

# 数学的対称性を用いた紋様作成ソフトウェアの開発

人間環境科学研究科 環境情報学専攻 環境情報システム学分野

顔紅梅

## 第1章 はじめに

紋様は生活に密着している美の芸術である。衣類、食器、壁紙、ビルの壁、お店の飾り、等等、私たちの身の周りには紋様があふれている。それらの紋様の中でも特に同じ図形が四方に繰り返しあらわれ、あたかも全平面を覆うように広がっている紋様を繰り返し紋様と呼ぶ。同じ形を繰り返し描くことにより、平面全体に果てしなく広がっていると、見る人に想像させている。

繰り返し紋様は平行移動、回転、鏡映<sup>(1)</sup>、すべり鏡映<sup>(2)</sup>などの対称性を持っている。この対称性は実用的な紋様を創作することに利用できる。すなわち、基本的なパターンから対称性を適用して、いろいろな紋様が創作される。

本研究の目的は数学的な理論とコンピュータ技術の結合によって新しい紋様デザインの創出を図ることである。このソフトを身近な道具として提供し伝統産業の活性化のための新しい紋様のクリエイションツールとして役に立てもらいたい。その一方既存の繰り返し紋様を分析、認識し、その中の基本要素としての素領域<sup>(3)</sup>を抽出し、それを利用して、新しい繰り返し紋様を生成することを新しく提案する。

## 第2章 対称性

繰り返し紋様はその紋様の対称性の群によって扱うことができる。平面上の繰り返し紋様に対応する対称性の群は17タイプに限られることが数学的に証明されている。それらは平面の「平行移動」、ある一点を中心とする「回転」、平面上のある直線を軸とする「鏡映」、「すべり鏡映」に関する紋様の持つ性質により分類することができる。対称性の群について解説した文献が多数発表されているがそれらの中で、ここでは、D.Schattschneiderの論文を参考にし、その中の分析法(表1)を使用した。また、17タイプの対称性群の表記は現在数学の分野で標準的に使用されている国際記号を用いる。

## 第3章 プログラムの作成

### 3.1 Java について

最初 VB(Visual Basic)を使って繰り返し紋様を生成するプログラムを作った。数学的な対称性理論を生かして1つのパターンから17タイプのすべてを生成するプログラムを実現した。そのほか生成した紋様の保存、印刷などの機能もできた。京都西陣織産業の人たちに試用してもらった。

VBは命令を順次記述していくタイプのプログラム言語と違い、まずプログラムのユーザインタフェースである画面を設計する。その画面上のボタンなどに対して機能を割り付ける形でプログラムを作成することができる。同社のWindows95などのOSと統一的な操作性を持ったプログラムを簡単に作成することもひとつの特徴で、使いやすい言語である。しかし、VBやCなどのプログラミング言語ではプロセッサ、多くの場合さらにオペレーティングシステムに固有なコンパイラを使って、ソースコードを実行可能コードにコンパイルする。コンパイル後の実行可能ファイルは単体で、コンパイルを行ったマシン上で実行することができる。この方法の欠点の1つは、移植性がないことである。あるオペレーティングシステム上でコンパイルされたコードは、それ以外のオペレーティングシステム上で実行することはできない。実行するシステムが異なれば、そのたびに再コンパイルが必要になる。このような問題を解決するための一つの方法はプラットフォームに依存しない言語Javaを使うことである。

Java言語の移植性の鍵はJavaコンパイラの出力が通常の実行可能コードではないことにある。Javaコンパイラは代わりにバイトコードプログラムと呼ばれる最適化された命令のセットを生成する。バイトコードは決まった形式の一連のバイトからなる。バイトコードプログラムは、Javaランタイムシステム(Java Virtual Machine(JVM)とも呼ばれる)

---

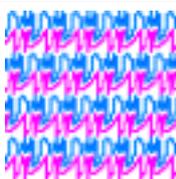
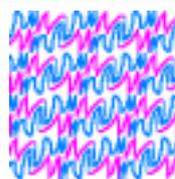
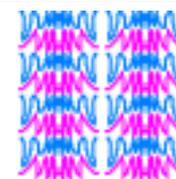
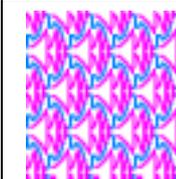
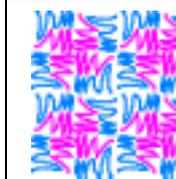
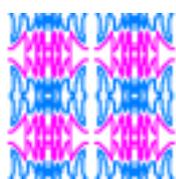
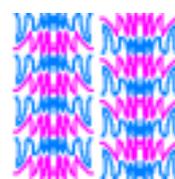
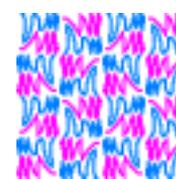
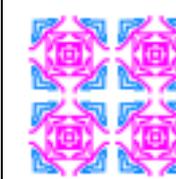
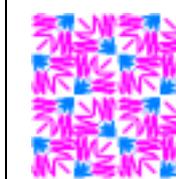
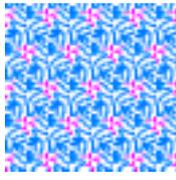
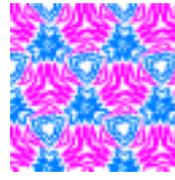
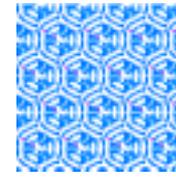
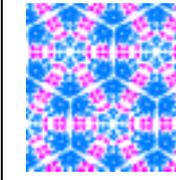
(1). 平面上に直線があるとき、その直線に関する「鏡映」とは平面上の各点をその直線に関して対称な点に移す1対1変換のことである。  
(2) 平面上に直線があるとき、その直線に関する「すべり鏡映」とはその直線に関して鏡映した後その直線の方向に、ある長さの平行移動をすることである。  
(3) .素領域: 平面を覆う繰り返し紋様の最も小さい対称性を持ってない領域である。プログラムで判断して入力パターンから抽出する。

表 1.17 タイプ繰り返し紋様群の分類判定条件表

タイプの名前	格子の形	回転の最小角度	鏡映軸	自明でないすべり鏡映軸	分類のための補助的性質
p1	平行四辺形	360°	なし	なし	
pg	長方形			あり	
pm	長方形		あり	なし	
cm	菱形			あり	
p2	平行四辺形	180°	なし	なし	
pgg	長方形			あり	
pmm	長方形		あり	なし	
pmg	長方形			あり	鏡映軸が全て平行である
cmm	菱形			あり	直交している鏡映軸がある
p3	六角形	120°	なし	なし	
p3m1	六角形		あり	あり	1/3 回転の中心は全て鏡映軸の上にある
p31m	六角形				鏡映軸の上でない 1/3 回転の中心がある
p4	正方形	90°	なし	なし	
p4m	正方形		あり	あり	1/4 回転の中心は全て鏡映軸の上にある
p4g	正方形				鏡映軸上でない 1/4 回転の中心がある
p6	六角形	60°	なし	なし	
p6m	六角形		あり	あり	

\* 繰り返し紋様はどんなタイプであるかは、表 1 に示すような格子の形、回転の角度、鏡映があるかどうか、すべり鏡映があるかどうかなどの性質によって分類できる。

表 2. 繰り返し紋様を生成する結果

 入力パターン					
					
p1	p2	pm	pg	cm	p4
					
pmm	pmg	pgg	cmm	p4m	p4g
					
p3	p3m1	p31m	p6	p6m	

によって解釈される。また、あるプラットフォーム上で生成されたバイトコードプログラムは、JVM がインストールされていればどのプラットフォーム上でも実行できる。すなわち、PC 上でコンパイルされた Java プログラムを、UNIX ワークステーションや Mac 上で実行することもできる。

### 3.2 プログラムについて

プログラムは機能によって 2 つに大きく分けられた。1 つは入力パターンから 17 タイプの繰り返し紋様を生成する。もう 1 つは後の章で詳述するように既存の繰り返し紋様を認識する。この 2 つの機能を応じて繰り返し紋様の生成と繰り返し紋様の認識のためにそれぞれ 2 つのインタフェイスを定義した。この 2 つのインタフェイスにより、独立的にそれ自身の機能が実現できる。それに 2 つのインタフェイスの間はパラメーターの転送より、繰り返し紋様を生成するインタフェイスから生成した繰り返し紋様は、認識するインタフェイスに利用できる。逆に繰り返し紋様を認識するインタフェイスから認識した結果は再び生成するインタフェイスに利用できる。

#### 3.2.1 繰り返し紋様の生成

第 2 章に述べた数学的な対称性を利用して繰り返し紋様を生成する。主に以下のような機能を持っている。

(a) 画像ファイル (.gif、.jpg) から、画像を読み込む。読み込んだ画像は入力パターンという。(b) 入力パターンに基づいて、17 タイプ繰り返し紋様を生成する。(c) 生成した繰り返し紋様は、パラメーターとして、繰り返し紋様を認識するインタフェイスに渡す。(d) 繰り返し紋様を認識するインタフェイスから渡すパラメーターを受け取って、入力パターンとして使う。

繰り返し紋様を生成するためには回転の計算、画像濃度を補間するための線形補間法、直線に対する鏡映点を求めるなどの数学的な手法を用いた。表 2 は生成する結果を示す。

#### 3.2.2 繰り返し紋様の認識

繰り返し紋様の認識と言うのは、繰り返し紋様を分析してこの繰り返し紋様はどんなタイプの繰り返し文様であるか、分析結果を出すことである。既存の繰り返し紋様を分類するには役に立つ。さらに、出した結果を利用して新しい紋様が生成できる。すなわち、既存の繰り返し紋様から新しい繰り返し紋様へ変化できる。

繰り返し紋様を認識する手順：

(a). 繰り返し紋様から繰り返しされた部分を抽出する。画像というのは人の目で見れば同じものでもデジタルした数値で完全には同じではないの場合もある。例えば RGB 値は(255、0、0)と(250、0、0)の場合、数値が違うけど人の目で見れば、赤い色で、同じ色と見える。プログラムで厳格的に数値と数値を比較すれば繰り返し部分の抽出は不可能である。それを認識するためにここで述べたあいまいさの部

分を考えて判定は  $s = \frac{\sum |f_{ij} - g_{ij}|}{\sum f_{ij}}$  の値を用いた。

(b). 繰り返し部分から、各タイプに応じて、素領域を抽出し、ユニットを形成する。

(c). (b)のユニットは(a)の繰り返し部分と比較し、どのくらい一致するのを判断する。式  $per = 100 - s \times 100$  で一致する比率を計算する。一番大きいのはこの繰り返し紋様を属するタイプと認める。

繰り返し紋様の認識インタフェイスは、主に、以下のような機能を持っている。

(a) 画像ファイル (.gif、.jpg) から、画像を読み込む。(b) 読み込んだ画像から、繰り返し部分を抽出する。(c) (b)に抽出した繰り返し部分について、17 タイプのどれかを分析し、認識する結果を出す。それに、繰り返し部分から得た素領域を画像の形式で表示する。(d) 繰り返し部分から得た素領域は、パラメーターとして、繰り返し紋様を生成するインタフェイスに渡す。(e) 繰り返し紋様を生成するインタフェイスから渡すパラメーターを受け取って、繰り返し紋様として表示する。この繰り返し紋様について、まだ、繰り返し部分を抽出、分析、認識結果を出す。

## 第 4 章 まとめ

### 4.1 実現されたこと

本研究は、VB で 17 タイプのすべてを生成するプログラムを完成した。初期のバージョンを西陣織産業の人たちに試用してもらい、紋様デザインのために活用され、いい評価を得た。実用的なソフトとして、大きな期待が集められた。Java でも 17 タイプのすべてを生成するプログラムを完成して、さらに、17 タイプの中に 10 タイプを認識するプログラムを完成した。

繰り返し紋様を生成する結果 (表 2) から見る

と、シンプルな画像から、予想以外の繰り返し紋様が生成できる。

認識結果(表3)から見ると、繰り返し紋様を生成するプログラムを使って生成した繰り返し紋様について認識する結果は100%を一致することが達成できる。既存の繰り返し紋様について一致する比率は前者より低いことが分かった。これは既存の繰り返し紋様の中にあるものはデジタル化の品質とある程度関係がある。それに、既存の繰り返し紋様の認識では、シンプルな繰り返し紋様を認識しやすい。

#### 4.2 解決すべき問題点

繰り返し紋様の認識では、また7タイプの認識ができない。それを解決する方法の一つはプログラムで対話的に座標を入力したり、範囲を決めたり、しながら認識していく。

#### 4.3 今後の課題と発展性

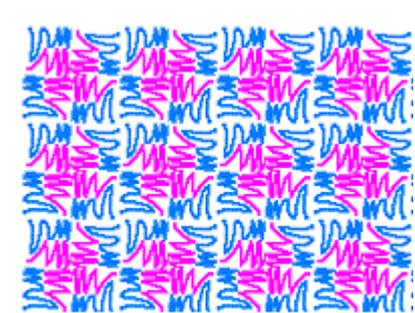
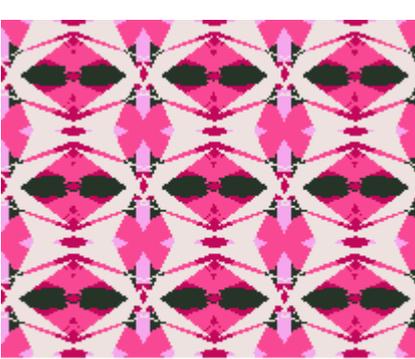
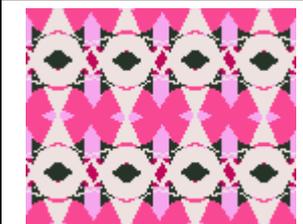
繰り返し紋様の生成と認識の最初の入力パターンは、画像ファイルからしか得られない。今のプログラムで画像を描画する機能が組み込んだら、さら

に使い勝手のよいものになる。また、同じタイプでも素領域、回転の中心点、鏡得軸などの取り方によって、形成した繰り返し文様が違う場合もある。これらを組み込めば、もっと豊かな紋様を形成できる。

#### 参考文献：

- 1) D.Schattschneider : The plane symmetry groups : Their recognition and notation、 American Mathematical Monthly 85(6)、439-50、1978
- 2) 難波誠：群と幾何学、現代数学社、1997
- 3) 川久保勝夫・宮西正直：現代数学序説( )、大阪大学出版会、1996
- 4) Brett Spell : プロフェッショナル Java 上・コンポーネント、スレッド、イベント処理編、株式会社インプレス、2001
- 5) Satyaraj Pantham : 速習 Java Swing プログラミング、ソフトバンク パブリッシング株式会社、1999
- 6) David M. Geary: グラフィック Java2Vo1.2 Swing 編(上、下)、株式会社アスキー、2000
- 7) Daniel Groner・K.C.Hopson・Harish Prabandham・Todd Sundsted : Java API スーパーバイブル1 標準クラスライブラリ編、株式会社 翔泳社、1997

表3.認識する結果について

繰り返し紋様	抽出した繰り返し部分	分析結果			
		タイプ	素領域	比率	再生成繰り返し紋様
 <p>p4.GIF</p>		p4		100	
		p4g		33	
		p4m		21	
 <p>pmm.gif</p>		pmm		78	
		pmg		29	

\* この表の比率と言うのは、各タイプに応じて、抽出した素領域でもう一度ユニットを生成して、このユニットと表の中に示した抽出した繰り返し部分と比較し、どのぐらい一致しているのが示している。