Pattern Design Using Cellular Automata and an Iterative Relocation System

Jae Kyun Shin, jkshin@yu.ac.kr
Seung Ryul Choi, srchoi@yu.ac.kr

School of Mechanical Engineering, Yeungnam University
Kyongsan, 712-749, South Korea

Abstract

We introduce a two-step method for generating decorative images. In the first step, a special kind of 2D Cellular Automata (CA) is used as an engine for generating various patterns in pixel format. In the second step, an iterative relocation system is applied to convert the pixel based pattern into a mesh based system. Example patterns show the power of the combined method. Also we introduce another way of using the mesh based system: a concept of magic canvas that maps simple figure into complex decorative patterns.

Introduction

The application of cellular automata as a tool for a computerized pattern design system is promising because CA can be interpreted as working on the smallest element of patterns, i.e. the pixel. We know famous patterns, such as snowflakes or mollusk skin patterns, for example, generated from CA. Many of these CA patterns were found by chance, not through a systematic application of the CA. However, such chance discoveries were enough to show the diversity of patterns that CA can generate.

The authors are trying to develop a generative art system based on a special class of CA, called a one-rule firing CA [1]. While a traditional CA fires and updates all the applicable cells in the system, only a subset of the applicable cells fires in the one-rule firing CA. A pattern generated from the one-rule system will be called an F-pattern, for convenience. One of the merits of the F-patterns is that the pattern can be completely defined by a series of numbers, called its F-code. For example, F=1 defines a Packard snow crystal or a line, depending on the cell type and neighborhood definition. The pattern can be decoded without a rule table.

Patterns from One-Rule Firing Cellular Automata

We can generate thousands of F-patterns in a couple of hours, using typical PCs today. We can control the category of patterns to be generated, simply by defining the number of symmetries and the number of colors used for the patterns. Figure 1 shows single F-patterns generated under different categories. F-codes for each of the patterns are quoted here simply to have an idea how an F-code looks like. In actuality, an F-code can be considered as a convenient way of storing rule table for a pattern. A complete rule table, for example, for a 6-colored CA with Moore neighborhood, will be composed of 6^8=1,679,616 individual rules. However, the number of rules actually used for a typical pattern will be less than 100. The F-code stores the small fraction of the rule table actually used for a pattern. The number of digits in an F-code corresponds to the number of rules actually used for the pattern.

While there are almost an infinite number of F-patterns, most of them are monotonous. For a generative art system, we need a method to create interesting patterns in a systematic way, not by simple trial and error. An F-pattern generally changes dramatically even at a single change of digit in its F-code, making it difficult to control the F-pattern through its F-code.

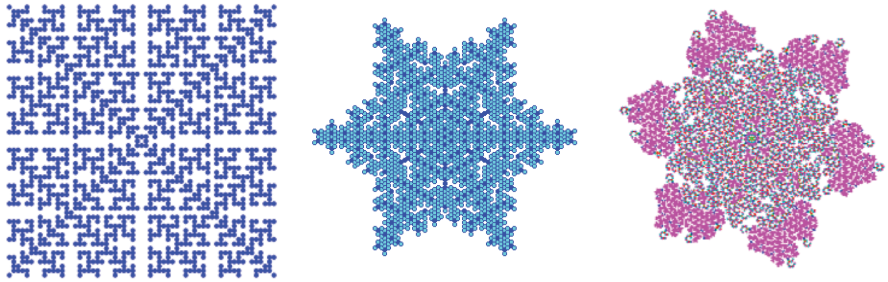


Figure 1: *Example of F-patterns from one-rule firing CA.*
(left) F=1(square cell) (middle) F=20212020 (right) F=652040430032650204605024105

To make a system more fruitful, a method of mixing multiple F-codes for a pattern was previously proposed [2]. Two different ways of mixing F-patterns were proposed. The first way is to mix several F-patterns time-step-wise. Starting from an initial seed, a pattern is grown up to time-step 100, for example, using pattern F_1. From time step 101 to 201, another pattern F_2 is applied, and so on. Second way of mixing is a hierarchical one. In a hierarchical mixing, an F-pattern F_1 is grown first to a certain time-step. And next, the canvas is expanded with respect to the center of canvas to get a sparsely populated canvas of isolated pixels. Then a second F-pattern F_2 is grown on each of these isolated pixels. Figure 2 shows various patterns generated through hierarchical mixing.

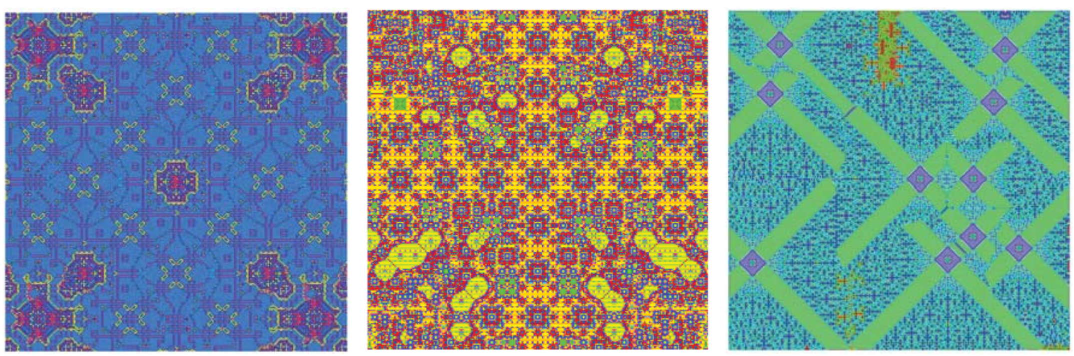


Figure 2: *CA patterns obtained by a rule-mixing technique*.

Iterative Relocation System

While CA based patterns can be mixed for a more diverse patterns, all the patterns are given in pixel format and have several limitations. For an increased scalability and for a more diverse application of the F-pattern, we introduce a method of iterative relocation. For the method, we start from pixel-based images (see Fig. 3(a)). A pixel-based image is interpreted as a grid system composed of nodes and edges. Initially, the nodes are located on each of the pixels and edges connect neighboring nodes. Thus the initial grid system constitutes a system of uniform net composed of unit square patches (Fig. 3(b)). On each of the nodes is associated a mass corresponding to the color codes of the pixels. From the initial uniform net, the nodes iteratively relocate their positions under an interaction with the neighboring nodes. When the position of *i*-th node at time-step *t* is denoted as $\mathbf{r}_i^t$, its position at time-step *t*+1 will be defined as the following equation.

$$\mathbf{r}_i^{t+1} = \sum_{j \in B_i} \omega_j \mathbf{r}_j^t$$

In the equation above, B_i denotes index set of neighbors of node i . The weight ω_j is defined in terms of the color codes of the set B_i. Throughout the present study, Moore neighborhood is used and a node has 8 neighbors except the boundary nodes. At a given time step, a node relocates its position, for example, to the center of mass of the neighboring nodes. The update is simultaneous throughout all the nodes and boundary conditions are applied along the edges of the canvas. When the nodes move, the shape of the patches changes, in general, from unit square to quadrilateral. It is because the amount of move is not identical in general. As the relocation process continues, the initially straight grid lines turns into natural looking curved lines as shown in Fig. 3(c).

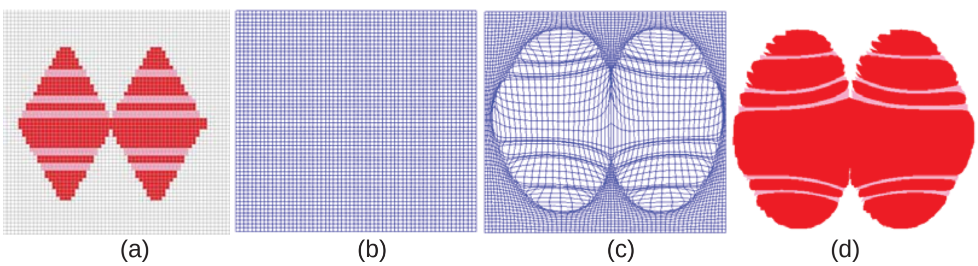


Figure 3: *(a) A simple pixel image drawn on sketchpad (60x60 pixel).*
(b) Uniform mesh system with 60x60 vertices. (c) Deformed mesh system. (d) Developed image.

The iterative relocation system can be applied for any pixel-based images. Especially when applied to the images generated from CA, the method can generate interesting patterns as shown in Fig. 4. For Fig. 4, CA images from rules mixing technique are used.

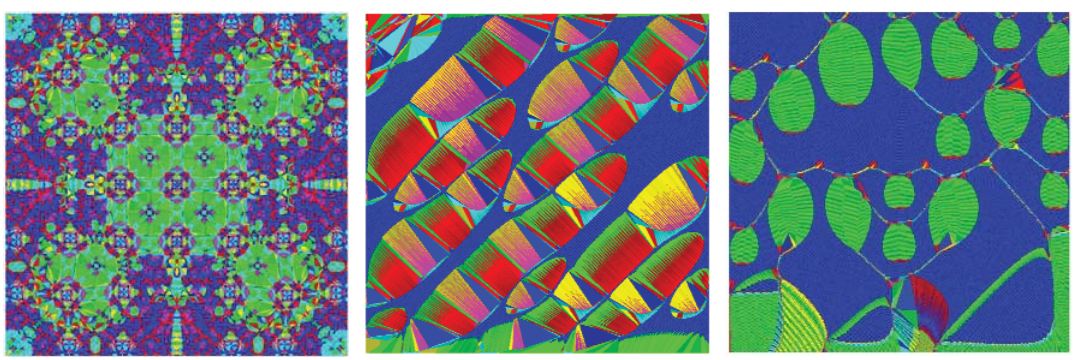
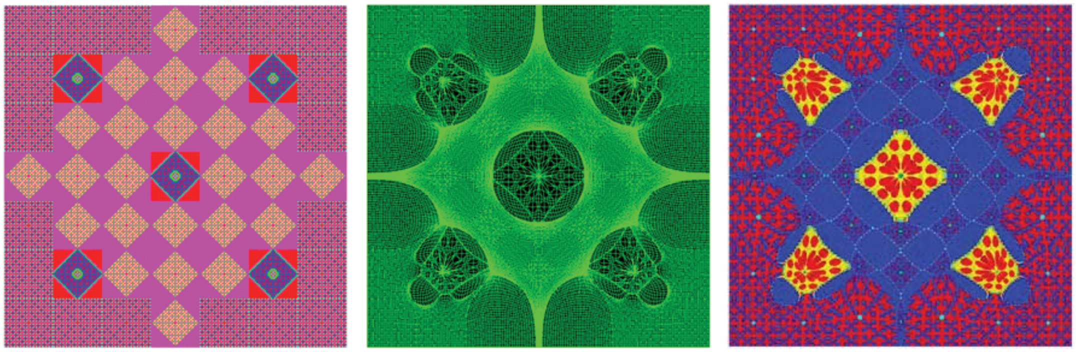


Figure 4: *Decorative patterns obtained applying iterative relocation system.*

One of the merits of the iterative relocation system is that the method can stitch several CA based patterns together to make a combined pattern. Figure 5 shows an example. Three component patterns of same size in pixel-format are arranged in a 7x7 array. Because of the seam lines between the component patterns, we can easily identify component patterns in Fig. 5(a). In the final quilt obtained (Fig. 5(b) and (c)), we cannot easily identify the seam lines.

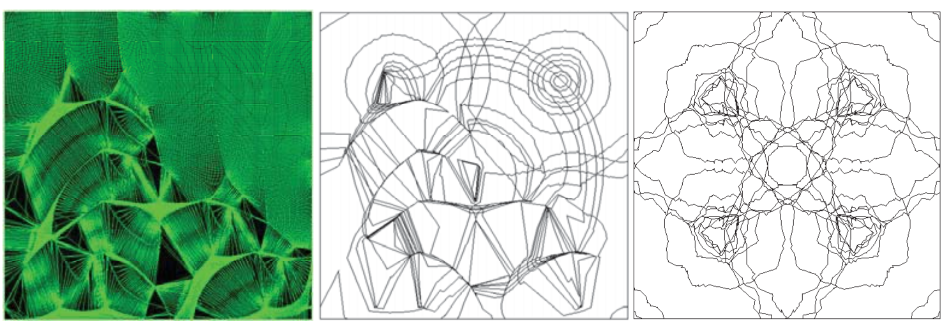


(a) (b) (c)

Figure 5: *Quilt made of component patterns*
(a) 3 different component CA patterns arranged in 7x7array.
(b)(c) Developed pattern through iterative relocation system.

Concept of Magic Canvas

The grid system developed through the iterative relocation system expresses images in terms of the vertical and horizontal lines. The grid system developed can be used for secondary images expressed in terms of closed curves. For example, if we draw a unit circle in Fig. 3(b), we can easily imagine that we will get a distorted image in Fig. 3(c). Figure 6 shows images obtained through transforming sets of concentric circles drawn in the initial uniform net. Initial uniform net is not shown here for simplicity. When we see a circle through a distorted glass, the circle will distort its shape. Like the distorted glass, the developed grid system distorts the image. It is like drawing images on a magic canvas.



(a) (b) (c)

Figure 6: *Concept of magic canvas.*
(a) Grid system obtained from iterative relocation system
(b) Mapping of four sets of concentric circles (c) Another image obtained from the same method.

References

[1] Shin, J. “Identifying patterns from one-rule firing cellular automata”, *Artificial Life 17(*1), 21-32, 2011.
[2] Shin, J. ”Application of cellular automata for a generative art system”, *Leonardo,* doi:10.1162 /LEON _a_00964, 2014.