterns and Images, 17–24, 2018.

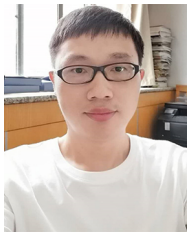
[33] Kaplan, C. S. Introductory tiling theory for computer graphics. *Synthesis Lectures on Computer Graphics and Animation* Vol. 4, No. 1, 1–113, 2009.

[34] Sá, R.; Medeiros e Sá, A. O livro SOBRE MALHAS ARQUIMEDIANAS. 2017. Available at https://www.cos.ufrj.br/seminarios/2018/slides/asla.pdf

Zhengzheng Yin is a master student at the School of Computer Science and Technology, Zhejiang Sci-Tech University, China. His research interests include computer graphics and generation of digital art graphics. E-mail: 202020503044@mails.zstu.edu.cn



Yao Jin is an associate professor at the School of Computer Science and Technology, Zhejiang Sci-Tech University, China. His research interests include computer graphics, digital geometry processing, and digital art graphics. E-mail: jinyao@zstu.edu.cn



Zhijian Fang is a lecturer at the School of Computer Science and Technology, Zhejiang Sci-Tech University. His research interests include big data and AI algorithms. E-mail: hptnt@zstu.edu.cn

Yun Zhang is currently a professor at Communication University of Zhejiang. His research interests include computer graphics, image and video editing, and computer vision. He is a member of CCF. E-mail: zhangyun@cuz.edu.cn

Huaxiong Zhang is a professor at the School of Computer Science and Technology, Zhejiang Sci-Tech University. His research interests include intelligent information processing and generation of digital art graphics. E-mail: zhxhz@zstu.edu.cn