Symmetrization of Quasi-regular Patterns with Periodic Tiling of Regular Polygons

Zhengzheng Yin
Zhejiang Sci-Tech University
Hangzhou, China
202020503044@mails.zstu.edu.cn

Yao Jin
Zhejiang Sci-Tech University
Hangzhou, China
jinyao@zstu.edu.cn

Zhijian Fang
Zhejiang Sci-Tech University
Hangzhou, China
hptnthptnt@zstu.edu.cn

Yun Zhang
Communication University of Zhejiang
Hangzhou, China
zhangyun@cuz.edu.cn

Huaxiong Zhang
Zhejiang Sci-Tech University
Hangzhou, China
zhxhz@zstu.edu.cn

Jiu Zhou
Zhejiang Sci-Tech University
Hangzhou, China
zhoujiu34@126.com

Lili He
Zhejiang Sci-Tech University
Hangzhou, China
llhellhe@zju.edu.cn

Abstract

Computer generated aesthetic patterns are widely used as design materials in many areas. Most common methods take fractals or dynamical systems as basic tools to create various patterns. To enhance aesthetics and controllability, some of them introduce symmetric layout along with these tools. One popular strategy employs dynamical systems compatible with symmetries, which constructs functions equivariant with respect to the desired symmetries. But it usually confines to simple planar symmetries. The other one generates symmetrical patterns under the constraints of tilings. Although a bit more flexible, it is restricted to some small ranges of tilings and lacks of texture variations. We thus proposed a new approach to generate aesthetic patterns by symmetrizing quasi-regular patterns with general k-uniform tilings. We adopted a unified strategy to construct invariant mappings on the k-uniform tilings, which can naturally eliminate texture seams across the tiling edges. Furthermore, we constructed three kinds of symmetries associates with patterns respectively, i.e., dihedral, rotational, and reflection symmetries. The proposed method can be easily implemented using GPU shaders, and is very efficient and suitable for any complicated tiling with regular polygons. Experiments show the advantages of our method over the state-of-the-arts in terms of the flexibility in controlling the generation of patterns with various parameters, as well as diversity of texture and style.

Keywords: Quasi-regular patterns, k-uniform tilings, Invariant mappings, Symmetry, Aethetic patterns

1. Introduction

A pattern is commonly considered as a type of repeated arrangement, and is used as basic materials with wide applications in designing various products, e.g., fabrics, neckties, jewelries, carpets, wallpapers, etc. Traditional way of pattern creation is by manual drawings which usually takes a long time to shape a single pattern even with the assistance of software such as Adobe Photoshop, Illustrator, et al. Thanks to various mathematical methods, patterns can be automatically and efficiently generated [1, 2]. Among these methods, shape grammar [3], fractals [4, 5], dynamical systems [6, 7] and their variants are the most commonly used tools. Although they are capable of creating splendidly aesthetic patterns, they always suffer from the uncontrollable problem, i.e., it is impossible to establish the relationship between the parameters of the mathematical models and their generated patterns, which prevents artists from wide applications. Quasi-regular pattern (QRP) is another kind of digital pattern generated by visualizing the smoothed form of Hamiltonian function [8]. Such patterns are usually characterised by translational and rotational symmetries with near-regular textures [9, 10, 11]. Though the parameters of its mathematical model have certain geometrical significance, it is still hard to predict the shape and texture of the pattern according to any given QRP model.

By contrast, it is much easier to control the spacial layout of the pattern than the texture details. Thus, to enhance aesthetics as well as controllability, patterns are usually created along with symmetric layouts. One way to generate such patterns is through constructing invariant functions with symmetries from dynamics [12, 13, 2]. The layout structure is determined by the designed function which is usually restricted to some simple planar symmetries. Another way to generate patterns is by using dynamical systems or fractals under the constraint of predefined tilings [14, 15].Although it is more flexible than the previous works, it usually confines to a small range of tiling structures, e.g., Penrose tilings [14] or Archimedes tilings [15]. In addition, their construction of continuous conditions across the tiling edges does reduce the texture richness, making the pattern visually monotonous.

In this paper, we proposed a new pattern generation method by symmetrizing quasi-regular patterns with constraints of general k-uniform tilings. We first reconstructed k-uniform tiling by the integer representation method [16]. Then we rearranged the basic elements of the tilings to ensure the texture continuity across the tiling edges. Next we constructed three kinds of invariant mappings with continuity conditions on the boundaries of fundamental regions. Finally, we colored the image space by QRP models with the help of the invariant mappings. 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 aethetic pattern by combining the tiling structures with the quasi-regular pattern models, which can flexibly control both the texture 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 large number of 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

There are large body of works on the generation of aesthetic patterns. Gieseke et al [17] recently conducted a survey on the control mechanisms of pattern generation. And here we only focused on the most related ones.

2.1. Pattern generation models

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

The model of quasi-regular pattern generates colorful patterns through visualizing a special kind of smooth functions [9, 10, 11]. It has several parameters with geometric significance, and the generated patterns usually have local symmetries and near-regular structures. Such patterns are widely used in the field of textile and garment design [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 spacial layout of the pattern and enhance aesthetic, researchers resorted to construct functions with symmetry property, which are used to generate patterns by dynamical systems. Chung et al. [12] constructed functions with all wallpaper symmetries by using Fourier series. Nathan et al. [22] extended the method to construct functions equivariant with respect to more symmetries, e.g. frieze groups and crystallographic groups. Zou et al [23] combined chaotic functions having cyclic and dihedral symmetries with orbit trap method to render aesthetic patterns. Lu et al. [13] generated patterns by constructively building trigonometric and polynomial functions with some specific wallpaper symmetries. Gdawiec et al. [24] applied fixed point theory to create symmetric functions. And they recently proposed a modified orbit trap method [2] which can obtain a great variety of interesting patterns. Besides, non-Euclidean geometries are also widely used in the generation of patterns with distinctive styles, such as spherical geometry [25] and hyperbolic geometry [26]. However, all these methods requires to analyze and design specific functions with certain symmetries, which are usually confined to a small range of symmetry types.

2.3. Patterns constrained by tilings

Another way to control the spatial layout of the pattern is by using predefined tilings. And the key of the method is to construct invariant mappings for the tiling so as to keep the texture continuity on the boundaries of polygons. One class of method uses dynamical systems based on several kinds of tilings, e.g. chair tiling [27],Penros tiling [14], spiral tiling [28] and Archimedean tilings [15]. Another class uses complicated fractal models instead with tiling structure possessing self-similarity property [29, 30, 31] . These methods are flexible in controlling the layout of the pattern. However, all of them suffer from at least two problems: 1)invariant mappings should be designed separately for each tiling; 2) the texture of the pattern is monotonous near the boundaries of the fundamental regions due to the simple continuous conditions. Our method combines 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 problems mentioned.

3. Preliminaries

To make the paper self-consistent, we introduce some preliminaries on the generation of quais-regular patterns and k-uniform tilings in this section.

3.1. Generation of quasi-regular patterns

Quasi-regular pattern is obtained by visualizing a specific kind of function whose geometry is a smooth 2-manifold surface, that is, a smooth height field defined in R^2. Its basic mathematical model is derived from the weak chaos theory [8] , and is known as the following equation,

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

where, q denotes the number of resonances, $(x, y) \in R^2$ are coordinates, and H_q is named as “QRP model” here. Please refer to [8] for the detail of the deduction. And the visualization of the surface defined by Eq. (1) is shown in Fig. 1(a).

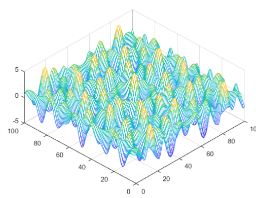
Noting that the contours of the QRP model H_q constitute a series of closed curves with various shapes, Zhang and Li [9] proposed a method to visualize contours of the surface H_q and produced quasi-regular pattern. They partitioned the height field of QRP model H_q into several disjoint intervals $h_1,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

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

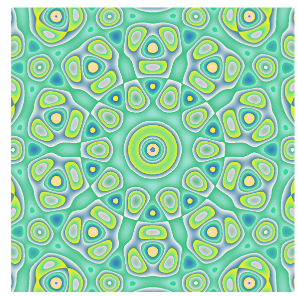
where W_x, W_y denote the width and height of the canvas,(x_t, y_t) is the vector for the translation and s is used to control the scale of the normalized space. Then, we set the color value for the pixel (n_x, n_y) of the connected region which corresponds to the its partitioned interval. See details of the algorithm in Alg. 1.

Algorithm 1 The 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



(a) The surface of the QRP model



(b) Colored pattern of the QRP model

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

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

While Archimedean tilings can be simply represented by vertex types, the representation of general k-uniform tiling is relatively complicated. Sánchez et al. [32] presented a simple representation for periodic tilings of the plane by regular polygons. The approach represents explicitly a minimal subset of the vertices from which they systematically

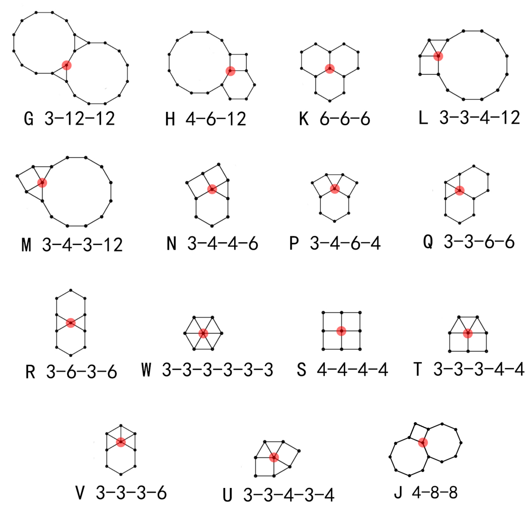


Figure 2. 15 different vertex types that appear in k-uniform tilings with regular polygons

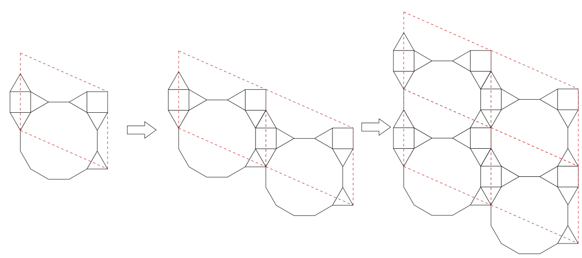


Figure 3. Illustration of tiling construction by translating grid.

generate all vertices of the tiling by translations. After that, Sánchez et al.[16] improved the method and proposed an integer representation for any k-uniform tiling.

By their definition, the coordinates of tiling's 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 12^{th} roots of unity. Thus a k-uniform tiling can be represented concretely by a $(2+n) \times 4$ integer matrix containing lattice coordinates for two translation vectors (the first two rows) and n seed vertices (remaining rows). One can refer [16] for the detail.

Given the integer representation of k-uniform tilings, the calculation of k-uniform tilings composed of regular polygon is straightforward, which can be summarized as 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).

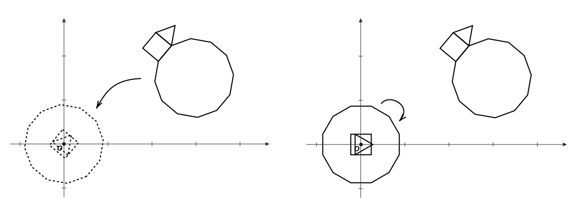
4. Symmetrization of Quasi-regular Patterns

4.1. Overview

In this paper, we aim to generate aesthetic patterns from quasi-regular patterns by controlling the spacial layout, and more precisely symmetrize the patterns under the constraint of any given k-uniform tiling. The key problem is to construct invariant mappings under certain symmetry group that can keep the continuity of the rendered pattern across the edges of the tiling and the axes of local symmetrical transformations.

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

4.2. Rearrangement of basic elements



(a) Translation (b) Rotation

Figure 4. Initialize the location of the basic elements.

Algorithm 2 Symmetrize QRP with k-uniform tilings

Input:
1. An integer matrix representing the given k-uniform tiling: T.
2. 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_n with Eq. (3).

for all P_n **do**
  for all $(P_x, P_y) \in P_n$ **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 Sec.4.3;
    Calculate the corresponding pixel color of (x, y) with the QRP algorithm (Alg. 1);
  end for
end for

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

To this end, instead of directly constructing the fundamental region for the tiling [14, 15], we proposed a scheme through rearranging all the basic elements to form the target domain, where the fundamental region can be defined. The goal of the scheme is to transform each basic element so that one edge of each basic element overlaps together, which helps to eliminate the texture seams between neighboring polygons naturally.

For convenience of constructing the fundamental region later, 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 so 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. 4). Therefore, the mapping can be expressed as follows:

$$\begin{pmatrix} P'_x \\ P'_y \end{pmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{pmatrix} P_x - C_x \\ P_y - C_x \end{pmatrix}, \tag{4}$$

where (C_x, C_y) is the centroid of the polygon, (P_x, P_y) and (P'_x, P'_y) are point pairs for the polygon before and after the mapping. And the rotation angle θ can be calculated by,

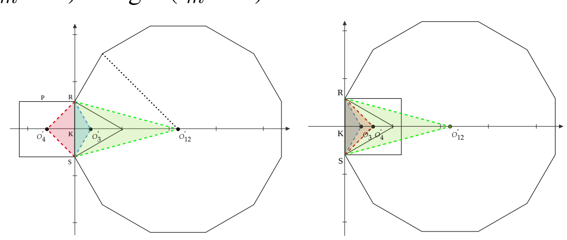
$$\theta = \alpha - \beta, \alpha = \begin{cases} 0 & n = 3 \\ \frac{\pi}{n} & n \neq 3 \end{cases}, \beta = \arccos\left(\frac{V_x}{\sqrt{V_x^2 + V_y^2}}\right) \tag{5}$$

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

Then, we rearranged all basic elements of a tiling to a state that one side of each element overlaps. In default, the overlapping edges and their perpendicular bisector line are placed coincident with y-axis and X-axis respectively. It can be noted that there are 2^n cases for such an arrangement, where n is the number of basic elements. And each case would lead to different fundamental region U, and thus produce very different results, which gives us another DOF to control the pattern. We took the tiling composed of three kinds of regular polygons as an example to explain the detail, where the basic elements include regular triangle, quadrilateral and dodecagon. It can be verified that 8 different arrangements could be made in total, and can be represented as $3^l4^l12^l$, $3^l4^l12^r$, $3^l4^r12^l$, $3^r4^l12^l$, $3^l4^r12^r$, $3^r4^l12^r$, $3^r4^r12^l$, $3^r4^r12^r$, where the digits denote the number of sides for the basic elements, and the superscript letter l and r denote whether the corresponding element is placed on the left (l) or the right (r). Fig. 5 displays two of the 8 cases, i.e., $3^l4^r12^r$ and $3^r4^r12^r$. All cases need to translate each polygon initially located in the center of the coordinate system(Fig. 4) along the x-axis to gain the new arrangement, and the displacement is calculated as,

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

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



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

Figure 5. 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.

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, which is marked as the colored area in Fig. 5.

4.3. Construction of Invariant Mappings

We construct three kinds of invariant mappings associated with general k-uniform tilings without strange steams to enrich the spatial structure of the quasi-regular patterns. And the generated patterns have expected local symmetries, whose texture is continuous across the edges of polygons.

The symmetry of a given tiling is an isometric transformation. And all symmetries of a tiling constitute a symmetry group, which 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 fundamental regions of all its basic elements,i.e., $U = \cup_m U_m$. We would build invariant mappings M upon the fundamental regions U. Generally, the invariant mapping is a transformation that keeps certain property of the object after it is applied. And the goal of the invariant mapping here is to make the mapped region and the fundamental region be congruent. Formally, let T be a symmetry transformation, then for any point (x, y) in the fundamental region, it holds $M(x, y) = M(T(x, y))$.

We take the fundamental region constructed in Fig. 6 as examples to describe how to construct three kinds of invariant mappings, which are denoted as M_1, M_2 and M_3 respectively. Let R_{AB} be the reflection transformation about the line $\overline{AB}$. To construct a symmetry group, we used different composite transformations of R_{AB} as its 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, which consists of m-fold rotations and m-fold reflections. Thus any k-uniform tiling equips with such a kind of symmetry naturally.

Without loss of generality, we took the GLM-type tiling illustrated in Fig. 6(a) as an example to construct the 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. Then the fundamental region corresponding to each polygon is $\triangle O'_3RK, \triangle O'_4RK$ and $\triangle O'_{12}RK$. By definition, for any point in the fundamental region U of the tiling, e.g., $A_1 \in \overline{RK}$ marked in Fig. 6(a), we could obtain its rotational symmetrical points,

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

as well as its reflectional symmetrical point,

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

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

And for any point $(x, y) \in P_m$, we could map it to its fundamental region U_m with the help of dihedral symmetry group D_m as follows,

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

Noting that all the symmetrical points leads to the same coordinate (x', y'), the mapping M_1 defined as $M_1(x, y) = H_q(x', y')$ is hence an invariant mapping, where H_q is the QRP model defined in Eq. (1).

The illustration of the mapping under the dihedral symmetry is shown in Fig. 6(d), where the red double arrows represent the symmetric relationship between the fundamental region U and others. An example of the generated pattern and it's corresponding surface can be seen in Fig. 7(a) and Fig. 7(c).

4.3.2 Invariant mappings with rotational symmetries

An n-fold rotational symmetry is a cyclic group C_n, which is a bit complicated to construct the invariant mapping than the previous case. To elaborate the mapping M_2, we took Fig. 6(b) as an example. In such a mapping, we have to consider the continuity of the seams across both the boundary edges of polygons and the fundamental region U. In Fig. 6(b), let $C_m(m = 3, 4, 12)$ be the cyclic symmetry group 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 them are the counterclockwise rotations about O_m with angle $2\pi/m$. To construct M_2, the fundamental region corresponding to each polygon becomes $\triangle O'_3RS$, $\triangle O'_4RS$ and $\triangle O'_{12}RS$. Then, we could obtain the symmetrical points of $\forall A_1 \in \overline{RS}$ as,

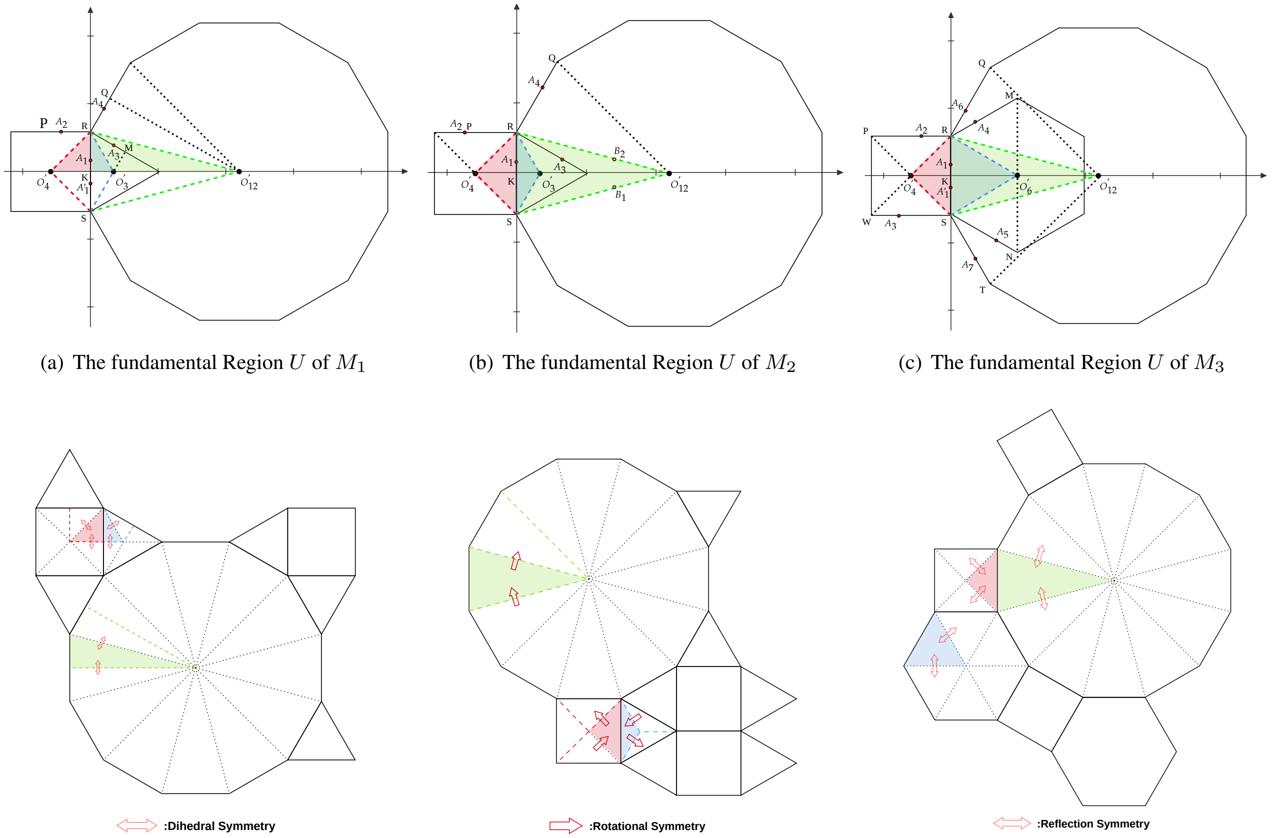
$$\begin{aligned} A_2 &= (\boldsymbol{R}_{O'_4K} \cdot \boldsymbol{R}_{O'_4R})^{-1}(A_1) \in \overline{RM}, \\ A_3 &= (\boldsymbol{R}_{O'_3K} \cdot \boldsymbol{R}_{O'_3R})^{-1}(A_1) \in \overline{RP}, \\ A_4 &= (\boldsymbol{R}_{O'_{12}K} \cdot \boldsymbol{R}_{O'_{12}R})^{-1}(A_1) \in \overline{RQ}, \end{aligned}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

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 fundamental regions for other polygons can be also deduced similarly from Eq. (10) and Eq. (11) by their corresponding generators.

Different from the dihedral symmetry, the definition of rotational symmetry cannot ensure the continuity across the boundaries of the fundamental regions. For symmetry transformation $\gamma_m \in C_m$ defined on P_m, we could map any



(a) The fundamental Region U of M_1

(b) The fundamental Region U of M_2

(c) The 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

Figure 6. 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).

point $(x, y) \in P_m$ to the fundamental region U_m under γ_m and obtain a new point $\mathbf{p} = (x_\gamma, y_\gamma) = \gamma_m(x, y) \in U_m$. We then constructed a composite mapping h applied on γ_m as follows to meet the continuity condition:

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

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

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

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

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

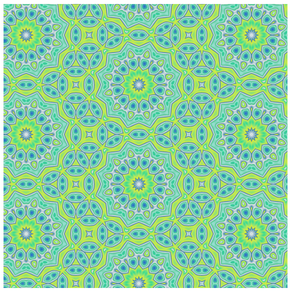
σ is a smooth function used to encourage each edge of the fundamental region U_m to have the same color after the rotation γ_m is applied, and it can be defined as a Sigmoid function:

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

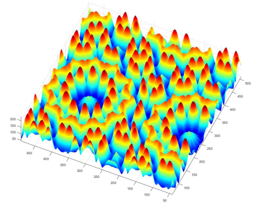
where b and ω are parameters of the linear function about x. It can be easily verified that $r = d_{\text{min}}(\mathbf{p})/d_{\text{sum}(\mathbf{p})} \in [0, 1/3]$. When the point $\mathbf{p} \in U$ approaches ∂U, $r \to 0$ and its mapped point is expected to approach to its symmetrical point about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

Since all the symmetrical points lead to the same coordinate (x', y') by 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 can be seen in Fig. 6(e), and the generated pattern and the surface of its QRP model are shown in Fig. 7(e) and Fig. 7(b).

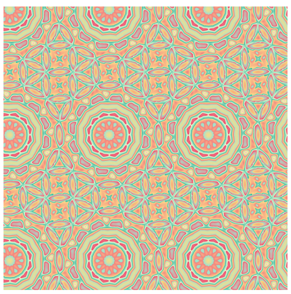
4.3.3 Invariant mappings with reflection symmetries



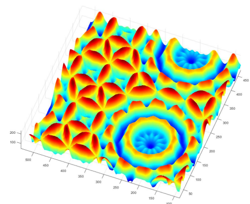
(a) Pattern with GLM tiling



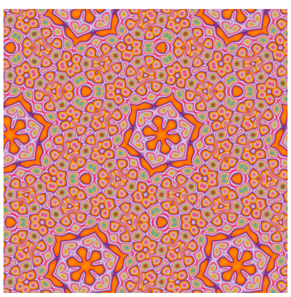
(b) Surface of GLM pattern



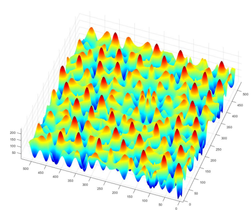
(c) Pattern with GLMT_1 tiling



(d) Surface of GLMT_1 pattern



(e) Pattern with H tiling



(f) Surface of H pattern

Figure 7. The aesthetic patterns of three kinds of invariant Mapping: (a), (b) and (c) 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. (d) (e) and (f) are the surface of pattern corresponding to (a), (b) and (c), respectively.

Patterns generated by using M_3 have local reflection symmetries, whose axes of symmetry are lines that end with the centroid of the polygon and one of its vertices. The construction of M_3 and M_1 shares some similarities. But unlike M_1, M_3 has a larger area of fundamental region and needs to consider the boundary continuity between polygons.

We took Fig. 6(c) as an example to demonstrate the detail of the construction. In Fig. 6(c), let D'_m be dihedral symmetry group of $P_m(m = 4, 6, 12)$. $\boldsymbol{R}_{O'_4R}$ and $\boldsymbol{R}_{O'_4S}$, $\boldsymbol{R}_{O'_6R}$ and $\boldsymbol{R}_{O'_6S}$, $\boldsymbol{R}_{O'_{12}R}$ and $\boldsymbol{R}_{O'_{12}S}$ are the generators of D'_4, D'_6 and D'_{12}, respectively. The fundamental region U corresponding to each polygon is $\triangle O'_4RS$, $\triangle O'_6RS$ and $\triangle O'_{12}RS$. Then, for any point $A_1 \in \overline{RK}$, it's symmetrical points under invariant mapping M_3 lying at P_4, P_6 and P_{12} are as follows:

$$\begin{aligned} A_2 &= \boldsymbol{R}_{O'_4R}(A_1) \in \overline{RP}, \\ A_3 &= \boldsymbol{R}_{O'_4S}(A_1) \in \overline{SW}, \\ A_4 &= \boldsymbol{R}_{O'_6R}(A_1) \in \overline{RM}, \\ A_5 &= \boldsymbol{R}_{O'_6S}(A_1) \in \overline{SN}, \\ A_6 &= \boldsymbol{R}_{O'_{12}R}(A_1) \in \overline{RQ}, \\ A_7 &= \boldsymbol{R}_{O'_{12}S}(A_1) \in \overline{ST}, \end{aligned} \tag{16}$$

Unlike M_1, the construction of M_3 has to eliminate the texture seams on the boundaries of the tilings. To this end, we just constructed a mapping that can make each edge of the 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 $\gamma_n \in D'_m$, i.e., $\mathbf{p} = (x_\gamma, y_\gamma) = \gamma_m(x, y) \in U_m$, and then applied a composite function to make it symmetrical:

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

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

Clearly, M_3 becomes an invariant mapping under the definition of Eq. (17). The illustration is shown in Fig. 6(f), and the generated pattern and it's QRP surface are shown in Fig. 7(d) and Fig. 7(f).

4.4. GPU Implementation

Due to the large amount of pixels, it is inefficient to render the pattern in CPU. We thus took advantages of the multi-pipeline feature of the GPU and rendered all pixels of a pattern in parallel, and applied OpenGL Shading Language (GLSL) for GPU computing.

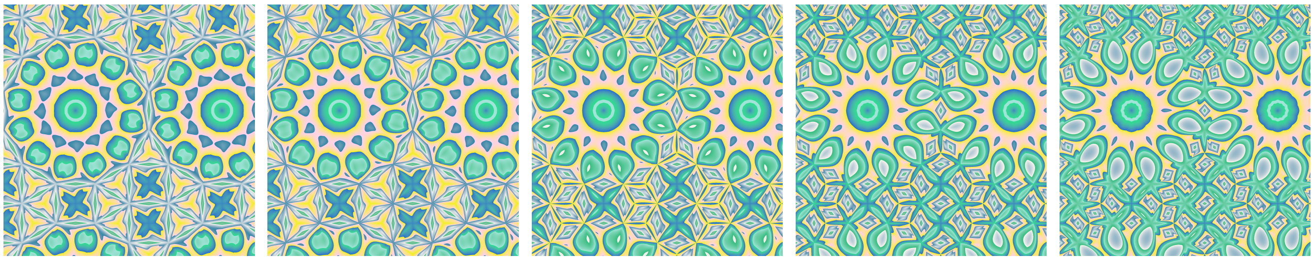
In the vertex shader, the projection matrix is received from the CPU memory. And the grids are translated by the matrix to make the tile covering the whole window, as is shown in Fig. 3.

The fragment shader obtains the vertex information passed in from the vertex shader, as well as other parameters. Substitute these parameters into Eq. (4)-Eq. (6) in Sec.4.2 to construct the fundamental region. At the same time, three different kinds of invariant mappings M_1, M_2 and M_3 as well as predefined QRP models are all implemented in the fragment shader Sec.4.3.

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

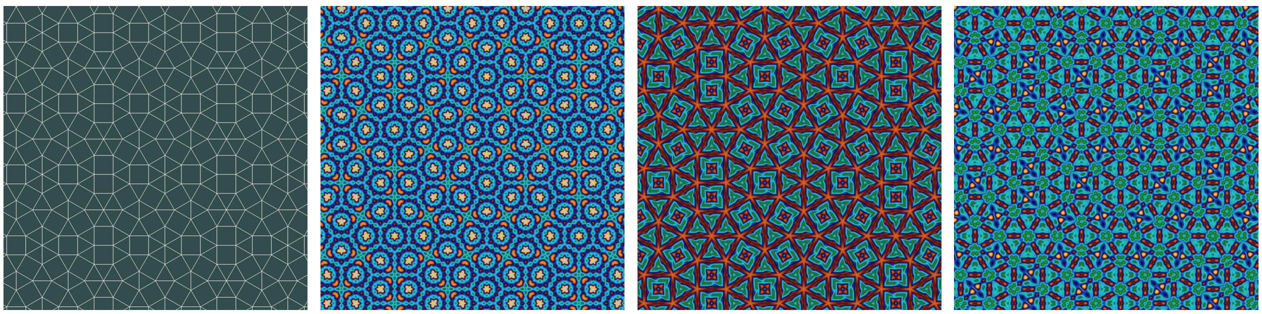
5. Experimental Results

The proposed pattern generation algorithm is implemented using GPU shaders and ran on a PC with Intel Xeon(2.40GHz) processor, NVDIA Quadro P620 (2GB), Windows 10 OS, and all patterns are rendered to image with

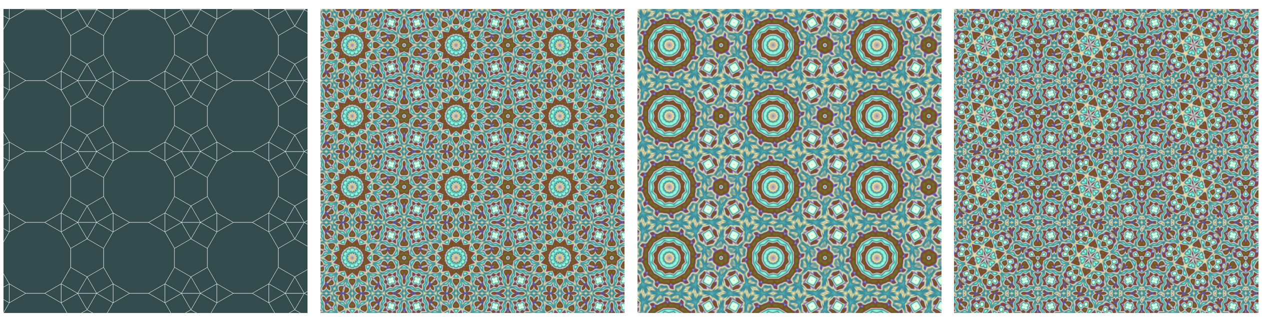


(a) $\eta = 0.2$ (b) $\eta = 0.3$ (c) $\eta = 0.5$ (d) $\eta = 0.7$ (e) $\eta = 1.0$

Figure 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

Figure 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.

a resolution of 1024×1024 pixels. The tiling dataset is collected by [34], which contains 212 different tillings. In default, we set the basic model (Eq. (1)) as the QRP model, with its parameters $q = 5, s = 18, x_t = y_t = 0$. The parameter q originally defined as a positive integer but could be set as any positive real number $q \geq 1$ in practice, which also works well.

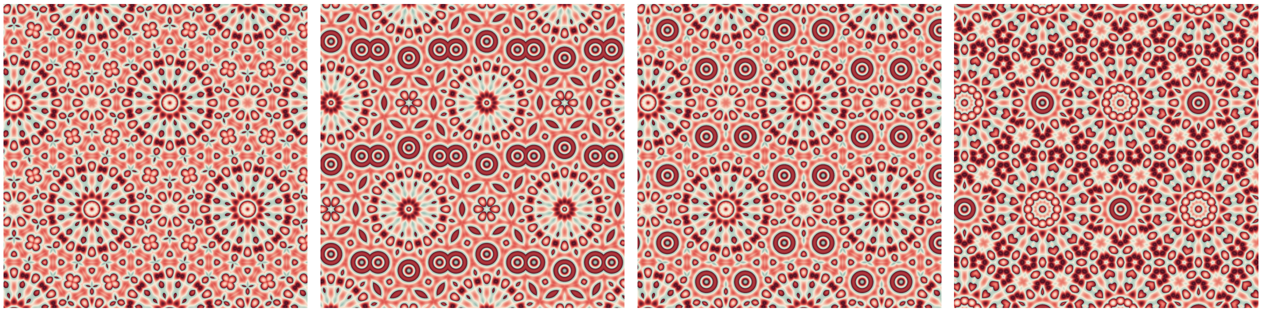
In the following, we experimentally demonstrate the effectiveness of the algorithm with several examples and comparisons from different aspects.

5.1. Influence factors of the algorithm

There are various ways to control the generated patterns for our algorithm, mainly including η in Fig. 8, type of local symmetries, way of overlaying basic polygons, models and parameters of quasi-regular patterns, type of tiling structures, et al. And all these factors affect the results greatly.

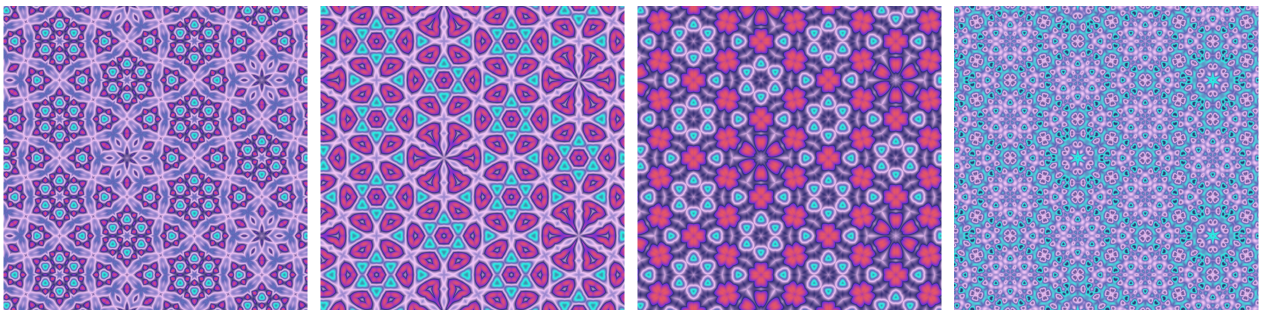
The parameter η in Eq. (12) is used to rescale the local detail of the texture, which is an extra degree to control the variation of the patterns. When $\eta = 1$, the position of the mapped point is kept still, and the pattern displays its original texture detail. And when $\eta < 1$, the position of the mapped point's y-axis is rescaled, and thus the detail of the pattern is meant to be magnified. The smaller the value of η, the larger the magnification, and the texture detail becomes clearer. The results generated by different values of η can be seen in Fig. 8. And we empirically set $\eta = 0.3$ in default.

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

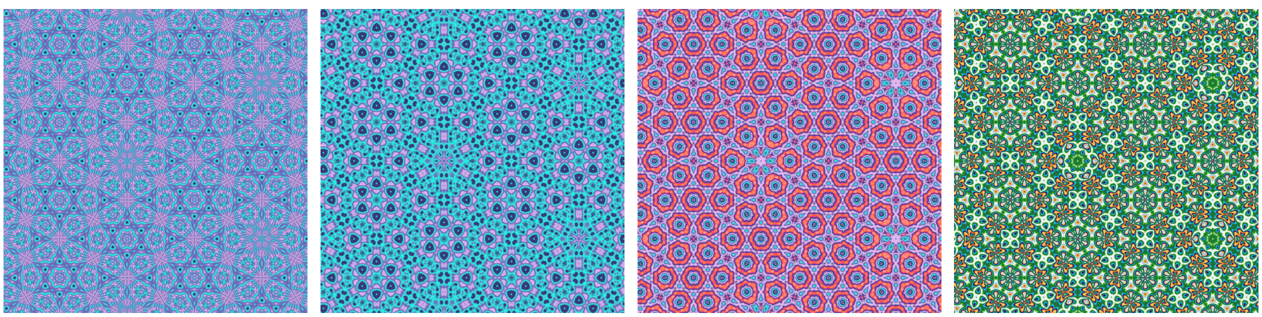


(a) $3^r4^r12^r$ (b) $3^l4^r12^r$ (c) $3^r4^l12^r$ (d) $3^r4^r12^l$

Figure 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)

Figure 11. Patterns generated with different QRP models: (a) basic model(Eq. (1)). (b)-(h) correspond to seven different models listed in Tab. 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

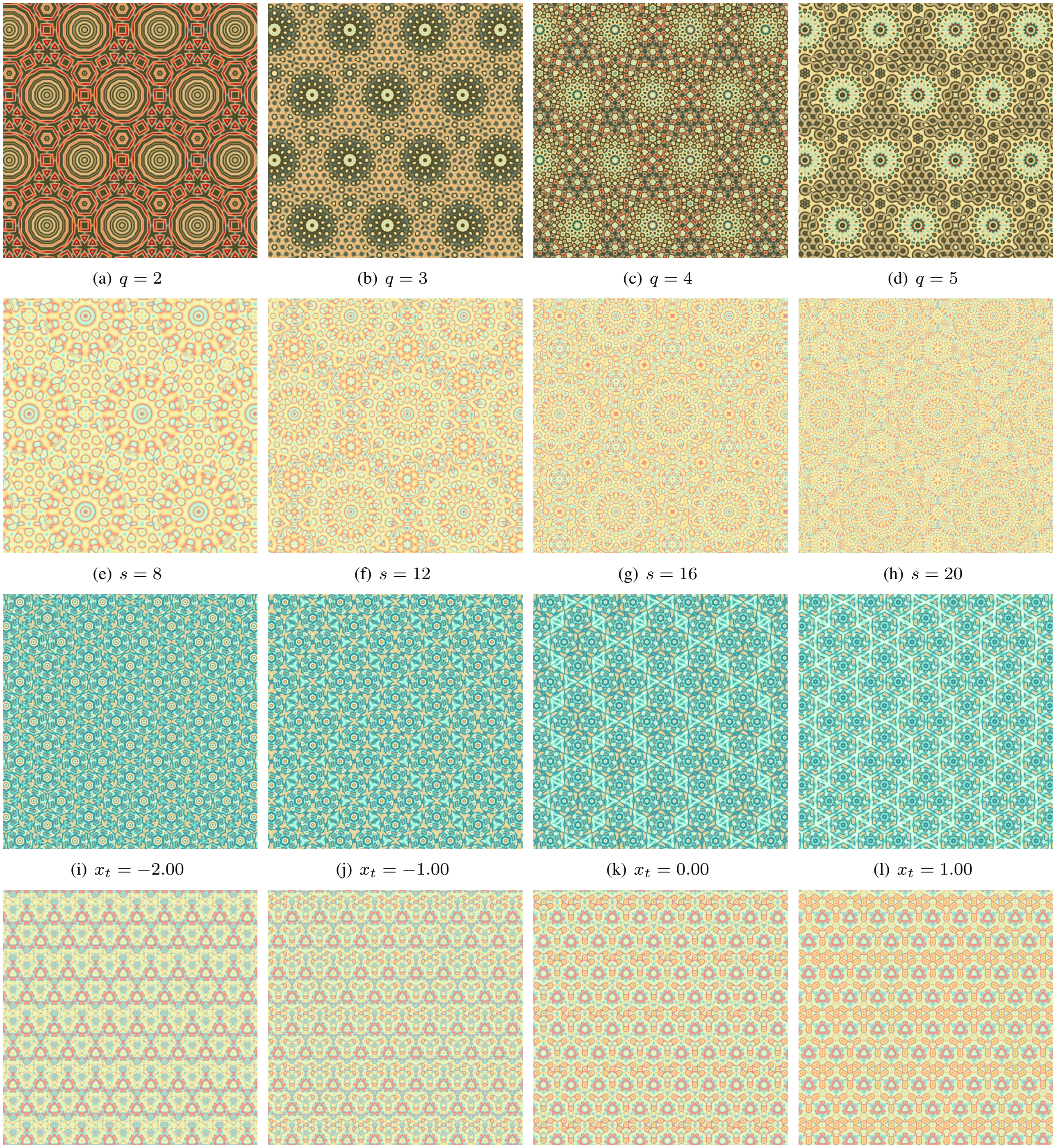
their own art styles respectively and differs greatly from each other in appearance. The reason behind this is that the constructed fundamental region of each type is different, which leads to different shapes of QRP defined in that region.

The third example in Fig. 10 shows the effectiveness of different ways to rearrange basic elements of the tiling. We used the same tiling structure (HLMP), QRP models, and local symmetry type (dihedral) but different rearrangements of basic elements. Results vividly shows that all of them share some similarities in global structure, but differs in local texture details. This is due to the fact that tiling structure and local symmetry type reflect the global features and the QRP model determines the local texture features.

Table 1. QRP models used in the experiments

QRP Models	e.g
$H = \sum_{i=1}^{\lfloor q \rfloor} \cos\left[x\cos^3\left(\frac{2\pi i}{q}\right) + y\sin^3\left(\frac{2\pi i}{q}\right)\right]$	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: $\Omega = x\cos\left(\frac{2\pi i}{q}\right) + y\sin\left(\frac{2\pi i}{q}\right)$



(a) $q = 2$ (b) $q = 3$ (c) $q = 4$ (d) $q = 5$

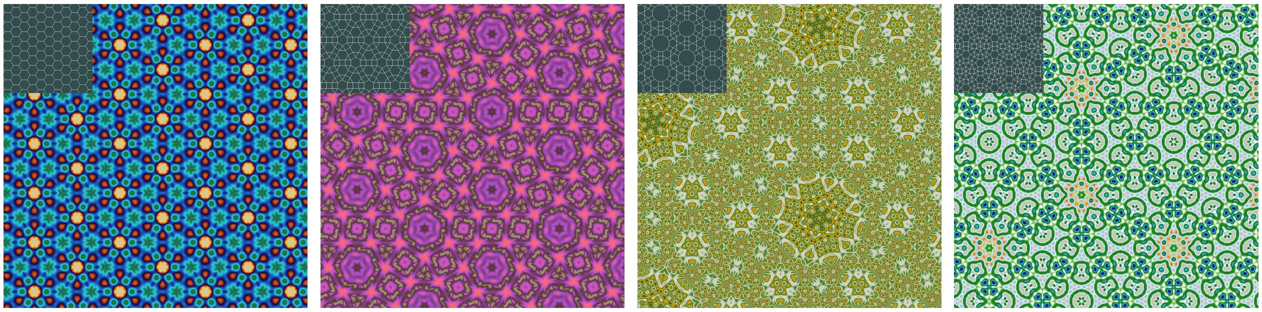
(e) $s = 8$ (f) $s = 12$ (g) $s = 16$ (h) $s = 20$

(i) $x_t = -2.00$ (j) $x_t = -1.00$ (k) $x_t = 0.00$ (l) $x_t = 1.00$

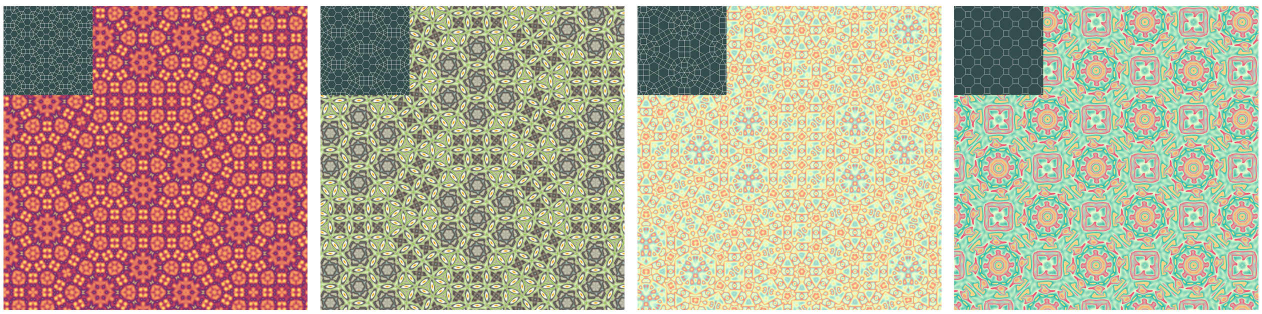
(m) $y_t = -2.00$ (n) $y_t = -1.00$ (o) $y_t = 0.00$ (p) $y_t = 1.00$

Figure 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.

Next, we show examples to demonstrate the colorful texture pattern of QRP models and its parameters with all the remaining factors fixed. Fig. 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 Tab. 1. Thanks to our scheme of building the fundamental region, any QRP model defined is contin-

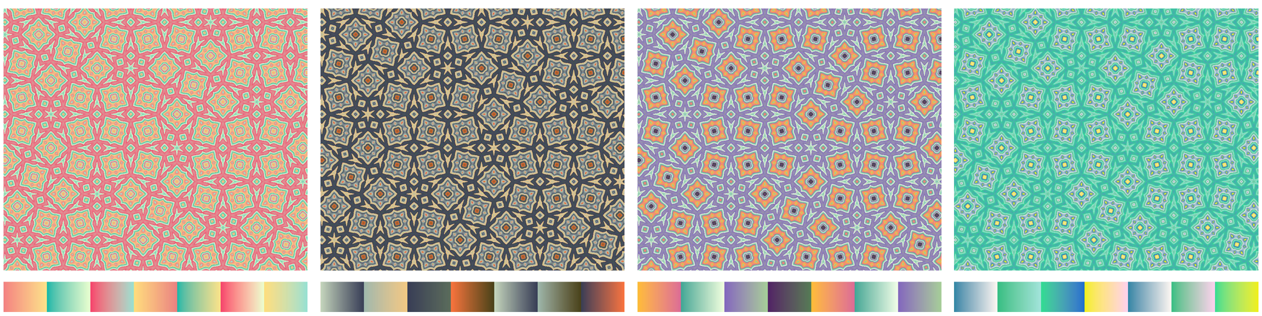


(a) one vertex type (b) two vertex types (c) three vertex types (d) four vertex types

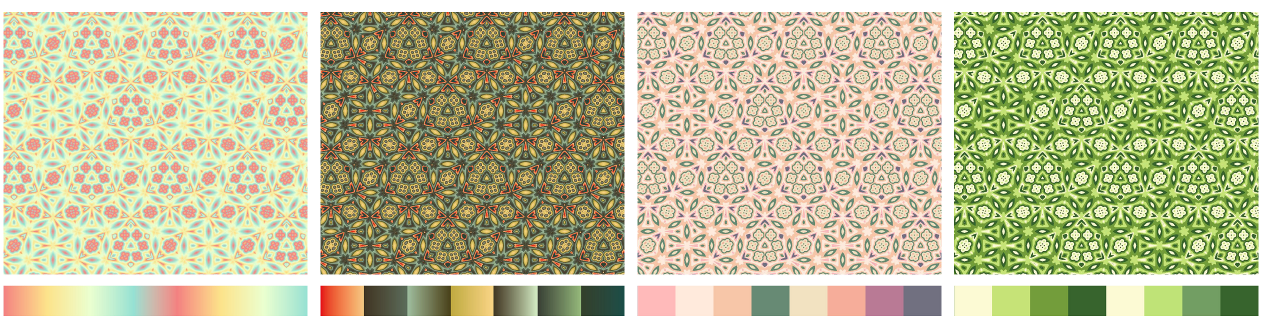


(e) five vertex types (f) six vertex types (g) seven vertex types (h) special vertex type

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



(a) Palette 1 (b) Palette 2 (c) Palette 3 (d) Palette 4



(e) 5000 colors (f) 1000 colors (g) 8 colors (h) 4 colors

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

uous across the edges of the tiling, which naturally eliminates the texture seams on those edges. And the patterns generated with different QRP models (Fig. 11) differs dramatically from each other both in local textures and global structures. We also investigated the parameters (q, s, x_t, y_t) of the QRP model and the corresponding results are shown in Fig. 12. Since the parameter q reflects the complexity of the model, the local texture detail becomes more complicated as q increases (Fig. 12(a)- Fig. 12(d)). Parameters x_t, y_t confine the domain of definition and thus affect the contours' shape of the QPR models. The texture details of the pattern undergo changes as these two parameters vary (Fig. 12(e)- Fig. 12(l)). s is used to control the scale of the texture pattern, and the size of the texture decreases as the value of s increases. The translation period gradually increases for the rendered image with the same resolution (Fig. 12(m)- Fig. 12(p)).

The tiling structure determines the layout of the pattern. Moreover, the fundamental regions are different for different tiling structures and therefore the QRP models defined therein are distinct. Fig. 13 displays pattern generation results with different tiling structures of different number of vertex types while keeping the remaining factors unchanged. It can be seen from the figure that tiling structures greatly affect the global features as well as local details of the generated patterns.

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

5.2. Comparisons

In this section, we demonstrated the advantages of our algorithm over the state-of-the-art methods by comparisons.

Dynamical system is one of the most popular tools used to generate aesthetic patterns with symmetry property. Thus, we first compared it to those generating patterns with simple planar symmetries from dynamical systems, e.g. [24, 2]. The results can be seen in Fig. 15(a)–Fig. 15(h). Note in general, their patterns are regular and monotonous due to the simple wallpaper symmetric layout. Besides, the methods [24, 2] requires to design specific functions with symmetry property for the dynamical system models, which is difficult and tedious work. By comparison, it is flexible for users to control the pattern from various aspects with our method and the generated patterns have variety in both space structures and textures. And meanwhile

We also compared with tilings-constrained pattern generation methods based on dynamical systems [14, 15]. We implemented both the algorithms and presented their results in Fig. 16 and Fig. 17. Comparisons show that our algorithm outperforms others [14, 15] in several aspects. Firstly, ours can generate patterns for any k-uniform tiling while the methods [14, 15] are limited to some simple types of tilings (Penrose and Archimedean) and should design invariant mappings for each tiling case by case. Secondly, ours eliminates the seams across the tiling edges naturally and thus the generated patterns are rich in variation near these edges. However, both the methods [14, 15] need to construct explicit continuous conditions across the tiling edges and visible seems can be usually seen in their patterns. Thirdly, the QRP model used in our algorithm has several fixed and meaningful parameters which can conveniently control the variation of the pattern. But the parameters of the dynamical models are integrated into the functions, which have different geometrical significance for different functions. Thus, one has to frequently change the function models in order to obtain patterns with different styles.

5.3. Performance

Table 2. Graphics generation speed at different resolutions

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

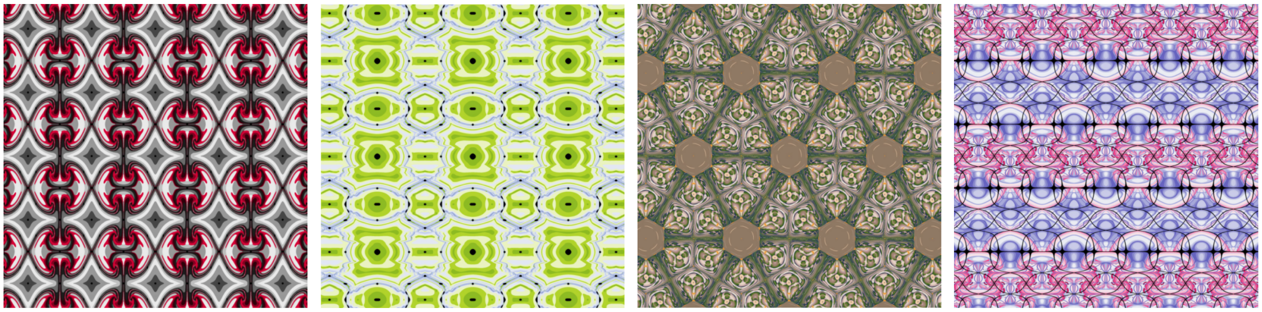
Thanks to the parallel implementation on GPU, our algorithm achieves high performance even for rendering images of large resolutions. Tab. 2 shows the time statistics for rendering images of different resolutions. Note that the cost time nearly keeps the same as the resolution increases.

5.4. Application

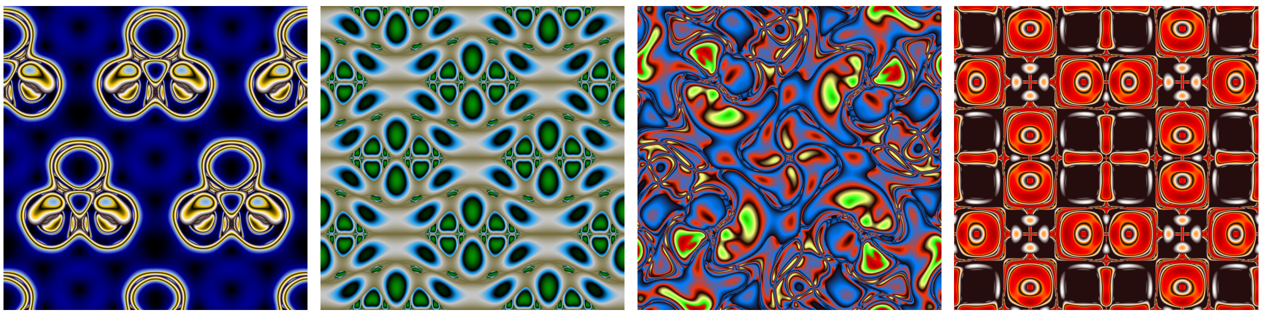
The patterns generated by our method could be used as materials in some design fields, such as pattern design in fabrics and ties. Here we showed an application of such patterns used in tie design Fig. 18. Since the style and feature of our patterns have similarities with that of ties, slight alterations are needed for final applications, which could greatly assist the artists to improve the efficiency of design flow.

6. Conclusions

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

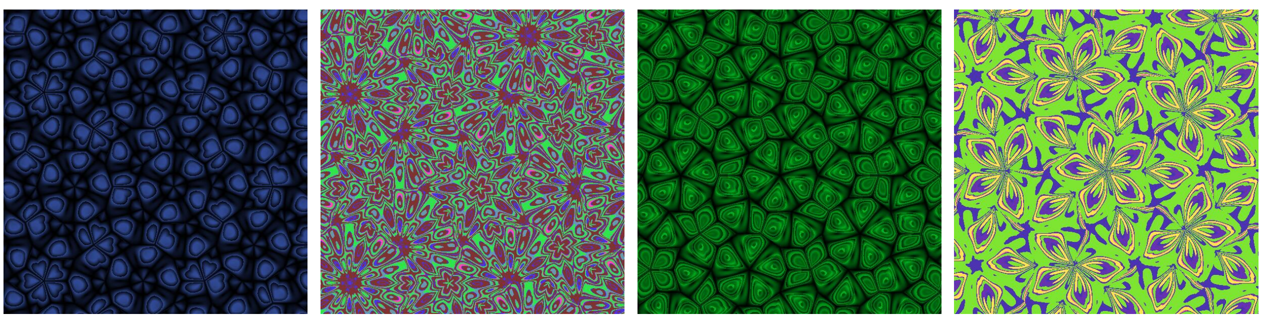


(a) (b) (c) (d)



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

Figure 15. Symmetrical patterns generated from dynamical systems.(a)-(d): Results of the method [24]; (e)-(f): Results of the method [2]. (All the figures are extracted from the paper [24] and [2])



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

Figure 16. Patterns generated with Penrose tilings [14]

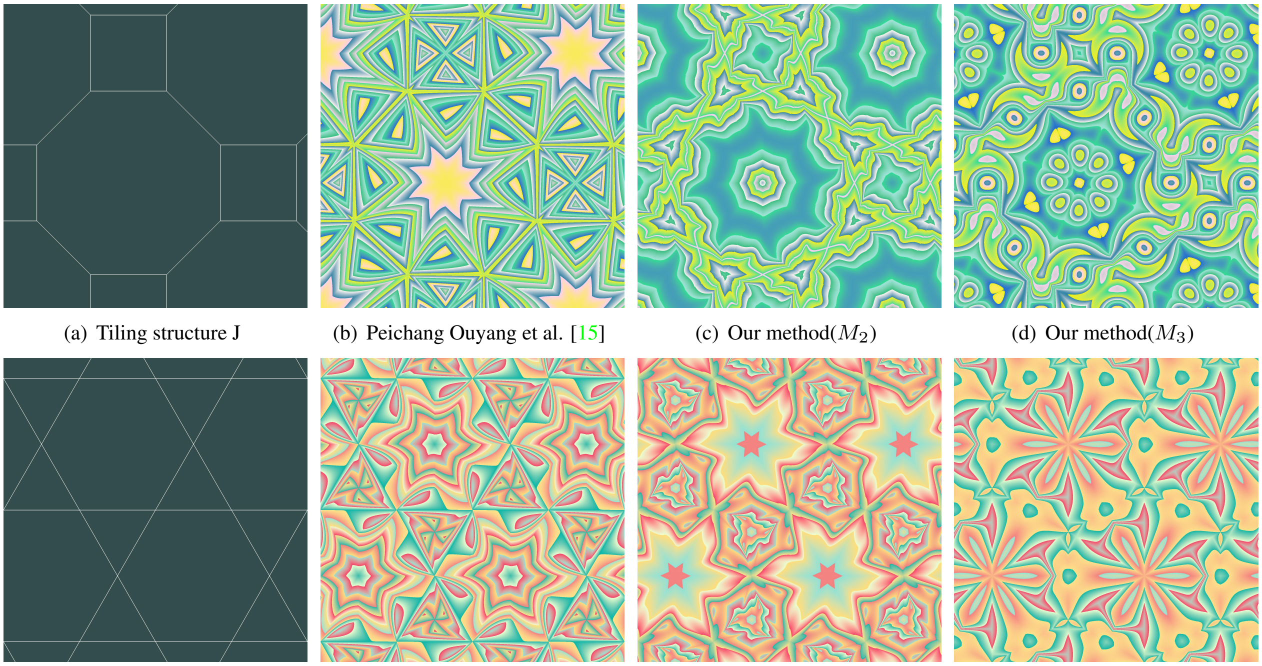
Although the overall performance of our method is promising, it still has a few limitations. we did not take into account the aesthetic evaluation of specific domains for the generated patterns, which may sometimes hinder practical applications in design. There are mainly two important factors affecting the aesthetics. One is the strategy for partitioning the height filed of QRP models that has a great impact on the shape of the generated pattern. The other one is the color compatibility and its spatial arrangements, which always requires artists to carry out secondary design for practical applications. In the future, we plan to integrate aesthetic model learnt from the specific art design field into the method, and study the method to generate quasi-regular patterns with the constraints of aperiodic tilings.

Acknowledgements

This work was supported by Key R&D Programs of Zhejiang Province (NO. 2023C01224, 2022C01220) and National Natural Science Foundation of China (No. 61702458). Yun Zhang was partially supported by the Zhejiang Province Public Welfare Technology Application Research (No. LGG22F020009), Key Lab of Film and TV Media Technology of Zhejiang Province (No. 2020E10015), and Teaching Reform Project of Communication University of Zhejiang (No. jgxm202131).

References

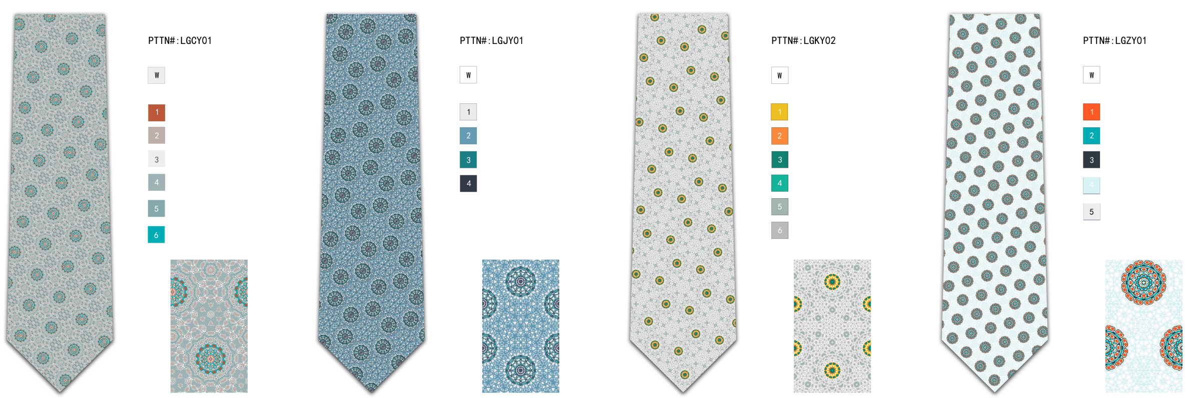
[1] P. Ouyang, K. W. Chung, A. Nicolas, and K. Gdawiec. Self-similar fractal drawings inspired by mc escher's print square limit. *ACM Transactions on Graphics (TOG)*, 40(3):1–34,



(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)

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



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

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

2021. 1

[2] K. Gdawiec and H. Adewinbi. Procedural generation of artistic patterns using a modified orbit trap method. *Applied Sciences*, 12(6):2923, 2022. 1, 2, 13, 14

[3] Z. Sayed, H. Ugail, I. J. Palmer, J. Purdy, and C. Reeve. Auto-parameterized shape grammar for constructing islamic geometric motif-based structures. *Transactions on Computational Science*, 28:146–162, 2016. 1, 2

[4] B. B. Mandelbrot and M. Aizenman. Fractals: form, chance, and dimension. *Physics Today*, 32(5):65–66, 1979. 1, 2

[5] G. Helt. Extending mandelbox fractals with shape inversions. In *Proceedings of the Bridges 2018: Mathematics, Art, Music, Architecture, Education, Culture*, pages 547–550, 2018. 1

[6] M. Szyszkowicz. Computer art from numerical methods. *Computer graphics forum*, 10(3):255–259, 1991. 1, 2

[7] K. Gdawiec, W. Kotarski, and A. Lisowska. Automatic generation of aesthetic patterns with the use of dynamical systems. In *Advances in Visual Computing: 7th International Symposium, ISVC 2011, Las Vegas, NV, USA, September 26-28, 2011. Proceedings, Part II 7*, pages 691–700. Springer, 2011. 1

[8] G. M. Zaslavsky, R. Sagdeev, D. Usikov, and A. Chernikov. *Weak chaos and quasi-regular patterns*. Cambridge University Press, 1992. 1, 3

[9] Y. Zhang and D. Li. Researches on fabric patterns designing with visual information in non-linear dynamical systems (in chinese). *Journal of Textile Research*, 23(5):27–28, 2002. 1, 2, 3

[10] S. Liu. The generation system of textile pattern draft based on quasi-regular pattern theory. *International Journal of Information Engineering and Electronic Business*, 1(1):50–57, 2009. 1, 2

[11] Y. Zhang. *The art of quasi-regular patterns (in Chinese)*. China Textile Press, 2017. 1, 2

[12] K. W. Chung and H. S. Y. Chan. Symmetrical patterns from dynamics. *Computer Graphics Forum*, 12(1):33–40, 1993. 2

[13] J. Lu, Z. Ye, and Y. Zou. Automatic generation of colorful patterns with wallpaper symmetries from dynamics. *The Visual Computer.*, 23(6):445–449, 2007. 2

[14] K. W. Chung and B. Wang. Automatic generation of aesthetic patterns on aperiodic tilings by means of dynamical systems. *International Journal of Bifurcation and Chaos*, 14(09):3249–3267, 2004. 2, 3, 4, 5, 13, 14

[15] P. Ouyang, W. Zhao, and X. Huang. Beautiful math, part 5: colorful archimedean tilings from dynamical systems. *IEEE computer graphics and applications*, 35(6):90–96, 2015. 2, 3, 4, 5, 13, 15

[16] J. E. S. Sánchez, T. Weyrich, A. M. e Sá, and L. H. de Figueiredo. An integer representation for periodic tilings of the plane by regular polygons. *Computers & Graphics*, 95:69–80, 2021. 2, 4

[17] L. Gieseke, P. Asente, R. Měch, B. Benes, and M. Fuchs. A survey of control mechanisms for creative pattern generation. *Computer Graphics Forum*, 40(2):585–609, 2021. 2

[18] T. H. Speller, D. Whitney, and E. Crawley. Using shape grammar to derive cellular automata rule patterns. *Complex Systems*, 17(1/2):79, 2007. 2

[19] K. Gdawiec. Fractal patterns from the dynamics of combined polynomial root finding methods. *Nonlinear Dynamics*, 90(1):2457—-2479, 2017. 2

[20] Y. Zhang and Y.-y. FU. Fabric pattern design based on quasi-regular pattern theory (in chinese). *Journal of Textile research*, 26(5):58–62, 2005. 2

[21] F. Jia and Y. Zhang. Methods of clothing pattern designing based on quais-regular patterns with shadow effect. *Journal of Textile Research*, 38(7):124–129, 2017. 2

[22] N. C.Carter, R. L.Eagles, S. M.Grimes, A. C.Hahn, and C. A.Reiter. Chaotic attractors with discrete planar symmetries. *Chaos Solitons & Fractals*, 9(12):2031–2054, 1998. 2

[23] Y. Zou, W. Li, J. Lu, and R. Ye. Orbit trap rendering method for generating artistic images with cyclic or dihedral symmetry. *Computer & Graphics*, 30(3):470–473, 2006. 2

[24] K. Gdawiec. Procedural generation of aesthetic patterns from dynamics and iteration processes. *International Journal of Applied Mathematics and Computer Science*, 27(4), 2017. 2, 13, 14

[25] S. Liu, M. Leng, and P. Ouyang. The visualization of spherical patterns with symmetries of the wallpaper group. *Complexity*, 2018:1–8, 2018. 2

[26] P. Ouyang, R. W. Fathauer, K.-w. Chung, and X. Wang. Automatic generation of hyperbolic drawings. *Applied Mathematics and Computation*, 347:653–663, 2019. 2

[27] K. Chung, H. Chan, and B. Wang. Automatic generation of nonperiodic patterns from dynamical systems. *Chaos, Solitons & Fractals*, 19(5):1177–1187, 2004. 3

[28] P. Ouyang, X. Tang, K. Chung, and T. Yu. Spiral patterns of color symmetry from dynamics. *Nonlinear Dynamics*, 94(1):261–272, 2018. 3

[29] P. Ouyang and R. W. Fathauer. Beautiful math, part 2: aesthetic patterns based on fractal tilings. *IEEE Computer Graphics and applications*, 34(1):68–76, 2014. 3

[30] P. Ouyang, H. Yi, Z. Deng, X. Huang, and T. Yu. Boundary dimensions of fractal tilings. *Fractals*, 23(04):1550035, 2015. 3

[31] X. Wang, P. Ouyang, K. Chung, X. Zhan, H. Yi, and X. Tang. Fractal tilings from substitution tilings. *Fractals*, 27(02):1950009, 2019. 3

[32] A. M. e Sá, L. H. de Figueiredo, and J. E. S. Sánchez. Synthesizing periodic tilings of regular polygons. In *2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 17–24. IEEE, 2018. 3

[33] C. S. Kaplan. Introductory tiling theory for computer graphics. *Synthesis Lectures on Computer Graphics and Animation*, 4(1):1–113, 2009. 6

[34] R. Sá. O livro sobre malhas arquimedianas, 2017. https://www.cos.ufrj.br/seminarios/2018/slides/asla.pdf. 9