BRIDGES
Mathematical Connections
in Art, Music, and Science

The Art and Science of Symmetric Design

Michael Field
Department of Mathematics
University of Houston
Houston, TX 77204-3476
E-mail: mf@uh.edu
URL: nothung.math.uh.edu/~mike/

Abstract

We describe some of the issues – both philosophical and technical – involved in the evolution of computer software used for the design and realization of aesthetically appealing symmetric patterns. We illustrate how the choice of algorithm can dramatically affect the final design.

1. Introduction

The images shown in this paper were all designed and constructed using a software package called *prism* (an acronym for PRograms for the Interactive Study of Maps) that I started to develop about eleven years ago. In recent years I have used *prism* as the basis for a course on 'Patterns, Designs and Symmetry' that I teach to Junior/Senior level art students at the University of Houston, and also in a seminar for teachers held under the auspices of the *Houston Teachers Institute.* These teaching experiences, and the possibilities for using this computer software in the classroom, are described in more detail in [3] and, more especially, [4]. Although working with *prism* in a teaching situation has often provided a strong stimulus to make the interface more user friendly, the main impetus for developing *prism* has always been my personal interest in finding new and effective ways to design, color and realize attractive symmetric patterns.

In this article, I want to share some speculations and experiences generated by recent attempts to produce 'art' from silicon and chaos. All of the images that are shown result from an ongoing project to develop a library of algorithms for the design and artistic realization of the one- and two-color wallpaper patterns. Necessarily, the pictures shown here are grey scale versions of the color originals[1]. For this reason, the emphasis will be on algorithms for *one*-color designs rather than on coloring algorithms for 2-color designs. Throughout, we shall follow the standard notational conventions for wallpaper patterns as described, for example, in Washburn and Crowe [6]. We refer to [1,5] for general background on symmetry and the use of methods from chaos and dynamical systems in design. For 2-color designs, see [1,2]. Many colored images can be found at the *URL:* nothung.math.uh.edu/~mike/. Colored versions of some of the images shown in this article may be found at the *URL:* nothung.math.uh.edu/~mike/bridges2000.

In Figure 1, we show a characteristic wallpaper pattern designed using *prism*. I feel that the development of the program that led to the creation of this image as an *integral* part of the

[1]The images prepared for the article were, however, *colored* in grey scale rather than being transformed into grey scale from the original color image.

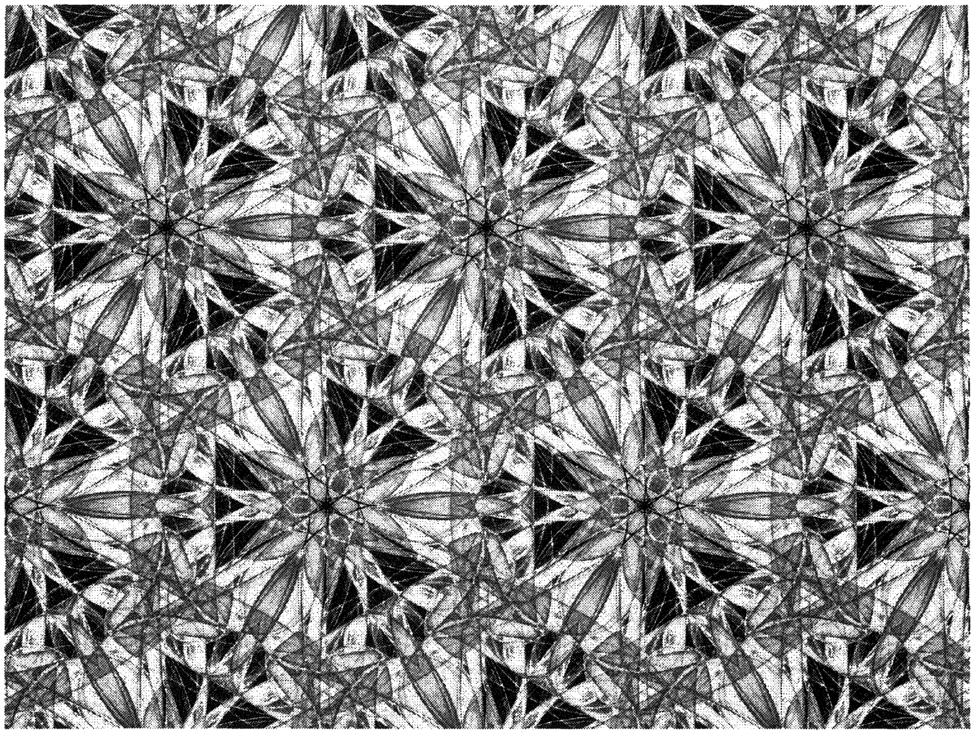

Figure 1: *A wallpaper pattern of type* **p31m**

creative process. That is, the mathematical algorithms, the computer program, and the design and reproduction (realization) of the images are just different facets of the same underlying process. In this regard, I am not so interested in creating pictures using a generic graphics software package. For me, the process of creation and design involves developing the tools and techniques in parallel with their application. Indeed, the development of these tools is largely governed by the character of the image I want to design and the way in which I want to realize the final image (for example, on photographic paper). Of course, sculptors and artists have always developed their own technology – whether it be pigments or the engineering and machining of contemporary sculpture. In spite of the apparent demise of the Renaissance man and the increasing compartmentalization of knowledge, I believe a creative artist – or scientist – cannot be detached from the tools with which he works. Sometimes, he or she needs to be an engineer or computer programmer.

In the following sections, we show some of effects that we can achieve with different types of algorithm. Partly for reasons of space, we avoid any detailed description of the explicit mathematical form of the algorithm (but see [3,5]).

2. Algorithms for symmetric design: I

Let us start with a hypothetical situation. Suppose that a designer wants to create a wallpaper pattern based on lions. What type of algorithm should be used? Is it possible to design a 'lion algorithm' that will reliably produce 'lion-like' images and which can be adjusted to vary the shape and textures of the lion? What type of symmetry should the designer use to convey, for example, a sense of motion or speed?

In one sense it is relatively straightforward to implement a lion algorithm. Take a scanned image of a lion and symmetrize over the appropriate symmetry group. This is the kaleidoscopic method employed so effectively in Kevin Lee's *KaleidoMania!* program. However, symmetric images produced by symmetrization techniques typically lack global coherence. More precisely, the process of symmetrization typically introduces discontinuities in the image – unless one avoid overlaps (see [3]). This, of course, is not a criticism of the kaleidoscopic method. Rather, it an observation that to create artistically satisfying images one needs more than a kaleidescope.

Although our approach has not yet led to a simple recipe for producing a 'lion algorithm', one class of algorithms we have experimented with can lead to images with a pronounced naturalistic flavor. In Figure 2 we show one such pattern that would resonate with any visitor to the Australian outback.

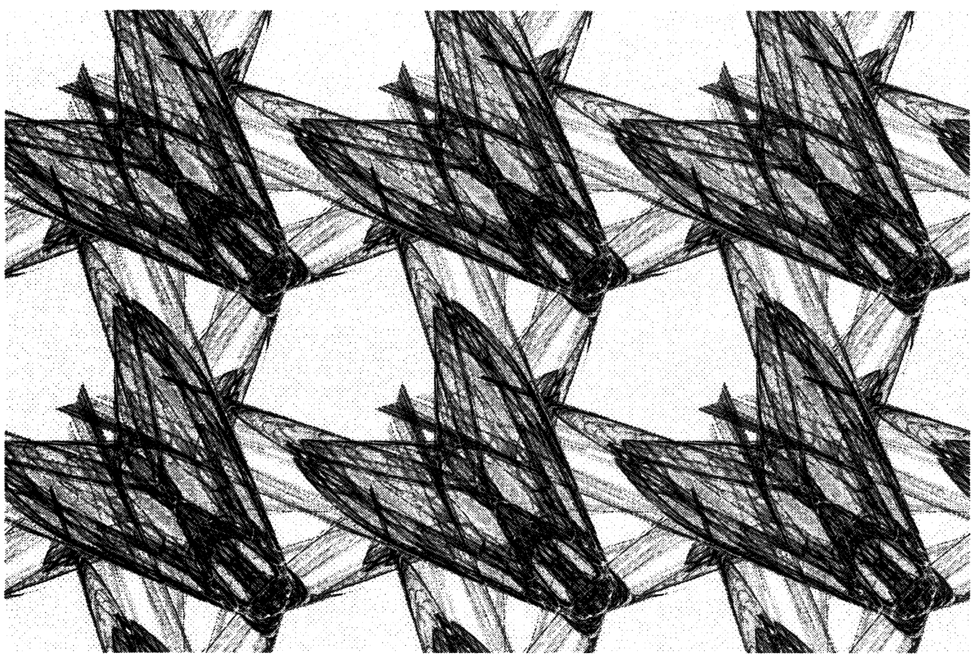

Figure 2: *Fly Quilt*

The quilt shown in Figure 2 is a wallpaper pattern of type **cm** and was constructed using ideas based on iterated function systems. In particular, the algorithm used was non-deterministic. The choice of symmetry **cm** is apposite as this pattern has just one direction of symmetry. We might also have used the symmetry **pm** – then the flies would have lined up wing-tip to wing-tip. The generating function used was a polynomial map in two variables. By making small changes in the coefficients of the generating function, it is easy to make incremental changes in the shape of the flies. We show one possible evolutionary sequence in Figure 3.

3. Algorithms for symmetric design: II

In another direction, we have developed algorithms that can generate abstract angular patterns.

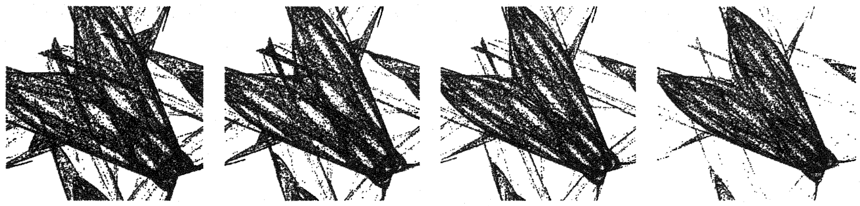

Figure 3: *Evolving flies*

In Figure 4, we show two examples of these patterns. The pattern on the left is of type **p4**, that on the right of type **pmm**. Both patterns were produced using a non-deterministic algorithm and used the same continuous piecewise linear generating function. However, when we attempt to color this type of image, we find that there are drawbacks to using a non-deterministic algorithm. The reason for this is somewhat technical to explain but is related to the fact that the associated invariant measures (see [3]) are often rather uniform. This can make effective coloring quite difficult. It turns out that this problem is less severe when we use a *deterministic* algorithm. Initial investigations of piecewise linear deterministic algorithms with symmetry **p4m** have resulted in a number of quite promising angular designs – some of which I hope to show at *Bridges 2000.*

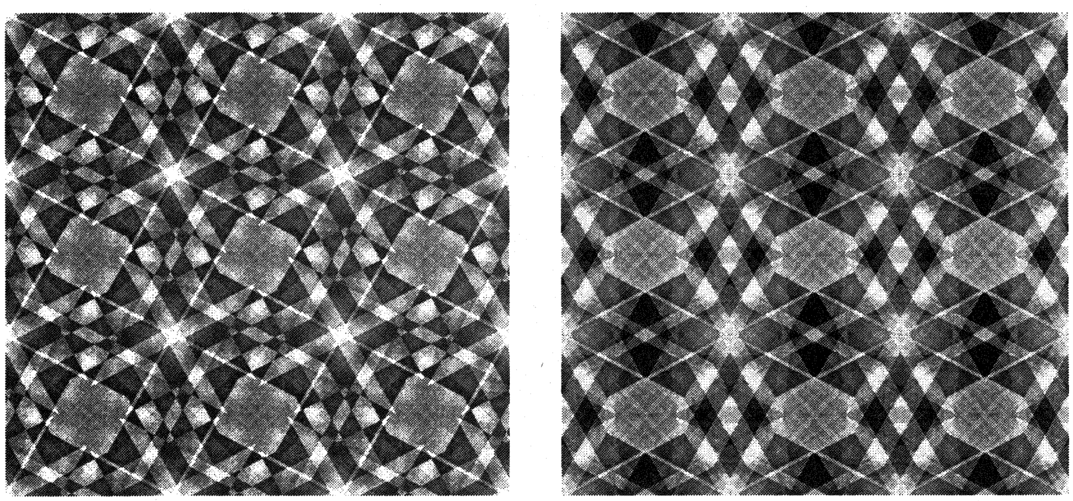

Figure 4: *Abstract designs of types* **p4** *and* **pmm**

4. Algorithms for symmetric design: III

Although it is relatively easy to construct quilt patterns using methods based on iterated function systems, there is no doubt that the most aesthetically pleasing patterns are generated using deterministic algorithms associated to (trigonometric) polynomial mappings. In part this is because deterministic algorithms of this type generally lead to images with sharply defined edges. In addition, there is usually an abundance of fine structure in the image - often highly localized. Experimentation with the algorithm, symmetry and coloring, leads to a symmetrically related set of curves associated to the singular set of the mapping. In Figure 5 we shown an image of type **pm** that was constructed using a deterministic algorithm. The symmetry type **pm** is well adapted to giving a sense of direction and motion to the image - in this case from left to right.

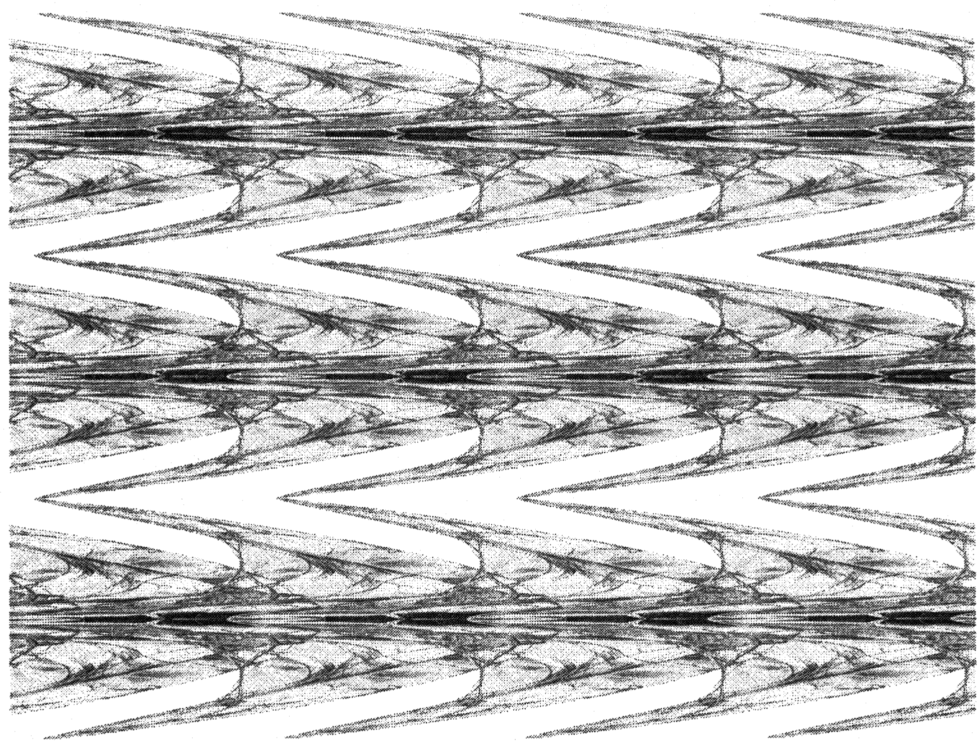

Figure 5: *A wallpaper pattern of type* **pm**

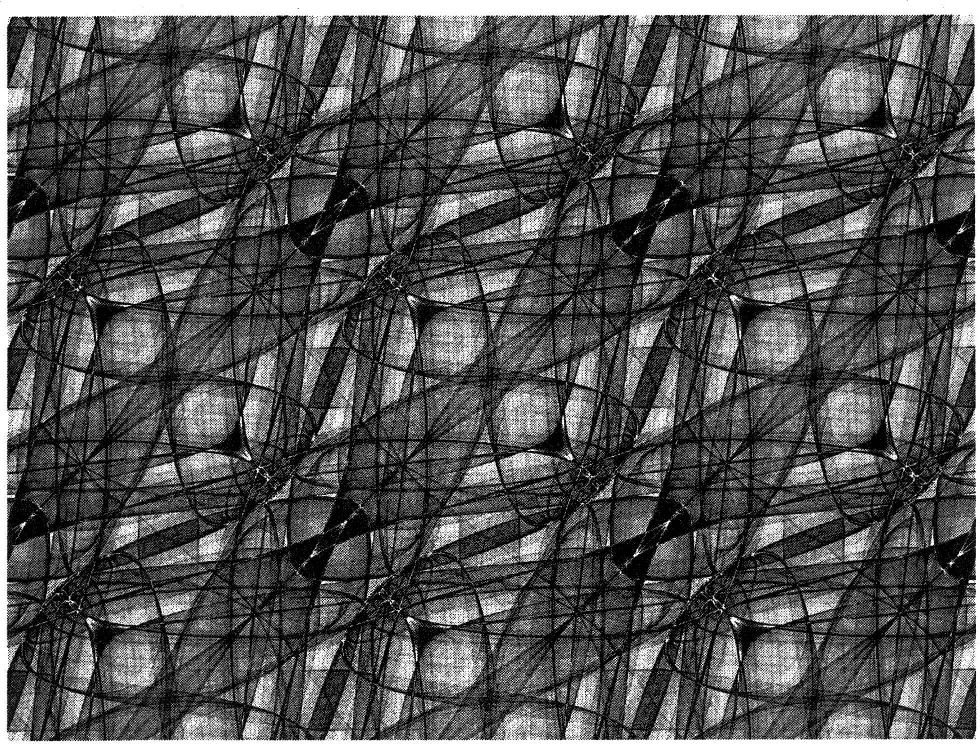

Figure 6: *A wallpaper pattern of type* **cmm**

In Figure 6, we show a deterministic quilt pattern of type **cmm**. This example has a particularly rich geometric structure that is strikingly shown using a grey scale coloring.

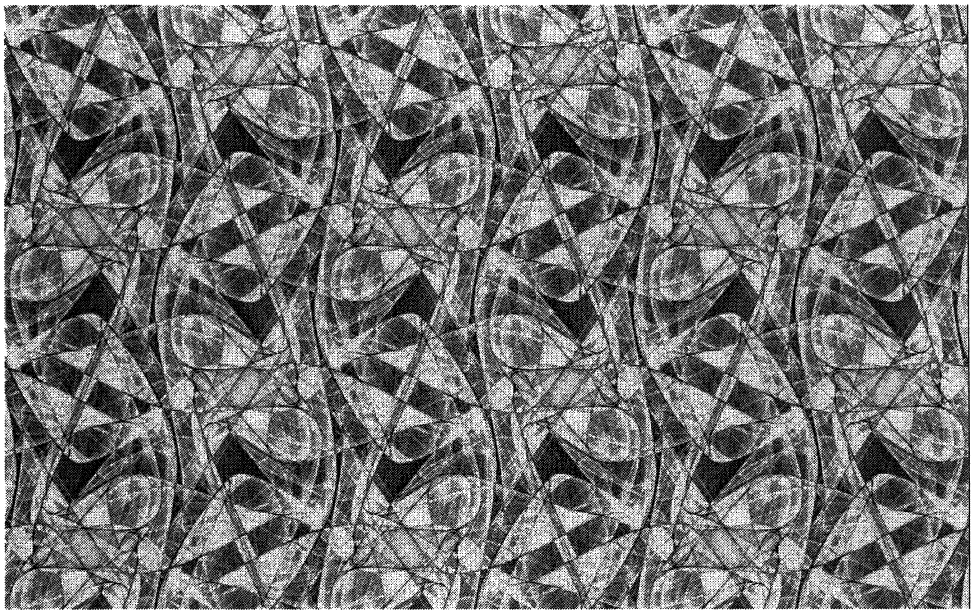

Figure 7: *A wallpaper pattern of type* **pgg**

The design of wallpaper patterns offers additional challenges. It is well known that some symmetry types seem much more attractive than others. Further, there appears to be a cultural bias in our response to symmetry. (We refer the reader to [6, Chapter 1] for a more extensive discussion of cultural and psychological responses to different types of symmetry.) In Western culture, designs with symmetry **p4m** (square tiling) or **p6m** (hexagonal tiling) are ubiquitous. On the other hand, patterns of type **pgg** are rather rare. One can speculate that it is the absence of reflection symmetry in **pgg** patterns that is responsible for their lack of appeal. However, patterns of type **pg** – also without reflection symmetry – are quite common. Generally, we have been rather unsuccessful in designing attractive patterns of type **pgg** using non-deterministic algorithms. We have had more success using deterministic algorithms – indeed, the search for algorithms that would yield appealing versions of patterns of type **pgg** was one of the reasons for our implementation of deterministic algorithms for all the 1- and 2-color quilt patterns. In Figure 7, we show an example of a quilt pattern of type **pgg** designed using a deterministic algorithm (a colored version of this quilt was shown at the 1999 Bridges meeting). Although the symmetry and fine structure of the pattern is quite complex, the overall effect is quite pleasing.

5. Conclusions

Perhaps the main criticism levelled against those who create art using computer graphics or other modern technology is that somehow what is done is not art. My own view is that creative artists are often innovative engineers who develop their chosen medium (I do not claim the converse). Thinking about computer graphics and art in this way leads to many intriguing and challenging problems. In art, technology and science. *Plus ça change, c'est le même chose.*

References

[1] M J Field. 'Harmony and Chromatics of Chaos', In: Proc. of the 1999 Bridges Conference, Southwestern College, Kansas.
[2] M J Field, 'Color symmetries in chaotic quilt patterns', In: Proc. ISAMA 99 (eds: N Friedman

and J Barrallo), Universidad del Pais Vasco, 1999, 181-187.
[3] M J Field. 'Designer Chaos', to appear in special issue of CAD.
[4] M J Field. 'Mathematics through Art - Art through Mathematics', preprint, 2000.
[5] M J Field and M Golubitsky, *Symmetry in Chaos*, Oxford University Press, New York and London, 1992.
[6] D Washburn and D Crowe. *Symmetries of Culture*, University of Washington Press, Seattle, 1988.